ADAPTIVE LINK SELECTION STRATEGIES FOR DISTRIBUTED ESTIMATION IN DIFFUSION WIRELESS NETWORKS

Songcen Xu, Rodrigo C. de Lamare and H. Vincent Poor

Communications Research Group, Department of Electronics, University of York, U.K.
Dept. of Electrical Engineering, Princeton University, Princeton, NJ 08544, USA
Emails: sx520@york.ac.uk, rcdl500@york.ac.uk, poor@princeton.edu

ABSTRACT

In this work, we propose adaptive link selection strategies for distributed estimation in diffusion-type wireless networks. We develop an exhaustive search-based link selection algorithm and a sparsity-inspired link selection algorithm that can exploit the topology of networks with poor-quality links. In the exhaustive search-based algorithm, we choose the set of neighbors that results in the smallest mean square error (MSE) for a specific node. In the sparsity-inspired link selection algorithm, a convex regularization is introduced to devise a sparsity-inspired link selection algorithm. The proposed algorithms have the ability to equip diffusion-type wireless networks and to significantly improve their performance. Simulation results illustrate that the proposed algorithms have lower MSE values, a better convergence rate and significantly improve the network performance when compared with existing methods.

Index Terms— Adaptive link selection, diffusion networks, wireless sensor networks, distributed processing.

1. INTRODUCTION

Distributed strategies have become very popular for parameter estimation in wireless networks and applications such as sensor networks and smart grids [1, 2, 3, 4]. In this context, for each specific node a set of neighbor nodes collect their local information and transmit the estimates to a specific node. Several works in the literature have proposed strategies for distributed processing: incremental [1], diffusion [2], sparsity-aware [3] and consensus-based strategies [4]. With diffusion strategies [2], the neighbors for each node are fixed and the combined coefficients are pre-calculated after the network is generated. This approach may not provide an optimized estimation performance for each specified node because there are links that are more severely affected by noise or fading. Moreover, when the number of neighbor nodes is large, each node requires a large network bandwidth and transmit power. A key problem with the strategies reported so far in the literature is that they do not exploit the topology of wireless networks and the knowledge about the poor links to improve the performance of estimation algorithms.

In order to optimize the performance of distributed estimation techniques in wireless networks and minimize the mean-square error (MSE) associated with the estimates, we propose two adaptive link selection algorithms. The proposed techniques exploit the knowledge about the poor links and the topology of the network to select a subset of links that results in an improved estimation performance. The first approach consists of an exhaustive search-based link selection (ESLS) algorithm, whereas the second technique is based on a sparsity–inspired link selection (SILS) algorithm. For the ESLS algorithm, we consider all possible combinations for each node with its neighbors and choose the combination associated with the smallest MSE value. For the SILS algorithm, we incorporate a reweighted zero attraction (RZA) strategy into the adaptive link selection algorithm. The RZA approach is usually employed in applications dealing with sparse systems in such a way that it shrinks the small values in the parameter vector to zero, which results in better convergence rate and steady-state performance. Unlike prior work with sparsity-aware algorithms [3, 5, 6, 7], the proposed SILS algorithm exploits the possible sparsity of the MSE associated with each of the links in a different way. Unlike existing methods that shrink the signal samples to zero, SILS shrinks to zero the links that have a poor performance. We introduce a convex penalty, i.e., an ℓ_1–norm term to adjust the combined coefficients for each node with its neighbors in order to select the neighbor nodes that yield the smallest MSE values. For a specified node, we calculate the MSE at all its neighbor nodes including the specified node itself through the previous estimate. For the node with the maximum MSE, we impose a penalty and give a reward to the node with the smallest MSE. The proposed SILS algorithm performs this process automatically. By using the SILS algorithm some nodes with unsatisfactory performance will be eliminated and some poor nodes will be taken into account when their performance improves, which means the network topology will change automatically as well. Simulation results for an application to distributed estimation illustrate that the proposed ESLS and SILS algorithms have better convergence rates and lower MSE values when compared with the existing diffusion least-mean square (LMS) strategies in [2].

This paper is organized as follows. Section 2 describes the distributed processing in the networks and the problem statement. In section 3, the proposed link selection algorithms are introduced. The numerical simulation results are provide in section 4. Finally, we conclude the paper in section 5.

Notation: We use boldface upper case letters to denote matrices and boldface lower case letters to denote vectors. We use $(\cdot)^T$ and $(\cdot)^{-1}$ denote the transpose and inverse operators respectively, and $(\cdot)^*$ for conjugate transposition.

2. DISTRIBUTED PROCESSING IN WIRELESS NETWORKS AND PROBLEM STATEMENT

We consider a diffusion–type wireless network with N nodes which have limited processing capabilities. At every time instant i, each node k takes a scalar measurement $d_k^{(i)}$ according to

$$d_k^{(i)} = \boldsymbol{\omega}_0^* \boldsymbol{x}_k^{(i)} + n_k^{(i)}, \quad i = 1, 2, \ldots, N, \tag{1}$$

where $\boldsymbol{x}_k^{(i)}$ is the $M \times 1$ input signal vector, $n_k^{(i)}$ is the noise sample at each node with zero mean and variance $\sigma_{n,k}^2$. Through (1), we can see that the measurements for all nodes are related to an unknown vector $\boldsymbol{\omega}_0$. Fig.1 shows an example for a diffusion–type wireless network with 20 nodes. The aim for a diffusion–type network is to compute an estimate of $\boldsymbol{\omega}_0$ in a distributed fashion, which can minimize the cost function

$$J_\omega(\boldsymbol{\omega}) = \mathbb{E}|d_k^{(i)} - \boldsymbol{\omega}^* \boldsymbol{x}_k^{(i)}|^2, \tag{2}$$

where $\mathbb{E}$ denotes the expectation operator. To solve this problem, one possible basic diffusion technique is the adapt–then–combine (ATC) diffusion strategy [2]

$$\begin{cases} \boldsymbol{\psi}_k^{(i)} = \boldsymbol{\omega}_k^{(i-1)} + \mu_k \boldsymbol{x}_k^{(i)} [d_k^{(i)} - \boldsymbol{\omega}_k^{(i-1)*} \boldsymbol{x}_k^{(i)}]^*, \\ \boldsymbol{\omega}_k^{(i)} = \sum\limits_{l \in \mathcal{N}_k} c_{kl} \boldsymbol{\psi}_l^{(i)}, \end{cases} \tag{3}$$

where c_{kl} is the combination coefficient, which is calculated under the Metropolis rules

$$\begin{cases} c_{kl} = \frac{1}{max(n_k, n_l)}, & \text{if } k \neq l \text{ are linked} \\ c_{kl} = 0, & \text{for } k \text{ and } l \text{ not linked} \\ c_{kk} = 1 - \sum\limits_{l \in \mathcal{N}_k / k} c_{kl}, & \text{for } k = l \end{cases} \tag{4}$$

and should satisfy

$$\sum_l c_{kl} = 1, l \in \mathcal{N}_k \forall k. \tag{5}$$

Another combination rule named Hastings rule, which has been reported recently in the literature [8], could also be employed here to calculate the combination coefficients. For this kind of strategy, the choice of the neighbors for each node is fixed, this situation will cause a problem when some of the neighbor nodes have a poor performance, and there is no chance for the node to discard the poorly performing neighbors instead of continue to use their information. In order to solve this problem and optimize the distributed processing, we need to provide each node with the ability to select its links.

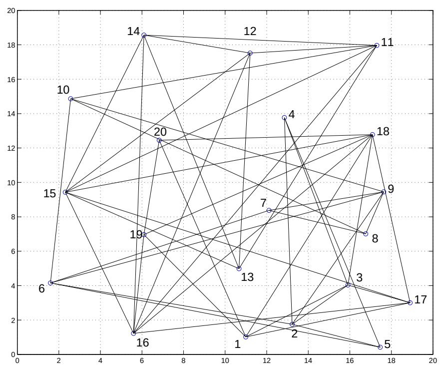

Fig. 1. Network topology with 20 nodes

3. PROPOSED ADAPTIVE LINK SELECTION ALGORITHMS

In order to optimize the distributed processing and improve the performance of the network, we propose the ESLS and the SILS algorithms. These two algorithmic strategies give the nodes the ability to choose their neighbors based on their MSE performance.

3.1. Exhaustive search-based link selection (ESLS)

The ESLS employs an exhaustive search to select the links that yield the best performance in terms of MSE. For our proposed ESLS algorithm, we employ the adaptation strategy given by

$$\boldsymbol{\psi}_k^{(i)} = \boldsymbol{\omega}_k^{(i-1)} + \mu_k \boldsymbol{x}_k^{(i)} [d_k^{(i)} - \boldsymbol{\omega}_k^{(i-1)*} \boldsymbol{x}_k^{(i)}]^*$$

Table 1. The ESLS Algorithm

Initialize: $\boldsymbol{\omega}_k^{(-1)}$=0
For each time instant i=1,2, . . . , n
For each node k=1,2, . . . , N
$\quad \boldsymbol{\psi}_k^{(i)} = \boldsymbol{\omega}_k^{(i-1)} + \mu_k \boldsymbol{x}_k^{(i)} [d_k^{(i)} - \boldsymbol{\omega}_k^{(i-1)*} \boldsymbol{x}_k^{(i)}]^*$
end
For each node k=1,2, . . . , N
$\quad$ find all possible sets of Ω_s
$\quad e_{\Omega_s}^{(i)} = d_k^{(i)} - \boldsymbol{x}_k^{(i)*} \sum_{l \in \Omega_s} c_{kl} \boldsymbol{\psi}_l^{(i)}$
$\quad \widehat{\Omega_s} = \arg\min_{\Omega_s} e_{\Omega_s}^{(i)}$
$\quad \boldsymbol{\omega}_k^{(i)} = \sum_{l \in \widehat{\Omega_s}} c_{kl} \boldsymbol{\psi}_l^{(i)}$
end
end

and redefine the diffusion step.

First, we introduce a tentative set Ω_s using a combinatorial approach described by

$$\Omega_s \triangleq C_M^j, \quad j = 1, 2, \ldots, M, \tag{6}$$

where $\{M\}$ is the total number of nodes linked to node k including node k itself. This combinatorial strategy will cover all combination choices for each node k with its neighbors.

After the tentative set Ω_s is defined, we redefine the cost function as

$$J_\psi(\boldsymbol{\psi}) \triangleq \mathbb{E}|d_k^{(i)} - \boldsymbol{x}_k^{(i)*} \boldsymbol{\psi}|^2, \tag{7}$$

where

$$\boldsymbol{\psi} \triangleq \sum_{l \in \Omega_s} c_{kl} \boldsymbol{\psi}_l^{(i)}, \tag{8}$$

and the error pattern is introduced as

$$e_{\Omega_s}^{(i)} \triangleq d_k^{(i)} - \boldsymbol{x}_k^{(i)*} \sum_{l \in \Omega_s} c_{kl} \boldsymbol{\psi}_l^{(i)}. \tag{9}$$

For each node k, the strategy that finds the best set Ω_s should solve the following optimization

$$\widehat{\Omega_s} = \arg\min_{\Omega_s} J_\psi(\boldsymbol{\psi}), \tag{10}$$

which is equivalent to minimizing the error $e_{\Omega_s}^{(i)}$. After all steps have been completed, the diffusion step in (3) can be modified as

$$\boldsymbol{\omega}_k^{(i)} = \sum_{l \in \widehat{\Omega_s}} c_{kl} \boldsymbol{\psi}_l^{(i)}. \tag{11}$$

At this point, the main steps of the ESLS algorithm have been completed. The proposed ESLS algorithm finds the set $\widehat{\Omega_s}$ that minimizes the cost function in (7) and then uses this set of nodes to obtain $\boldsymbol{\omega}_k^{(i)}$ through (11). The ESLS algorithm is summarized in Table 1. When the ESLS is implemented in networks with small and low-power sensors, the cost may become expensive, as the algorithm in (6) requires an exhaustive search and needs more communication resources to examine all the possible sets Ω_s.

3.2. Sparsity-inspired link selection (SILS)

The ESLS algorithm outlined above needs to examine all possible sets to find a solution, which might result in an unacceptable computational complexity for large networks operating in time-varying scenarios. To solve the combinatorial problem with a low complexity, we propose the SILS algorithm which is as simple as a standard diffusion LMS algorithm and is suitable for adaptive implementations and scenarios where the parameters to be estimated are slowly time-varying.

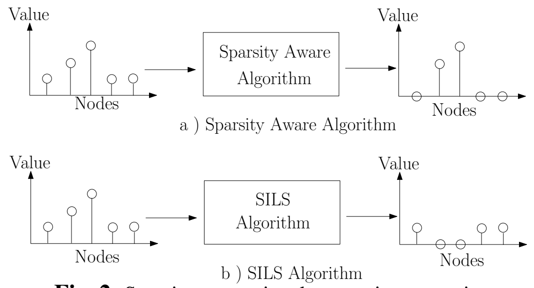

Fig. 2. Sparsity aware signal processing strategies

Fig. 2 shows the difference between the existing sparsity-aware methods and the proposed sparsity-inspired strategy. In Fig. 2 (a), we can see that, after being processed by a sparsity-aware algorithm, the nodes with small error values will be shrunk to zero. In contrast, the sparsity-inspired algorithm will shrink the nodes with large error values to zero as illustrated in Fig. 2 (b).

In the proposed SILS algorithm, we introduce the convex penalty term ℓ_1–norm into the diffusion step in (3) to perform the link selection. Different penalty terms have been considered for this task. We have adopted the reweighted zero–attracting strategy [3] into the diffusion step in (3) because this strategy has shown an excellent performance and is simple to implement.

First, we consider the following regularization function

$$f_1(e_l^{(i)})) = \sum_{l \in \mathcal{N}_k} \log(1 + \varepsilon|e_l^{(i)}|), \tag{12}$$

where the error pattern $e_l^{(i)}$ is defined as

$$e_l^{(i)} \triangleq d_k^{(i)} - \boldsymbol{x}_k^{(i)*} \boldsymbol{\psi}_l^{(i)} \qquad (l \in \mathcal{N}_k) \tag{13}$$

and ε is the shrinkage magnitude. Then, the diffusion step in (3) can be transformed into the link selection method as

$$\boldsymbol{\omega}_k^{(i)} = \sum_{l \in \mathcal{N}_k} [c_{kl} - \rho \frac{\partial f_1}{\partial e_l}(e_l^{(i)})] \boldsymbol{\psi}_l^{(i)}, \tag{14}$$

where ρ is used to control the shrinkage intensity of the algorithm and it is updated by using

$$\frac{\partial f_1(e_l^{(i)})}{\partial e_l} = \varepsilon \frac{\text{sign}(e_l^{(i)})}{1 + \varepsilon|\xi_{\min}|}. \tag{15}$$

In (15), the parameter $\xi_{\min}$ stands for the minimum value of $e_l^{(i)}$ in each group of nodes including each node k and its neighbors. The function $\text{sign}(x)$ is defined as

$$\text{sign}(x) = \begin{cases} x/|x| & x \neq 0 \\ 0 & x = 0. \end{cases} \tag{16}$$

To further simplify the expression in (14), we introduce the vector and matrix quantities required to describe the adaptation process. We first define a vector $\boldsymbol{c}$ that contains the combination coefficients for each group of nodes including node k and its neighbors as described by

$$\boldsymbol{c} \triangleq [c_{(k,l)}]_{l \in \mathcal{N}_k}. \tag{17}$$

Then, we introduce a matrix $\boldsymbol{\Psi}$ that includes all the estimated vectors which are generated after the adaptation step in (3) for each group as given by

$$\boldsymbol{\Psi} \triangleq [\boldsymbol{\psi}_l^{(i)}]_{l \in \mathcal{N}_k}. \tag{18}$$

An error vector $\boldsymbol{e}$ that contains all the error values calculated through (13) for each group is expressed by

$$\boldsymbol{e} \triangleq [e_l^{(i)}]_{l \in \mathcal{N}_k}. \tag{19}$$

To employ the sparsity–inspired approach, we have modified the vector $\boldsymbol{e}$ in the following way. The maximum value $e_l^{(i)}$ in $\boldsymbol{e}$ will be set to $|e_l^{(i)}|$, while the minimum value $e_l^{(i)}$ will be set to $-|e_l^{(i)}|$. For the remaining entries, they will be set to zero. Finally, by inserting (15)-(19) into (14), the diffusion step will be changed to

$$\begin{aligned} \boldsymbol{\omega}_k^{(i)} &= \sum_{j=1}^{\mathcal{N}_k} [\boldsymbol{c}_j - \rho \frac{\partial f_1}{\partial \boldsymbol{e}_j}(\boldsymbol{e}_j)] \boldsymbol{\Psi}_j \\ &= \sum_{j=1}^{\mathcal{N}_k} [\boldsymbol{c}_j - \rho\varepsilon \frac{\text{sign}(\boldsymbol{e}_j)}{1 + \varepsilon|\xi_{\min}|}] \boldsymbol{\Psi}_j. \end{aligned} \tag{20}$$

The proposed SILS algorithm performs link selection by the adjustment of the combination coefficients through $\boldsymbol{c}$ in (20). For the neighbor node with the largest MSE value, after our modifications for $\boldsymbol{e}$, its $e_l^{(i)}$ value in $\boldsymbol{e}$ will be a positive number which will lead to the term $\rho\varepsilon \frac{\text{sign}(\boldsymbol{e}_j)}{1+\varepsilon|\xi_{\min}|}$ in (20) being positive too. This means that the combining coefficient for this node will be reduced and the weight for this node to build the $\boldsymbol{\omega}_k^{(i)}$ is reduced too. In contrast, for the neighbor node with the minimum MSE, as its $e_l^{(i)}$ value in $\boldsymbol{e}$ will be a negative number, the term $\rho\varepsilon \frac{\text{sign}(\boldsymbol{e}_j)}{1+\varepsilon|\xi_{\min}|}$ in (20) will be negative too. As a result, the weight for this node associated with the minimum MSE to build the $\boldsymbol{\omega}_k^{(i)}$ is increased. For the remaining neighbor nodes, the $e_l^{(i)}$ value in $\boldsymbol{e}$ is zero, which means the term $\rho\varepsilon \frac{\text{sign}(\boldsymbol{e}_j)}{1+\varepsilon|\xi_{\min}|}$ in (20) is zero and there is no change for their weights to build the $\boldsymbol{\omega}_k^{(i)}$. The process for the combination coefficients in (5) is still satisfied. The SILS algorithm is summarized in Table 2.

For the ESLS and SILS algorithms, we redesign the diffusion step and employ the same adaptation procedure, which means these two algorithms have the ability to equip any diffusion–type wireless networks operating with other than the LMS algorithm. This includes the diffusion RLS strategy [9] and the diffusion conjugate gradient strategy [10].

4. SIMULATION RESULTS

In this section, we compare our proposed diffusion link selection algorithms, ESLS and SILS, with the traditional diffusion ATC algorithm [2] based on the performance of MSE. With the network topology structure in Fig. 1, we introduce N=20 nodes in this system. The length for the unknown parameter ω_0 is M=10 and it is generated randomly. The input signal is generated as $\boldsymbol{x}_k^{(i)} = [x_k^{(i)} \;\; x_k^{(i-1)} \;\; \ldots \;\; x_k^{(i-M+1)}]$ and $x_k^{(i)} = u_k^{(i)} + \alpha_k x_k^{(i-1)}$, where α_k

Table 2. The SILS Algorithm

Initialize: $\boldsymbol{\omega}_k^{(-1)}$=0
For each time instant i=1,2, . . . , n
For each node k=1,2, . . . , N

$$\boldsymbol{\psi}_k^{(i)} = \boldsymbol{\omega}_k^{(i-1)} + \mu_k \boldsymbol{x}_k^{(i)} [d_k^{(i)} - \boldsymbol{\omega}_k^{(i-1)*} \boldsymbol{x}_k^{(i)}]^*$$

end
For each node k=1,2, . . . , N

$$e_l^{(i)} = d_k^{(i)} - \boldsymbol{x}_k^{(i)*} \boldsymbol{\psi}_l^{(i)} \qquad (l \in \mathcal{N}_k)$$
$$\boldsymbol{c} = [c_{(k,l)}]_{l \in \mathcal{N}_k}$$
$$\boldsymbol{\Psi} = [\boldsymbol{\psi}_l^{(i)}]_{l \in \mathcal{N}_k}$$
$$\boldsymbol{e} = [e_l^{(i)}]_{l \in \mathcal{N}_k}$$

Find the maximum and minimum terms in $\boldsymbol{e}$
Modified $\boldsymbol{e}$ as $\boldsymbol{e}$=$[0 \cdots 0, \underbrace{|e_l^{(i)}|}_{\max}, 0 \cdots 0, \underbrace{-|e_l^{(i)}|}_{\min}, 0 \cdots 0]$

$$\xi_{\min} = \min(e_l^{(i)})$$
$$\boldsymbol{\omega}_k^{(i)} = \sum_{j=1}^{\mathcal{N}_k} [\boldsymbol{c}_j - \rho\varepsilon \frac{\mathrm{sign}(\boldsymbol{e}_j)}{1+\varepsilon|\xi_{\min}|}] \boldsymbol{\Psi}_j$$

end
end

is a correlation coefficient and $u_k^{(i)}$ is a white noise process with variance $\sigma_{u,k}^2 = 1 - |\alpha_k|^2$, to ensure the variance of $\boldsymbol{x}_k^{(i)}$ is $\sigma_{x,k}^2 = 1$. The noise samples are modeled as complex Gaussian noise with variance of $\sigma_{n,k}^2 = 0.001$. The step size for all these four algorithms is $\mu = 0.045$. For the static scenario, the sparsity parameters of the SILS algorithm are set to $\rho = 4 * 10^{-3}$ and $\varepsilon = 10$. The results are averaged over 100 independent runs. From Fig. 3, we can see that ESLS has the best performance for both the MSE and the convergence rate, and obtains a 5 dB gain over the traditional diffusion ATC algorithm. SILS is a bit worse than the ESLS, but is still significantly better than the standard diffusion ATC algorithm by about 4 dB. For the complexity and processing time, SILS is as simple as the standard diffusion ATC algorithm, while ESLS is more complex. For the time–varying scenario, the sparsity parameters of the SILS algorithm are set to $\rho = 6 * 10^{-3}$ and $\varepsilon = 10$. The unknown vector $\boldsymbol{\omega}_0$ is defined by the first–order Markov vector process

$$\boldsymbol{\omega}_0^{(i+1)} = a\boldsymbol{\omega}_0^{(i)} + \boldsymbol{z}^{(i)}, \tag{21}$$

where $\boldsymbol{z}^{(i)}$ is an independent zero–mean Gaussian vector process with variance $\sigma_z^2 = 0.01$ and $a = 0.98$. Fig. 4 shows that, for the time–varying scenario, ESLS still performs best, while SILS has the second best performance.

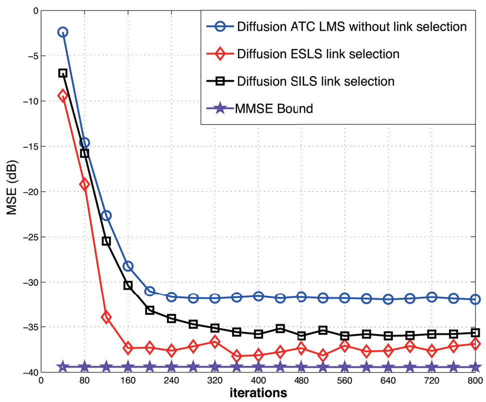

Fig. 3. Network MSE curves in a static scenario

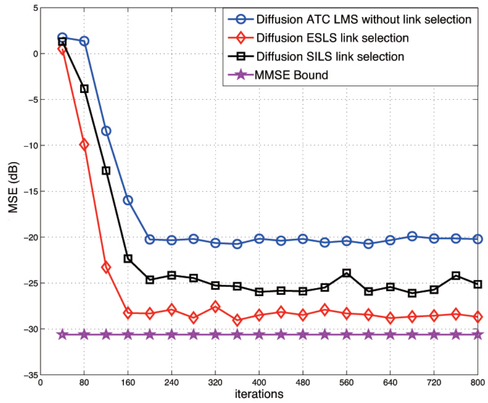

Fig. 4. Network MSE curves in a time–varying scenario

5. CONCLUSION

In this paper, two adaptive link selection strategies have been proposed for distributed estimation in diffusion–type wireless networks. The ESLS algorithm uses an exhaustive search to perform the link selection, and SILS employs a sparsity-inspired approach with the ℓ_1–norm penalization. Numerical results have shown that the two proposed algorithms achieve a better convergence rate and lower MSE values than the algorithms in [2]. These results hold also when employing other algorithms including RLS and CG techniques. The ESLS and SILS algorithms can be used in any kind of diffusion–type wireless networks and can also be applied to problems of statistical inference in smart grids.

6. REFERENCES

[1] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 48, no. 8, pp. 223–229, Aug 2007.

[2] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, July 2008.

[3] Y. Chen, Y. Gu, and A.O. Hero, "Sparse LMS for system identification," *Proc. ICASSP*, pp. 3125–3128, Taipei, May 2009.

[4] L. Xie, D.-H. Choi, S. Kar, and H. V. Poor, "Fully Distributed State Estimation for Wide-Area Monitoring Systems," *IEEE Trans. on Smart Grid*, vol. 3, no. 3, pp. 1154–1169, September 2012.

[5] R. C. de Lamare and R. Sampaio-Neto, "Adaptive Reduced-Rank Processing Based on Joint and Iterative Interpolation, Decimation and Filtering," *IEEE Trans. Signal Proc.*, vol. 57, no. 7, pp. 2503 – 2514, July 2009.

[6] R. C. de Lamare and P. S. R. Diniz, "Set-Membership Adaptive Algorithms based on Time-Varying Error Bounds for CDMA Interference Suppression," *IEEE Trans. Vehicular Technology*, vol. 58, no. 2, pp. 644–654, February 2009.

[7] L. Guo and Y. F. Huang, "Frequency-Domain Set-Membership Filtering and Its Applications," *IEEE Trans. Signal Proc.*, vol. 55, no. 4, pp. 1326–1338, April 2007.

[8] X. Zhao and A. H. Sayed, "Performance limits for distributed estimation over lms adaptive networks," *IEEE Trans. Signal Proc.*, vol. 60, no. 10, pp. 5107–5124, October 2012.

[9] F. Cattivelli, C. G. Lopes, and A. H. Sayed, "A diffusion RLS scheme for distributed estimation over adaptive networks," *Proc. IEEE SPAWC*, pp. 1–5, June 2007.

[10] S. Xu and R. C. de Lamare, "Distributed Conjugate Gradient Strategies for Distributed Estimation Over Sensor Networks," *Proc. Sensor Signal Processing for Defence 2012*, London, UK, 2012.