

Quaternion-Valued Nonlinear Adaptive Filtering

Bukhari Che Ujang, *Student Member, IEEE*, Clive Cheong Took, *Member, IEEE*, and Danilo P. Mandic, *Senior Member, IEEE*

Abstract—A class of nonlinear quaternion-valued adaptive filtering algorithms is proposed based on locally analytic nonlinear activation functions. To circumvent the stringent standard analyticity conditions which are prohibitive to the development of nonlinear adaptive quaternion-valued estimation models, we use the fact that stochastic gradient learning algorithms require only local analyticity at the operating point in the estimation space. It is shown that the quaternion-valued exponential function is locally analytic, and, since local analyticity extends to polynomials, products, and ratios, we show that a class of transcendental nonlinear functions can serve as activation functions in nonlinear and neural adaptive models. This provides a unifying framework for the derivation of gradient-based learning algorithms in the quaternion domain, and the derived algorithms are shown to have the same generic form as their real- and complex-valued counterparts. To make such models second-order optimal for the generality of quaternion signals (both circular and noncircular), we use recent developments in augmented quaternion statistics to introduce widely linear versions of the proposed nonlinear adaptive quaternion valued filters. This allows full exploitation of second-order information in the data, contained both in the covariance and pseudocovariances to cater rigorously for second-order noncircularity (improperness), and the corresponding power mismatch in the signal components. Simulations over a range of circular and noncircular synthetic processes and a real world 3-D noncircular wind signal support the approach.

Index Terms—Augmented quaternion statistics, \mathbb{H} -circularity, nonlinear adaptive filtering, quaternion least mean square, widely linear modeling, widely linear quaternion least mean square, wind prediction.

I. INTRODUCTION

THE recent resurgence of quaternion-valued signal processing is due to their great potential in the modeling of 3- and 4-D data. Owing to their convenience over real-valued vectorial models, they have found application across the areas of engineering, computer graphics [1], and robotics [2]. In the statistical signal processing field, quaternions have been employed in adaptive filtering, including Kalman filtering [3] and stochastic gradient algorithms, such as the quaternion least mean square (QLMS) [4]. However, they are still relatively underexplored in nonlinear filtering, neural networks, and blind source separation communities, mainly because of

problems arising due to the lack of analytic nonlinear functions in the quaternion domain \mathbb{H} . Although quaternion nonlinear functions have been implemented, for example, the quaternion independent component analysis (ICA) algorithm [5], [6], the analyticity of such function has not been rigorously examined. The very stringent Cauchy–Riemann–Fueter (CRF) conditions [7] ensure that the only globally analytic quaternion-valued functions are linear functions and constants. This is a serious obstacle as the CRF conditions prevent us from choosing the standard nonlinear activation functions (tanh, logistic) as the nonlinearities in nonlinear quaternion-valued adaptive estimation. To partially overcome this issue, a suboptimal solution in the form of a “split” nonlinear quaternion function that treats each channel separately (as a real channel) passed through a real smooth nonlinearity was employed in [8].

Early nonlinear learning algorithms that used the “split” nonlinear quaternion activation function include the quaternion multilayer perceptron (QMLP) [8], which, owing to the power of quaternion algebra, exhibited enhanced performance over vector-based algorithms [9], [10]. However, the training of QMLP neglects the non-commutativity aspect of the quaternion algebra and thus does not exploit the full potential of the processing in the quaternion domain; this issue was addressed with the split quaternion nonlinear adaptive filtering algorithm (SQAFA) [11]. The nonlinearities used in SQAFA were still standard real activation functions applied componentwise, thus prohibiting a rigorous treatment of the cross-information across the data channels.

It is important to note that most practical learning algorithms are gradient-based [8]–[11], where the operating point moves at every sample interval, to build nonlinear adaptive models and we only require local analyticity at a point. In analogy to the complex domain, where the so-called fully complex nonlinearities (elementary transcendental functions) provide means for generic extensions of real neural networks [12], [13], our aim is to show that the class of elementary transcendental functions, such as tanh, are locally analytic in \mathbb{H} and thus permit generalization of neural networks to the quaternion domain. This is not possible to achieve using the standard CRF conditions [7], which are too restrictive. To this end, we exploit some recent results on local analyticity [14], and, through a cumbersome derivation, we show analytically the possibility of building generic quaternion-valued nonlinear adaptive filters for the most commonly used activation functions, such as tanh. The derivation involves proving local analyticity of exponential functions and their ratios, thus enabling the local analyticity for transcendental nonlinear activation functions in \mathbb{H} . This set of results allows us to establish the nonlinear adaptive filtering and neural network paradigm in \mathbb{H} , in the

Manuscript received July 16, 2010; revised April 4, 2011; accepted May 12, 2011. Date of publication June 27, 2011; date of current version August 3, 2011.

The authors are with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2BT, U.K. (e-mail: che.che-ujang07@imperial.ac.uk; c.cheong-took@imperial.ac.uk; d.mandic@imperial.ac.uk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TNN.2011.2157358

same way they are established in \mathbb{R} and \mathbb{C} [13], [15]–[20], as a natural generalization.

In this paper, we introduce a class of fully quaternion locally analytic nonlinear functions suitable for quaternion-valued nonlinear adaptive filtering. We also show that full second-order statistical information in the quaternion domain can be exploited by combining the proposed nonlinear models with the so-called augmented quaternion statistics and the widely linear model [21], [22]. For simplicity, the analysis and derivations are provided for a single nonlinear perceptron and its widely linear counterpart. Extension to large-scale architectures is readily achieved in the same way as the corresponding extensions in \mathbb{R} and \mathbb{C} , and are out of the scope of this paper.

The rest of this paper is organized as follows. Section II introduces basic operations of quaternion algebra. Section III reviews the basic concept behind augmented quaternion statistics. This is followed by a review of the quaternion analyticity conditions along with the analysis of the quaternion exponential function and quaternion tanh function in Section IV. Section V derives the proposed learning algorithms and their widely linear counterparts, followed by their convergence analyzes. Section VI compares the performances of the proposed algorithms against the existing algorithms of the kind. The results are discussed in Section VII, and this paper concludes in Section VIII.

II. QUATERNION ALGEBRA

Quaternions are an associative algebra defined over the field of real numbers \mathbb{R} with a quaternion variable q given by

$$q = [q_a, \bar{q}] = q_a + q_b \iota + q_c J + q_d \kappa, \quad \{q_a, q_b, q_c, q_d \in \mathbb{R}\} \quad (1)$$

where \bar{q} is the vector part and ι, J, κ are both orthogonal unit vectors and imaginary units. These orthogonal unit vectors are related by

$$\begin{aligned} \iota J &= \kappa & J \kappa &= \iota & \kappa \iota &= J \\ \iota J \kappa &= \iota^2 = J^2 = \kappa^2 &= -1. \end{aligned} \quad (2)$$

The addition and subtraction operations in quaternion algebra are defined similar to complex algebra (componentwise) and are given by

$$\begin{aligned} w + x &= [w_a + x_a, \bar{w} + \bar{x}] \\ &= (w_a + x_a) + (w_b + x_b)\iota + (w_c + x_c)J \\ &\quad + (w_d + x_d)\kappa. \end{aligned} \quad (3)$$

The multiplication operates quite differently the multiplication of quaternions is given by

$$wx = [w_a, \bar{w}][x_a, \bar{x}] = [w_a x_a - \bar{w} \cdot \bar{x}, w_a \bar{x} + x_a \bar{w} + \bar{w} \times \bar{x}] \quad (4)$$

where the symbols “ \cdot ” and “ \times ” denote, respectively, the dot product and cross product. The presence of the cross product makes the quaternion multiplication noncommutative, i.e., $wx \neq xw$. The conjugate of a quaternion q is

$$q^* = [q_a, \bar{q}]^* = [q_a, -\bar{q}] = q_a - q_b \iota - q_c J - q_d \kappa \quad (5)$$

and the norm square is

$$\|q\|_2^2 = qq^* = q^*q = q_a^2 + q_b^2 + q_c^2 + q_d^2. \quad (6)$$

Operators of equivalence important to this paper are the three quaternion involutions (self-inverse mappings)

$$\begin{aligned} q^\iota &= -\iota q \iota = q_a + q_b \iota - q_c J - q_d \kappa \\ q^J &= -J q J = q_a - q_b \iota + q_c J - q_d \kappa \\ q^\kappa &= -\kappa q \kappa = q_a - q_b \iota - q_c J + q_d \kappa. \end{aligned} \quad (7)$$

In the sequel, all the constants and variables are assumed quaternion-valued, unless stated otherwise.

III. AUGMENTED QUATERNION STATISTICS

The concept of augmented statistics in division algebra was first introduced to define the notion of second-order noncircularity, or improperness, for complex random normal vectors [23], and was subsequently extended to non-normal vectors [24]. In the complex domain \mathbb{C} , the second-order properness of a complex random vector can be fully characterized by its covariance $C_{\mathbf{z}\mathbf{z}}$ and pseudocovariance $\mathcal{P}_{\mathbf{z}\mathbf{z}}$, defined as [23]

$$C_{\mathbf{z}\mathbf{z}} = E(\mathbf{z}\mathbf{z}^H) \quad \mathcal{P}_{\mathbf{z}\mathbf{z}} = E(\mathbf{z}\mathbf{z}^T) \quad (8)$$

where $(\cdot)^H$ and $(\cdot)^T$ denote, respectively, the Hermitian and transpose vector operator, and $\mathbf{z} = \mathbf{x} + \mathbf{y}\iota$ where \mathbf{x} and \mathbf{y} are real-valued. A complex random vector is termed “circular” if its probability distribution is rotation-invariant. In the second-order sense, this implies that the real and imaginary components have equal variance and are not correlated: that is, the pseudocovariance $\mathcal{P}_{\mathbf{z}\mathbf{z}}$ vanishes [13], [20].

A. \mathbb{C}^η -Circular Quaternion Random Variables

The concept of augmented statistics was subsequently extended to the quaternion domain in [25], albeit with the restriction of a single rotation axis of ι, J, κ . A quaternion random variable q that obeys this condition is said to be \mathbb{C}^η -circular, and is defined as

$$q \triangleq q e^{j\theta} \quad \forall \theta \quad (9)$$

for one and only one pure imaginary unit η , where $\eta \in \{\iota, J, \kappa\}$. The symbol \triangleq denotes equality in terms of the probability distribution function (pdf) and the symbol θ represents the angle of rotation.

B. \mathbb{H} -Circular Quaternion Random Variables

The restriction of a single rotation axis for \mathbb{C}^η -circular random variable has proven to be too rigid in practical scenarios and a generalization, allowing for a pdf along any two arbitrary axis of rotation to be circular, was introduced in [26]. A quaternion random variable q that satisfies this condition is said to be \mathbb{H} -circular, or \mathbb{Q} -proper, and is defined as

$$q \triangleq q e^{j\theta} \quad \forall \theta \quad (10)$$

for all the pure imaginary units $\eta \in \{\iota, J, \kappa\}$. An \mathbb{H} -circular quaternion random variable is circular in all its dimensions, meaning that the scatterplot of any two components of $\{1, \iota, J, \kappa\}$ is circular. We can now define a \mathbb{Q} -proper (second-order circular) random variable q as the one that has equal powers in all the components, q_a, q_b, q_c , and q_d .

C. Augmented Second-Order Statistics of Quaternion Random Vectors

Similar to the complex case, in general the covariance alone is not sufficient to fully describe the complete second-order information within the quaternion random vector. To provide a generic framework for second-order statistical modeling of quaternion vectors, i.e., to deal with \mathbb{Q} -improper signals, we also need to employ complementary covariance matrices (pseudocovariances). These complementary covariance matrices are termed the ι -covariance $C_{\mathbf{q}\iota}$, J -covariance $C_{\mathbf{q}J}$, and κ -covariance $C_{\mathbf{q}\kappa}$, and are given by [21] and [22]

$$C_{\mathbf{q}\iota} = E \left\{ \mathbf{q}\mathbf{q}^{\iota H} \right\} \quad C_{\mathbf{q}J} = E \left\{ \mathbf{q}\mathbf{q}^{JH} \right\} \quad C_{\mathbf{q}\kappa} = E \left\{ \mathbf{q}\mathbf{q}^{\kappa H} \right\}. \quad (11)$$

Thus, the complete second-order characteristics of the quaternion random vector are described by the augmented covariance matrix $C_{\mathbf{q}}^a$ of an augmented vector $\mathbf{q}^a = [\mathbf{q}^T \mathbf{q}^{\iota T} \mathbf{q}^{JT} \mathbf{q}^{\kappa T}]^T$, given by¹

$$C_{\mathbf{q}}^a = E \{ \mathbf{q}^a \mathbf{q}^{aH} \} = \begin{bmatrix} C_{\mathbf{q}\mathbf{q}} & C_{\mathbf{q}\iota} & C_{\mathbf{q}J} & C_{\mathbf{q}\kappa} \\ C_{\mathbf{q}\iota}^H & C_{\mathbf{q}\iota\iota} & C_{\mathbf{q}\iota J} & C_{\mathbf{q}\iota\kappa} \\ C_{\mathbf{q}J}^H & C_{\mathbf{q}J\iota} & C_{\mathbf{q}JJ} & C_{\mathbf{q}J\kappa} \\ C_{\mathbf{q}\kappa}^H & C_{\mathbf{q}\kappa\iota} & C_{\mathbf{q}\kappa J} & C_{\mathbf{q}\kappa\kappa} \end{bmatrix} \quad (12)$$

where the submatrices in (12) are calculated according to²

$$C_{\delta} = E \{ \mathbf{q}\delta^H \} \quad C_{\alpha\beta} = E \{ \alpha\beta^H \} \\ \delta \in \{ \mathbf{q}^{\iota}, \mathbf{q}^J, \mathbf{q}^{\kappa} \} \quad \alpha, \beta \in \{ \mathbf{q}, \mathbf{q}^{\iota}, \mathbf{q}^J, \mathbf{q}^{\kappa} \}. \quad (13)$$

A quaternion random vector \mathbf{q} is said to be \mathbb{C}^{ι} -circular when the J -covariance $C_{\mathbf{q}J}$ and κ -covariance $C_{\mathbf{q}\kappa}$ vanish [22]. Similar definitions hold for \mathbb{C}^J -circular and \mathbb{C}^{κ} -circular quaternion random vectors. The semi-widely linear model, based on the statistics of \mathbb{C}^{η} circularity, is described in [22]. On the other hand, an \mathbb{H} -circular quaternion random vector \mathbf{q} has the property that it is not correlated with its quaternion involutions \mathbf{q}^{ι} , \mathbf{q}^J , and \mathbf{q}^{κ}

$$E \left\{ \mathbf{q}\mathbf{q}^{\iota H} \right\} = \mathbf{0} \quad E \left\{ \mathbf{q}\mathbf{q}^{JH} \right\} = \mathbf{0} \quad E \left\{ \mathbf{q}\mathbf{q}^{\kappa H} \right\} = \mathbf{0} \quad (14)$$

yielding the augmented covariance matrix $C_{\mathbf{q}}^a$ in (12) of a \mathbb{H} -circular random vector in the form³

$$C_{\mathbf{q}}^a = E \{ \mathbf{q}^a \mathbf{q}^{aH} \} = \begin{bmatrix} C_{\mathbf{q}\mathbf{q}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & C_{\mathbf{q}\mathbf{q}}^{\iota} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & C_{\mathbf{q}\mathbf{q}}^J & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & C_{\mathbf{q}\mathbf{q}}^{\kappa} \end{bmatrix}. \quad (15)$$

¹As long as the covariance matrix $C_{\mathbf{q}\mathbf{q}}$ is nonsingular, then it follows immediately that the other covariance matrices $C_{\mathbf{q}\iota\iota}$, $C_{\mathbf{q}JJ}$, $C_{\mathbf{q}\kappa\kappa}$ have inverses. Therefore, the augmented $C_{\mathbf{q}}^a$ is full rank and nonsingular.

²The matrices $C_{\mathbf{q}\eta\eta}$ are an involution of $C_{\mathbf{q}\mathbf{q}}$ over η and therefore can be simplified to $C_{\mathbf{q}\mathbf{q}}^{\eta}$ where $\eta \in \{ \iota, J, \kappa \}$ [22].

³Any other basis comprising four combinations out of $\{ \mathbf{q}, \mathbf{q}^{\iota}, \mathbf{q}^J, \mathbf{q}^{\kappa} \}$, and their conjugates is equally valid. The basis proposed in [21] and used here, $\mathbf{q}^a = [\mathbf{q}^T \mathbf{q}^{\iota T} \mathbf{q}^{JT} \mathbf{q}^{\kappa T}]^T$, provides most convenient representation, as shown in the augmented covariance structure for \mathbb{H} -circular signals in (12) and (15).

To exploit the complete second-order statistics of quaternion-valued signals, a filtering model similar to the widely linear model in \mathbb{C} needs to be considered [13], [27]. The quaternion widely linear model is based on the augmented basis that builds the matrix $C_{\mathbf{q}}^a$ (12), and can be described by [21], [28] and [22]

$$y = \mathbf{w}^{aT} \mathbf{x}^a = \mathbf{g}^T \mathbf{x} + \mathbf{h}^T \mathbf{x}^{\iota} + \mathbf{u}^T \mathbf{x}^J + \mathbf{v}^T \mathbf{x}^{\kappa} \quad (16)$$

where \mathbf{g} , \mathbf{h} , \mathbf{u} , and \mathbf{v} are the weight vectors, \mathbf{x} is the input signal, \mathbf{x}^{ι} , \mathbf{x}^J , and \mathbf{x}^{κ} are, respectively, its ι , J , and κ involutions, $\mathbf{w}^a = [\mathbf{g}^T \mathbf{h}^T \mathbf{u}^T \mathbf{v}^T]^T$ is the augmented weight vector, and $\mathbf{x}^a = [\mathbf{x}^T \mathbf{x}^{\iota T} \mathbf{x}^{JT} \mathbf{x}^{\kappa T}]^T$ is the augmented random input vector. Another benefit of the quaternion widely linear model is the possibility to determine the degree of properness of quaternion random vectors [29].

IV. NONLINEAR FUNCTIONS IN \mathbb{H}

In \mathbb{C} , the analyticity of a complex function $f(z) = u(x, y) + v(x, y)\iota$ is governed by the Cauchy–Riemann (CR) equations, given by

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y} \quad \frac{\partial v}{\partial x} = -\frac{\partial u}{\partial y}. \quad (17)$$

That is, for a complex function $f(z)$ to be analytic in \mathbb{C} , the derivatives along the real and imaginary axes have to be equal

$$\frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}\iota = 0 \Leftrightarrow \frac{\partial f}{\partial z^*} = 0 \quad (18)$$

where $z = x + y\iota$. By continuity, the analyticity in the quaternion domain can be defined by the generalized Cauchy–Riemann (GCR) conditions, given by [30]

$$\frac{\partial f}{\partial q_a} = -\frac{\partial f}{\partial q_b}\iota \quad \frac{\partial f}{\partial q_a} = -\frac{\partial f}{\partial q_c}J \quad \frac{\partial f}{\partial q_a} = -\frac{\partial f}{\partial q_d}\kappa \quad (19)$$

where $q = q_a + q_b\iota + q_cJ + q_d\kappa$. Only a special form of quaternion linear functions and constants satisfy the GCR conditions, they are therefore too prohibitive for any practical application such as in neural networks, where typically nonlinear neuron models are involved. The GCR conditions were initially proposed for a 4-D domain, with Clifford algebra as their basis, making them unsuitable for applications in \mathbb{H} [31]. To circumvent this issue, Fueter further relaxed these conditions by redefining them based on a quaternion basis, resulting in the CRF conditions given by [7]

$$\frac{\partial f}{\partial q_a} + \frac{\partial f}{\partial q_b}\iota + \frac{\partial f}{\partial q_c}J + \frac{\partial f}{\partial q_d}\kappa = 0 \Leftrightarrow \frac{\partial f}{\partial q^*} = 0. \quad (20)$$

Unlike the GCR conditions, the CRF conditions are defined by a single quaternion partial differentiation which leads to a close analogue of Cauchy’s theorem, Cauchy’s integral formula, and the Laurent expansion [32]. It can be shown that only linear quaternion functions and constants satisfy the CRF conditions [7], limiting the scope for nonlinear adaptive filtering in \mathbb{H} which requires differentiable nonlinear functions. To further relax the quaternion differentiation condition, a “local” analyticity condition was proposed in [14], by using a complex representation of a quaternion to give

$$\frac{\partial f}{\partial q_a} = -\frac{\partial f}{\partial \alpha} \hat{\zeta} \quad (21)$$

where $\hat{\zeta}$ and α are given by

$$\hat{\zeta} = \frac{q_b \iota + q_c J + q_d \kappa}{\alpha} \quad \alpha = \sqrt{q_b^2 + q_c^2 + q_d^2}. \quad (22)$$

The term ‘‘local’’ here refers to the fact that this representation uses ‘‘imaginary’’ unit $\hat{\zeta}$ which depends on the values of q_b , q_c , and q_d [14]. The local analyticity condition only guarantees the first-order differentiability of the single variable quaternion functions at the current operating point. This is perfectly adequate for quaternion-valued gradient descent adaptive filtering algorithms, as they only require the information about the gradient value at a point.

A. Fully Quaternion Functions

For quaternion-valued nonlinear quaternion functions to be suitable for adaptive filtering applications [33], they should share some properties of fully complex nonlinearities, given by:

- 1) $f(z) = u(x, y) + v(x, y)\iota$ is nonlinear in x and y ;
- 2) $f(z)$ has no singularities and is always bounded for all values of z ;
- 3) the partial derivatives $\partial u/\partial x$, $\partial v/\partial y$, $\partial v/\partial x$ and $\partial u/\partial y$ are continuous and bounded;
- 4) $(\partial u/\partial x)(\partial v/\partial y) \neq (\partial v/\partial x)(\partial u/\partial y)$ to ensure continuous learning.

The so-called fully complex activation functions exhibit locally all the above characteristics [12]. Note that fulfilling the third and fourth characteristic is equivalent to fulfilling the CR conditions in (18).

To provide a rigorous basis for nonlinear quaternion-valued adaptive filtering, we need to identify fully quaternion nonlinearities in \mathbb{H} . We shall call the function that satisfies the local analyticity condition in (21) a ‘‘fully quaternion nonlinearity,’’ in the sense of local analyticity. We can now evaluate the analyticity of a function at a given point by analyzing the local derivative within the $\hat{\zeta}$ -plane (with $\hat{\zeta}$ fixed) to obtain the relationship [14]

$$\begin{aligned} \frac{\partial f}{\partial \alpha} &= \frac{\partial q_b}{\partial \alpha} \frac{\partial f}{\partial q_b} + \frac{\partial q_c}{\partial \alpha} \frac{\partial f}{\partial q_c} + \frac{\partial q_d}{\partial \alpha} \frac{\partial f}{\partial q_d} \\ \alpha \frac{\partial f}{\partial \alpha} &= q_b \frac{\partial f}{\partial q_b} + q_c \frac{\partial f}{\partial q_c} + q_d \frac{\partial f}{\partial q_d}. \end{aligned} \quad (23)$$

Based on this relationship, along with $\hat{\zeta}$ and α in (22), the right-hand side of the analyticity condition in (21) is expanded along the orthogonal-axis vectors ι , J , and κ as

$$\begin{aligned} & - \left(\frac{\partial f}{\partial \alpha} \right) \left(\hat{\zeta} \right) \\ &= - \left(\frac{q_b}{\alpha} \frac{\partial f}{\partial q_b} + \frac{q_c}{\alpha} \frac{\partial f}{\partial q_c} + \frac{q_d}{\alpha} \frac{\partial f}{\partial q_d} \right) \left(\frac{q_b \iota + q_c J + q_d \kappa}{\alpha} \right). \end{aligned} \quad (24)$$

By analogy with \mathbb{C} , this yields the characteristics of a fully quaternion locally analytic nonlinearity suitable for gradient-based learning, given by:

- 1) $f(q) = u(q_a, \alpha) + v(q_a, \alpha)\hat{\zeta}$ is nonlinear in q_a and α ;
- 2) $f(q)$ has no singularities and is always bounded for all values of q ;

- 3) the partial derivatives $\partial u/\partial q_a$, $\partial v/\partial \alpha$, $\partial v/\partial q_a$, and $\partial u/\partial \alpha$ are continuous and bounded;
- 4) $(\partial u/\partial q_a)(\partial v/\partial \alpha) \neq (\partial v/\partial q_a)(\partial u/\partial \alpha)$ to ensure continuous learning.

We next focus on the analyticity of the quaternion exponential function e^q , as it serves as a building block to construct transcendental nonlinear quaternion functions, typically used as nonlinear activation functions.

B. Quaternion Exponential Function

The notion of exponential function in \mathbb{H} is not straightforward. Due to the non-commutativity of the quaternion product, there exist several definitions of the quaternion exponential [34], for convenience, we consider the following exponential function [35, p. 9]

$$e^q = e^{q_a + q_b \iota + q_c J + q_d \kappa} = e^{q_a} e^{q_b \iota + q_c J + q_d \kappa}. \quad (25)$$

Expanding the term e^q using the Euler formula leads to

$$\begin{aligned} e^q &= e^{q_a} \left(\cos(\alpha) + \sin(\alpha)\hat{\zeta} \right) \\ &= e^{q_a} \left(\cos(\alpha) + \frac{q_b \sin(\alpha)\iota}{\alpha} + \frac{q_c \sin(\alpha)J}{\alpha} + \frac{q_d \sin(\alpha)\kappa}{\alpha} \right) \end{aligned} \quad (26)$$

where α and $\hat{\zeta}$ are defined in (22). This quaternion exponential function satisfies the analyticity condition in (21), giving the local derivative of the exponential function as

$$\frac{\partial e^q}{\partial q} = \frac{\partial e^q}{\partial q_a} = e^q. \quad (27)$$

Observe that, as desired, this result represents a generic extension of the real and complex derivatives of an exponential. In addition, as gradient-based learning algorithms are local, this result provides a basis for introducing other nonlinearities, such as the elementary transcendental functions, as a vehicle for a class of fully quaternion nonlinear adaptive filters.

C. Local Analyticity of the Quaternion tanh Function

Similar to the complex domain, $\tanh(q)$ in \mathbb{H} can be defined as

$$\tanh(q) = \frac{e^{2q} - 1}{e^{2q} + 1}. \quad (28)$$

Proceeding in a similar manner as when addressing the analyticity of e^q , we first expand $\tanh(q)$ using the Euler formula in (26), leading to (full derivation is given in Appendix A)

$$\tanh(q) = \frac{e^{4q_a} - 1 + 2e^{2q_a} \sin(2\alpha)\hat{\zeta}}{e^{4q_a} + 1 + 2e^{2q_a} \cos(2\alpha)}. \quad (29)$$

To prove the local analyticity, the left-hand side of (21) is obtained by differentiating (29) with respect to q_a , and the right-hand side of (21) is obtained by differentiating (29) with

respect to α , resulting in (a detailed derivation is given in Appendix B)

$$\frac{\partial \tanh(q)}{\partial q_a} = \frac{4e^{6q_a} \cos(2\alpha) + 8e^{4q_a} + 4e^{2q_a} \cos(2\alpha)}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} + \frac{(4e^{2q_a} \sin(2\alpha) - 4e^{6q_a} \sin(2\alpha)) \hat{\zeta}}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} \quad (30)$$

$$= -\frac{\partial \tanh(q)}{\partial \alpha} \hat{\zeta} \quad (31)$$

thus illustrating that $\tanh(q)$ is a locally analytic quaternion function. The expression for a local derivative of $\tanh(q)$ is obtained analogously to the complex case, to this end, we shall first define $\text{sech}(q)$ as

$$\text{sech}(q) = \frac{2}{e^q + e^{-q}}. \quad (32)$$

By expanding (32) into its Euler form and then squaring (full derivation can be found in Appendix C), we have

$$\text{sech}^2(q) = \frac{4e^{6q_a} \cos(2\alpha) + 8e^{4q_a} + 4e^{2q_a} \cos(2\alpha)}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} + \frac{-4e^{6q_a} \sin(2\alpha) + 4e^{2q_a} \sin(2\alpha) \hat{\zeta}}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2}. \quad (33)$$

A comparison of the definition for $\text{sech}^2(q)$ in (33) with $(\partial \tanh(q)/\partial q_a) = -(\partial \tanh(q)/\partial \alpha) \hat{\zeta}$ in (31) shows that they are equivalent; therefore, we have introduced a generic extension of the real and complex tanh function to the quaternion domain, whose derivative is

$$\frac{\partial \tanh(q)}{\partial q} = \text{sech}^2(q). \quad (34)$$

V. NONLINEAR ADAPTIVE FILTERING IN \mathbb{H}

The cost function in quaternion-valued adaptive filtering is usually given by a real function of quaternion variables

$$E(n) = e_a^2(n) + e_b^2(n) + e_c^2(n) + e_d^2(n) = e(n)e^*(n) \quad (35)$$

where the error $e(n) = d(n) - y(n)$ with $d(n)$ and $y(n)$ denoting, respectively, the desired signal and output signal. The terms $e_a(n)$, $e_b(n)$, $e_c(n)$, and $e_d(n)$ denote, respectively, the error component in the real part, ι part, j part, and κ part.

A. Split Quaternion Algorithms

All existing nonlinear quaternion-based adaptive filtering algorithms employ a ‘‘split’’ quaternion nonlinear function, i.e., a real function such as \tanh applied componentwise. In this case, the output signal $y(n)$ is defined as [8], [11]

$$\begin{aligned} y(n) &= \Phi_s(\mathbf{w}^T(n)\mathbf{x}(n)) \\ &= \Phi_a(\mathbf{w}^T(n)\mathbf{x}(n)) + \Phi_b(\mathbf{w}^T(n)\mathbf{x}(n))\iota \\ &\quad + \Phi_c(\mathbf{w}^T(n)\mathbf{x}(n))j + \Phi_d(\mathbf{w}^T(n)\mathbf{x}(n))\kappa \end{aligned} \quad (36)$$

where $\Phi_s(\cdot)$ denotes the ‘‘split’’ quaternion nonlinearity, $\mathbf{w}(n)$ is the weight vector of the adaptive filter, and $\mathbf{x}(n)$ is the filter input. The term $\mathbf{w}^T(n)\mathbf{x}(n)$ can be calculated as $\sum_{i=1}^L w_i(n)x(n-i)$ and is expanded in Appendix D.

Function Φ_a is a real-valued nonlinear activation function applied to the real part of $\mathbf{w}^T(n)\mathbf{x}(n)$, Φ_b to the ι part, Φ_c to the j part, and Φ_d to the κ part. Since this ‘‘split’’ quaternion function is analytic only componentwise, we are not fully exploiting the couplings between the $\{1, \iota, j, \kappa\}$ axes (channels). Note that the odd-symmetry property still applies to the split quaternion function, i.e., $\Phi_s'(\mathbf{w}^T(n)\mathbf{x}(n)) = \Phi_s'(\mathbf{x}^H(n)\mathbf{w}(n))$.

Existing nonlinear quaternion-based algorithms minimize the cost function (35) through a gradient descent weight update specified by

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu \nabla_{\mathbf{w}} E(n) \quad (37)$$

where the gradient $\nabla_{\mathbf{w}} E(n)$ is given by [32] and [36]

$$\nabla_{\mathbf{w}} E(n) = \frac{\partial E(n)}{\partial \mathbf{w}^*} = \frac{\partial E}{\partial \mathbf{w}_a} + \frac{\partial E}{\partial \mathbf{w}_b} \iota + \frac{\partial E}{\partial \mathbf{w}_c} j + \frac{\partial E}{\partial \mathbf{w}_d} \kappa. \quad (38)$$

We shall next employ the proposed fully quaternion activation functions to introduce structurally simpler yet more powerful algorithms than the existing ones.

B. Derivation of Fully Quaternion Algorithms

To introduce the quaternion nonlinear gradient descent (QNGD) algorithm that employs a fully quaternion nonlinear activation function, consider the output $y(n)$ given by

$$y(n) = \Phi(\mathbf{w}^T(n)\mathbf{x}(n)) \quad (39)$$

where $\Phi(\cdot)$ is the fully quaternion nonlinearity such as the $\tanh(q)$ introduced in Section IV. In order to derive the QNGD, we shall express the cost function (35) as

$$\begin{aligned} E(n) &= \left(d(n) - y(n) \right) \left(d^*(n) - y^*(n) \right) \\ &= d(n)d^*(n) - d(n)y^*(n) - y(n)d^*(n) \\ &\quad + y(n)y^*(n). \end{aligned} \quad (40)$$

The error gradient $\nabla_{\mathbf{w}} E(n)$ of QNGD is then calculated as

$$\begin{aligned} \nabla_{\mathbf{w}} E(n) &= -d(n)\nabla_{\mathbf{w}} y^*(n) - \nabla_{\mathbf{w}} y(n)d^*(n) \\ &\quad + y(n)\nabla_{\mathbf{w}} y^*(n) + \nabla_{\mathbf{w}} y(n)y^*(n) \end{aligned} \quad (41)$$

and the expressions for $\nabla_{\mathbf{w}} y(n)$ and $\nabla_{\mathbf{w}} y^*(n)$ are given by (the full derivation is given in Appendix D)

$$\begin{aligned} \nabla_{\mathbf{w}} y(n) &= -\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))2\mathbf{x}^*(n) \\ \nabla_{\mathbf{w}} y^*(n) &= \Phi'^*(\mathbf{w}^T(n)\mathbf{x}(n))4\mathbf{x}^*(n). \end{aligned} \quad (42)$$

Substitute the terms $\nabla_{\mathbf{w}} y^*(n)$ and $\nabla_{\mathbf{w}} y(n)$ into (41) to obtain the QNGD weight update in the form

$$\begin{aligned} \mathbf{w}(n+1) &= \mathbf{w}(n) + \mu \left(2e(n)\Phi'(\mathbf{x}^H(n)\mathbf{w}^*(n))\mathbf{x}^*(n) \right. \\ &\quad \left. - \Phi'(\mathbf{w}^T(n)\mathbf{x}(n))\mathbf{x}^*(n)e^*(n) \right) \end{aligned} \quad (43)$$

where $\Phi'(\cdot)$ is the local derivative of the fully quaternion function.

We shall now extend the QNGD to fully capture the second-order statistics of the signal by incorporating the quaternion widely linear model [21], [22], [28] into its derivation, resulting in the augmented quaternion nonlinear gradient

descent (AQNGD) algorithm.⁴ The output $y(n)$ of AQNGD is defined as

$$y(n) = \Phi(\text{net}(n)) \quad (44)$$

where $\text{net}(n) = \mathbf{g}^T(n)\mathbf{x}(n) + \mathbf{h}^T(n)\mathbf{x}'(n) + \mathbf{u}^T(n)\mathbf{x}^j(n) + \mathbf{v}^T(n)\mathbf{x}^k(n)$ is calculated using the widely linear model in Section III, to which the fully quaternion function $\Phi(\cdot)$ is applied.

The weight updates of the AQNGD are made gradient-adaptive according to

$$\begin{aligned} \mathbf{g}(n+1) &= \mathbf{g}(n) - \mu \nabla_{\mathbf{g}} E(n) & \mathbf{h}(n+1) &= \mathbf{h}(n) - \mu \nabla_{\mathbf{h}} E(n) \\ \mathbf{u}(n+1) &= \mathbf{u}(n) - \mu \nabla_{\mathbf{u}} E(n) & \mathbf{v}(n+1) &= \mathbf{v}(n) - \mu \nabla_{\mathbf{v}} E(n). \end{aligned} \quad (45)$$

The error gradient $\nabla_{\mathbf{w}} E(n)$ in (41) is equivalent to $\nabla_{\mathbf{g}} E(n)$, hence

$$\begin{aligned} \mathbf{g}(n+1) &= \mathbf{g}(n) + \mu \left(2e(n)\Phi'(\text{net}(n))\mathbf{x}^*(n) \right. \\ &\quad \left. - \Phi'(\text{net}(n))\mathbf{x}^*(n)e^*(n) \right) \end{aligned} \quad (46)$$

where $\Phi'(\text{net}(n))$ is a fully quaternion locally analytic derivative, while the error gradient $\nabla_{\mathbf{h}} E(n)$ is given by

$$\begin{aligned} \nabla_{\mathbf{h}} E(n) &= -d(n)\nabla_{\mathbf{h}} y^*(n) - \nabla_{\mathbf{h}} y(n)d^*(n) \\ &\quad + y(n)\nabla_{\mathbf{h}} y^*(n) + \nabla_{\mathbf{h}} y(n)y^*(n). \end{aligned} \quad (47)$$

In the same manner, the terms $\nabla_{\mathbf{h}} y(n)$ and $\nabla_{\mathbf{h}} y^*(n)$ are calculated as

$$\begin{aligned} \nabla_{\mathbf{h}} y(n) &= -\Phi'(\text{net}(n))2\mathbf{x}'^*(n) \\ \nabla_{\mathbf{h}} y^*(n) &= \Phi'^*(\text{net}(n))4\mathbf{x}'^*(n). \end{aligned} \quad (48)$$

Substituting $\nabla_{\mathbf{h}} y(n)$ and $\nabla_{\mathbf{h}} y^*(n)$ into the error gradient $\nabla_{\mathbf{h}} E(n)$ in (47) yields

$$\begin{aligned} \mathbf{h}(n+1) &= \mathbf{h}(n) + \mu \left(2e(n)\Phi'(\text{net}(n))\mathbf{x}'^*(n) \right. \\ &\quad \left. - \Phi'(\text{net}(n))\mathbf{x}'^*(n)e^*(n) \right). \end{aligned} \quad (49)$$

Proceeding in a similar manner, the weight updates for $\mathbf{u}(n)$ and $\mathbf{v}(n)$ are found to be

$$\begin{aligned} \mathbf{u}(n+1) &= \mathbf{u}(n) + \mu \left(2e(n)\Phi'(\text{net}(n))\mathbf{x}^j{}^*(n) \right. \\ &\quad \left. - \Phi'(\text{net}(n))\mathbf{x}^j{}^*(n)e^*(n) \right) \\ \mathbf{v}(n+1) &= \mathbf{v}(n) + \mu \left(2e(n)\Phi'(\text{net}(n))\mathbf{x}^k{}^*(n) \right. \\ &\quad \left. - \Phi'(\text{net}(n))\mathbf{x}^k{}^*(n)e^*(n) \right). \end{aligned} \quad (50)$$

⁴A comprehensive account of widely linear modeling in the complex domain is given in [13].

For convenience, the final weight update of the AQNGD can be written in an augmented form as⁵

$$\begin{aligned} \mathbf{w}^a(n+1) &= \mathbf{w}^a(n) + \mu \left(2e(n)\Phi'(\text{net}(n))\mathbf{x}^{a*}(n) \right. \\ &\quad \left. - \Phi'(\text{net}(n))\mathbf{x}^{a*}(n)e^*(n) \right). \end{aligned} \quad (51)$$

C. Convergence Analysis of QNGD and AQNGD

To proceed with the analysis, we will make three widely used general assumptions [37]:

- 1) the learning rate μ is sufficiently small;
- 2) at convergence, the *a priori* output error $\tilde{z}(n)$ is statistically independent of the input vector $\mathbf{x}(n)$, i.e., $E\{\tilde{z}(n)\mathbf{x}(n)\} = 0$;
- 3) both the *a posteriori* output error $\bar{z}(n)$ and *a priori* output error $\tilde{z}(n)$ are Gaussian.

Applying these assumptions, the final condition for the convergence of QNGD becomes (the full derivation can be found in Appendix E)

$$0 < \mu < \frac{1}{10E\{\mathbf{x}^T(n)\mathbf{x}^*(n)\|\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))\|_2^2}} \quad (52)$$

whereas the condition for AQNGD is

$$0 < \mu < \frac{1}{10E\{\mathbf{x}^{aT}(n)\mathbf{x}^{a*}(n)\|\Phi'(\text{net}(n))\|_2^2}}. \quad (53)$$

Both the upper bounds of (52) and (53) are governed by the expected value of the random input vector and the gradient of the fully quaternion nonlinearity. Note that the upper bound of μ for the AQNGD in (53) is smaller than that of QNGD in (52), due to the larger size of the augmented input vector $\mathbf{x}^a(n)$.

VI. SIMULATIONS

A comprehensive comparison of the performances is provided between the training algorithm for the feedforward QMLP [8], [38] and the nonlinear finite impulse response filters trained with the QMLP learning algorithm (QMLP-FIR) [11], adaptive amplitude split quaternion adaptive filtering algorithm (AASQAF) [11], real-valued nonlinear gradient descent (NGD) [37], and the proposed algorithms based on fully quaternion nonlinear functions, i.e., QNGD and AQNGD. The quaternion multilayer perceptron-finite impulse response (QMLP-FIR), AASQAF, NGD, QNGD, and AQNGD were implemented with a filter length L , whereas the QMLP had one hidden layer comprising of L input neurons, three hidden neurons, and one output neuron. The $\tanh(q)$ nonlinear activation function was used for all the algorithms. The performance was measured using the prediction gain R_p defined as [37]

$$R_p = 10 \log_{10} \frac{\sigma_x^2}{\sigma_e^2} \quad (54)$$

where σ_x^2 and σ_e^2 denote, respectively, the estimated variance of the input and error. The three quaternion-valued processes

⁵The QNGD could also be readily extended to incorporate the semi-widely linear model [22], however this is beyond the scope of this paper.

TABLE I
CLASSES OF QUATERNION WHITE GAUSSIAN NOISE

WGN	\mathbb{H} -Circular	\mathbb{C}^l -Circular	Noncircular
ϵ_a	$\mathcal{N}(0, 1)$	$\mathcal{N}(0, 1)$	$\mathcal{N}(0, 1)$
ϵ_b	$\mathcal{N}(0, 1)$	$\mathcal{N}(0, 1)$	$-0.6\epsilon_a + \mathcal{N}(0, 1)$
ϵ_c	$\mathcal{N}(0, 1)$	$0.4\epsilon_a + 0.8\epsilon_b + \mathcal{N}(0, 1)$	$0.8\epsilon_b + \mathcal{N}(0, 1)$
ϵ_d	$\mathcal{N}(0, 1)$	$0.8\epsilon_a - 0.4\epsilon_b + \mathcal{N}(0, 1)$	$0.8\epsilon_a - 0.4\epsilon_b + \mathcal{N}(0, 1)$

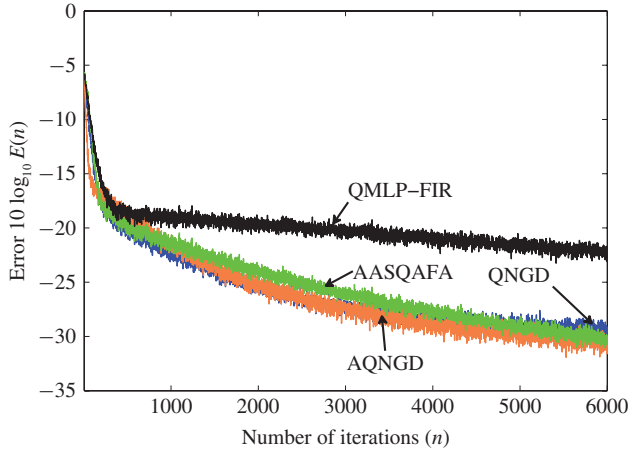


Fig. 1. Learning curves for QMLP-FIR [11], AASQAFA [11], QNGD, and AQNGD on the prediction of linear AR (4) signal (56) driven by \mathbb{H} -circular white Gaussian noise.

considered were the synthetic linear autoregressive (AR) (4) process [13] with a varying degree of circularity, the noncircular chaotic 4-D Saito signal [39], and the real-world 3-D wind field.

A. Linear AR (4)

For this experiment, the input tap length was chosen to be $L = 3$, prediction horizon $M = 1$, and the learning rate $\mu = 5 \times 10^{-3}$.

In the first set of simulations, the performances of AQNGD, QNGD, AASQAFA, and QMLP-FIR were analyzed for a linear AR (4) process with varying degrees of circularity of the driving quaternion quadruply white Gaussian noise (QWGN) $\epsilon(n)$. The QWGN is described by

$$\epsilon(n) = \epsilon_a(n) + \epsilon_b(n)i + \epsilon_c(n)j + \epsilon_d(n)k \quad (55)$$

where ϵ_a , ϵ_b , ϵ_c , and ϵ_d are realizations of real-valued white Gaussian noises (WGN). The properties of noises used to generate different classes of QWGN are shown in Table I. Note that the properties for \mathbb{C}^j -circular and \mathbb{C}^k -circular noises are similar to those of the \mathbb{C}^l -circular input noise, and their descriptions are omitted because of space limitation.

A total of 100 independent simulation trials were conducted and averaged for the linear AR (4) process given by

$$r(n) = 1.79r(n-1) - 1.85r(n-2) + 1.27r(n-3) - 0.41r(n-4) + \epsilon(n). \quad (56)$$

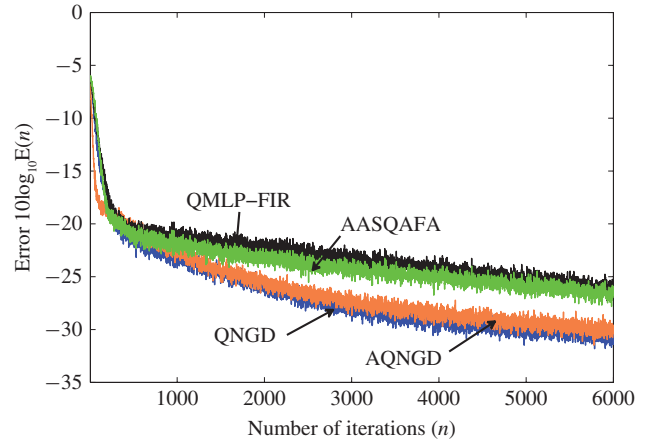


Fig. 2. Learning curves for QMLP-FIR [11], AASQAFA [11], QNGD, and AQNGD on the prediction of linear AR (4) signal (56) driven by \mathbb{C}^l -circular white Gaussian noise.

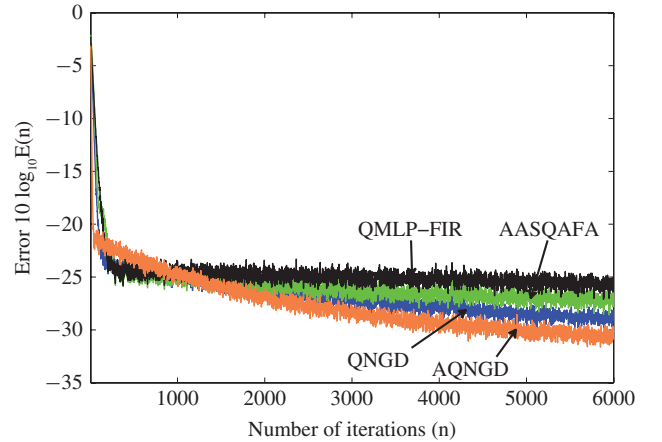


Fig. 3. Learning curves for QMLP-FIR [11], AASQAFA [11], QNGD, and AQNGD on the prediction of linear AR (4) signal (56) driven by noncircular white Gaussian noise.

Fig. 1 shows the learning curves for an \mathbb{H} -circular quaternion white Gaussian noise as the driving noise of the linear AR (4) process. Observe that the proposed AQNGD and QNGD had the fastest convergence, followed by the AASQAFA and QMLP-FIR. It can be seen that the steady-state performances for AQNGD, QNGD, and AASQAFA were similar owing to the matched power of the components of the \mathbb{H} -circular linear AR (4) signal. Fig. 2 depicts the learning curves for the input \mathbb{C}^l -circular white Gaussian noise⁶ for all of the algorithms considered. Similar to the previous case, the AQNGD and QNGD had the fastest convergence, and, as desired, the steady-state results for AQNGD and QNGD were equivalent. In the case of \mathbb{C}^j and \mathbb{C}^k WGN, similar performances were obtained and are omitted here for conciseness. Fig. 3 shows learning curves for all the algorithms considered using a noncircular white Gaussian noise as the input, the AQNGD and QNGD had superior performances over the AASQAFA and QMLP-FIR. It can also be seen that the steady-state performance of AQNGD was superior to that of QNGD, as

⁶The notion of \mathbb{C}^l circularity refers to only having a pair of axes exhibiting complex circularity.

TABLE II
PREDICTION GAIN R_p FOR A LINEAR AR (4) PROCESS WITH VARYING DEGREES OF NONCIRCULARITY

Algorithms	\mathbb{H} -Circular	\mathbb{C}^I -Circular	\mathbb{C}^J -Circular	\mathbb{C}^K -Circular	Noncircular
AQNGD (dB)	20.22	20.93	20.91	20.88	21.58
QNGD (dB)	19.46	20.04	19.99	20.01	20.45
AASQAFA (dB)	18.09	15.75	15.35	15.66	17.01
QMLP-FIR (dB)	16.58	18.11	18.11	18.05	18.04

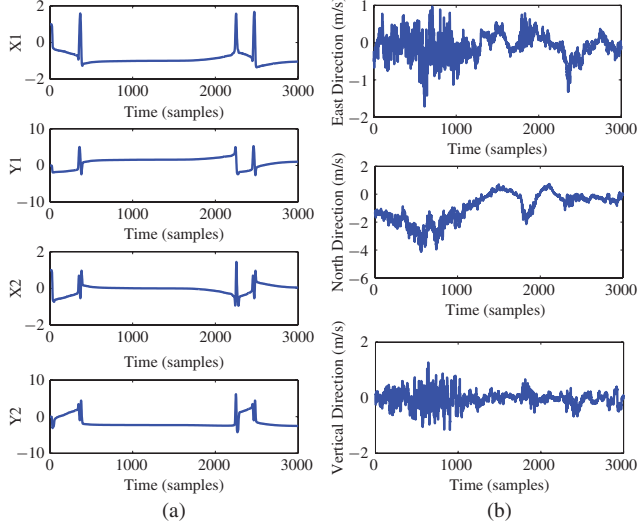


Fig. 4. Noncircular signals used in simulations. (a) 4-D Saito signal. (b) 3-D wind signal.

it was designed to cater for any noncircular AR type of processes.

Table II compares prediction gains R_p of the AQNGD, QNGD, AASQAFA, and QMLP-FIR for the prediction of linear AR (4) process with varying classes of input circularity, with $\mu = 10^{-2}$. The prediction gain was obtained from an average of 100 Monte Carlo trials. In all the cases, the proposed algorithms, i.e., AQNGD and QNGD, had superior performance over the AASQAFA and QMLP-FIR, illustrating the power of the fully quaternion function over the “split” quaternion function. Also from Table II, the use of the quaternion widely linear model for noncircular data is fully justified, as indicated by a higher prediction gain of AQNGD over the QNGD for noncircular sources.

B. 4-D Saito's Chaotic Circuit

The four state variables and five parameters that govern the noncircular Saito's chaotic process are given by [39]

$$\begin{aligned} \begin{bmatrix} \frac{\partial x_1}{\partial \tau} \\ \frac{\partial y_1}{\partial \tau} \end{bmatrix} &= \begin{bmatrix} -1 & 1 \\ -\alpha_1 & -\alpha_1 \beta_1 \end{bmatrix} \begin{bmatrix} x_1 - \eta \rho_1 h(z) \\ y_1 - \eta \frac{\rho_1}{\beta_1} h(z) \end{bmatrix} \\ \begin{bmatrix} \frac{\partial x_2}{\partial \tau} \\ \frac{\partial y_2}{\partial \tau} \end{bmatrix} &= \begin{bmatrix} -1 & 1 \\ -\alpha_2 & -\alpha_2 \beta_2 \end{bmatrix} \begin{bmatrix} x_2 - \eta \rho_2 h(z) \\ y_2 - \eta \frac{\rho_2}{\beta_2} h(z) \end{bmatrix} \end{aligned} \quad (57)$$

where τ is the time constant of the chaotic circuit and $h(z)$ is the normalized hysteresis value given by [39]

$$h(z) = \begin{cases} 1, & z \geq -1 \\ -1, & z \leq 1. \end{cases} \quad (58)$$

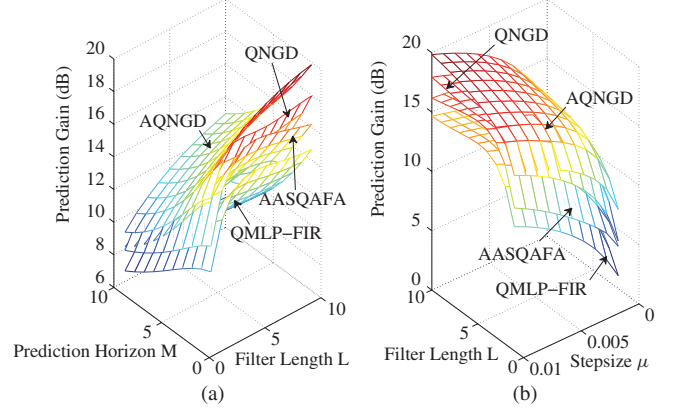


Fig. 5. Performance of AQNGD, QNGD, AASQAFA, and QMLP-FIR on the prediction of the noncircular 4-D Saito signal. (a) Dependence of the prediction gain on M and L. (b) Dependence of the prediction gain on μ on L.

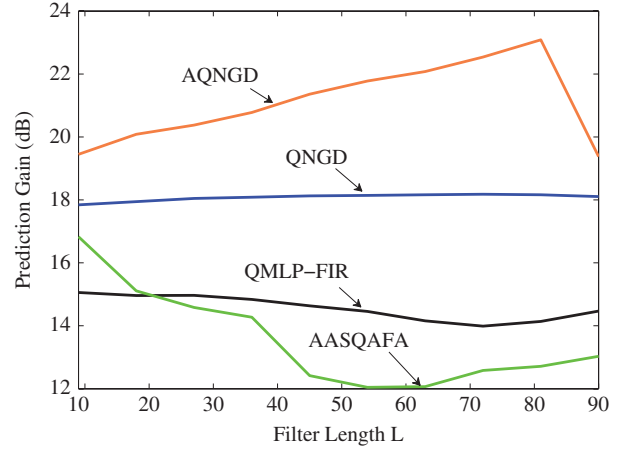


Fig. 6. Performance of AQNGD, QNGD, AASQAFA, and QMLP-FIR on the prediction of the noncircular 4-D Saito signal over a range of filter lengths.

The parameters z , ρ_1 , and ρ_2 are given as $z = x_1 + x_2$, $\rho_1 = (\beta_1/1 - \beta_1)$, and $\rho_2 = (\beta_2/1 - \beta_2)$. The Saito chaotic signal was initialized with the following parameters: $\eta = 1.3$, $\alpha_1 = 7.5$, $\alpha_2 = 15$, $\beta_1 = 0.16$, and $\beta_2 = 0.097$, and is noncircular, as shown dimensionwise in Fig. 4(a). Fig. 5 depicts the performances of the algorithms considered in terms of prediction horizon M (with fixed step size $\mu = 10^{-2}$) and step size μ (with fixed prediction horizon M = 1). Observe that the AQNGD outperformed all the other algorithms by a margin greater than 2 dB. For all the cases, increasing the step size led to a higher prediction gain provided that the upper bounds of QNGD in (52) and AQNGD in (53) were satisfied.

Fig. 6 illustrates the dependence of the prediction gain on filter length L for all algorithms with a fixed prediction horizon

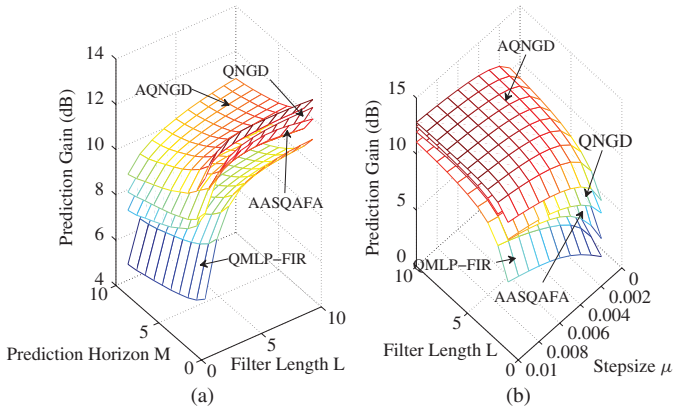


Fig. 7. Performance of AQNGD, QNGD, AASQAFA, and QMLP-FIR on the prediction of a 3-D wind signal. (a) Dependence of the prediction gain on M and L. (b) Dependence of the prediction gain on μ on L.

$M = 1$ and step size $\mu = 10^{-2}$. Observe that the prediction gain for the AQNGD was the largest, followed closely by the QNGD. However, increasing the filter length above $L = 80$ taps would lead to a significant performance degradation of the AQNGD, whereas the performance of the QNGD remains almost constant for higher filter length L . This is because increasing the filter length would proportionally increase the value of the term $\mathbf{x}^{aT}(n)\mathbf{x}^{a*}(n)$, which controls the maximum allowable μ , thus violating the upper bound of μ for AQNGD, specified in (53). However, this value is still within the upper bound of μ for QNGD given in (52).

C. Wind Forecasting

In this set of simulations, a single realization of 3-D wind field was used as the input.⁷ Fig. 4(b) shows the wind field signal dimensionwise, and Fig. 7 illustrates the performances of AQNGD, QNGD, AASQAFA, and QMLP-FIR as a function of prediction horizon M and step size μ . The performance of AQNGD was better than that of QNGD, this was closely followed by AASQAFA, whereas the performance of the QMLP-FIR was the poorest.

Fig. 8 shows a comparison of the proposed QNGD with the existing QMLP and three real-valued NGD as a function of prediction horizon M with a fixed step size $\mu = 10^{-2}$. From Fig. 8, observe that the QNGD outperformed the other algorithms considered. Also, observe that QMLP prediction gain was almost constant with the increase of the prediction horizon due to the structural richness of the feedforward multilayer neural network, which conforms to our earlier studies in [11].

VII. DISCUSSION

The performances of the filters that use the proposed locally analytic fully quaternion activation functions were generally better than those of the existing AASQAFA and QMLP-FIR. The widely linear version outperformed the QNGD as a result of the implementation of the quaternion widely linear model

⁷The wind data were sampled at 32 Hz and recorded by the 3-D WindMaster anemometer provided by Gill Instruments.

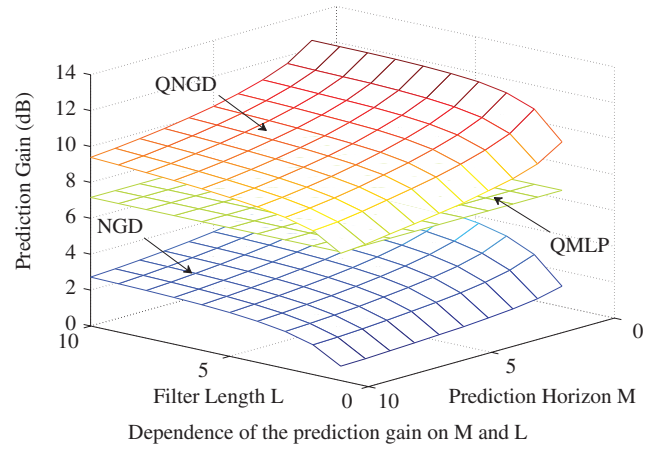


Fig. 8. Performance of QNGD, QMLP, and NGD on the prediction of a 3-D wind signal.

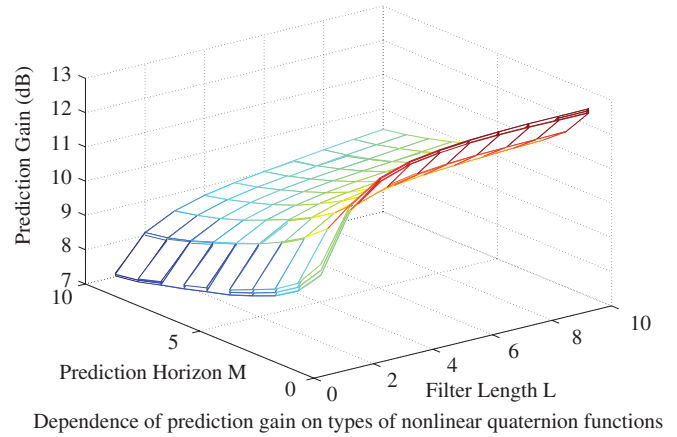


Fig. 9. Prediction gains of QNGD for $\tan(q)$, $\sin(q)$, $\arctan(q)$, $\arcsin(q)$, $\sinh(q)$, $\arctanh(q)$, and $\operatorname{arcsinh}(q)$ for the prediction of 3-D wind signal.

that fully captures the second-order statistics of quaternion signals. In order to create a class of fully quaternion function that is suitable for quaternion-valued adaptive filtering, it is essential to examine the possibility of employing other fully complex transcendental functions [12] as locally analytic fully quaternion functions. In Section IV, we have established that the exponential function e^q is locally analytic and, given that summations and products of analytic functions are analytic as well as quotients (provided the denominator does not vanish), the $\tanh(q)$ function is also locally analytic because it can be expressed in terms of e^q as

$$\tanh(q) = \frac{\sinh(q)}{\cosh(q)} = \frac{e^q - e^{-q}}{e^q + e^{-q}} = \frac{e^{2q} - 1}{e^{2q} + 1}. \quad (59)$$

This was verified by a rigorous derivation given in Appendix B. By continuity, the other quaternion transcendental functions are also locally analytic. In the complex domain, it has been shown in [40] that these performances based on a set of fully analytic transcendental functions were similar. In the same spirit, Fig. 9 confirms by simulations that the other elementary transcendental functions give similar performance to that of the locally analytic function $\tanh(q)$. We have therefore shown that the fully complex transcendental activation

TABLE III

COMPUTATIONAL COMPLEXITIES OF THE ALGORITHMS CONSIDERED

Algorithms	Multiplications	Additions
1 × QMLP-FIR	36L+20	28L+15
1 × AASQAFA	68L+36	54L+19
1 × QMLP	108L+216	96L+168
3 × NGD	9L+3	6L+3
1 × QNGD	68L+36	54L+24
1 × AQNGD	272L+144	208L+38

functions from \mathbb{C} can be extended to fully quaternion functions in \mathbb{H} , this is consistent with the observations in [40].

For convenience, the class of locally analytic fully quaternion functions and their derivatives are given below

$$\tanh(q) : \frac{\partial \tanh(q)}{\partial q} = \text{sech}^2(q) \quad (60)$$

$$\tan(q) : \frac{\partial \tan(q)}{\partial q} = \sec^2(q) \quad (61)$$

$$\sin(q) : \frac{\partial \sin(q)}{\partial q} = \cos(q) \quad (62)$$

$$\arctan(q) : \frac{\partial \arctan(q)}{\partial q} = (1 + q^2)^{-1} \quad (63)$$

$$\arcsin(q) : \frac{\partial \arcsin(q)}{\partial q} = (1 - q^2)^{-\frac{1}{2}} \quad (64)$$

$$\sinh(q) : \frac{\partial \sinh(q)}{\partial q} = \cosh(q) \quad (65)$$

$$\text{arctanh}(q) : \frac{\partial \text{arctanh}(q)}{\partial q} = (1 - q^2)^{-1} \quad (66)$$

$$\text{arcsinh}(q) : \frac{\partial \text{arcsinh}(q)}{\partial q} = (1 + q^2)^{-1}. \quad (67)$$

Another factor to consider is the computational complexity of the algorithms, which is summarized in Table III. The computational complexity of the AASQAFA and QNGD is $\mathcal{O}(68L)$, the NGD has the lowest computational complexity of $\mathcal{O}(9L)$ and the AQNGD has the highest computational complexity of $\mathcal{O}(272L)$. Computational complexities of the QMLP-FIR is $\mathcal{O}(36L)$ and for the QMLP it is $\mathcal{O}(108L)$. The QNGD algorithm thus represents an improvement from our previous proposed algorithm AASQAFA [11] in terms of performance and simplicity, while maintaining similar computational complexity.

In summary, the advantages of proposed class of QNGD and AQNGD algorithms based on fully quaternion locally analytic nonlinearities are as follows.

- 1) The performances of algorithms based on fully quaternion locally analytic functions, QNGD and AQNGD, were superior to those based on the split quaternion functions AASQAFA and QMLP-FIR, as the fully quaternion nonlinearities (60)–(67) operate directly in the quaternion domain instead of the channelwise processing in \mathbb{R} .
- 2) The widely linear model (16) enables the AQNGD to fully capture the quaternion second-order statistics suitable for noncircular signals (improper), and hence offers a further performance enhancement over the

standard linear model employed in QNGD, AASQAFA, and QMLP-FIR.

- 3) The fully quaternion-based QNGD is a reasonable choice as it allows for a tradeoff between performance and computational complexity.

VIII. CONCLUSION

A class of quaternion-valued nonlinear functions suitable for stochastic gradient-based training of quaternion-valued nonlinear adaptive filters has been proposed. The existing learning algorithms either completely neglect the non-commutativity aspect of quaternion, thus proving inadequate for the modeling of 3- and 4-D processes, or are unable to provide an accurate estimate due to the use of the suboptimal split-quaternion function that applies real nonlinearities componentwise. A class of fully quaternion activation functions has been derived according to the local analyticity condition, which enables the extension of fully complex nonlinear activation functions to the quaternion domain. The proposed fully quaternion algorithms (QNGD and AQNGD) have been shown to exhibit excellent performance on the prediction of 4-D synthetic and 3-D real-world vector signals. The widely linear AQNGD has been shown to achieve enhanced performance due to the utilization of the quaternion widely linear model and the associated augmented quaternion statistics, which fully captures the second-order information within quaternion-valued signals and enables the processing of both second-order circular (proper) and noncircular (improper) processes. Simulations over a range of noncircular synthetic signals and real-world 3-D wind recordings illustrate the benefits of the proposed approach.

APPENDIX A

EULER FORM OF $\tanh(q)$

The function $\tanh(q)$ in terms of the Euler formula is given by

$$\begin{aligned} \tanh(q) &= \frac{e^{2q_a} \cos(2\alpha) - 1 + e^{2q_a} \sin(2\alpha)\hat{\zeta}}{e^{2q_a} \cos(2\alpha) + 1 + e^{2q_a} \sin(2\alpha)\hat{\zeta}} \\ &= \frac{e^{4q_a} (\cos^2(2\alpha) + \sin^2(2\alpha)) - 1 + 2e^{2q_a} \cos(2\alpha)\hat{\zeta}}{e^{4q_a} (\cos^2(2\alpha) + \sin^2(2\alpha)) + 1 + 2e^{2q_a} \cos(2\alpha)} \\ &= \frac{e^{4q_a} - 1 + 2e^{2q_a} \sin(2\alpha)\hat{\zeta}}{e^{4q_a} + 1 + 2e^{2q_a} \cos(2\alpha)}. \end{aligned} \quad (68)$$

APPENDIX B

LOCAL ANALYTICITY OF $\tanh(q)$

To examine the local analyticity of $\tanh(q)$, we first apply the quaternion local analyticity condition in (21)–(68) to show that

$$\begin{aligned} \frac{\partial \tanh(q)}{\partial q_a} &= - \left(\frac{q_b}{\alpha} \frac{\partial \tanh(q)}{\partial q_b} + \frac{q_c}{\alpha} \frac{\partial \tanh(q)}{\partial q_c} \right. \\ &\quad \left. + \frac{q_d}{\alpha} \frac{\partial \tanh(q)}{\partial q_d} \right) \left(\frac{q_{bt} + q_{cJ} + q_{dK}}{\alpha} \right). \end{aligned} \quad (69)$$

Similar to the case of quaternion exponential functions, we obtain the term $(\partial \tanh(q)/\partial q_a)$ by differentiating (68) with

respect to q_a , to give

$$\begin{aligned} \frac{\partial \tanh(q)}{\partial q_a} &= \frac{\partial}{\partial q_a} \left(\frac{e^{4q_a} - 1}{e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1} \right. \\ &\quad \left. + \frac{2e^{2q_a} \sin(2\alpha) \hat{\zeta}}{e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1} \right) \\ &= \frac{4e^{6q_a} \cos(2\alpha) + 8e^{4q_a} + 4e^{2q_a} \cos(2\alpha)}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} \\ &\quad + \frac{(4e^{2q_a} \sin(2\alpha) - 4e^{6q_a} \sin(2\alpha)) \hat{\zeta}}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2}. \end{aligned} \quad (70)$$

In order to determine the remaining terms in (69), define

$$u = 2e^{2q_a} \sin(2\alpha) \quad v = e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1. \quad (71)$$

We can then substitute u and v into (68) and expand $\hat{\zeta}$ according to (22) to yield

$$\begin{aligned} \tanh(q) &= \frac{e^{4q_a} - 1 + u \hat{\zeta}}{v} \\ &= \frac{e^{4q_a} - 1}{v} + \frac{uq_b I}{v\alpha} + \frac{uq_c J}{v\alpha} + \frac{uq_d \kappa}{v\alpha}. \end{aligned} \quad (72)$$

Proceeding in a manner similar to when determining the analyticity of e^q , the term $(\partial \tanh(q)/\partial q_b)$ is obtained by differentiating (69) with respect to q_b , resulting in

$$\begin{aligned} \frac{\partial \tanh(q)}{\partial q_b} &= \frac{\partial}{\partial q_b} \left(\frac{e^{4q_a} - 1}{v} + \frac{uq_b I}{v\alpha} + \frac{uq_c J}{v\alpha} + \frac{uq_d \kappa}{v\alpha} \right) \\ &= \frac{(e^{4q_a} - 1)(4e^{2q_a} q_b \sin(2\alpha))}{v^2} \\ &\quad + \left(\frac{v\alpha u + v4e^{2q_a} q_b^2 \cos(2\alpha) - \frac{uvq_b^2}{\alpha} + uq_b^2 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) I \\ &\quad + \left(\frac{v4e^{2q_a} q_b q_c \cos(2\alpha) - \frac{uvq_b q_c}{\alpha} + uq_b q_c 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) J \\ &\quad + \left(\frac{v4e^{2q_a} q_b q_d \cos(2\alpha) - \frac{uvq_b q_d}{\alpha} + uq_b q_d 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) \kappa. \end{aligned} \quad (73)$$

Noticing that u , v , and α are functions of the variables q_b , q_c , and q_d , the terms $(\partial \tanh(q)/\partial q_c)$ and $(\partial \tanh(q)/\partial q_d)$ become

$$\begin{aligned} \frac{\partial \tanh(q)}{\partial q_c} &= \frac{(e^{4q_a} - 1)(4e^{2q_a} q_c \sin(2\alpha))}{v^2} \\ &\quad + \left(\frac{v4e^{2q_a} q_b q_c \cos(2\alpha) - \frac{uvq_b q_c}{\alpha} + uq_b q_c 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) I \\ &\quad + \left(\frac{v\alpha u + v4e^{2q_a} q_c^2 \cos(2\alpha) - \frac{uvq_c^2}{\alpha} + uq_c^2 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) J \\ &\quad + \left(\frac{v4e^{2q_a} q_c q_d \cos(2\alpha) - \frac{uvq_c q_d}{\alpha} + uq_c q_d 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) \kappa \end{aligned} \quad (74)$$

$$\begin{aligned} \frac{\partial \tanh(q)}{\partial q_d} &= \frac{(e^{4q_a} - 1)(4e^{2q_a} q_d \sin(2\alpha))}{v^2} \\ &\quad + \left(\frac{v4e^{2q_a} q_b q_d \cos(2\alpha) - \frac{uvq_b q_d}{\alpha} + uq_b q_d 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) I \\ &\quad + \left(\frac{v4e^{2q_a} q_c q_d \cos(2\alpha) - \frac{uvq_c q_d}{\alpha} + uq_c q_d 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) J \\ &\quad + \left(\frac{v\alpha u + v4e^{2q_a} q_d^2 \cos(2\alpha) - \frac{uvq_d^2}{\alpha} + uq_d^2 4e^{2q_a} \sin(2\alpha)}{(v\alpha)^2} \right) \kappa. \end{aligned} \quad (75)$$

Replacing (73)–(75) to the right-hand of side of (69), expanding the terms u and v (71), and simplifying it further by employing $\sin^2(\alpha) + \cos^2(\alpha) = 1$ along with substituting $\hat{\zeta}$ and α in (22) give

$$\begin{aligned} & - \frac{\partial \tanh(q)}{\partial \alpha} \hat{\zeta} \\ &= \left(\frac{(4e^{6q_a} \sin(2\alpha) - 4e^{2q_a} \sin(2\alpha))}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} \right. \\ &\quad \left. + \frac{4e^{6q_a} \cos(2\alpha) + 8e^{4q_a} + 4e^{2q_a} \cos(2\alpha)}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} \hat{\zeta} \right) \left(-\hat{\zeta} \right) \\ &= \frac{4e^{6q_a} \cos(2\alpha) + 8e^{4q_a} + 4e^{2q_a} \cos(2\alpha)}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} \\ &\quad + \frac{4e^{2q_a} \sin(2\alpha) - 4e^{6q_a} \sin(2\alpha)}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} \hat{\zeta}. \end{aligned} \quad (76)$$

APPENDIX C

LOCAL DERIVATIVE OF $\tanh(q)$

We shall first expand (32) into its Euler formula to give

$$\begin{aligned} \operatorname{sech}(q) &= \frac{2}{e^{q_a} (\cos(\alpha) + \sin(\alpha) \hat{\zeta}) + e^{-q_a} (\cos(\alpha) - \sin(\alpha) \hat{\zeta})} \\ &= \frac{2e^{3q_a} (\cos(\alpha) - \sin(\alpha) \hat{\zeta}) + 2e^{q_a} (\cos(\alpha) + \sin(\alpha) \hat{\zeta})}{e^{4q_a} + 2e^{2q_a} (\cos^2(\alpha) - \sin^2(\alpha)) + 1} \end{aligned} \quad (77)$$

and apply the identity $\cos^2(\alpha) - \sin^2(\alpha) = \cos(2\alpha)$ to give

$$\begin{aligned} \operatorname{sech}(q) &= \frac{2e^{3q_a} (\cos(\alpha) - \sin(\alpha) \hat{\zeta}) + 2e^{q_a} (\cos(\alpha) + \sin(\alpha) \hat{\zeta})}{e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1}. \end{aligned} \quad (78)$$

Squaring (78) and substituting $2 \sin(\alpha) \cos(\alpha) = \sin(2\alpha)$ yield

$$\begin{aligned} \operatorname{sech}^2(q) &= \frac{4e^{6q_a} \cos(2\alpha) + 8e^{4q_a} + 4e^{2q_a} \cos(2\alpha)}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} \\ &\quad + \frac{-4e^{6q_a} \sin(2\alpha) + 4e^{2q_a} \sin(2\alpha)}{(e^{4q_a} + 2e^{2q_a} \cos(2\alpha) + 1)^2} \hat{\zeta}. \end{aligned} \quad (79)$$

APPENDIX D
DERIVATION OF $\nabla_{\mathbf{w}} y^*(n)$ AND $\nabla_{\mathbf{w}} y(n)$

The term $\mathbf{w}^T(n)\mathbf{x}(n)$ can be expanded as (due to space limitation, the time index “n” has been dropped)

$$\mathbf{w}^T(n)\mathbf{x}(n) = \begin{bmatrix} \mathbf{w}_a^T \mathbf{x}_a - \mathbf{w}_b^T \mathbf{x}_b - \mathbf{w}_c^T \mathbf{x}_c - \mathbf{w}_d^T \mathbf{x}_d \\ \mathbf{w}_a^T \mathbf{x}_b + \mathbf{w}_b^T \mathbf{x}_a + \mathbf{w}_c^T \mathbf{x}_d - \mathbf{w}_d^T \mathbf{x}_c \\ \mathbf{w}_a^T \mathbf{x}_c + \mathbf{w}_c^T \mathbf{x}_a + \mathbf{w}_d^T \mathbf{x}_b - \mathbf{w}_b^T \mathbf{x}_d \\ \mathbf{w}_a^T \mathbf{x}_d + \mathbf{w}_d^T \mathbf{x}_a + \mathbf{w}_b^T \mathbf{x}_c - \mathbf{w}_c^T \mathbf{x}_b \end{bmatrix}. \quad (80)$$

The gradient $\nabla_{\mathbf{w}} y(n)$ is defined as [36]

$$\begin{aligned} \nabla_{\mathbf{w}} y(n) &= \frac{\partial y(n)}{\partial \mathbf{w}^*} \\ &= \nabla_{\mathbf{w}_a} y(n) + \nabla_{\mathbf{w}_b} y(n)\iota + \nabla_{\mathbf{w}_c} y(n)J + \nabla_{\mathbf{w}_d} y(n)\kappa. \end{aligned} \quad (81)$$

The odd-symmetry property also applies to the fully quaternion function and is given by $\Phi'^*(\mathbf{w}^T(n)\mathbf{x}(n)) = \Phi'(\mathbf{x}^H(n)\mathbf{w}(n))$. The derivatives in (81) can be calculated from the expansions (80) while using the odd-symmetry property, resulting in

$$\begin{aligned} \nabla_{\mathbf{w}_a} y(n) &= \Phi'(\mathbf{w}^T(n)\mathbf{x}(n))(\mathbf{x}_a + \mathbf{x}_b\iota + \mathbf{x}_cJ + \mathbf{x}_d\kappa) \\ \nabla_{\mathbf{w}_b} y(n)\iota &= \Phi'(\mathbf{w}^T(n)\mathbf{x}(n))(-\mathbf{x}_a - \mathbf{x}_b\iota + \mathbf{x}_cJ + \mathbf{x}_d\kappa) \\ \nabla_{\mathbf{w}_c} y(n)J &= \Phi'(\mathbf{w}^T(n)\mathbf{x}(n))(-\mathbf{x}_a + \mathbf{x}_b\iota - \mathbf{x}_cJ + \mathbf{x}_d\kappa) \\ \nabla_{\mathbf{w}_d} y(n)\kappa &= \Phi'(\mathbf{w}^T(n)\mathbf{x}(n))(-\mathbf{x}_a + \mathbf{x}_b\iota + \mathbf{x}_cJ - \mathbf{x}_d\kappa) \end{aligned} \quad (82)$$

where the symbol $\Phi'(\cdot)$ denotes the derivative of the fully quaternion function. Finally, substituting (82) into (81) yields

$$\nabla_{\mathbf{w}} y(n) = -\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))2\mathbf{x}^*(n). \quad (83)$$

Similarly, for $\nabla_{\mathbf{w}} y^*(n)$ we obtain

$$\nabla_{\mathbf{w}} y^*(n) = \Phi'^*(\mathbf{w}^T(n)\mathbf{x}(n))4\mathbf{x}^*(n) \quad (84)$$

which is used in the derivation of QNGD and AQNGD. Similar gradient derivations hold for the weight vectors \mathbf{h} , \mathbf{u} , and \mathbf{v} .

APPENDIX E
CONVERGENCE OF QNGD AND AQNGD

The convergence criterion employed in this paper is given by

$$E\{\|\bar{\tilde{e}}(n)\|_2^2\} \leq E\{\|\tilde{e}(n)\|_2^2\} \quad (85)$$

where $\bar{\tilde{e}}$ and \tilde{e} are, respectively, the *a posteriori* and the *a priori* output error, given by

$$\begin{aligned} \bar{\tilde{e}}(n) &= d(n) - \Phi(\mathbf{w}^T(n+1)\mathbf{x}(n)) + \bar{\tilde{e}}(n) \\ \tilde{e}(n) &= d(n) - \Phi(\mathbf{w}^T(n)\mathbf{x}(n)) + \tilde{e}(n) \end{aligned} \quad (86)$$

where symbols $\bar{\tilde{e}}$ and \tilde{e} denote quaternion QWGN is defined as

$$\epsilon(n) = \epsilon_a(n) + \epsilon_b(n)\iota + \epsilon_c(n)J + \epsilon_d(n)\kappa \quad (87)$$

where ϵ_a , ϵ_b , ϵ_c , and ϵ_d are realisations of real-valued WGN, independent and identically distributed. The terms $\bar{\tilde{e}}$ and \tilde{e} in (86) can be related by the first-order Taylor series expansion as [41]

$$\|\bar{\tilde{e}}(n)\|_2^2 = \|\tilde{e}(n)\|_2^2 + \Delta \mathbf{w}^H(n) \frac{\partial \|\tilde{e}(n)\|_2^2}{\partial \mathbf{w}^*(n)} \quad (88)$$

where $(\partial \|\tilde{e}(n)\|_2^2)/(\partial \mathbf{w}^*(n))$ is effectively the error gradient of the cost function. We shall first evaluate the term $\|\tilde{e}(n)\|_2^2$ as

$$\begin{aligned} \|\tilde{e}(n)\|_2^2 &= \left(d(n) - y(n) + \tilde{e}(n) \right) \left(d^*(n) - y^*(n) + \tilde{e}^*(n) \right) \\ &= d(n)d^*(n) - d(n)y^*(n) + d(n)\tilde{e}^*(n) - y(n)d^*(n) \\ &\quad + y(n)y^*(n) - y(n)\tilde{e}^*(n) + \tilde{e}(n)d^*(n) - \tilde{e}(n)y^*(n) \\ &\quad + \tilde{e}(n)\tilde{e}^*(n). \end{aligned} \quad (89)$$

Then, the error gradient $(\partial \|\tilde{e}(n)\|_2^2)/(\partial \mathbf{w}^*(n))$ can be calculated as

$$\begin{aligned} \frac{\partial \|\tilde{e}(n)\|_2^2}{\partial \mathbf{w}^*(n)} &= -d(n)\nabla_{\mathbf{w}} y^*(n) - \nabla_{\mathbf{w}} y(n)d^*(n) + y(n)\nabla_{\mathbf{w}} y^*(n) \\ &\quad + \nabla_{\mathbf{w}} y(n)y^*(n) - \nabla_{\mathbf{w}} y(n)\tilde{e}^*(n) - \tilde{e}(n)\nabla_{\mathbf{w}} y^*(n) \\ &= \left(-d(n) + y(n) - \tilde{e}(n) \right) \nabla_{\mathbf{w}} y^*(n) \\ &\quad + \nabla_{\mathbf{w}} y(n) \left(-d^*(n) + y^*(n) - \tilde{e}^*(n) \right) \\ &= -\tilde{e}(n)\nabla_{\mathbf{w}} y^*(n) - \nabla_{\mathbf{w}} y(n)\tilde{e}^*(n) \\ &= -[4\tilde{e}(n)\Phi'(\mathbf{x}^H(n)\mathbf{w}^*(n))\mathbf{x}^*(n) \\ &\quad - 2\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))\mathbf{x}^*(n)\tilde{e}^*(n)]. \end{aligned} \quad (90)$$

The term $\Delta \mathbf{w}^H(n) = -\mu(\partial \|\tilde{e}(n)\|_2^2/\partial \mathbf{w}^*(n))^H$, where $(\partial \|\tilde{e}(n)\|_2^2/\partial \mathbf{w}^*(n))$ is given in (90), and can be calculated as

$$\begin{aligned} \Delta \mathbf{w}^H &= \mu [2\mathbf{x}^T(n)\Phi'^*(\mathbf{x}^H(n)\mathbf{w}^*(n))\tilde{e}^*(n) \\ &\quad - \tilde{e}(n)\mathbf{x}^T(n)\Phi'^*(\mathbf{w}^T(n)\mathbf{x}(n))]. \end{aligned} \quad (91)$$

Substitute (90)–(91) into the Taylor series expansion (88) and apply the expectation operators on both sides to yield

$$\begin{aligned} &E\left\{\|\bar{\tilde{e}}(n)\|_2^2\right\} \\ &= E\left\{\|\tilde{e}(n)\|_2^2 - \mu \left([2\mathbf{x}^T(n)\Phi'^*(\mathbf{x}^H(n)\mathbf{w}^*(n))\tilde{e}^*(n) \right. \right. \\ &\quad \left. \left. - \tilde{e}(n)\mathbf{x}^T(n)\Phi'^*(\mathbf{w}^T(n)\mathbf{x}(n))] [4\tilde{e}(n)\Phi'(\mathbf{x}^H(n)\mathbf{w}^*(n))\mathbf{x}^*(n) \right. \right. \\ &\quad \left. \left. - 2\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))\mathbf{x}^*(n)\tilde{e}^*(n)] \right)\right\}. \end{aligned} \quad (92)$$

Applying the assumptions of small μ and statistical independence between the $\tilde{e}(n)$ and $\mathbf{x}(n)$, followed by the factorization of the term $\|\tilde{e}(n)\|_2^2$, gives

$$\begin{aligned} &E\{\|\bar{\tilde{e}}(n)\|_2^2\} \\ &= E\left\{\|\tilde{e}(n)\|_2^2 \left[1 - 10\mu \mathbf{x}^T(n)\mathbf{x}^*(n)\|\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))\|_2^2 \right]\right\} \\ &= E\{\|\tilde{e}(n)\|_2^2\} \\ &\quad E\left\{\left[1 - 10\mu \mathbf{x}^T(n)\mathbf{x}^*(n)\|\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))\|_2^2 \right]\right\}. \end{aligned} \quad (93)$$

The two terms can be separated since they are independent of each other corresponding to the statistical independence between the $\tilde{e}(n)$ and $\mathbf{x}(n)$. Therefore, the condition for convergence in (85) is satisfied for

$$0 < 10\mu E\{\mathbf{x}^T(n)\mathbf{x}^*(n)\|\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))\|_2^2\} < 1. \quad (94)$$

Solving for μ , we obtain the range of the step size for QNGD to converge as

$$0 < \mu < \frac{1}{10E\{\mathbf{x}^T(n)\mathbf{x}^*(n)\|\Phi'(\mathbf{w}^T(n)\mathbf{x}(n))\|_2^2}}. \quad (95)$$

Similarly, the bounds on μ so that the AQNGD converges are given by

$$0 < \mu < \frac{1}{10E\{\mathbf{x}^{aT}(n)\mathbf{x}^{a*}(n)\|\Phi'(\text{net}(n))\|_2^2}}. \quad (96)$$

ACKNOWLEDGMENT

The authors would like to thank Gill Instruments, Hampshire, U.K., for kindly providing them with the 3-D Wind-Master anemometer.

REFERENCES

- [1] S. B. Choe and J. J. Faraway, "Modeling head and hand orientation during motion using quaternions," *J. Aeros.*, vol. 113, no. 1, pp. 186–192, 2004.
- [2] J. C. K. Chou, "Quaternion kinematic and dynamic differential equations," *IEEE Trans. Robot. Autom.*, vol. 8, no. 1, pp. 53–64, Feb. 1992.
- [3] D. Choukkroun, I. Y. Bar-Itzhack, and Y. Ohsman, "Novel quaternion Kalman filter," *IEEE Trans. Aeros. Electron. Syst.*, vol. 42, no. 1, pp. 174–190, Jan. 2006.
- [4] C. C. Took and D. P. Mandic, "The quaternion LMS algorithm for adaptive filtering of hypercomplex processes," *IEEE Trans. Signal Process.*, vol. 57, no. 4, pp. 1316–1327, Apr. 2009.
- [5] N. Le Bihan and S. Buchholz, "Quaternionic independent component analysis using hypercomplex nonlinearities," presented at the International Conference on Mathematics in Signal Processing, 2006.
- [6] J. Via, D. P. Palomar, L. Vielva, and I. Santamaria, "Quaternion ICA from second-order statistics," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1586–1600, Apr. 2011.
- [7] A. Sudbery, "Quaternionic analysis," *Math. Proc. Camb. Phil. Soc.*, vol. 85, no. 2, pp. 199–225, 1979.
- [8] P. Arena, L. Fortuna, G. Muscato, and M. G. Xibilia, *Neural Networks in Multidimensional Domains* (Lecture Notes in Control and Information Sciences), vol. 234. New York: Springer-Verlag, 1998.
- [9] T. Nitta, "A backpropagation algorithm for neural networks based on 3-D vector product," in *Proc. Int. Joint Conf. Neural Netw.*, vol. 1. Oct. 1993, pp. 589–592.
- [10] T. Nitta and H. de Garis, "A 3-D vector version of the back-propagation algorithm," in *Proc. Int. Joint Conf. Neural Netw.*, vol. 2. 1992, pp. 511–516.
- [11] B. C. Ujang, C. C. Took, and D. P. Mandic, "Split quaternion nonlinear adaptive filtering," *Neural Netw.*, vol. 23, no. 3, pp. 426–434, Apr. 2010.
- [12] T. Kim and T. Adali, "Approximation by fully complex multilayer perceptrons," *Neural Comput.*, vol. 15, no. 7, pp. 1641–1666, Jul. 2003.
- [13] D. P. Mandic and V. S. L. Goh, *Complex Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models*. New York: Wiley, 2009.
- [14] S. De Leo and P. Rotelli, "Quaternion analyticity," *Appl. Math. Lett.*, vol. 16, no. 7, pp. 1077–1081, 2003.
- [15] I. Aizenberg, N. N. Aizenberg, and J. P. L. Vandewalle, *Multi-Valued and Universal Binary Neurons*. New York: Springer-Verlag, 2000.
- [16] I. Aizenberg and C. Moraga, "Multilayer feedforward neural network based on multi-valued neurons (MLMVN) and a backpropagation learning algorithm," *Soft Comput.*, vol. 11, no. 2, pp. 169–183, Sep. 2006.
- [17] A. Hirose, *Complex-Valued Neural Networks: Theories and Applications*. Singapore: World Scientific, 2003.
- [18] A. Hirose and H. Onishi, "Proposal of relative-minimization learning for behavior stabilization of complex-valued recurrent neural networks," *Neurocomputing*, vol. 24, nos. 1–3, pp. 163–171, Feb. 1999.
- [19] S. L. Goh, M. Chen, D. H. Popovic, K. Aihara, D. Obradovic, and D. P. Mandic, "Complex-valued forecasting of wind profile," *Renew. Ener.*, vol. 31, no. 11, pp. 1733–1750, Sep. 2006.
- [20] D. P. Mandic, S. Javidi, S. L. Goh, A. Kuh, and K. Aihara, "Complex-valued prediction of wind profile using augmented complex statistics," *Renew. Ener.*, vol. 34, no. 1, pp. 196–210, Jan. 2009.
- [21] C. C. Took and D. P. Mandic, "Augmented second-order statistics of quaternion random signals," *Signal Process.*, vol. 91, no. 2, pp. 214–224, Feb. 2011.
- [22] J. Via, D. Ramirez, and I. Santamaria, "Properness and widely linear processing of quaternion random vectors," *IEEE Trans. Inf. Theory*, vol. 56, no. 7, pp. 3502–3515, Jul. 2010.
- [23] F. D. Neeser and J. L. Massey, "Proper complex random processes with applications to information theory," *IEEE Trans. Inf. Theory*, vol. 39, no. 4, pp. 1293–1302, Jul. 1993.
- [24] B. Picinbono, "On circularity," *IEEE Signal Process. Lett.*, vol. 42, no. 12, pp. 3473–3482, Dec. 1994.
- [25] N. N. Vakhania, "Random vectors with values in quaternion Hilbert spaces," *Theor. Probab. Appl.*, vol. 43, no. 1, pp. 99–115, 1999.
- [26] P. O. Amblard and N. Le Bihan, "On properness of quaternion valued random variables," in *Proc. Int. Conf. Math. Signal Process.*, 2004, pp. 23–26.
- [27] B. Picinbono and P. Chevalier, "Widely linear estimation with complex data," *IEEE Trans. Signal Process.*, vol. 43, no. 8, pp. 2030–2033, Aug. 1995.
- [28] C. C. Took and D. P. Mandic, "A quaternion widely linear adaptive filter," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4427–4431, Aug. 2010.
- [29] J. Via, D. P. Palomar, and L. Vielva, "Generalized likelihood ratios for testing the properness of quaternion Gaussian vectors," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1356–1370, Apr. 2011.
- [30] E. Stiefel, "On Cauchy–Riemann equations in higher dimensions," *J. Res. Nat. Bureau Stand.*, vol. 48, no. 5, pp. 395–398, May 1952.
- [31] R. E. S. Watson, "The generalized Cauchy–Riemann–Fueter equation and handedness," *Compl. Variab.*, vol. 48, no. 7, pp. 555–568, Jul. 2003.
- [32] C. A. Deavours, "The quaternion calculus," *Amer. Math. Monthly*, vol. 80, no. 9, pp. 995–1008, Nov. 1973.
- [33] C. You and D. Hong, "Nonlinear blind equalization schemes using complex-valued multilayer feedforward neural networks," *IEEE Trans. Neural Netw.*, vol. 9, no. 6, pp. 1442–1455, Nov. 1998.
- [34] F. F. Brackx, "The exponential function of a quaternion variable," *Appl. Anal.*, vol. 8, no. 3, pp. 265–276, 1979.
- [35] L. Shi, "Exploration in quaternion colour," M.S. thesis, Dept. Comput. Sci., Simon Fraser Univ., Burnaby, BC, Canada, 2005.
- [36] D. P. Mandic, C. Jahanchahi, and C. C. Took, "A quaternion gradient operator and its applications," *IEEE Signal Process. Lett.*, vol. 18, no. 1, pp. 47–50, Jan. 2011.
- [37] D. P. Mandic and J. A. Chambers, *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability*. New York: Wiley, 2001.
- [38] S. Buchholz and N. Le Bihan, "Polarized signal classification by complex and quaternionic multilayer perceptrons," *Int. J. Neural Syst.*, vol. 18, no. 2, pp. 75–85, 2008.
- [39] K. Mitsubori and T. Saito, "Torus doubling and hyperchaos in a five dimensional hysteresis circuit," in *Proc. IEEE Int. Symp. Circuit Syst.*, vol. 6. London, U.K., May–Jun. 1994, pp. 113–116.
- [40] W. Duch and N. Jankowski, "Survey of neural transfer functions," *Neural Comput. Surv.*, vol. 2, pp. 163–212, 1999.
- [41] E. Soria-Olivas, J. Calpe-Maravilla, J. F. Guerrero-Martinez, M. Martinez-Sober, and J. Espi-Lopez, "An easy demonstration of the optimum value of the adaptation constant in the LMS algorithm [FIR filter theory]," *IEEE Trans. Educ.*, vol. 41, no. 1, p. 81, Feb. 1998.



Bukhari Che Ujang (S'11) received the B.E. degree in electrical engineering from Multimedia University, Cyberjaya, Malaysia, in 2006. He is currently pursuing the Ph.D. degree in adaptive signal processing at Imperial College London, London, U.K., (funded by a joint scholarship from the Ministry of Higher Education Malaysia and the Universiti Putra Malaysia, Selangor, Malaysia).

His current research interests include adaptive filters, nonlinear signal processing, and quaternion-valued analysis.



Clive Cheong Took (S'06–M'07) received the B.S. degree in telecommunication engineering from King's College, London University, London, U.K. (where he was the top departmental graduate), in 2004, and the Ph.D. degree in blind signal processing from Cardiff University, Cardiff, U.K., in 2007.

He is currently a Research Associate at Imperial College London, London, U.K. His current research interests include adaptive, blind, and multidimensional signal processing.



Danilo P. Mandic (M'99–SM'03) is a Reader in signal processing at Imperial College London, London, U.K., where he has been working in the area of nonlinear adaptive signal processing and nonlinear dynamics. He has been a Guest Professor at Katholieke Universiteit Leuven, Leuven, Belgium, the Tokyo University of Agriculture and Technology, Tokyo, Japan, and Westminster University, London, and a Frontier Researcher in RIKEN, Tokyo. His publication record includes two research monographs titled *Recurrent Neural Networks for*

Prediction (West Sussex, U.K.: Wiley, August 2001) and *Complex Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models* (West Sussex, U.K.: Wiley, April 2009), an edited book titled *Signal Processing for Information Fusion* (New York: Springer, 2008), and more than 200 publications on signal and image processing.

He has been a member of the IEEE Technical Committee on Machine Learning for Signal Processing, an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS, and the *International Journal of Mathematical Modelling and Algorithms*. He has produced award winning papers and products from his collaboration with the industry. He is a member of the London Mathematical Society.