

Analysis of incremental augmented affine projection algorithm for distributed estimation of complex signals

Azam Khalili^a, Wael M. Bazzi^b, Amir Rastegarnia^{a,*}

^a*Department of Electrical Engineering, Malayer University, Malayer, 65719-95863, Iran*

^b*Electrical and Computer Engineering Department, American University in Dubai, Dubai, United Arab Emirates*

Abstract

This paper considers the problem of distributed estimation in an incremental network when the measurements taken by the node follow a widely linear model. The proposed algorithm which we refer to it as incremental augmented affine projection algorithm (incAAPA) utilizes the full second order statistical information in the complex domain. Moreover, it exploits spatio-temporal diversity to improve the estimation performance. We derive steady-state performance metric of the incAAPA in terms of the mean-square deviation (MSD). We further derive sufficient conditions to ensure mean-square convergence. Our analysis illustrate that the proposed algorithm is able to process both second order circular (proper) and noncircular (improper) signals. The validity of the theoretical results and the good performance of the proposed algorithm are demonstrated by several computer simulations.

Keywords: adaptive networks, incremental, complex data, affine.

1. Introduction

In many practical applications the ultimate goal is to estimate an unknown parameter of interest by using observations acquired by spatially distributed nodes [1, 2]. Different solutions (algorithms) have been introduced in the literature to solve the decentralized estimation problem. In some algorithms

*corresponding author, Email: a_rastegar@ieee.org

(known as consensus methods) such as [3, 4] the nodes collect all the data first, perform local estimation and then interact iteratively with their neighbors. Networks that rely on in-network processing at each node while allowing the node to learn with new observations are known as adaptive networks [5, 6]. An adaptive network consists of a collection of spatially distributed nodes that are able to communicate with each other through a topology. Two major class of adaptive networks, based on the network topology are incremental networks [7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] or diffusion algorithms [20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]. In the incremental algorithms, a cyclic path through the network is established, and nodes communicate with neighbors within this path. In adaptive diffusion implementations, information is processed locally at the nodes and then diffused in real-time across the network.

1.1. Motivation for current work

An important factor in learning abilities of adaptive networks is the adaptive filter that is embedded at the nodes. The mentioned adaptive networks use different types of adaptive filters such as least mean-square (LMS) [8, 14], recursive least-squares (RLS) [7, 21], affine projection algorithm (APA) [10, 11] and normalized least mean squares (NLMS) [31]. The LMS algorithm is a popular choice due to its stability and low complexity. On the other hand, the algorithms based on the least squares criterion such as the RLS algorithm, converge faster. However, they suffer from the high computational complexity issue [32]. The affine projection algorithm has the benefits of both approaches that is stability, low computational complexity and fast convergence [32, 33].

The APA was originally introduced for real-value signals [34, 35]. The complex domain provides a natural processing framework for signals with intensity and direction components [36, 37]. Statistical signal processing in \mathbb{C} has traditionally been viewed as a straightforward extension of the corresponding algorithms in the real domain \mathbb{R} . However, recent developments in augmented complex statistics show that they do not make full use of the algebraic structure of the complex domain [36]. For example, it was shown that the covariance

matrix is not sufficient to model the statistics of noncircular signals and it is necessary to introduce the pseudocovariance matrix to fully capture the relation between the real and imaginary components of random vectors. It is also shown that the standard linear model is only sufficient for modelling 'proper' signals, whereas an optimal model for 'improper' signals is provided by a widely linear model [38, 39].

1.2. Contributions

The aforementioned adaptive networks rely on linear model and signal processing in \mathbb{R} domain. In this paper, we develop and analyze an incremental adaptive network based on the widely linear model which employs the affine projection algorithm as learning entity to process the complex-valued signals. The proposed algorithm, which we refer to as incremental augmented affine projection algorithm (incAAPA), exploits spatio-temporal diversity to improve the estimation performance. Moreover, at the same time, it employs augmented complex statistics which it turn enables it to process both types of proper and improper signals. To analyze the steady-state performance of the IncAAPA by resorting to the weighted energy conservation approach and derive a closed-form expression, in terms of the mean-square deviation (MSD) that explains how the algorithm performs in the steady-state. We further derive the stability bound of the proposed algorithm to ensure mean-square convergence. We use different synthetic benchmark signals including circular complex-valued signal, the non-circular complex-valued signal and noncircular chaotic Ikeda map signal in our simulations to evaluate the performance of the proposed algorithm. Simulation results validate the theoretical results and reveal the advantage of using the proposed algorithm for processing of both second order circular (proper) and noncircular (improper) signals.

1.3. Paper organization and notation

The remainder of this paper is organized as follows. In Section 2, we formulate the estimation problem and derive the proposed algorithm. In Section 3,

Table 1: Symbols and their descriptions.

$x, \mathbf{x}, \mathbf{X}$	Scalar, column vector, matrix;
$\text{Tr}[\mathbf{A}]$	Trace of matrix \mathbf{A} ,
$(\cdot)^T, (\cdot)^H, (\cdot)^*$	Transpose, Hermitian transpose, Complex conjugation
$E[\cdot]$	Statistical expectation operator;
\otimes	Kronecker product
$\Re\{\cdot\}, \Im\{\cdot\}$	Real, Imaginary parts of a complex number
$\text{vec}\{\cdot\}$	This notation will be used in two ways: $\mathbf{x} = \text{vec}\{\Sigma\}$ is a $M^2 \times 1$ column vector whose entries are formed by stacking the successive columns of an $M \times M$ matrix on top of each other, and $\mathbf{X} = \text{vec}\{\mathbf{x}\}$ is a matrix whose entries are recovered from \mathbf{x} .
$\ \mathbf{u}\ _{\Sigma}^2 = \mathbf{u}^H \Sigma \mathbf{u}$	the weighted square norm of \mathbf{u} .
$\lambda_{\max}(\mathbf{A})$	The largest eigenvalue of matrix \mathbf{A} .

performance analysis of the incAAPA algorithm is provided. In Section 4, we present simulation results to verify our theoretical analysis, and we conclude in Section 5.

We adopt small boldface letters for vectors and bold capital letters for matrices. The other symbols used in this paper are listed in Table. 1.

2. Proposed Algorithm

2.1. The estimation problem

Let denote by set $\mathcal{K} = \{1, 2, \dots, N\}$ a network with N nodes which communicate according to the incremental protocol. This protocol require a Hamiltonian cycle through which local information are sequentially circulated from node to node (See Fig. 1). At any time i , node k measures data $\{d_k(i), \mathbf{x}_{k,i}\}$ where $d_k(i) \in \mathbb{C}$ is the desired signal and $\mathbf{x}_{k,i} = [x_{k,i}(1), x_{k,i}(2), \dots, x_{k,i}(L)]^T \in \mathbb{C}^{L \times 1}$ denotes the input vector at node k . we consider the following assumptions on our data model.

Assumption 1. The desired signal $d_k(i)$ and the input vector $\mathbf{x}_{k,i}$ are related through a widely linear model as [36]

$$d_k(i) = \mathbf{x}_{k,i}^T \mathbf{h}^o + \mathbf{x}_{k,i}^H \mathbf{g}^o + v_k(i), \quad (1)$$

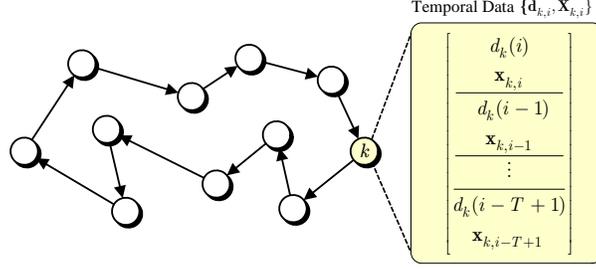


Figure 1: An incremental network with N nodes.

where $\{\mathbf{h}^o, \mathbf{g}^o\} \in \mathbb{R}^{L \times 1}$ are the unknown parameters, and $v_k(i) \in \mathbb{C}$ denotes samples of doubly white noise with variance $\sigma_{v,k}^2$. The measurement noises $\{v_k(i)\}$ for all nodes $k = 1, \dots, N$, and all observation times $i \geq 1$, are independent of each other and the regression vectors $\mathbf{x}_{k,i}$.

Assumption 2. For $k = 1, \dots, N$ and $i \geq 1$, the regression vectors $\mathbf{x}_{k,i}$ are independent over node indices k and observation times i .

The measurements model (1) appears in many practical applications, such as the frequency estimation problem in three-phase power systems [40], and in simultaneous modelling and forecasting of wind signals [41, 42].

Definition 1. (*Noncircular signals*) To define the noncircular (or improper) signals, let define the augmented vector $\mathbf{x}_{k,i}^a$ as

$$\mathbf{x}_{k,i}^a \triangleq \begin{bmatrix} \mathbf{x}_{k,i} \\ \mathbf{x}_{k,i}^* \end{bmatrix}, \quad (2L \times 1)$$

The covariance matrix $\mathbf{C}_{\mathbf{xx},k}^a$ for the augmented vector $\mathbf{x}_{k,i}^a$ is given by

$$\mathbf{C}_{\mathbf{xx},k}^a = \mathbb{E}[\mathbf{x}_{k,i}^a \mathbf{x}_{k,i}^{aH}] = \begin{bmatrix} \mathbf{C}_{\mathbf{xx},k} & \mathbf{P}_{\mathbf{xx},k} \\ \mathbf{P}_{\mathbf{xx},k}^* & \mathbf{C}_{\mathbf{xx},k}^* \end{bmatrix} \quad (2)$$

when the pseudo-covariance matrix $\mathbf{P}_{\mathbf{xx},k} = \mathbf{0}$, the complex random vector is called circular (proper).

2.2. Algorithm derivation

To develop the proposed algorithm we denote by $\{\mathbf{d}_{k,i}, \mathbf{X}_{k,i}\}$ the T most recent observations at node k as

$$\mathbf{X}_{k,i} = [\mathbf{x}_{k,i}, \dots, \mathbf{x}_{k,i-T+1}], \quad (L \times T) \quad (3)$$

$$\mathbf{d}_{k,i} = [d_k(i), \dots, d_k(i-T+1)]^\top, \quad (T \times 1) \quad (4)$$

Our objective is to develop a distributed and adaptive algorithm that is able to estimate the unknown parameters $\{\mathbf{h}^o, \mathbf{g}^o\}$ at every node $k \in \mathcal{K}$. To achieve this objective, we formulate the desired estimation problem as the following constrained optimization problem which is based on the minimum disturbance [43]

$$\begin{aligned} & \text{minimize } f(\mathbf{h}_i, \mathbf{g}_i) \triangleq \left(\|\mathbf{h}_i - \mathbf{h}_{i-1}\|^2 + \|\mathbf{g}_i - \mathbf{g}_{i-1}\|^2 \right) \quad (5) \\ & \text{subject to } \sum_{k=1}^K \Re \left\{ (\mathbf{d}_{k,i} - \mathbf{X}_{k,i}^\top \mathbf{h}_i - \mathbf{X}_{k,i}^H \mathbf{g}_i)^H \right\} = 0 \end{aligned}$$

In the following theorem we discuss the solution of the problem (5).

Theorem 1. *Let $\mathbf{h}_{k,i}$ and $\mathbf{g}_{k,i}$ denote the local estimates of \mathbf{h}^o and \mathbf{g}^o at node k at time i respectively. The optimal weights $\{\mathbf{h}^o, \mathbf{g}^o\}$ can be recursively estimated at every node k via the following update equations*

$$\mathbf{h}_{k,i} = \mathbf{h}_{k-1,i} + \mu_k \mathbf{X}_{k,i}^* \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (6)$$

$$\mathbf{g}_{k,i} = \mathbf{g}_{k-1,i} + \mu_k \mathbf{X}_{k,i} \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (7)$$

where

$$\mathbf{A}_{k,i} \triangleq \mathbf{X}_{k,i}^H \mathbf{X}_{k,i} + \mathbf{X}_{k,i}^\top \mathbf{X}_{k,i}^* \quad (8)$$

$$\mathbf{B}_{k,i} \triangleq (\mathbf{A}_{k,i} + \delta \mathbf{I})^{-1} \quad (9)$$

Moreover, $\delta > 0$ is the regularisation parameter, $\mu > 0$ is the step-size parameter and $\mathbf{e}_{k,i}$ is the local error signal which is defined as

$$\mathbf{e}_{k,i} \triangleq \mathbf{d}_{k,i} - \mathbf{X}_{k,i}^\top \mathbf{h}_{k-1,i} - \mathbf{X}_{k,i}^H \mathbf{g}_{k-1,i} \quad (10)$$

PROOF. See [45].

Remark 1. It must be noted that a non-cooperative solution for the problem (5) based on the affine projection algorithm is given by

$$\mathbf{h}_{k,i} = \mathbf{h}_{k,i-1} + \mu_k \mathbf{X}_{k,i}^* \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (11)$$

$$\mathbf{g}_{k,i} = \mathbf{g}_{k,i-1} + \mu_k \mathbf{X}_{k,i} \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (12)$$

where reveals that in this solution every node uses solely its local information.

An important question now is, how well does the adaptive incremental solution (6) and (7) perform? That is, how close does each $\{\mathbf{h}_{k,i}, \mathbf{g}_{k,i}\}$ get to the desired solutions $\{\mathbf{h}^o, \mathbf{g}^o\}$ as time evolves? In the next section, we provide a framework for studying the performance of such network by examining the flow of energy through the network both in time and space.

3. Performance Analysis

In order to study the performance of the IncAAPA algorithm, we extend the weighted energy conservation argument for the stand alone adaptive filters [32] to the case of incremental networks. we evaluate the steady-state performances at each individual node in terms of mean-square deviation (MSD). To begin the analysis we define the following weight error vectors as

$$\tilde{\mathbf{g}}_{k,i} = \mathbf{g}_0 - \mathbf{g}_{k,i}, \quad \tilde{\mathbf{h}}_{k,i} = \mathbf{h}_0 - \mathbf{h}_{k,i} \quad (13)$$

using (13), the update equations in (6) and (7) can be rewritten in terms of the weight error vectors as

$$\tilde{\mathbf{h}}_{k,i} = \tilde{\mathbf{h}}_{k-1,i} - \mu_k \mathbf{X}_{k,i}^* \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (14)$$

$$\tilde{\mathbf{g}}_{k,i} = \tilde{\mathbf{g}}_{k-1,i} - \mu_k \mathbf{X}_{k,i} \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (15)$$

Multiplying both sides of (14) and (15) by $\mathbf{X}_{k,i}^\top \Sigma_k$ and $\mathbf{X}_{k,i}^\mathbf{H} \Sigma_k$ respectively gives

$$\mathbf{X}_{k,i}^\top \Sigma_k \tilde{\mathbf{h}}_{k,i} = \mathbf{X}_{k,i}^\top \Sigma_k \tilde{\mathbf{h}}_{k-1,i} - \mu_k \mathbf{X}_{k,i}^\top \Sigma_k \mathbf{X}_{k,i}^* \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (16)$$

$$\mathbf{X}_{k,i}^\mathbf{H} \Sigma_k \tilde{\mathbf{g}}_{k,i} = \mathbf{X}_{k,i}^\mathbf{H} \Sigma_k \tilde{\mathbf{g}}_{k-1,i} - \mu_k \mathbf{X}_{k,i}^\mathbf{H} \Sigma_k \mathbf{X}_{k,i} \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (17)$$

The weighted a-posteriori and weighted a-priori error vectors $\{\mathbf{e}_{p,k}^{\Sigma_k}, \mathbf{e}_{a,k}^{\Sigma_k}\}$ are introduced as

$$\mathbf{e}_{p,k}^{\Sigma_k} = \mathbf{X}_{k,i}^T \Sigma_k \tilde{\mathbf{h}}_{k,i} + \mathbf{X}_{k,i}^H \Sigma_k \tilde{\mathbf{g}}_{k,i} \quad (18)$$

$$\mathbf{e}_{a,k}^{\Sigma_k} = \mathbf{X}_{k,i}^T \Sigma_k \tilde{\mathbf{h}}_{k-1,i} + \mathbf{X}_{k,i}^H \Sigma_k \tilde{\mathbf{g}}_{k-1,i} \quad (19)$$

If we add (16) and (17) and use the definitions of $\mathbf{e}_{p,k}^{\Sigma_k}$ and $\mathbf{e}_{a,k}^{\Sigma_k}$ we get

$$\mathbf{e}_{p,k}^{\Sigma_k} = \mathbf{e}_{a,k}^{\Sigma_k} - \mu_k \mathbf{F}_{k,i} \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (20)$$

where

$$\mathbf{F}_{k,i} \triangleq \mathbf{X}_{k,i}^H \Sigma_k \mathbf{X}_{k,i} + \mathbf{X}_{k,i}^T \Sigma_k \mathbf{X}_{k,i}^* \quad (21)$$

using (20), we can be expressed $\mathbf{e}_{k,i}$ as

$$\mathbf{e}_{k,i} = \frac{1}{\mu_k} \mathbf{B}_{k,i}^{-1} \mathbf{F}_{k,i}^{-1} (\mathbf{e}_{a,k}^{\Sigma_k} - \mathbf{e}_{p,k}^{\Sigma_k}) \quad (22)$$

Now we define the augmented quantities $\tilde{\mathbf{w}}_{k,i}$ and $\mathbf{U}_{k,i}$ as

$$\tilde{\mathbf{w}}_{k,i} \triangleq \begin{bmatrix} \tilde{\mathbf{h}}_{k,i} \\ \tilde{\mathbf{g}}_{k,i} \end{bmatrix}_{2L \times 1}, \quad \mathbf{U}_{k,i} \triangleq \begin{bmatrix} \mathbf{X}_{k,i}^* \\ \mathbf{X}_{k,i} \end{bmatrix}_{2L \times T}, \quad (23)$$

Corollary 1. Using the definition in (23) we have

$$\mathbf{e}_{p,k}^{\Sigma_k} = \mathbf{U}_{k,i}^H \Sigma_k \tilde{\mathbf{w}}_{k,i}, \quad (24a)$$

$$\mathbf{e}_{a,k}^{\Sigma_k} = \mathbf{U}_{k,i}^H \Sigma_k \tilde{\mathbf{w}}_{k-1,i} \quad (24b)$$

$$\mathbf{F}_{k,i} = \mathbf{U}_{k,i}^H \Sigma_k \mathbf{U}_{k,i} \quad (24c)$$

This definitions in (23) allows us to rewrite equations (14) and(15) as the following equivalent form

$$\tilde{\mathbf{w}}_{k,i} = \tilde{\mathbf{w}}_{k-1,i} - \mathbf{U}_{k,i} \mathbf{F}_{k,i}^{-1} (\mathbf{e}_{a,k}^{\Sigma_k} - \mathbf{e}_{p,k}^{\Sigma_k}) \quad (25)$$

To derive the desired steady-state performance metric, we need to derive the mean-square variance relation that explains how $\{\mathbf{h}_{k,i}, \mathbf{g}_{k,i}\}$ evolves in time. Thus, we assert the following theorem on the mean-square variance relation of the incAAPA.

Theorem 2. *The mean-square variance relation of the incAAPA is given by the following expression.*

$$\mathbf{E}[\|\tilde{\mathbf{w}}_{k,i}\|_{\Sigma_k}^2] = \mathbf{E}[\|\tilde{\mathbf{w}}_{k-1,i}\|_{\Sigma'_k}^2] + \mu_k^2 \mathbf{E}[\mathbf{v}_{k,i}^H \mathbf{D}_{k,i} \Sigma_k \mathbf{D}_{k,i} \mathbf{v}_{k,i}] \quad (26)$$

$$\Sigma'_k = \Sigma_k - \mu_k \Sigma_k \mathbf{E}[\mathbf{D}_{k,i}] - \mu_k \mathbf{E}[\mathbf{D}_{k,i}] \Sigma_k + \mu_k^2 \mathbf{E}[\mathbf{D}_{k,i} \Sigma_k \mathbf{D}_{k,i}] \quad (27)$$

PROOF. See Appendix B for details.

To derive the desired metric we apply the $\text{vec}(\cdot)$ operator on both sides of (26) and (27) and get¹

$$\text{vec}\{\Sigma_k \mathbf{E}[\mathbf{D}_{k,i}]\} = (\mathbf{E}[\mathbf{D}_{k,i}^T] \otimes \mathbf{I}) \boldsymbol{\sigma}_k \quad (28)$$

$$\text{vec}\{\mathbf{E}[\mathbf{D}_{k,i}] \Sigma_k\} = (\mathbf{I} \otimes \mathbf{E}[\mathbf{D}_{k,i}]) \boldsymbol{\sigma}_k \quad (29)$$

$$\text{vec}\{\mathbf{E}[\mathbf{D}_{k,i} \Sigma_k \mathbf{D}_{k,i}]\} = (\mathbf{E}[\mathbf{D}_{k,i}^T] \otimes \mathbf{E}[\mathbf{D}_{k,i}]) \boldsymbol{\sigma}_k \quad (30)$$

Therefore, using (28)-(30), we obtain a linear relation between the corresponding vectors $\{\boldsymbol{\sigma}'_k, \boldsymbol{\sigma}_k\}$, namely

$$\boldsymbol{\sigma}'_k = \mathcal{F}_k \boldsymbol{\sigma}_k \quad (31)$$

where \mathcal{F}_k is given by

$$\mathcal{F}_k = \mathbf{I} - \mu_k (\mathbf{E}[\mathbf{D}_{k,i}^T] \otimes \mathbf{I}) - \mu_k (\mathbf{I} \otimes \mathbf{E}[\mathbf{D}_{k,i}]) + \mu_k^2 (\mathbf{E}[\mathbf{D}_{k,i}^T] \otimes \mathbf{E}[\mathbf{D}_{k,i}]) \quad (32)$$

¹For any matrices of compatible dimensions $\{\mathbf{Z}_1, \mathbf{Z}_2, \Sigma_k\}$, it holds that

$$\text{vec}\{\mathbf{Z}_1 \Sigma_k \mathbf{Z}_2\} = (\mathbf{Z}_2^T \otimes \mathbf{Z}_1) \text{vec}\{\Sigma_k\}$$

The second term in right hand side of (26) can be rewritten as

$$\begin{aligned}
\mathbf{s}_k &= \mathbb{E}[\mathbf{v}_{k,i}^H \mathbf{D}_{k,i} \boldsymbol{\Sigma}_k \mathbf{D}_{k,i} \mathbf{v}_{k,i}] \\
&= \text{Tr} \left[\mathbb{E}[\mathbf{v}_{k,i}^H \mathbf{D}_{k,i} \boldsymbol{\Sigma}_k \mathbf{D}_{k,i} \mathbf{v}_{k,i}] \right] \\
&= \mathbb{E} \left[\text{Tr}[\mathbf{v}_{k,i}^H \mathbf{D}_{k,i} \boldsymbol{\Sigma}_k \mathbf{D}_{k,i} \mathbf{v}_{k,i}] \right] \\
&= \mathbb{E} \left[\text{Tr}[\mathbf{D}_{k,i} \mathbf{v}_{k,i} \mathbf{v}_{k,i}^H \mathbf{D}_{k,i} \boldsymbol{\Sigma}_k] \right] \\
&= \sigma_{v,k}^2 \boldsymbol{\gamma}_k^T \boldsymbol{\sigma}_k
\end{aligned} \tag{33}$$

where $\mathbf{v}_{k,i} = [v_k(i), \dots, v_k(i - T + 1)]^T$, $\mathbf{D}_{k,i} = \mathbf{U}_{k,i} \mathbf{B}_{k,i} \mathbf{U}_{k,i}^H$ and

$$\boldsymbol{\gamma}_k = \text{vec}\{\mathbb{E}[\mathbf{D}_{k,i}^2]\} \tag{34}$$

As a result, expression (26) becomes

$$\mathbb{E}[\|\tilde{\mathbf{w}}_{k,i}\|_{\boldsymbol{\sigma}_k}^2] = \mathbb{E}[\|\tilde{\mathbf{w}}_{k-1,i}\|_{\boldsymbol{\sigma}'_k}^2] + \mu_k^2 \delta_{v,k}^2 \boldsymbol{\gamma}_k^T \boldsymbol{\sigma}_k \tag{35}$$

Observe, that (35) is a coupled equation: it involves both $\mathbb{E}[\|\tilde{\mathbf{w}}_{k,i}\|_{\boldsymbol{\sigma}_k}^2]$ and $\mathbb{E}[\|\tilde{\mathbf{w}}_{k-1,i}\|_{\boldsymbol{\sigma}'_k}^2]$, i.e., information from two spatial locations. The ring topology together with the weighting matrices can be exploited to resolve this difficulty [8] where a similar equation is solved for incremental LMS algorithm. If we follow the steps given in [8] we can obtain the following equation for incAAPA algorithm

$$\begin{aligned}
\mathbb{E} \left[\|\tilde{\mathbf{w}}_{k-1}\|_{\boldsymbol{\sigma}_{k-1}}^2 \right] &= \mathbb{E} \left[\|\tilde{\mathbf{w}}_{k-1}\|_{\mathcal{F}_k \dots \mathcal{F}_N \mathcal{F}_1 \dots \mathcal{F}_{k-1} \boldsymbol{\sigma}_{k-1}}^2 \right] \\
&\quad + \mathbf{s}_k \mathcal{F}_{k+1} \dots \mathcal{F}_N \mathcal{F}_1 \dots \mathcal{F}_{k-1} \boldsymbol{\sigma}_{k-1} \\
&\quad + \mathbf{s}_{k+1} \mathcal{F}_{k+2} \dots \mathcal{F}_N \mathcal{F}_1 \dots \mathcal{F}_{k-1} \boldsymbol{\sigma}_{k-1} \\
&\quad + \mathbf{s}_{k-2} \mathcal{F}_{k-1} \boldsymbol{\sigma}_{k-1} + \mathbf{s}_{k-1} \boldsymbol{\sigma}_{k-1}
\end{aligned} \tag{36}$$

Let

$$\mathcal{P}_{k,l} = \mathcal{F}_{k+l-1} \dots \mathcal{F}_N \mathcal{F}_1 \dots \mathcal{F}_{k-1}, \quad l = 1, 2, \dots, N \tag{37}$$

$$\mathbf{f}_k = \mathbf{s}_k \mathcal{P}_{k,2} + \mathbf{s}_{k+1} \mathcal{P}_{k,3} + \dots + \mathbf{s}_{k-2} \mathcal{P}_{k,N} + \mathbf{s}_{k-1} \tag{38}$$

where in (37) the subscripts are all *mod N*. Using (37) and (38), we can represent (36) in the following form

$$\mathbb{E} \left[\|\tilde{\mathbf{w}}_{k-1}\|_{(\mathbf{I}-\mathcal{P}_{k,1})\boldsymbol{\sigma}_{k-1}}^2 \right] = \mathbf{f}_k \boldsymbol{\sigma}_{k-1} \quad (39)$$

Expression (39) can be exploited to evaluate the desired performance measures at node k . In fact, since we are free to select the weight vector $\boldsymbol{\sigma}_{k-1}$, choosing $\boldsymbol{\sigma}_{k-1} = (\mathbf{I} - \mathcal{P}_{k,1})^{-1} \text{vec}\{\mathbf{I}\}$ results in the expression for the steady-state MSD at node k as follows

$$\text{MSD}_k = \mathbf{f}_k (\mathbf{I} - \mathcal{P}_{k,1})^{-1} \text{vec}\{\mathbf{I}\} \quad (40)$$

$$(41)$$

To derive the required mean-square stability condition we refer to (35). The recursion is stable if the matrix \mathcal{F}_k is stable. From (32), we can rewrite \mathcal{F}_k as

$$\mathcal{F}_k = \mathbf{I} - \mu_k \mathcal{M}_k + \mu_k^2 \mathcal{N}_k \quad (42)$$

where

$$\mathcal{M}_k = (\mathbb{E}[\mathbf{D}_{k,i}^T] \otimes \mathbf{I}) + (\mathbf{I} \otimes \mathbb{E}[\mathbf{D}_{k,i}]) \quad (43)$$

$$\mathcal{N}_k = (\mathbb{E}[\mathbf{D}_{k,i}^T] \otimes \mathbb{E}[\mathbf{D}_{k,i}]) \quad (44)$$

Following the same approach in [44], the condition on step-size that guarantees the convergence of the proposed algorithm in the mean-square is given as

$$0 < \mu_k < \min \left\{ \frac{1}{\lambda_{\max}(\mathcal{M}_k^{-1} \mathcal{N}_k)}, \frac{1}{\lambda_{\max}(\mathcal{H}_k)} \right\} \quad (45)$$

where

$$\mathcal{H}_k = \begin{bmatrix} \frac{1}{2} \mathcal{M}_k - \frac{1}{2} \mathcal{N}_k \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \quad (46)$$

4. Simulation Results

In order to illustrate the performance of the proposed algorithm, in this section we present the simulation results for a network with $N = 10$ nodes. We consider three different models for the input vectors including

- The synthesized circular complex-valued signal which is generated at every node by a AR(1) model as ²

$$x_{k,i}(t) = 0.5x_{k,i}(t-1) + q_{k,i} \quad (47)$$

where $q_{k,i}$ is complex valued doubly white Gaussian noise with unit variance.

- The synthesized noncircular complex-valued signal which is generated at every node by a ARMA model as

$$x_{k,i}(t) = 0.5x_{k,i}(t-1) + 2q_{k,i} + 0.5q_{k,i}^* + q_{k,i-1} + 0.9q_{k,i-1}^* \quad (48)$$

- The noncircular chaotic Ikeda map signal which is given by

$$\begin{aligned} a(t) &= 1 + 0.9(a(t-1) \cos(r_k(t-1)) - b(t-1) \sin(r_k(t-1))); \\ b(t) &= 0.9(a(t-1) \sin(r_k(t-1)) + b(t-1) \cos(r_k(t-1))); \end{aligned} \quad (49)$$

where $x_r(t) = \Re\{x_{k,i}(t)\}$, $b(t) = \Im\{x_{k,i}(t)\}$, $x_{k,i}(t) = a(t) + jb(t)$ and

$$r_k(t) = 0.4 - (6/(1 + a(t)^2 + b(t)^2));$$

Fig. 2 shows the scatter plots of the synthesized circular complex-valued signal, synthesized noncircular complex-valued signal and the nonlinear and noncircular chaotic Ikeda map signal. We also consider the measurement model (1) with $L = 4$ and $\mathbf{h}^o = \mathbf{g}^o = [1 \ 1 \ 1 \ 1]^T$. The measurement noise profile $\sigma_{v,k}^2$ is plotted in Fig. 3. We set $\mu = 0.2$ for the proposed algorithm, however, to compare the MSD of non-cooperative scheme with the proposed algorithm, we need to replace μ with $N\mu$ in (11). The MSD learning curves for non-cooperative scheme (11) and the proposed algorithm have been plotted in Fig. 4. All the graphs were produced by averaging 100 independent trials. For the proposed algorithm, we have averaged MSD over all of the nodes, i.e.

$$\text{MSD} = \frac{1}{N} \sum_{k=1}^N \text{MSD}_k \quad (50)$$

²Note that $\mathbf{x}_{k,i} = [x_{k,i}(1), x_{k,i}(2), \dots, x_{k,i}(L)]$

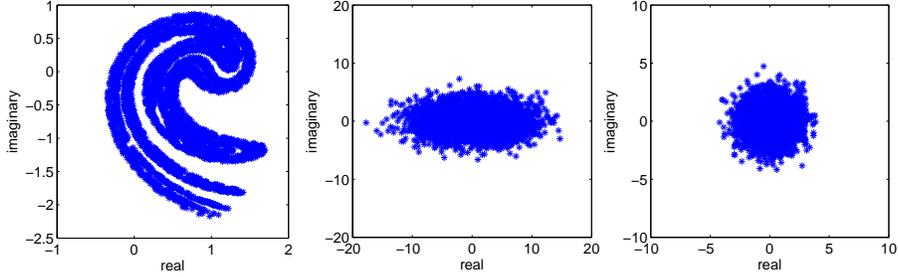


Figure 2: Geometric view of the synthesized circular complex-valued signal (left), synthesized noncircular complex-valued signal (middle) and the nonlinear and noncircular chaotic Ikeda map signal (right).

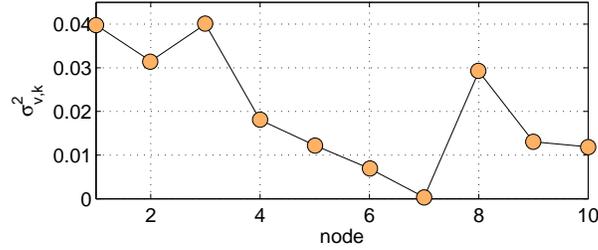


Figure 3: The measurement noise profile $\sigma_{v,k}^2$.

We observe that the proposed algorithm provides smaller MSD than the non-cooperative solution for all kind of the test signals.

In the next experiment we compare the theoretical and simulation result values, which are provided by the proposed algorithm at individual nodes for the synthesized noncircular complex-valued signal. The steady-state values are obtained by averaging over 100 independent trials and over 50 time samples after convergence. Fig. 5 shows the numerical results for both simulation and the theoretical expression. We can observe that the simulation results match well the theoretical expressions. Fig. 6 shows the steady-state MSD of proposed algorithm for different values of T . We observe that increasing T leads to a large steady-state MSD, while at the same time improves the convergence speed.

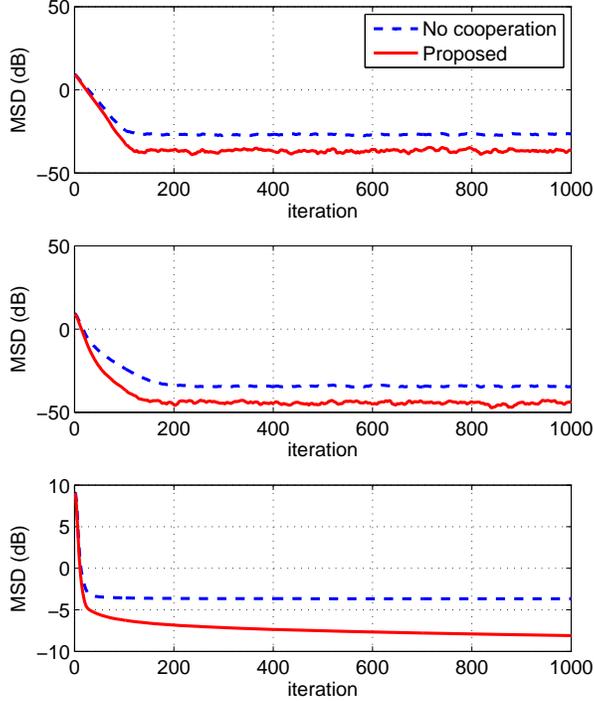


Figure 4: The MSD learning curves for non-cooperative scheme and the proposed algorithm for different signals: circular complex-valued (top), synthesized noncircular complex-valued signal (middle) and the nonlinear and noncircular chaotic Ikeda map signal (bottom).

5. Conclusion

In this paper we proposed an adaptive estimation algorithm for in-network processing of complex signals over distributed networks. In the proposed algorithm (incAAPA), nodes cooperate to exploit the spatio-temporal diversity; while the affine projection learning rules enables the network (algorithm) to process proper and improper signals. We extracted closed-form expressions that show how the proposed algorithm performs in the steady-state. We further have derived the required conditions for mean-square stability. Simulation results showed validity of the theoretical results and the good performance of the

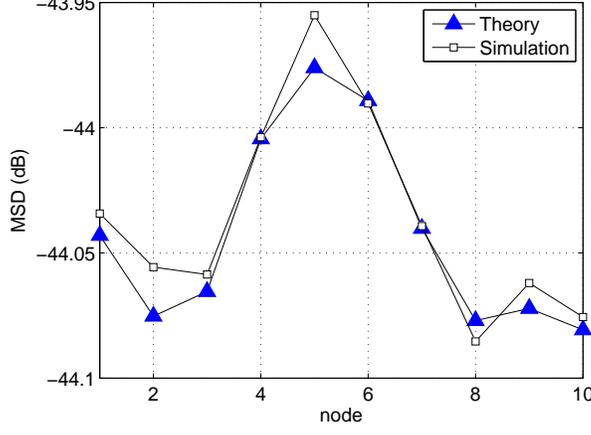


Figure 5: The theoretical and simulation result values provided by the proposed at different nodes.

proposed algorithm.

Appendix A

If we equate the weighted energies of both sides of (25), we arrive the following space-time version of the weighted energy conservation relation for IncAAPA as:

$$\|\tilde{\mathbf{w}}_{k,i}\|_{\Sigma_k}^2 + \mathbf{e}_{a,k}^{\Sigma_k H} \mathbf{F}_{k,i}^{-1} \mathbf{e}_{a,k}^{\Sigma_k} = \|\tilde{\mathbf{w}}_{k-1,i}\|_{\Sigma_k}^2 + \mathbf{e}_{p,k}^{\Sigma_k} \mathbf{F}_{k,i}^{-1} \mathbf{e}_{p,k}^{\Sigma_k} \quad (51)$$

Substituting (20) into (51) and rearranging the result, we obtain

$$\|\tilde{\mathbf{w}}_{k,i}\|_{\Sigma_k}^2 = \|\tilde{\mathbf{w}}_{k-1,i}\|_{\Sigma_k}^2 - \mu_k \mathbf{e}_{a,k}^{\Sigma_k H} \mathbf{B}_{k,i} \mathbf{e}_{k,i} - \mu_k \mathbf{e}_{k,i}^H \mathbf{B}_{k,i} \mathbf{e}_{a,k}^{\Sigma_k} + \mu_k^2 \mathbf{e}_{k,i}^H \mathbf{B}_{k,i} \mathbf{F}_{k,i} \mathbf{B}_{k,i} \mathbf{e}_{k,i} \quad (52)$$

By using the error signal $\mathbf{e}_{k,i} = \mathbf{U}_{k,i}^H \tilde{\mathbf{w}}_{k-1,i} + \mathbf{v}_{k,i}$ we have

$$\begin{aligned} \|\tilde{\mathbf{w}}_{k,i}\|_{\Sigma_k}^2 &= \|\tilde{\mathbf{w}}_{k-1,i}\|_{\Sigma_k}^2 - \mu_k \tilde{\mathbf{w}}_{k-1,i}^H \Sigma_k \mathbf{U}_{k,i} \mathbf{B}_{k,i} (\mathbf{U}_{k,i}^H \tilde{\mathbf{w}}_{k-1,i} + \mathbf{v}_{k,i}) \\ &\quad - \mu_k (\tilde{\mathbf{w}}_{k-1,i}^H \mathbf{U}_{k,i} + \mathbf{v}_{k,i}^H) \mathbf{B}_{k,i} \mathbf{U}_{k,i}^H \Sigma_k \tilde{\mathbf{w}}_{k-1,i} \\ &\quad + \mu_k^2 (\tilde{\mathbf{w}}_{k-1,i}^H \mathbf{U}_{k,i} + \mathbf{v}_{k,i}^H) \mathbf{B}_{k,i} \mathbf{F}_{k,i} \mathbf{B}_{k,i} (\mathbf{U}_{k,i}^H \tilde{\mathbf{w}}_{k-1,i} + \mathbf{v}_{k,i}) \end{aligned} \quad (53)$$

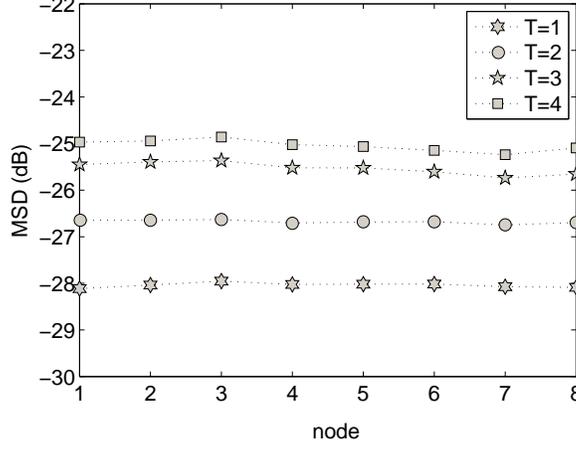


Figure 6: The steady-state MSD values of proposed algorithm at every node k for different values of T .

Taking expectations of both sides of (53) and applying the Assumptions. 1 and 2 we obtain

$$\mathbb{E}[\|\tilde{\mathbf{w}}_{k,i}\|_{\Sigma_k}^2] = \mathbb{E}[\|\tilde{\mathbf{w}}_{k-1,i}\|_{\Sigma'_k}^2] + \mu_k^2 \mathbb{E}[\mathbf{v}_{k,i}^H \mathbf{B}_{k,i} \mathbf{F}_{k,i} \mathbf{B}_{k,i} \mathbf{v}_{k,i}] \quad (54)$$

where

$$\Sigma'_k = \Sigma_k - \mu_k \mathbb{E}[\Sigma_k \mathbf{U}_{k,i} \mathbf{B}_{k,i} \mathbf{U}_{k,i}^H] - \mu_k \mathbb{E}[\mathbf{U}_{k,i} \mathbf{B}_{k,i} \mathbf{U}_{k,i}^H \Sigma_k] + \mu_k^2 \mathbb{E}[\mathbf{U}_{k,i} \mathbf{B}_{k,i} \mathbf{U}_{k,i}^H \Sigma_k \mathbf{U}_{k,i} \mathbf{B}_{k,i} \mathbf{U}_{k,i}^H] \quad (55)$$

Then, (54) and (55) can be rewritten as

$$\mathbb{E}[\|\tilde{\mathbf{w}}_{k,i}\|_{\Sigma_k}^2] = \mathbb{E}[\|\tilde{\mathbf{w}}_{k-1,i}\|_{\Sigma'_k}^2] + \mu_k^2 \mathbb{E}[\mathbf{v}_{k,i}^H \mathbf{D}_{k,i} \Sigma_k \mathbf{D}_{k,i} \mathbf{v}_{k,i}] \quad (56)$$

$$\Sigma'_k = \Sigma_k - \mu_k \Sigma_k \mathbb{E}[\mathbf{D}_{k,i}] \mu_k \mathbb{E}[\mathbf{D}_{k,i} \Sigma_k] + \mu_k^2 \mathbb{E}[\mathbf{D}_{k,i} \Sigma_k \mathbf{D}_{k,i}] \quad (57)$$

and the proof is complete.

References

References

- [1] S. Barbarossa, S. Sardellitti, P. D. Lorenzo, Distributed detection and estimation in wireless sensor networks, CoRR abs/1307.1448.

- [2] J.-J. Xiao, A. Ribeiro, Z.-Q. Luo, G. Giannakis, Distributed compression-estimation using wireless sensor networks, *Signal Processing Magazine, IEEE* 23 (4) (2006) 27–41.
- [3] I. Schizas, A. Ribeiro, G. Giannakis, Consensus in ad hoc wsns with noisy links :part i: Distributed estimation of deterministic signals, *Signal Processing, IEEE Transactions on* 56 (1) (2008) 350–364.
- [4] I. Schizas, G. Giannakis, S. Roumeliotis, A. Ribeiro, Consensus in ad hoc wsns with noisy links: Part ii: Distributed estimation and smoothing of random signals, *Signal Processing, IEEE Transactions on* 56 (4) (2008) 1650–1666.
- [5] A. H. Sayed, S.-Y. Tu, J. Chen, X. Zhao, Z. Towfic, Diffusion strategies for adaptation and learning over networks, *IEEE Signal Processing Magazine* 30 (3) (2013) 155–171.
- [6] A. Sayed, Adaptive networks, *Proceedings of the IEEE* 102 (4) (2014) 460–497.
- [7] A. H. Sayed, C. G. Lopes, Distributed recursive least-squares strategies over adaptive networks, in: *Proc. Asilomar Conf. Signals, Systems, Computers, Monterey, CA, 2006*, pp. 233–237.
- [8] C. G. Lopes, A. H. Sayed, Incremental adaptive strategies over distributed networks, *IEEE Trans. Signal Processing* 55 (8) (2007) 4064–4077.
- [9] N. Takahashi, I. Yamada, Incremental adaptive filtering over distributed networks using parallel projection onto hyperslabs, *IEICE Technical Report* 108 (2008) 17–22.
- [10] L. Li, J. A. Chambers, A new incremental affine projection-based adaptive algorithm for distributed networks, *Signal Processing* 88 (10) (2008) 2599 – 2603.

- [11] L. Li, J. A. Chambers, C. G. Lopes, A. H. Sayed, Distributed estimation over an adaptive incremental network based on the affine projection algorithm, *IEEE Trans. Signal Process.* 58 (1) (2010) 151–164.
- [12] C. G. Lopes, A. H. Sayed, Randomized incremental protocols over adaptive networks, in: *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Dallas, TX, 2010, pp. 3514–3517.
- [13] A. Rastegarnia, M. A. Tinati, A. Khalili, Performance analysis of quantized incremental LMS algorithm for distributed adaptive estimation, *Signal Processing* 90 (8) (2010) 2621 – 2627.
- [14] F. Cattivelli, A. H. Sayed, Analysis of spatial and incremental LMS processing for distributed estimation, *IEEE Trans. on Signal Process.* 59 (4) (2011) 1465–1480.
- [15] M. Saeed, A. U. H. Sheikh, A new lms strategy for sparse estimation in adaptive networks, in: *Personal Indoor and Mobile Radio Communications (PIMRC)*, 2012 IEEE 23rd International Symposium on, 2012, pp. 1722–1733.
- [16] M. S. E. Abadi, A.-R. Danaee, Low computational complexity family of affine projection algorithms over adaptive distributed incremental networks, *AEU - International Journal of Electronics and Communications* 68 (2) (2014) 97 – 110.
- [17] N. Bogdanovic, J. Plata-Chaves, K. Berberidis, Distributed incremental-based lms for node-specific parameter estimation over adaptive networks, in: *Acoustics, Speech and Signal Processing (ICASSP)*, 2013 IEEE International Conference on, 2013, pp. 5425–5429.
- [18] Y. Liu, W. K. Tang, Enhanced incremental LMS with norm constraints for distributed in-network estimation, *Signal Processing* 94 (0) (2014) 373 – 385.

- [19] N. Bogdanovic, J. Plata-Chaves, K. Berberidis, Distributed incremental-based lms for node-specific parameter estimation over adaptive networks, in: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, 2013, pp. 5425–5429.
- [20] C. G. Lopes, A. H. Sayed, Diffusion least-mean squares over adaptive networks: Formulation and performance analysis, *IEEE Trans. on Signal Process.* 56 (7) (2008) 3122–3136.
- [21] F. S. Cattivelli, C. G. Lopes, A. H. Sayed, Diffusion recursive least-squares for distributed estimation over adaptive networks, *IEEE Trans. on Signal Process.* 56 (5) (2008) 1865–1877.
- [22] F. S. Cattivelli, A. H. Sayed, Multilevel diffusion adaptive networks, in: Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing, Taipei, Taiwan, 2009.
- [23] N. Takahashi, I. Yamada, A. H. Sayed, Diffusion least-mean-squares with adaptive combiners, in: Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing, Taipei, Taiwan, 2009, pp. 2845–2848.
- [24] F. Cattivelli, A. Sayed, Diffusion lms strategies for distributed estimation, *Signal Processing, IEEE Transactions on* 58 (3) (2010) 1035–1048.
- [25] O. Gharehshiran, V. Krishnamurthy, G. Yin, Distributed energy-aware diffusion least mean squares: Game-theoretic learning, *Selected Topics in Signal Processing, IEEE Journal of* 7 (5) (2013) 821–836.
- [26] P. Di Lorenzo, S. Barbarossa, Distributed least mean squares strategies for sparsity-aware estimation over gaussian markov random fields, in: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on, 2014, pp. 5472–5476.
- [27] J. Chen, A. H. Sayed, Diffusion adaptation strategies for distributed optimization and learning over networks, *IEEE Trans. Signal Process.* 60 (8) (2012) 4289–4305.

- [28] C. Li, P. Shen, Y. Liu, Z. Zhang, Diffusion information theoretic learning for distributed estimation over network, *Signal Processing, IEEE Transactions on* 61 (16) (2013) 4011–4024.
- [29] R. Arablouei, S. Werner, Y.-F. Huang, K. Dogancay, Distributed least mean-square estimation with partial diffusion, *Signal Processing, IEEE Transactions on* 62 (2) (2014) 472–484.
- [30] F. Wen, Diffusion lmp algorithm with adaptive variable power, *Electronics Letters* 50 (5) (2014) 374–376.
- [31] S. Baqi, A. Zerguine, M. Bin Saeed, Diffusion normalized least mean squares over wireless sensor networks, in: *Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International*, 2013, pp. 1454–1457.
- [32] A. H. Sayed, *Adaptive Filters*, Wiley, NJ, 2008.
- [33] Y. Xia, C. C. Took, D. P. Mandic, An augmented affine projection algorithm for the filtering of noncircular complex signals, *Signal Processing* 90 (6) (2010) 1788 – 1799.
- [34] J. Benesty, P. Duhamel, Y. Grenier, A multichannel affine projection algorithm with applications to multichannel acoustic echo cancellation, *Signal Processing Letters, IEEE* 3 (2) (1996) 35–37.
- [35] R. Arablouei, K. Doanay, Affine projection algorithm with selective projections, *Signal Processing* 92 (9) (2012) 2253 – 2263.
- [36] D. Mandic, V. S. L. Goh, *Complex Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models*, Wiley Publishing, 2009.
- [37] T. Adali, P. Schreier, L. Scharf, Complex-valued signal processing: The proper way to deal with impropriety, *Signal Processing, IEEE Transactions on* 59 (11) (2011) 5101–5125.

- [38] D. Mandic, S. Javidi, G. Souretis, V. Goh, Why a complex valued solution for a real domain problem, in: Machine Learning for Signal Processing, 2007 IEEE Workshop on, 2007, pp. 384–389.
- [39] X.-L. Li, T. Adali, Complex-valued linear and widely linear filtering using mse and gaussian entropy, Signal Processing, IEEE Transactions on 60 (11) (2012) 5672–5684.
- [40] S. Kanna, S. Talebi, D. Mandic, Diffusion widely linear adaptive estimation of system frequency in distributed power grids, in: Energy Conference (ENERGYCON), 2014 IEEE International, 2014, pp. 772–778.
- [41] Y. Xia, D. Mandic, A. Sayed, An adaptive diffusion augmented clms algorithm for distributed filtering of noncircular complex signals, Signal Processing Letters, IEEE 18 (11) (2011) 659–662.
- [42] D. Mandic, S. Javidi, S. Goh, A. Kuh, K. Aihara, Complex-valued prediction of wind profile using augmented complex statistics, Renewable Energy 34 (1) (2009) 196 – 201.
- [43] F. Castoldi, M. de Campos, Application of a minimum-disturbance description to constrained adaptive filters, Signal Processing Letters, IEEE 20 (12) (2013) 1215–1218.
- [44] H.-C. Shin, A. Sayed, Mean-square performance of a family of affine projection algorithms, Signal Processing, IEEE Transactions on 52 (1) (2004) 90–102.
- [45] A. Khalili, A. Rastegarnia, and, W. M. Bazzi, Incremental augmented affine projection algorithm, Submitted.