

An Augmented Extended Kalman Filter Algorithm for Complex-Valued Recurrent Neural Networks

Su Lee Goh

su.goh@imperial.ac.uk

Danilo P. Mandic

d.mandic@imperial.ac.uk

Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K.

An augmented complex-valued extended Kalman filter (ACEKF) algorithm for the class of nonlinear adaptive filters realized as fully connected recurrent neural networks is introduced. This is achieved based on some recent developments in the so-called augmented complex statistics and the use of general fully complex nonlinear activation functions within the neurons. This makes the ACEKF suitable for processing general complex-valued nonlinear and nonstationary signals and also bivariate signals with strong component correlations. Simulations on benchmark and real-world complex-valued signals support the approach.

1 Introduction ---

Recent progress in biomedicine, wireless and mobile communications, seismics, and sonar and radar signal processing has brought to the light new problems, where data models are often complex valued or have a higher-dimensional compact representation (Mandic & Chambers, 2001; Haykin, 1994). To process such signals, research has largely been directed toward extending the results from real-valued adaptive filters to those operating in the complex plane, \mathbb{C} . One such algorithm is the complex least mean square (CLMS) algorithm for linear finite impulse response (FIR) adaptive filters, introduced in 1975 (Widrow, McCool, & Ball, 1975). More recently, complex-valued learning algorithms have been introduced for training of neural networks (NNs) (Kim & Adali, 2001; Leung & Haykin, 1991; Treichler, Johnson, & Larimore, 1987; Goh & Mandic, 2004).

Fully connected recurrent neural networks (FCRNNs) have been employed as nonlinear adaptive filters¹ (Mandic & Chambers, 2001; Medsker

¹Nonlinear autoregressive (NAR) processes can be modeled using feedforward networks, whereas nonlinear autoregressive moving-average (NARMA) processes can be represented using RNNs.

& Jain, 2000) where for real-time applications, the real-time recurrent learning (RTRL) algorithm (Williams & Zipser, 1989) has been widely used to train FCRNNs. In the complex domain, a recently proposed fully complex real-time recurrent learning (CRTRL)² algorithm (Goh & Mandic, 2004; Goh, Popovic, & Mandic, 2004) has been applied to forecasting of the complex-valued wind field. These initial results were promising, but it was also realized that gradient-based learning may experience problems when processing signals with rich dynamics (for instance, intermittent signals).

Following the established approaches from the real domain \mathbb{R} , a possible solution to this problem may be based on Kalman filters (Puskorius & Feldkamp, 1994), which have been shown to exhibit superior performance in several applications, including state estimation for road navigation (Obradovic, Lenz, & Schupfner, 2004), parameter estimation for time-series modeling, and neural network training (Puskorius & Feldkamp, 1991; Julier & Uhlmann, 2004; Haykin, 2001). Kalman filtering (Kalman, 1960) is known to give the optimal solution to the linear gaussian sequential state estimation problem in the domain of second order statistics. However, for a nonlinear or nongaussian state tracking problem with possibly an infinite number of states, no such general optimal algorithm exists. Extensions of the class of Kalman filter algorithms that cater to this case are termed the extended Kalman filter (EKF). The EKF (Feldkamp & Puskorius, 1998; Grewal & Andrews, 2001) is based on a truncated Taylor series expansion, which linearizes the system model locally, and the subsequent use of a linear Kalman filter. The EKF algorithms have been used to train temporal neural networks (Obradovic, 1996; Baltersee & Chambers, 1998; Mandic, Baltersee, & Chambers, 1998), with promising results.

Recently, EKF training of NNs has been extended to the complex domain (Huang & Chen, 2000). Notice that in order to design an algorithm suitable for the complex domain, we need a precise mathematical model that describes the evolution of system parameters. Hence, extensions of learning algorithms to the complex domain are not trivial and often involve some constraints, for instance, a simplified model of both complex statistics and complex nonlinearities within neurons. This might be suboptimal for classes of signals with significant correlation between the real and imaginary parts, which affects both the choice of the complex activation function and complex statistics.

To tackle this problem, we first provide mathematical foundations for complex-valued second-order statistics and highlight the need to consider

² The CRTRL algorithm uses a fully complex activation function (AF) that is analytical and bounded almost everywhere in \mathbb{C} . In a split-complex AF, the real and imaginary components of the input signal x are separated and fed through the real-valued activation function $f_R(x) = f_I(x)$, $x \in \mathbb{R}$. A split-complex activation function is therefore given as $f(x) = f_R(\text{Re}(x)) + j f_I(\text{Im}(x))$, for example, $f(x) = \frac{1}{1+e^{-\beta(\text{Re}(x))}} + j \frac{1}{1+e^{-\beta(\text{Im}(x))}}$.

the so-called augmented statistics in the derivation of such learning algorithms. Next, both the augmented complex-valued Kalman filter (ACKF) and the the augmented extended complex-valued Kalman filter (ACEKF) algorithm are derived, whereby the recently developed CRTRL algorithm is used to compute the Jacobian matrix within the ACEKF. For rigor, this is achieved for a fully complex nonlinear activation function of a neuron. The potential of such an ACEKF for training of FCRNNs is analyzed, and a comparison with the CRTRL is provided. The analysis is comprehensive and is supported by simulation examples on benchmark complex-valued nonlinear and colored signals, together with simulations on complex-valued real-world radar and environmental measurements. Finally, possible directions for future research are highlighted.

2 Complex-Valued Second-Order Statistics

Before deriving the ACEKF algorithm for complex FCRNNs, we introduce some important notations from complex second-order statistics. The covariance matrix for two real-valued RVs \mathbf{x} and \mathbf{y} is defined as (Anton, 2003)

$$\mathbf{P}_{xy} = cov[\mathbf{x}, \mathbf{y}] = E \left[(\mathbf{x} - E[\mathbf{x}]) (\mathbf{y} - E[\mathbf{y}])^T \right], \quad (2.1)$$

where $E(\cdot)$ represents the statistical expectation operator. Similarly, the four covariance matrices of two complex-valued RVs, $\mathbf{x} = \mathbf{x}^r + j\mathbf{x}^i$ and $\mathbf{y} = \mathbf{y}^r + j\mathbf{y}^i$, are given by (Neeser & Massey, 1992)

$$\begin{aligned} \mathbf{P}_{\mathbf{x}^r \mathbf{y}^r} &= cov[\mathbf{x}^r, \mathbf{y}^r] & \mathbf{P}_{\mathbf{x}^r \mathbf{y}^i} &= cov[\mathbf{x}^r, \mathbf{y}^i] \\ \mathbf{P}_{\mathbf{x}^i \mathbf{y}^r} &= cov[\mathbf{x}^i, \mathbf{y}^r] & \mathbf{P}_{\mathbf{x}^i \mathbf{y}^i} &= cov[\mathbf{x}^i, \mathbf{y}^i], \end{aligned} \quad (2.2)$$

where $j = \sqrt{-1}$, $(\cdot)^T$ denotes the vector transpose operator, and superscripts $(\cdot)^r$ and $(\cdot)^i$ denote, respectively, the real and imaginary part of a complex number or complex vector. These four real-valued matrices are equivalent to the following two complex-valued matrices, given by (Neeser & Massey, 1992),

$$\begin{aligned} \mathbf{P}_{\mathbf{xy}} &= E \left[(\mathbf{x} - E[\mathbf{x}]) (\mathbf{y} - E[\mathbf{y}])^H \right] \\ \mathbf{P}_{\mathbf{xy}}^{\#} &= E \left[(\mathbf{x} - E[\mathbf{x}]) (\mathbf{y} - E[\mathbf{y}])^T \right] \end{aligned} \quad (2.3)$$

where

$$\mathbf{P}_{\mathbf{xy}} = \mathbf{P}_{\mathbf{x}^r \mathbf{y}^r} + \mathbf{P}_{\mathbf{x}^i \mathbf{y}^i} + j(\mathbf{P}_{\mathbf{x}^i \mathbf{y}^r} - \mathbf{P}_{\mathbf{x}^r \mathbf{y}^i})$$

$$\mathbf{P}_{\mathbf{x}\mathbf{y}}^{\xi} = \mathbf{P}_{\mathbf{x}'\mathbf{y}'} - \mathbf{P}_{\mathbf{x}^i\mathbf{y}^i} + j(\mathbf{P}_{\mathbf{x}^i\mathbf{y}'} + \mathbf{P}_{\mathbf{x}'\mathbf{y}^i}), \quad (2.4)$$

and the symbol $(\cdot)^H$ denotes the Hermitian transpose operator. We can solve for $\mathbf{P}_{\mathbf{x}'\mathbf{y}'}$, $\mathbf{P}_{\mathbf{x}^i\mathbf{y}^i}$, $\mathbf{P}_{\mathbf{x}^i\mathbf{y}'}$, and $\mathbf{P}_{\mathbf{x}'\mathbf{y}^i}$ to obtain

$$\begin{aligned} \mathbf{P}_{\mathbf{x}'\mathbf{y}'} &= \frac{1}{2} \Re(\mathbf{P}_{\mathbf{x}\mathbf{y}} + \mathbf{P}_{\mathbf{x}\mathbf{y}}^{\xi}) \\ \mathbf{P}_{\mathbf{x}^i\mathbf{y}^i} &= \frac{1}{2} \Re(\mathbf{P}_{\mathbf{x}\mathbf{y}} - \mathbf{P}_{\mathbf{x}\mathbf{y}}^{\xi}) \\ \mathbf{P}_{\mathbf{x}^i\mathbf{y}'} &= \frac{1}{2} \Im(\mathbf{P}_{\mathbf{x}\mathbf{y}} + \mathbf{P}_{\mathbf{x}\mathbf{y}}^{\xi}) \\ \mathbf{P}_{\mathbf{x}'\mathbf{y}^i} &= \frac{1}{2} \Im(-\mathbf{P}_{\mathbf{x}\mathbf{y}} + \mathbf{P}_{\mathbf{x}\mathbf{y}}^{\xi}), \end{aligned} \quad (2.5)$$

where symbols \Re and \Im denote, respectively, the real and imaginary part of a complex quantity. It is clear that the four real-valued covariance matrices, equation 2.5, are in a one-to-one relationship with the two complex-valued covariance matrices in equation 2.4. In the literature, nearly always only $\mathbf{P}_{\mathbf{x}\mathbf{y}}$ is considered and is referred to as the *covariance matrix*, whereas $\mathbf{P}_{\mathbf{x}\mathbf{y}}^{\xi}$ is termed the *pseudocovariance matrix*.

2.1 Augmented Covariance Matrix. It is often assumed that the theory of complex-valued random vectors (RVs) is no different from that of the real RVs, as long as the definition of the covariance matrix of an RV \mathbf{x} is changed from $E[\mathbf{x}\mathbf{x}^T]$ in the real case to $E[\mathbf{x}\mathbf{x}^H]$ in the complex case (Schreier & Scharf, 2003; Picinbono, 1996). This assumption, however, is not justified since the covariance matrix $E[\mathbf{x}\mathbf{x}^H]$ will not completely describe the second-order statistical behavior of \mathbf{x} .

For complex-valued gaussian random variables, we therefore need to consider both the variable \mathbf{x} and its complex conjugate \mathbf{x}^* in order to make full use of the available statistical information. This additional information is contained in the cross-moments, and to design mathematically well-founded complex learning algorithms, a study of these moments and their implications on learning is required (Neeser & Massey, 1992).

We therefore set out to derive a complex-valued Kalman filter that takes into account the augmented complex statistics and thus provides a general framework for nonlinear adaptive filtering in \mathbb{C} . To achieve this, instead of a complex RV \mathbf{x} , we consider an ‘‘augmented’’ $(2n \times 1)$ -dimensional vector $\mathbf{x}^a = [\mathbf{x}, \mathbf{x}^*]$. In addition, for ‘‘improper’’ (see appendix A for more detail) vectors, it is the augmented $(2n \times 2n)$ -dimensional covariance matrix $\mathbf{P}_{\mathbf{x}^a\mathbf{x}^a} = E[\mathbf{x}^a(\mathbf{x}^a)^T]$ (rather than the $(n \times n)$ -dimensional matrix $\mathbf{P}_{\mathbf{x}\mathbf{x}} = E[\mathbf{x}\mathbf{x}^H]$) that contains the complete second-order statistical information. Such an augmented covariance matrix is given by (Schreier & Scharf,

2003)

$$\mathbf{P}_{\mathbf{x}^a \mathbf{x}^a} = E \begin{bmatrix} \mathbf{x} \\ \mathbf{x}^* \end{bmatrix} [\mathbf{x}^T \ \mathbf{x}^H] = \begin{bmatrix} \mathbf{P}_{\mathbf{xx}} & \mathbf{P}_{\mathbf{xx}}^\xi \\ \mathbf{P}_{\mathbf{xx}}^{\xi*} & \mathbf{P}_{\mathbf{xx}}^* \end{bmatrix}. \tag{2.6}$$

Notice that the covariance matrix $\mathbf{P}_{\mathbf{x}^a \mathbf{x}^a}$ is invertible and thus positive definite. Besides $\mathbf{P}_{\mathbf{xx}}$ being positive semidefinite (PSD) and $\mathbf{P}_{\mathbf{xx}}^{\xi*}$ being symmetric, the Schur complement $\mathbf{P}_{\mathbf{xx}}^* - \mathbf{P}_{\mathbf{xx}}^{\xi*} \mathbf{P}_{\mathbf{xx}}^{-1} \mathbf{P}_{\mathbf{xx}}^\xi$ must be PSD to ensure that $\mathbf{P}_{\mathbf{x}^a \mathbf{x}^a}$ is PSD and thus a valid covariance matrix for \mathbf{x}^a (Neuser & Massey, 1992).

3 The Augmented Complex-Valued Kalman Filter Algorithm _____

Based on the analysis of augmented complex statistics, we first derive a Kalman filter learning algorithm for complex-valued inputs using the augmented states and augmented covariance matrix. Consider a general state-space model given by (Haykin, 2001),

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{F}_{k+1} \mathbf{x}_k + \boldsymbol{\omega}_k \\ \mathbf{y}_k &= \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k, \end{aligned} \tag{3.1}$$

where $\boldsymbol{\omega}_k$ and \mathbf{v}_k are independent, zero-mean, complex-valued gaussian processes of covariance matrices \mathbf{Q}_k and \mathbf{R}_k , respectively, and \mathbf{F} and \mathbf{H} are the transition and measurement matrix. From equation 3.1, the augmented state-space model is obtained as

$$\begin{aligned} \mathbf{x}_{k+1}^a &= \mathbf{F}_{k+1}^a \mathbf{x}_k^a + \boldsymbol{\omega}_k^a \\ \mathbf{y}_k^a &= \mathbf{H}_k^a \mathbf{x}_k^a + \mathbf{v}_k^a, \end{aligned} \tag{3.2}$$

where $\mathbf{x}_k^a = [\mathbf{x}_k, \mathbf{x}_k^*]$, $\mathbf{y}_k^a = [\mathbf{y}_k, \mathbf{y}_k^*]$, $\mathbf{F}_k^a = [\mathbf{F}_k, \mathbf{F}_k^*]$, $\mathbf{H}_k^a = [\mathbf{H}_k, \mathbf{H}_k^*]$, $\boldsymbol{\omega}_k^a = [\boldsymbol{\omega}_k, \boldsymbol{\omega}_k^*]$, and $\mathbf{v}_k^a = [\mathbf{v}_k, \mathbf{v}_k^*]$. The augmented covariance matrices of the zero-mean complex-valued gaussian noise processes are denoted respectively by \mathbf{Q}_k^a and \mathbf{R}_k^a .

In the initialization of the algorithm ($k = 0$), we set

$$\begin{aligned} \hat{\mathbf{x}}_0^a &= E [\mathbf{x}_0^a], \\ \mathbf{P}_0 &= E \left[(\mathbf{x}_0^a - E [\mathbf{x}_0^a]) (\mathbf{x}_0^a - E [\mathbf{x}_0^a])^T \right]. \end{aligned} \tag{3.3}$$

The computation steps for $k = 1, \dots, L$ are given below (for clarity, we use notation similar to that from (Haykin, 2001):

State estimate propagation:

$$\hat{\mathbf{x}}_k^{a-} = \mathbf{F}_{k,k-1}^a \hat{\mathbf{x}}_{k-1}^{a-} \quad (3.4)$$

Error covariance propagation:

$$\mathbf{P}_k^- = \mathbf{F}_{k,k-1}^a \mathbf{P}_{k-1} (\mathbf{F}_{k,k-1}^a)^H + \mathbf{Q}_{k-1}^a \quad (3.5)$$

Kalman gain matrix:

$$\mathbf{G}_k = \mathbf{P}_k^- (\mathbf{H}_k^a)^H \left[\mathbf{H}_k^a \mathbf{P}_k (\mathbf{H}_k^a)^H + \mathbf{R}_k^a \right]^{-1} \quad (3.6)$$

State estimate update:

$$\hat{\mathbf{x}}_k^a = \hat{\mathbf{x}}_k^{a-} + \mathbf{G}_k (\mathbf{y}_k^a - \mathbf{H}_k^a \hat{\mathbf{x}}_k^{a-}) \quad (3.7)$$

Error covariance update:

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{G}_k \mathbf{H}_k^a) \mathbf{P}_k^- \quad (3.8)$$

This completes the description of the augmented complex-valued Kalman filter.

4 FCRNN Trained with Augmented Extended Complex-Valued Kalman Filter

4.1 FCRNN Architecture. Figure 1 shows an FCRNN, which consists of N neurons with p external inputs and N feedback connections. The network has two distinct layers: the external input-feedback layer and a layer of processing elements. Let $y_{l,k}$ denote the complex-valued output of a neuron, $l = 1, \dots, N$ at time index k and \mathbf{s}_k the $(1 \times p)$ external complex-valued input vector. The overall input to the network \mathbf{u}_k then represents a concatenation of vectors \mathbf{y}_k , \mathbf{s}_k and the bias input $(1 + j)$, and is given by

$$\begin{aligned} \mathbf{u}_k &= [s_{k-1}, \dots, s_{k-p}, 1 + j, y_{1,k-1}, \dots, y_{N,k-1}]^T \\ u_{n,k} \in \mathbf{u}_k &= u_{n,k}^r + j u_{n,k}^i, \quad n = 1, \dots, p + N + 1. \end{aligned} \quad (4.1)$$

For the l th neuron, its weights form a $(p + N + 1) \times 1$ -dimensional weight vector $\mathbf{w}_l^T = [w_{l,1}, \dots, w_{l,p+N+1}]$, $l = 1, \dots, N$, which are encompassed in the complex-valued weight matrix of the network $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_N]$.

The output of every neuron can be expressed as

$$y_{l,k} = \Phi(ne t_{l,k}), \quad l = 1, \dots, N, \quad (4.2)$$

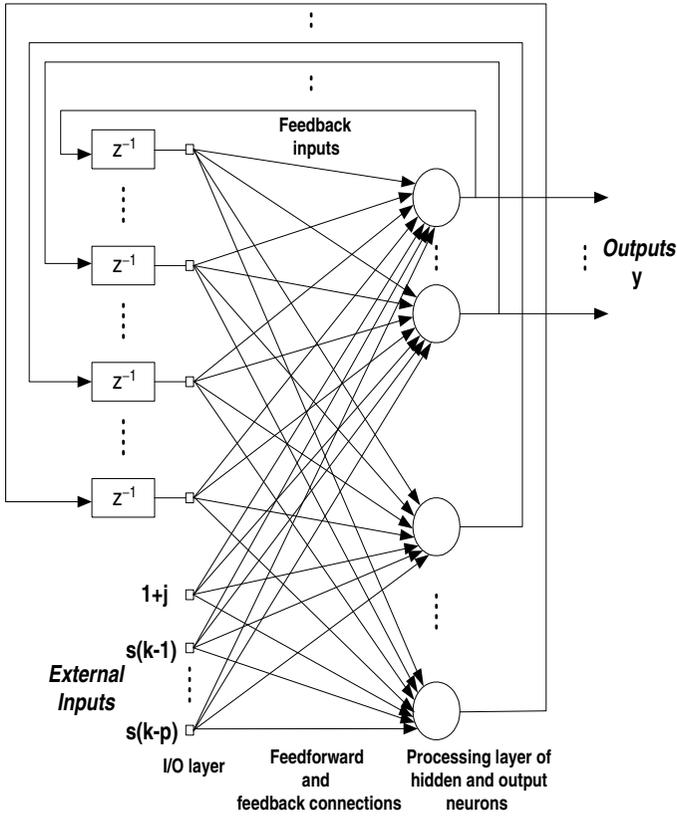


Figure 1: A fully connected recurrent neural network for prediction.

where Φ is a complex nonlinear activation function of a neuron and

$$net_{l,k} = \sum_{n=1}^{p+N+1} w_{l,n} u_{n,k} \tag{4.3}$$

is the net input to l th node at time index k .

In order to establish a mathematical framework for Kalman filter-based training of FCRNNs, the dynamical behavior of the FCRNN by a state-space model is given by

$$\mathbf{w}_{k+1}^a = \mathbf{w}_k^a + \boldsymbol{\omega}_k^a \tag{4.4}$$

$$\mathbf{y}_k^a = \mathbf{h}(\mathbf{w}_k^a, \mathbf{u}_k) + \mathbf{v}_k^a, \tag{4.5}$$

where \mathbf{h} is a nonlinear operator associated with observations, \mathbf{w}_k^a is the augmented weight vector, and \mathbf{y}_k^a is the augmented output of the network. The underlying idea here is to linearize the state-space model for every time instant k . Once such a local linear model is obtained, the standard ACKF approach can be applied. In practice, the process noise covariance \mathbf{Q}_k^a and measurement noise covariance \mathbf{R}_k^a matrices might be time varying, but here we assume they are constant. Notice that the Jacobian \mathbf{H}_k^a is defined as a set of partial derivatives of the outputs with respect to the weights, \mathbf{w}_k^a , and needs to be updated at every time instant during learning.

4.2 Derivation of the Augmented Complex-Valued Extended Kalman Filter Algorithm. When operating in the complex domain, similar to the real EKF, the nonlinear state and measurement equations ought to be linearized about the current state in order to subsequently employ a standard Kalman filter. Since this linearization is dynamical in its nature, the performance of such EKF cannot be assessed beforehand. Indeed, there is a chance that the updated trajectory estimate will be poorer than the nominal one, which leads to inaccuracy in the estimates, causing further error accumulation (Brown & Hwang, 1997). The derivation of the required complex-valued Jacobian matrix is nontrivial, and the linearization can lead to a highly unstable performance (Julier & Uhlmann, 2004). Therefore, careful parameter selection is required when using the ACEKF algorithm. With this in mind, we proceed with the derivation of the ACEKF as an extension from the ACKF presented in the previous section.

The following expressions summarize the proposed ACEKF:

$$\mathbf{G}_k = \mathbf{P}_k^- (\mathbf{H}_k^a)^H [\mathbf{H}_k^a \mathbf{P}_k^- (\mathbf{H}_k^a)^H + \mathbf{R}_k^a]^{-1} \quad (4.6)$$

$$\hat{\mathbf{w}}_k^a = \hat{\mathbf{w}}_k^{a-} + \mathbf{G}_k [\mathbf{y}_k^a - \mathbf{h}(\hat{\mathbf{w}}_k^{a-}, \mathbf{u}_k)] \quad (4.7)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{G}_k \mathbf{H}_k^a) \mathbf{P}_k^- + \mathbf{Q}_k^a. \quad (4.8)$$

The ACEKF is initialized by

$$\begin{aligned} \hat{\mathbf{w}}_0^a &= E[\mathbf{w}_0] \\ \mathbf{P}_0 &= E[(\mathbf{w}_0^a - E[\mathbf{w}_0^a]) (\mathbf{w}_0^a - E[\mathbf{w}_0^a])^T]. \end{aligned} \quad (4.9)$$

For generality, the augmented Jacobian matrix \mathbf{H}_k^a is computed using the augmented CRTL algorithm³ for recurrent networks (Goh & Mandic, 2004) (using fully complex nonlinearities within the network; see appendix B). The vector $\hat{\mathbf{w}}_k^a$ denotes the estimate of the augmented state of

³ Matrix \mathbf{H}_k^a denotes the matrix of partial derivatives of the network's augmented output \mathbf{y}_k^a with respect to weight parameters (see appendix B).

the system update at step k . The Kalman gain matrix \mathbf{G}_k is a function of the estimated error covariance matrix \mathbf{P}_k , the Jacobian matrix \mathbf{H}_k^a , and a global scaling matrix $\mathbf{H}_k^a \mathbf{P}_k^- (\mathbf{H}_k^a)^H + \mathbf{R}_k^a$.

5 Simulations

For the experiments, the nonlinearity at the neuron was chosen to be the complex tanh function,

$$\Phi(x) = \frac{e^{\beta x} - e^{-\beta x}}{e^{\beta x} + e^{-\beta x}}, \quad (5.1)$$

where $x \in \mathbb{C}$. The value of the slope of $\Phi(x)$ was $\beta = 1$. The architecture of the FCRNN (see Figure 1) consisted of $N = 3$ neurons with the tap input length of $p = 5$. To support the analysis, we tested the ACEKF on a wide range of signals, including complex linear and complex nonlinear signals. To further illustrate the approach and verify the advantage of using the ACEKF over CEKF and the CRTRL, additional single-trial experiments were performed on real-world complex-valued wind⁴ and radar⁵ data.

In the first experiment, the input signal was a stable complex linear $AR(4)$ process given by

$$r(k) = 1.79r(k-1) - 1.85r(k-2) + 1.27r(k-3) + 0.41r(k-4) + n(k), \quad (5.2)$$

with complex white gaussian noise (CWGN) $n(k) \sim \mathcal{N}(0,1)$ as the driving input. The CWGN can be expressed as $n(k) = n^r(k) + jn^i(k)$. The real and imaginary components of CWGN are mutually independent sequences having equal variances, so that $\sigma_n^2 = \sigma_{n^r}^2 + \sigma_{n^i}^2$. For the second experiment, we used a complex benchmark nonlinear signal (Narendra & Parthasarathy, 1990),

$$z(k) = \frac{z^2(k-1)(z(k-1) + 2.5)}{1 + z(k-1) + z^2(k-2)} + r(k-1). \quad (5.3)$$

⁴ Publicly available online from <http://mesonet.agron.iastate.edu/>. The wind vector can be expressed in the complex domain \mathbb{C} as $v(t)e^{j\theta(t)} = v_E(t) + jv_N(t)$. Here, the two wind components, the speed v and direction θ , which are of different natures, are modeled as a single quantity in a complex representation space (Mandic and Goh, 2005).

⁵ Publicly available online from <http://soma.ece.mcmaster.ca/ipix/>.

Table 1: Comparison of Prediction Gains R_p , MSE, and SNR for the Various Classes of Signals.

Signal	Nonlinear	AR4	Wind	Radar
R_p [dB] (ACEKF)	6.24	4.77	12.65	10.58
R_p [dB] (standard CEKF)	5.55	3.98	10.24	9.91
R_p [dB] (CRTRL)	3.76	3.54	6.12	7.22
SNR [dB]	5	5	3	3

To assess the performance, standard prediction gain R_p was employed and is given by (Haykin & Li, 1995),

$$R_p(k) \triangleq 10 \log_{10} \left(\frac{\sigma_x^2}{\hat{\sigma}_e^2} \right) [dB], \quad (5.4)$$

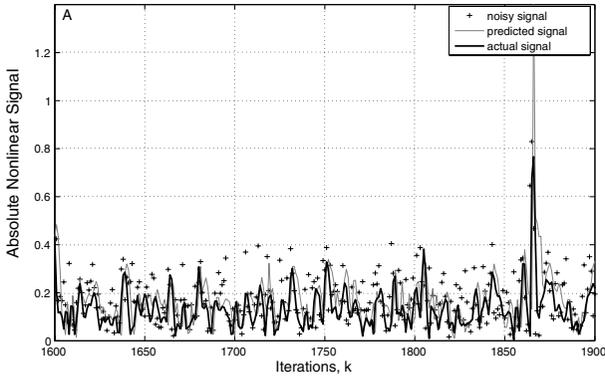
where σ_x^2 denotes the variance of the input signal $x(k)$, and $\hat{\sigma}_e^2$ denotes the estimated variance of the forward prediction error $e(k)$.

Table 1 shows a comparison of the prediction gains R_p [dB] between the ACEKF, standard complex-valued (CEKF) (without considering the augmented states), and CRTRL for various classes of signals, and also the signal-to-noise ratio (SNR) for various experiments. In all the cases, there was a significant improvement in the performance when ACEKF was employed over that of CRTRL and CEKF. Figures 2 and 3 show a subsegment of the predictions generated by the ACEKF and CRTRL for complex-valued colored (see equation 5.2) and nonlinear (see equation 5.3) signals. The ACEKF outperformed CRTRL for both cases.

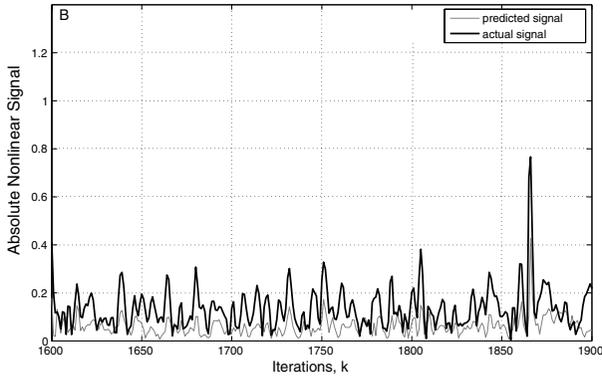
To further illustrate the advantage of ACEKF over CRTRL, we compared the performances of FCRNNs trained with these algorithms in experiments on real-world radar and wind data. Figure 4 shows the prediction performance of the ACEKF applied to the complex-valued real-world (velocity and angle components) wind signal. Simulation results for radar data are shown in Figure 5. In both cases, the AECKF outperformed the other algorithms considered (for a quantitative performance comparison, see Table 1).

6 Discussion and Conclusion

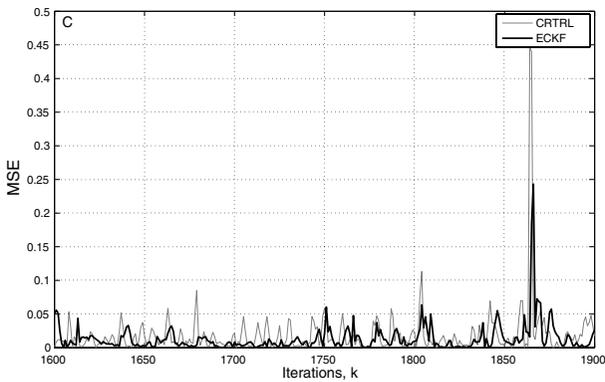
The augmented complex-valued extended Kalman filter (ACEKF) has been introduced for nonlinear adaptive filtering in the complex domain. For generality, this has been achieved for nonlinear adaptive filters realized as fully connected recurrent neural networks (FCRNNs), and the ACEKF has been derived using the augmented complex-valued statistics, whereby



(a) ACEKF Algorithm

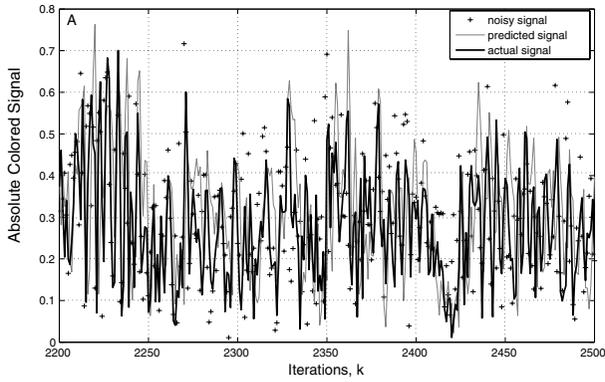


(b) CRTRL Algorithm

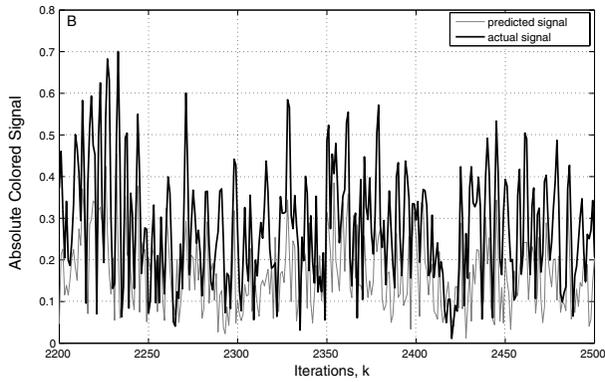


(c) Mean Square Error

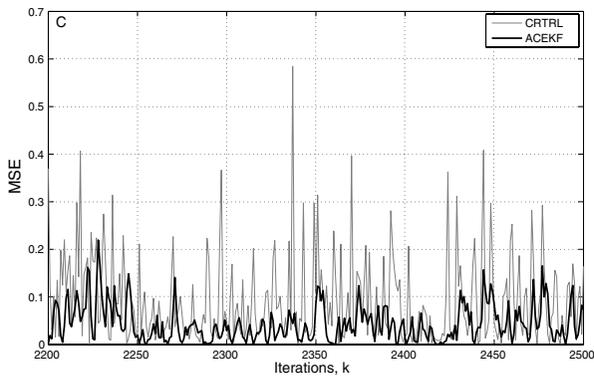
Figure 2: Comparison of one-step-ahead prediction performance between the ACEKF and CRTRL for nonlinear signal (see equation 5.3).



(a) ACEKF Algorithm



(b) CRTRL Algorithm



(c) Mean Square Error

Figure 3: Comparison of one-step-ahead prediction performance between the ACEKF and CRTRL for colored signal (see equation 5.2).

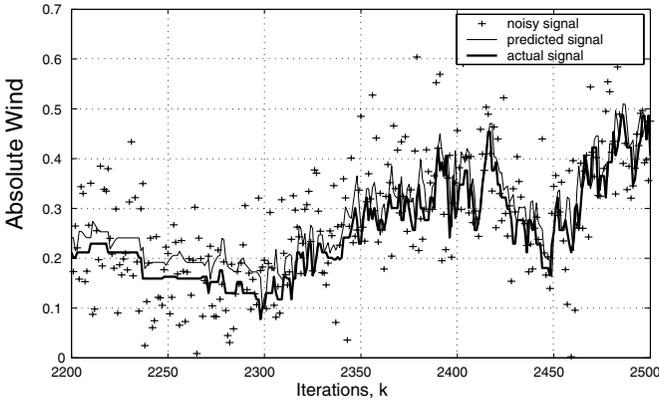


Figure 4: Prediction performance for complex wind signal using ACEKF.

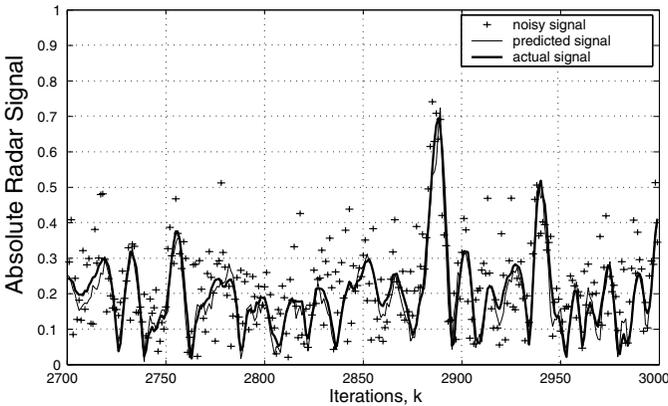


Figure 5: Prediction performance for complex radar signal using ACEKF.

a complete second-order statistics for complex-valued quantities is taken into account. The performance of the ACEKF has been evaluated on benchmark complex-valued nonlinear and colored signals and also on real-life complex-valued wind and radar signals. Experimental results have justified the potential of ACEKF in nonlinear complex-valued neural adaptive filtering applications. It should be noted that the computational cost of the ACEKF is relatively more expensive than that of the CRTRL. To mitigate this problem, a decoupled extended Kalman filter (DEKF) algorithm (Puskorius et al. 1991) may be used; however, the reduction in computational complexity may lead to poorer performance of the algorithm in the domain of augmented complex statistics. This is due to the fact that the DEKF is derived from EKF by approximating the second-order information between

weights so that they belong only to mutually exclusive groups; correlations between weights estimates can be neglected, which leads to a sparser error covariance matrix \mathbf{P}_k . The decoupling methods for processes described by augmented complex statistics are an area for future research, especially if the main concern is the computational complexity of augmented EKF.

Appendix A: Proper and Improper Complex Random Vectors _____

A complex RV \mathbf{x} is called proper if its pseudocovariance $\mathbf{P}_{\mathbf{xx}}^{\xi}$ vanishes (Neeser & Massey, 1992; Schreier & Scharf, 2003). For a proper complex-valued random vector \mathbf{x} , we have

$$\mathbf{P}_{\mathbf{x}^r \mathbf{x}^r} = \mathbf{P}_{\mathbf{x}^i \mathbf{x}^i}, \quad \mathbf{P}_{\mathbf{x}^i \mathbf{x}^r} = -\mathbf{P}_{\mathbf{x}^r \mathbf{x}^i}^T. \quad (\text{A.1})$$

This means that the real and imaginary parts of \mathbf{x} are uncorrelated. The vanishing of $\mathbf{P}_{\mathbf{xx}}^{\xi}$ does not imply that the real part of $x_m \in \mathbf{x}$ and the imaginary part of $x_n \in \mathbf{x}$ are uncorrelated for $m \neq n$. When RVs are proper, the probability density function (PDF) of a gaussian RV takes a form similar to that from the real case. Let $\mathbf{x} \in \mathbb{C}^n$ be a proper complex-valued gaussian random variable with mean $\boldsymbol{\mu}$ and nonsingular covariance matrix $\mathbf{P}_{\mathbf{xx}}$. Then the PDF of \mathbf{x} is given by (Picinbono, 1998)

$$f(\mathbf{x}) = \frac{1}{\det(\pi \mathbf{P}_{\mathbf{xx}})} e^{-(\mathbf{x}-\boldsymbol{\mu})^T \mathbf{P}_{\mathbf{xx}}^{-1} (\mathbf{x}-\boldsymbol{\mu})}, \quad (\text{A.2})$$

where $\mathbf{P}_{\mathbf{xx}}^{-1}$ is some Hermitian and positive-definite matrix. For convenience, in many applications, complex-valued RVs are treated as proper. However, $\mathbf{P}_{\mathbf{xx}}^{\xi}$ may not be necessarily zero, and the results obtained this way are suboptimal.

Appendix B: An Augmented CRTRL Algorithm for the FCRNN _____

The augmented CRTRL algorithm for training the FCRNN is briefly described. The cost function of the recurrent network is given by $J_k = \frac{1}{2} \mathbf{e}_k \mathbf{e}_k^T$. The instantaneous output error is defined as

$$\mathbf{e}_k = \mathbf{d}_k^a - \mathbf{y}_k^a, \quad (\text{B.1})$$

where \mathbf{d}_k^a denotes the augmented desired output vector. The augmented weight matrix update becomes

$$\Delta \mathbf{w}_k^a = -\eta \frac{\partial J_k}{\partial \mathbf{w}_k^a} = -\eta \mathbf{e}_k \frac{\partial \mathbf{y}_k^a}{\partial \mathbf{w}_k^a}. \quad (\text{B.2})$$

We introduce three new matrices—the $2N \times (2N + p + 1)$ matrix $\Pi_{l,k}^a$, the $2N \times (2N + p + 1)$ matrix $\mathbf{U}_{l,k}$, and the $2N \times 2N$ diagonal matrix \mathbf{S}_k —as (Haykin, 1994)

$$\Pi_{l,k}^a = \frac{\partial \mathbf{y}_k^a}{\partial \mathbf{w}_k^a}, \quad \mathbf{y}^a = [y_{1,k}, \dots, y_{N,k}, y_{1,k}^*, \dots, y_{N,k}^*], \quad l = 1, 2, \dots, N \tag{B.3}$$

$$\mathbf{U}_{l,k} = \begin{bmatrix} 0 \\ \vdots \\ \mathbf{u}_k \\ \vdots \\ 0 \end{bmatrix} \leftarrow l\text{th row}, \quad l = 1, \dots, N \tag{B.4}$$

$$\mathbf{S}(k) = \text{diag} (\Phi' (\mathbf{u}_k^T \mathbf{w}_{1,k}^a), \dots, \Phi' (\mathbf{u}_k^T \mathbf{w}_{N,k}^a)) \tag{B.5}$$

$$\Pi_{l,k+1}^a = \mathbf{S}_k^* \left[\mathbf{U}_{l,k}^* + (\mathbf{W}_{f,k}^a)^H \Pi_{l,k}^a \right], \tag{B.6}$$

where \mathbf{W}_f denotes the set of those entries in \mathbf{W} that correspond to the feedback connections.

Acknowledgments _____

We thank Dragan Obradovic from Siemens Corporate Research for his critical comments and lengthy discussions.

References _____

Anton, H. (2003). *Contemporary linear algebra*. New York: Wiley.

Baltersee, J., & Chambers, J. A. (1998). Nonlinear adaptive prediction of speech signals using a pipelined recurrent network. *IEEE Transactions on Signal Processing*, 46(8), 2207–2216.

Brown, R. B., & Hwang, P. Y. C. (1997). *Introduction to random signals and applied Kalman filtering*. New York: Wiley.

Feldkamp, L. A., & Puskorius, G. V. (1998). A signal processing framework based on dynamical neural networks with application to problems in adaptation, filtering and classification. *IEEE Transactions on Neural Networks*, 86, 2259–2277.

Goh, S. L., & Mandic, D. P. (2004). A complex-valued RTRL algorithm for recurrent neural networks. *Neural Computation*, 16(12), 2699–2713.

Goh, S. L., Popovic, D. H., & Mandic, D. P. (2004). Complex-valued estimation of wind profile and wind power. In *Proceedings of the 12th IEEE Mediterranean Electrotechnical Conference, MELECON 2004* (Vol. 3, pp. 1037–1040). Piscataway, NJ: IEEE Press.

- Grewal, M. S., & Andrews, A. P. (2001). *Kalman filtering: Theory and practice*. New York: Wiley.
- Haykin, S. (1994). *Neural networks: A comprehensive foundation*. Upper Saddle River, NJ: Prentice Hall.
- Haykin, S. (2001). *Kalman Filtering and neural networks*. New York: Wiley.
- Haykin, S., & Li, L. (1995). Nonlinear adaptive prediction of nonstationary signals. *IEEE Transactions on Signal Processing*, 43(2), 526–535.
- Huang, R. C., & Chen, M. S. (2000). Adaptive equalization using complex-valued multilayered neural network based on the extended Kalman filter. In *Proceedings of Signal Processing, International Conference on WCCC-ICSP* (Vol. 1, pp. 519–524). Piscataway, NJ: IEEE Press.
- Julier, S. J., & Uhlmann, J. K. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3), 401–422.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transaction of the ASME—Journal of Basic Engineering*, 82, 35–45.
- Kim, T., & Adali, T. (2001). Approximation by fully complex MLP using elementary transcendental activation functions. In *Proceedings of XI IEEE Workshop on Neural Networks for Signal Processing* (pp. 203–212). Piscataway, NJ: IEEE Press.
- Leung, H., & Haykin, S. (1991). The complex backpropagation algorithm. *IEEE Transactions on Signal Processing*, 39(9), 2101–2104.
- Mandic, D., Baltersee, J., & Chambers, J. (1998). Non-linear adaptive prediction of speech with a pipelined recurrent neural network and advanced learning algorithms. In A. Prochazka, J. Uhlir, P. W. Rayner, & N. G. Kingsbury (Eds.), *Signal analysis and prediction*. Boston: Birkhauser.
- Mandic, D. P., & Chambers, J. A. (2001). *Recurrent neural networks for prediction: Learning algorithms, architectures and stability*. New York: Wiley.
- Mandic, D. P., & Goh, S. L. (2005). Sequential data fusion via vector spaces: Complex modular neural network approach. In *Proceedings of the IEEE Workshop on Machine Learning Signal Processing* (pp. 147–151). Piscataway, NJ: IEEE Press.
- Medsker, L. R., & Jain, L. (2000). *Recurrent neural networks: Design and applications*. Boca Raton, FL: CRC Press.
- Narendra, K. S., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1), 4–27.
- Neeser, F., & Massey, J. (1992). Proper complex random processes with applications to information theory. *IEEE Transactions on Information Theory*, 39(4), 1293–1302.
- Obradovic, D. (1996). On-line training of recurrent neural networks with continuous topology adaptation. *IEEE Transactions on Neural Networks*, 7(1), 222–228.
- Obradovic, D., Lenz, H., & Schupfner, M. (2004). Sensor fusion in Siemens car navigation system. In *Proceedings of the IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing* (pp. 655–664). Piscataway, NJ: IEEE Press.
- Picinbono, B. (1996). Second-order complex random vectors and normal distributions. *IEEE Transactions on Signal Processing*, 44(10), 2637–2640.
- Picinbono, B. (1998). On circularity. *IEEE Transactions on Signal Processing*, 42(12), 3473–3482.
- Puskorius, G. V., & Feldkamp, L. A. (1991). Decoupled extended Kalman filter training of feedforward layered networks. *IJCNN91 Settle International Joint Conference on Neural Networks* (Vol. 1, pp. 771–777). Piscataway, NJ: IEEE Press.

- Puskorius, G. V., & Feldkamp, L. A. (1994). Neurocontrol of nonlinear dynamical systems with Kalman filter trained recurrent networks. *IEEE Transactions on Neural Networks*, 5(2), 279–297.
- Schreier, P. J., & Scharf, L. L. (2003). Second-order analysis of improper complex random vectors and processes. *IEEE Transactions on Signal Processing*, 51(3), 714–725.
- Treichler, J. R., Johnson, J. C. R., & Larimore, M. (1987). *Theory and design of adaptive filters*. New York: Wiley.
- Widrow, B., McCool, J., & Ball, M. (1975). The complex LMS algorithm. *Proceedings of the IEEE*, 63, 712–720.
- Williams, R. J., & Zipser, D. A. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2), 270–280.

Received August 22, 2005; accepted July 9, 2006.