Typo Corrections for: "Complex Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models"

Danilo P. Mandic
Communications and Signal Processing Research Group,
Department of Electrical and Electronic Engineering,
Imperial College London, London, SW7 2AZ, U.K.
email: {d.mandic}@imperial.ac.uk
URL: http://www.commsp.ee.ic.ac.uk/~mandic

Last Updated: 16 June 2010

I. Typos In Sections 15.4 and 15.5

The version below correct some of the minor typos and uses a simplified notation (e.g. $\mathbf{F}_{k,k-1} = \mathbf{F}_{k-1}$).

II. The Augmented Kalman Filter Algorithm for RNNs

Amongst recursive filters in the domain of second order statistics, Kalman filters are optimal sequential state estimators for nonstationary signals [112, 141]. They have also been used in several modern applications including state estimation for car navigation systems [193, 223], parameter estimation for time series modelling [245], and the training of neural networks [112, 139, 246]. To discuss Kalman filter based algorithms for the training of complex valued RNNs, we shall first introduce an augmented state space model and the corresponding updates for the Kalman filter. Similarly to ACLMS and ACRTRL, these updates have the same generic form as the standard updates. Due to the augmentation all the vectors have two times the size and matrices four times the size of the corresponding vectors and matrices within the standard algorithm.

Consider a general state-space model, given by [112]

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

where $\mathbf{x}_k$ are the states to be estimated and $\mathbf{y}_k$ is the system output (usually one or a subset of the states). Variables $\boldsymbol{\omega}_k$ and $\mathbf{v}_k$ are independent, zero mean, complex valued Gaussian noise processes with covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$ respectively, and $\mathbf{F}$ and $\mathbf{H}$ are the transition and measurement matrices. The augmented state space model can be written as

$$\begin{aligned} \mathbf{x}^a_k &= \mathbf{F}^a_{k-1}\mathbf{x}^a_{k-1} + \boldsymbol{\omega}^a_k \\ \mathbf{y}^a_k &= \mathbf{H}^a_k\mathbf{x}^a_k + \boldsymbol{\nu}^a_k \end{aligned} \tag{2}$$

where $\mathbf{x}^a_k = \left[\mathbf{x}^T_k, \mathbf{x}^H_k\right]^T$, $\mathbf{y}^a_k = \left[\mathbf{y}^T_k, \mathbf{y}^H_k\right]^T$, $\mathbf{F}^a_k = \begin{bmatrix} \mathbf{F}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{F}^*_k \end{bmatrix}$, $\mathbf{H}^a_k = \begin{bmatrix} \mathbf{H}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{H}^*_k \end{bmatrix}$, $\boldsymbol{\omega}^a_k = \left[\boldsymbol{\omega}^T_k, \boldsymbol{\omega}^H_k\right]^T$ and $\boldsymbol{\nu}^a_k = \left[\boldsymbol{\nu}^T_k, \boldsymbol{\nu}^H_k\right]^T$. The augmented equivalents of $\mathbf{Q}_k$ and $\mathbf{R}_k$ are denoted respectively by $\mathbf{Q}^a_k$ and $\mathbf{R}^a_k$.

To initialise the algorithm for the time instant $k = 0$, set

$$\begin{aligned} &\hat{\mathbf{x}}^a_0 = E\left[\mathbf{x}^a_0\right], \\ &\mathbf{P}_0 = E\left[\left(\mathbf{x}^a_0 - E\left[\mathbf{x}^a_0\right]\right)\left(\mathbf{x}^a_0 - E\left[\mathbf{x}^a_0\right]\right)^h\right] \end{aligned} \tag{3}$$

The updates within the Kalman filtering algorithms are given below[1]

State estimate propagation:

$$\hat{\mathbf{x}}^{a-}_k = \mathbf{F}^a_{k-1}\hat{\mathbf{x}}^a_{k-1} \tag{4}$$

Error covariance propagation:

$$\mathbf{P}^-_k = \mathbf{F}^a_{k-1}\mathbf{P}_{k-1}(\mathbf{F}^a_{k-1})^H + \mathbf{Q}^a_k \tag{5}$$

Kalman gain matrix:

$$\mathbf{G}_k = \mathbf{P}^-_k(\mathbf{H}^a_k)^H\left[\mathbf{H}^a_k\mathbf{P}^-_k(\mathbf{H}^a_k)^H + \mathbf{R}^a_k\right]^{-1} \tag{6}$$

[1]For clarity, we use notation similar to that from [112].

State estimate update:

$$\hat{\mathbf{x}}_k^a = \hat{\mathbf{x}}_k^{a-} + \mathbf{G}_k \left(\mathbf{y}_k^a - \mathbf{H}_k^a \hat{\mathbf{x}}_k^{a-}\right) \tag{7}$$

Error covariance update:

$$\mathbf{P}_k = \left(\mathbf{I} - \mathbf{G}_k \mathbf{H}_k^a\right) \mathbf{P}_k^- \tag{8}$$

This completes the description of the augmented complex valued Kalman filter (ACKF).

A. EKF Based Training of Complex RNNs

To establish a mathematical framework for Kalman filter based training of complex RNNs, consider a nonlinear state space model[2]

$$\begin{aligned} \mathbf{w}_k^a &= \mathbf{w}_{k-1}^a + \boldsymbol{\omega}_k^a \\ \mathbf{y}_k^a &= \mathbf{h}(\mathbf{w}_k^a, \mathbf{u}_k^a) + \boldsymbol{\nu}_k^a \end{aligned} \tag{9}$$

where $\mathbf{h}(\cdot)$ is a nonlinear operator associated with observations, $\mathbf{w}_k^a$ is an augmented weight vector of the network, $\mathbf{u}_k^a$ is the overall input vector to the network, and $\mathbf{y}_k^a$ is the augmented vector of observations. From the first expression in (9), the complex weights within RNN are modelled as random walk. The idea behind the Extended Kalman Filter (EKF) is to linearise the state space model (9) locally (for every time instant k) based on a truncated Taylor series expansion around $\mathbf{h}$ [70, 102]. Once such a local linear model is obtained, standard ACKF updates (4) – (8) can be applied.

The Augmented Complex Extended Kalman Filtering Algorithm (ACEKF) for the training of complex RNNs can now be summarised as [95]

$$\begin{aligned} \mathbf{G}_k &= \mathbf{P}_k^- (\mathbf{H}_k^a)^H \left[\mathbf{H}_k^a \mathbf{P}_k^- (\mathbf{H}_k^a)^H + \mathbf{R}_k^a\right]^{-1} \\ \hat{\mathbf{w}}_k^a &= \hat{\mathbf{w}}_k^{a-} + \mathbf{G}_k \left[\mathbf{y}_k^a - \mathbf{h}(\hat{\mathbf{w}}_k^{a-}, \mathbf{u}_k^a)\right] \\ \mathbf{P}_k &= \left(\mathbf{I} - \mathbf{G}_k \mathbf{H}_k^a\right) \mathbf{P}_k^- + \mathbf{Q}_k^a \end{aligned} \tag{10}$$

and is initialised by

$$\begin{aligned} \hat{\mathbf{w}}_0^a &= E\left[\mathbf{w}_0\right] \\ \mathbf{P}_0 &= E\left[\left(\mathbf{w}_0^a - E\left[\mathbf{w}_0^a\right]\right)\left(\mathbf{w}_0^a - E\left[\mathbf{w}_0^a\right]\right)^H\right] \end{aligned} \tag{11}$$

The augmented Jacobian[3] matrix $\mathbf{H}_k^a$ of the partial derivatives of $\mathbf{h}$ is computed using the augmented CRTRL algorithm for recurrent networks [93](using fully-complex nonlinearities). 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 $\mathbf{H}_k^a \mathbf{P}_k^- (\mathbf{H}_k^a)^H + \mathbf{R}_k^a$.

III. AUGMENTED COMPLEX UNSCENTED KALMAN FILTER (ACUKF)

Since the higher order terms within the Taylor series expansion in the EKF model are often not negligible, the EKF is prone to accumulating error over time (10). To help solve this problem, the unscented Kalman filter (UKF) [139, 301] has been proposed, whereby nonlinear transforms are used to propagate the signal statistics. This way, the information from higher order moments of non–Gaussian processes is accounted for, and the approximations within the UKF scenario are accurate at least up to second order statistical moments[4] [139]. Within the CUKF, a series of so–called complex valued sigma vectors, that is, vectors selected to be representatives of the probability distribution, are used to calculate the crosscorrelation between the error in the estimated state and error in the estimated observations, as well as the correlation matrix of the error.

Within the CUKF framework, the information about the distributions of complex random variables is propagated through the system model (9) using $(2L+1)$ weighted particles, where L is the dimension of the state space of the system. The weighting for every such particle is given by

$$\begin{aligned} &\mathcal{W}_0^{(m)} = \frac{\lambda}{L+\lambda}, \\ &\mathcal{W}_0^{(c)} = \frac{\lambda}{L+\lambda} + 1 - \alpha^2 + \beta, \\ &\mathcal{W}_n^{(m)} = \mathcal{W}_n^{(c)} = \frac{\lambda}{2(L+\lambda)}, \quad n = 1, \ldots, 2L \end{aligned} \tag{12}$$

[2]EKF based algorithms have proven successful for the training of real valued temporal neural networks [181, 222].

[3]Matrix $\mathbf{H}_k^a$ is the matrix of partial derivatives of the augmented output $\mathbf{y}_k^a$ with respect to the weights.

[4]The EKF is accurate only up to first order statistics due to the first order linearisation in the truncated Taylor series expansion.

where $\lambda = \alpha^2(L+\kappa) - L$ is a scaling parameter, α is set to a small value (typically of order 10^{-3}) and is related to the spread of the sample points around the mean, κ is usually set to 0, whereas parameter β incorporates knowledge from *prior* distributions (in the case of complex valued Gaussian distributions, the optimal value is $\beta = 2$).

A. State Space Equations for the Complex Unscented Kalman Filter

The CUKF effectively aims at evaluating the Jacobian matrix within CEKF through the so called sigma–point propagation, hence not requiring any analytical calculation of the derivative. The complex valued weight vector within the network[5] and the error covariance matrix are initialised as

$$\hat{\mathbf{w}}_0 = E[\mathbf{w}], \quad \mathbf{P}_0 = E\left[(\mathbf{w} - \hat{\mathbf{w}}_0)(\mathbf{w} - \hat{\mathbf{w}}_0)^T\right] \tag{13}$$

whereas the sigma–point calculation is given by [301]

$$\begin{aligned} \boldsymbol{\mathcal{S}}_k &= (L+\lambda)(\mathbf{P}_k + \mathbf{Q}_k) \\ \mathsf{W}_k &= \left[\hat{\mathbf{w}}_k, \hat{\mathbf{w}}_k + \sqrt{\boldsymbol{\mathcal{S}}_k}, \hat{\mathbf{w}}_k - \sqrt{\boldsymbol{\mathcal{S}}_k}\right] \end{aligned} \tag{14}$$

These sigma point estimates are then passed through a nonlinear function $\mathbf{h}$, that is

$$\boldsymbol{\mathcal{Y}_k} = \mathbf{h}(\mathsf{W}_k, \mathbf{x}_k) \tag{15}$$

and their mean is computed as

$$\mathbf{y}_k = \sum_{n=0}^{2L} \mathcal{W}_n^m \boldsymbol{\mathcal{Y}}_{n,k} \tag{16}$$

to yield the measurement-update equations for the CUKF in the form

$$\begin{aligned} \mathbf{P}_{\mathbf{yy},k} &= \sum_{n=0}^{2L} \mathcal{W}_n^c \left((\boldsymbol{\mathcal{Y}}_{n,k} - \mathbf{y}_k)(\boldsymbol{\mathcal{Y}}_{n,k} - \mathbf{y}_k)^H\right) + \mathbf{R}_k \\ \mathbf{P}_{\mathbf{wy},k} &= \sum_{n=0}^{2L} \mathcal{W}_n^c \left((\mathsf{W}_{n,k} - \hat{\mathbf{w}}_k)(\boldsymbol{\mathcal{Y}}_{n,k} - \mathbf{y}_k)^H\right) \end{aligned} \tag{17}$$

Finally, the filter update recursions for the complex unscented Kalman filter are given by

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_{\mathbf{wy},k} \mathbf{P}_{\mathbf{yy},k}^{-1} \\ \hat{\mathbf{w}}_{k+1} &= \hat{\mathbf{w}}_k + \mathbf{K}_k \mathbf{e}_k \\ \mathbf{P}_{k+1} &= \mathbf{P}_k - \mathbf{K}_k \mathbf{P}_{\mathbf{yy},k} \mathbf{K}_k^H \end{aligned} \tag{18}$$

where the estimation error $\mathbf{e}_k = \mathbf{d}_k - \mathbf{y}_k$, and $\mathbf{d}_k$ is the desired output vector.

The conceptual differences between CUKF and complex valued EKF [95] are relatively minor but result in significant theoretical and practical advantages. For instance, the use of sigma vectors (14) to improve the estimation of the statistical properties of the signal in hand facilitates the processing of non–Gaussian processes, typically found in real world applications.

B. ACUKF Based Training of Complex RNNs

Consider the augmented state space model

$$\begin{aligned} \mathbf{w}_k^a &= \mathbf{w}_{k-1}^a + \boldsymbol{\omega}_k^a \\ \mathbf{y}_k^a &= \mathbf{h}(\mathbf{w}_k^a, \mathbf{x}_k^a) + \boldsymbol{\nu}_k^a \end{aligned} \tag{19}$$

with the augmented complex variables as in the ACKF. The augmented covariance matrices of zero mean complex valued Gaussian noise processes $\boldsymbol{\omega}$ and $\boldsymbol{\nu}$ are denoted respectively by $\mathbf{Q}_k^a$ and $\mathbf{R}_k^a$. After the state augmentation, based on (12) the

[5]The ACUKF training is derived for a general case of RNNs. The algorithms can be straightforwardly simplified to IIR and FIR filters, by removing nonlinearity or feedback.

$(4L+1)$ weighted particles for the augmented complex valued mean and covariance estimation become

$$\mathcal{W}_0^{(m)} = \frac{\lambda}{2L+\lambda},$$
$$\mathcal{W}_0^{(c)} = \frac{\lambda}{2L+\lambda} + 1 - \alpha^2 + \beta,$$
$$\mathcal{W}_n^{(m)} = \mathcal{W}_n^{(c)} = \frac{\lambda}{2(2L+\lambda)}, \quad n = 1, \ldots, 4L$$

where $\lambda = \alpha^2(2L+\kappa) - 2L$ is a scaling parameter.

The following expressions summarise the augmented CUKF for the training of complex valued RNNs

$$\begin{aligned}
\hat{\mathbf{w}}_0^a &= E[\mathbf{w}_0^a] \\
\mathbf{P}_0^a &= E\left[\left(\mathbf{w}_0^a - \hat{\mathbf{w}}_0^a\right)\left(\mathbf{w}_0^a - \hat{\mathbf{w}}_0^a\right)^T\right] \\
\boldsymbol{\mathcal{S}}_k^a &= (2L+\lambda)(\mathbf{P}_k^a + \mathbf{Q}_k^a) \\
\mathsf{W}_k^a &= \left[\hat{\mathbf{w}}_k^a, \hat{\mathbf{w}}_k^a + \sqrt{\boldsymbol{\mathcal{S}}_k^a}, \hat{\mathbf{w}}_k^a - \sqrt{\boldsymbol{\mathcal{S}}_k^a}\right]
\end{aligned} \tag{20}$$

whereby, based on (18) and (19), the recursive updates within ACUKF are given by

$$\begin{aligned}
\boldsymbol{\mathcal{Y}_k^a} &= \mathbf{h}\left(\mathsf{W}_k^a, \mathbf{x}_k^a\right) \\
\mathbf{y}_k^a &= \sum_{n=0}^{4L} \mathcal{W}_n^m \boldsymbol{\mathcal{Y}}_{n,k}^a \\
\mathbf{P}_{\mathbf{yy},k}^a &= \sum_{n=0}^{4L} \mathcal{W}_n^c \left(\left(\boldsymbol{\mathcal{Y}}_{n,k}^a - \mathbf{y}_k^a\right)\left(\boldsymbol{\mathcal{Y}}_{n,k}^a - \mathbf{y}_k^a\right)^H\right) + \mathbf{R}_k^a \\
\mathbf{P}_{\mathbf{wy},k}^a &= \sum_{n=0}^{4L} \mathcal{W}_n^c \left(\left(\mathsf{W}_{n,k}^a - \hat{\mathbf{w}}_k^a\right)\left(\boldsymbol{\mathcal{Y}}_{n,k}^a - \mathbf{y}_k^a\right)^H\right) \\
\mathbf{K}_k^a &= \mathbf{P}_{\mathbf{wy},k}^a \{\mathbf{P}_{\mathbf{yy},k}^a\}^{-1} \\
\hat{\mathbf{w}}_{k+1}^a &= \hat{\mathbf{w}}_k^a + \mathbf{K}_k^a \mathbf{e}_k^a \\
\mathbf{P}_{k+1}^a &= \mathbf{P}_k^a - \mathbf{K}_k^a \mathbf{P}_{\mathbf{yy},k}^a \{\mathbf{K}_k^a\}^H
\end{aligned} \tag{21}$$

ACKNOWLEDGEMENTS

The authers wish to thank Jesus Navarro Moreno and Dahir H. Dini for pointing out the typos in sections 15.4 and 15.5.