DISTRIBUTED PARAMETER ESTIMATION WITH SELECTIVE COOPERATION

Stefan Werner,[1] *Yih-Fang Huang,*[2] *Marcello L. R. de Campos,*[3] *and Visa Koivunen* [1]

[1]Helsinki University of Technology
Department of Signal Processing and Acoustics
Espoo, Finland
{stefan.werner, visa.koivunen}@tkk.fi

[2]University of Notre Dame
Department of Electrical Engineering
Notre Dame, USA
huang@nd.edu

[3]Federal University of Rio de Janeiro
Electrical Engineering Program
Rio de Janeiro, Brazil
campos@lps.ufrj.br

ABSTRACT

This paper proposes selective update and cooperation strategies for parameter estimation in distributed adaptive sensor networks. A set-membership filtering approach is employed that results in reduced complexity for updating parameter estimates at each network node, a significant reduction in information exchange between cooperating nodes, and an optimal strategy to obtain consensus estimates. The proposed strategies and the estimation algorithm offer a new way to explore cooperation in adaptive distributed sensor networks.

***Index Terms*—** Distributed estimation, adaptive signal processing, set-membership filtering, sensor networks

1. INTRODUCTION

In many practical problems, multiple displaced sensors are used to estimate and track an unknown common parameter, e.g., average temperature, level of water contaminants, or a target position, that characterizes the received signal at different locations. Signal collection through a distributed network of sensor nodes improves robustness of performance and reliability of the network due to redundancy and provides spatial diversity due to multiple viewing angles [1–4].

Parameter estimation in adaptive networks is typically solved by either a centralized approach or a decentralized approach. In a centralized approach, signals from all nodes in the network are collected and processed by a single *fusion center* to yield the parameter estimate. Clearly, if the network has a large number of nodes, centralized processing may be computationally prohibitive, and may require communications over longer range which leads to reduced battery life because higher transmit powers are needed to ensure required SNRs at the receiver. In decentralized estimation, spatially displaced estimators provide local estimates which are then combined to render a consensus parameter estimate. Comparing to the centralized estimation approach, decentralized estimation reduces the amount of data that each estimator needs to process and it improves the robustness of performance. However, it will require more communication bandwidth especially if cooperation among the nodes is to be considered.

This paper considers a decentralized estimation problem where the common parameter vector is estimated in a fully distributed manner [3]. This strategy becomes useful when nodes process the data without the explicit knowledge of network topology, and when the system's ability to react to the spatial characteristics of the sensor data is an important concern. In those situations, it is beneficial to consider cooperative schemes that enable neighboring nodes (or sensors in close proximity) to exchange data necessary to update local parameter estimates. Due to the spatial separation of nodes, diversity gains in estimation can be achieved. Each sensor offers a different perspective of the parameter of interest, e.g., each sensor experiences different fading impairments. Thereafter, local estimates are shared with neighboring nodes, and a local consensus estimate is obtained by combining all local estimates within the neighborhood of concern, see, e.g., [4]. However, the increased information sharing among nodes, data aggregation and fusion, may lead to increased energy consumption as well as additional bandwidth requirements [4]. Thus, it is important that the amount of data communication and local processing complexity at the nodes are kept to a minimum.

In this paper we propose diffusion strategies that feature reduced node complexity and *selective cooperation* for distributed sensor networks. The main objective is to make the entire network more energy- and bandwidth-efficient. Thus the nodes should *update their parameter estimates only when needed* and *cooperate only when such an action is "informative"*. The important point we make here is that *continual updates of parameter estimates and unnecessary/excessive cooperation may corrupt the network and lead to worse parameter estimates*. This selective cooperation strategy is particularly appealing to the following types of networks: (i) The network is comprised of a number of clustered neighboring nodes, each cluster has dedicated links between each pair of nodes; (ii) Each node in the network is able to broadcast data to a subset of network nodes (e.g., its neighbors). Naturally, in this scenario, the number of neighbors is limited by the available resources at each node. The benefits of the proposed selective cooperation strategy for the first type of networks is clear, namely, reduce the power and bandwidth required for communication between nodes. The benefits for the second type of network is primarily to ensure that the updates of parameter estimates are based on data that offer quality information.

To realize the aforementioned selective update and cooperation strategies, we propose to employ a set-membership adaptive filtering (SMAF) approach, see, e.g., [5–7], to solve the underlying estimation problems. Most, if not all, SMAF algorithms feature *sparse* data-dependent *selective* parameter updates. Specifically, these algorithms update parameter estimates only when the received data contain sufficiently fresh information, namely, *innovation*, to warrant an update of the estimate. Since received data often do not contain sufficient innovative information, the SMAF algorithm updates rather sparsely (often less than 10% of the time). As a consequence, the SMAF approach can provide us with novel strategies for centralized and decentralized estimation in adaptive networks that offer improved data processing flexibility, and reduced communication bandwidth and power requirements [8].

This work was partially funded by the Academy of Finland, Smart and Novel Radios (SMARAD) Center of Excellence, the Fulbright Nokia Scholarship, Faculty Research Program of University of Notre Dame, and by CAPES and CNPq (Brazil).

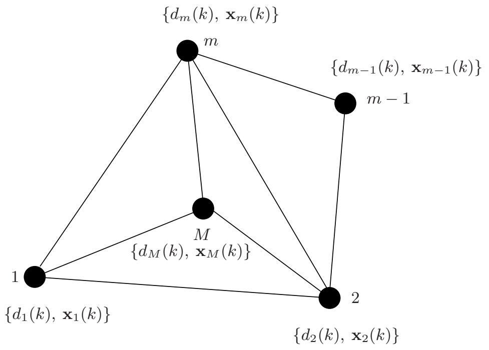
$\{d_m(k),\ \mathbf{x}_m(k)\}$
m
$\{d_{m-1}(k),\ \mathbf{x}_{m-1}(k)\}$
$m-1$
M
$\{d_M(k),\ \mathbf{x}_M(k)\}$
1
$\{d_1(k),\ \mathbf{x}_1(k)\}$
2
$\{d_2(k),\ \mathbf{x}_2(k)\}$

Fig. 1. Distributed network with M nodes.

Cooperative schemes developed based on enhanced SMAF principles will reduce not only the complexity at each sensor node, but the amount of data traffic between nodes. The optimal combining rule for the consensus estimate is non-trivial for conventional techniques. The SMAF approach, on the other hand, offers a well defined criterion on how to fuse different local estimates to arrive at an "optimum" consensus estimate.

2. BACKGROUND AND PROBLEM FORMULATION

Consider a sensor network of M nodes, and each node m has access to the input-output data pairs $\{d_m(k),\ \mathbf{x}_m(k)\}_{m=1}^{M}$, see Fig.1, where $d_m(k) \in \mathbb{R}$ and $\mathbf{x}_m(k) \in \mathbb{R}^{N\times 1}$ denote the respective desired output signal and input signal vector of a common (global) but unknown vector $\mathbf{w}_\text{o}$. Let us define the neighborhood $\mathcal{N}_m$ of node m as the set of nodes linked to it, including itself [4]. We define m to be the first element of the index set $\mathcal{N}_m$. Specifically, for the neighborhood of node m in Fig. 1, $\mathcal{N}_m = \{m,\ 1,\ 2,\ m-1,\ M\}$. Each node is supposed to transmit its data pair $\{d_m(k),\ \mathbf{x}_m(k)\}$ to its neighbors. The idea is to collect the data-pairs of neighbors and use them to produce a local update $\boldsymbol{\phi}_m(k)$ according to a specific SMF algorithm $f[\cdot]$, i.e.,

$$\boldsymbol{\phi}_m(k) = f[\mathbf{w}_m(k-1),\ \mathbf{x}_l(k),\ d_l(k);\ l \in \mathcal{N}_m] \qquad (1)$$

where $\mathbf{w}_m(k-1)$ is the consensus estimate of node m at time $k-1$. The local estimate phase is followed by an estimate-diffusion phase, in which the nodes diffuse their local estimates to their neighbors, if this is considered beneficial. Upon receiving the local estimates, a consensus estimate is obtained at each node by properly combining the local estimates of the neighborhood (aggregation phase)

$$\mathbf{w}_m(k) = g[\boldsymbol{\phi}_l(k);\ l \in \mathcal{N}_m]. \qquad (2)$$

Based on (1) and (2) we make the following observations:

1. To make a local update, each node needs to transmit (or broadcast) its data pair to all neighbors generating *feedforward traffic*.
2. The complexity of the update depends on the update rule, $f[\cdot]$, employed at each node. In [4], local processing complexity is kept low by employing least-mean squares (LMS) stochastic gradient algorithms.
3. To obtain the consensus estimate, (2), each node needs to share its local update with its neighbors. Thus some *feedback traffic* will be ensued.

To reduce the node complexity (Item 2), we propose to employ an SMAF technique [5], termed SM-NLMS (set-membership normalized least-mean squares). In SM-NLMS, the adaptive filter is designed such that the output estimation error is upper bounded by a predetermined threshold. We will show in the following that employment of SMAF algorithms enables us to reduce not only the node complexity, but also the feedforward and feedback traffic (Items 1 and 3). Furthermore, a consensus estimate can be obtained in such a way that is consistent with the SMAF framework. The principles of the proposed adaptive decentralized strategy can be summarized with the following basic steps: 1) Transmission: nodes transmit their data pairs to their neighbors, 2) Estimation: nodes make a local estimate based on all available data pairs, 3) Diffusion: nodes diffuse their local estimate to their neighbors, and 4) Aggregation: nodes combine all available estimates to form a local consensus estimate.

3. SET-MEMBERSHIP NLMS DIFFUSION SCHEMES

The SM-NLMS algorithm can be considered an SMAF counter part of the conventional normalized least-mean squares (NLMS) algorithm. Notably, the SM-NLMS algorithm features a data-dependent step size and, accordingly, selective update of parameter estimates. In particular, the step-size is optimized whenever the estimate is updated. This helps SM-NLMS algorithm to obtain better quality estimates with reduced complexity and expedited convergence, when compared to the NLMS algorithm. In this section, the SM-NLMS algorithm will be the core of our proposed diffusion scheme. We herewith develop three diffusion strategies that offer different levels of performance, complexity reduction and amount of cooperation between nodes.

3.1. Diffusion SM-NLMS

In order to reduce the average complexity of the local update and the amount of feedback arising from consensus updates, we employ the SM-NLMS algorithm [5] to obtain the local parameter updates. For this purpose, define the *constraint set* at node m, $\mathcal{H}_m(k)$, as the set of all vectors $\boldsymbol{\phi}$ that make the output error, at node m and time instant k, upper bounded in magnitude. In particular,

$$\mathcal{H}_m(k) = \{\boldsymbol{\phi} \in \mathbb{R}^N : |d_m(k) - \boldsymbol{\phi}^\text{T}\mathbf{x}_m(k)| \leq \gamma\}. \qquad (3)$$

Assume that each node has access to the input-signal and desired signal pairs of its neighborhood, i.e., $\{d_l(k),\ \mathbf{x}_l(k)\}_{l\in\mathcal{N}_m}$. Using an incremental update strategy like in [4], we can directly apply the SM-NLMS algorithm for obtaining the local estimate at time instant k, $\boldsymbol{\phi}_m(k)$. That is, each data-pair in the neighborhood is used in a sequential manner for the local update. In this way the SMAF strategy is employed to discard data for which $\boldsymbol{\phi}_m(k) \in \mathcal{H}_l(k),\ l \in \mathcal{N}_m$. After having obtained the local updates, the nodes retrieve the local estimates from all their neighbors and obtain the consensus estimate according to (2), see Section 3.4. As a result of this straightforward SMAF strategy, a reduction in the computational complexity is expected at each node. Furthermore, if none of the data pairs imply innovation at a certain time instant, i.e., no update is required, diffusion of the local estimate is unnecessary. On the other hand, each node should still transmit its data pair $\{d, \mathbf{x}\}$ to its neighbors.

An alternative to the above strategy is for the neighboring nodes to share only their local estimates, but not the data pairs. This alternative strategy is referred to here as SM-NLMS (NFF) algorithm, which still obtains the consensus estimate according to (2). The SM-NLMS (NFF) is similar to the *diffusion only* strategy considered in [9] with the LMS estimation algorithm that *updates parameter estimates continually*, regardless of the benefits of such updates. The

SMAF approach offers the distinct feature of *selective* broadcasting of the local estimates. Though it may slow down convergence (since some neighbors' data pairs are not exploited in local updates), it saves power which may be more important in wireless scenarios.

3.2. Diffusion SM-NLMS with Spatial Innovation Check

The first strategy outlined above requires the transmission of data pairs to all the neighbors, namely, complete feedforward traffic, irrespective of whether or not that offers innovation. Our second strategy aims to reduce this feedforward traffic by executing a preliminary innovation check, i.e., *spatial innovation check*, at node l to decide whether or not to communicate the data pair $\{d_l(k),\ \mathbf{x}_l(k)\}$ to node m. The main idea behind the spatial innovation check is the following. To perform the spatial update in (2), node m needs to know the local estimates of its neighbors $\{\boldsymbol{\phi}_l(k)\}_{l\in\mathcal{N}_m}$. When approaching steady-state, we will have $\mathbf{w}_m(k) \approx \boldsymbol{\phi}_m(k-1)$. Therefore, a good indicator that the data pair at node l will contribute to an update at node m is when $\alpha_l(k)$ in (6) is non-zero if it is computed with the vector $\boldsymbol{\phi}_m(k-1)$. The drawback of such a strategy is that we need to store locally all the neighbors' estimates, run the checks, and then unicast the data to each neighboring node we think should benefit from the data pair. In a typical wireless scenario, broadcasting data to nearby neighbors seems more realistic and the above outlined strategy may not reduce the feedforward traffic. Therefore, we propose to communicate $\{d_l(k),\ \mathbf{x}_l(k)\}$ to node m only if $\mathbf{w}_l(k-1) \notin \mathcal{H}_l(k)$. In other words, if a data pair implies innovation at a node (resulting in a local update) it is likely to imply innovation in neighboring nodes. This approach is likely to yield similar result as the one discussed above, since near convergence we will have $\mathbf{w}_l(k-1) \approx \boldsymbol{\phi}_m(k)$. During the transient, the approximation will not be accurate. However, since the solution shall be far from steady-state, innovation check based on either of the vectors, $\mathbf{w}_l(k-1)$ or $\boldsymbol{\phi}_m(k)$, will likely result in an update.

Let us now define the *spatial innovation set* $\mathcal{N}'_m(k)$ as the set of neighbors for which the following holds true,

$$\mathcal{N}'_m(k) = \{l \in \mathcal{N}_m : \boldsymbol{\phi}_l(k-1) \notin \mathcal{H}_l(k)\}. \tag{4}$$

That is, only the nodes that belong to the spatial innovation set will broadcast data pairs, which will reduce the broadcast traffic on the forward link. The incremental update is identical to the SM-NLMS diffusion approach in the previous section, but is now carried out over the reduced number of data pairs defined by $\mathcal{N}'_m(k)$. Note that the spatial innovation set is a function of k, because its members are only the neighbors that will perform an update. The recursions of the SM-NLMS diffusion algorithm with spatial innovation check, termed SM-NLMS (SIC), presented above are given by

$$\begin{aligned}
&\text{At each node } m:\\
&\quad \boldsymbol{\phi}_m(k) = \mathbf{w}_m(k-1),\ \sigma_m^2(k) = \sigma_m^2(k-1)\\
&\quad \text{For each } l \in \mathcal{N}'_m(k):\\
&\qquad e_l(k) = d_l(k) - \boldsymbol{\phi}_m^{\mathrm{T}}(k)\mathbf{x}_l(k)\\
&\qquad \boldsymbol{\phi}_m(k) \leftarrow \boldsymbol{\phi}_m(k) + \frac{\alpha_l(k)e_l(k)}{\|\mathbf{x}_l(k)\|^2}\mathbf{x}_l(k)\\
&\qquad \sigma_m^2(k) \leftarrow \sigma_m^2(k) - \frac{\alpha_l^2(k)e_l^2(k)}{\|\mathbf{x}_l(k)\|^2}
\end{aligned} \tag{5}$$

where $\sigma_m(k)$ is the SM-NLMS associated sphere radius [5] and

$$\alpha_l(k) = \begin{cases} 1 - \gamma/|e_l(k)| & \text{if } |e_l(k)| > \gamma \\ 0 & \text{Otherwise.} \end{cases} \tag{6}$$

is a data-dependent step size. Consensus (spatial) update is performed according to (2).

3.3. Diffusion SM-NLMS with Spatial Innovation and Reduced Feedback Traffic

The strategy in the previous section aims to reduce the feedforward traffic (number of data pairs communicated between nodes). Concerning the amount of feedback traffic (i.e., the diffusion of the local estimates $\boldsymbol{\phi}_m(k)$), it is only reduced if no update is performed or, equivalently, when $\alpha_l(k) = 0, \forall\, l \in \mathcal{N}'_m$. However, if a data pair of the neighborhood is exploited for an update, the local estimate needs to be diffused to all neighbors. To reduce the feedback traffic even further, we could choose to feed back the local estimate based on a local innovation test. That is, only if a local data pair implies innovation, $\mathbf{w}_m(k-1) \notin \mathcal{H}_m(k) = \{\mathbf{w} : |d_m(k) - \mathbf{w}^{\mathrm{T}}\mathbf{x}_m(k)| \leq \gamma\}$, we continue updating with all other data pairs belonging to the spatial innovation set. The resulting algorithm is termed SM-NLMS (SIC-RFB). We can expect SM-NLMS (SIC-RFB) to slower convergence. The recursions are given by

$$\begin{aligned}
&e_m(k) = d_m(k) - \mathbf{w}_m^{\mathrm{T}}(k-1)\mathbf{x}_m(k)\\
&\text{If } |e_m(k)| > \gamma\\
&\quad \boldsymbol{\phi}_m(k) = \mathbf{w}_m(k-1),\ \sigma_m^2(k) = \sigma_m^2(k-1)\\
&\quad \text{For each } l \in \mathcal{N}'_m(k)\\
&\qquad e_l(k) = d_l(k) - \boldsymbol{\phi}_m^{\mathrm{T}}(k)\mathbf{x}_l(k)\\
&\qquad \boldsymbol{\phi}_m(k) \leftarrow \boldsymbol{\phi}_m(k) + \frac{\alpha_l(k)e_l(k)}{\|\mathbf{x}_l(k)\|^2}\mathbf{x}_l(k)\\
&\qquad \sigma_m^2(k) \leftarrow \sigma_m^2(k) - \frac{\alpha_l^2(k)e_l^2(k)}{\|\mathbf{x}_l(k)\|^2}
\end{aligned} \tag{7}$$

3.4. Consensus Estimate

There are many different ways to implement (2) to combine the local estimates, see, e.g., [4]. The most common is to simply apply a weighted average, i.e., $\mathbf{w}_m(k) = \sum_{l\in\mathcal{N}_m} a_l(k)\boldsymbol{\phi}_m(k)$. Alternatively, we can consider convex combinations, where consensus building is done pair-wise sequentially. This may be beneficial if certain nodes perform better estimation than others [4]. For the strategies presented in this paper, consensus building can be done with a convex combination that is consistent with the SMAF framework.

An important difference between the SM-NLMS and the NLMS algorithms is that, at each recursion, the SM-NLMS algorithm renders a set of estimates. Each point in the bounding sphere $S_m = \{\boldsymbol{\phi} \in \mathbb{R}^N : \|\boldsymbol{\phi} - \boldsymbol{\phi}_k\|^2 \leq \sigma_m^2\}$ is considered a *feasible solution* to the underlying estimation problem. Consider the special case of combining the local estimate at node m and that at node $l \in \mathcal{N}_m$, which are contained, respectively, in hyper-spheres S_m and S_l. To obtain a consensus estimate, $\mathbf{w}_m^*$, we need to find a sphere S^* that tightly outer bounds the intersection of S_m and S_l. A sphere S_m^* that contains this intersection is obtained by the convex combination

$$\begin{aligned}
S_m^*(k) &= \{\mathbf{w} \in \mathbb{R}^N : \|\mathbf{w} - \mathbf{w}_m(k)\|^2 \leq \sigma_m^{2*}\}\\
&= \{\mathbf{w} \in \mathbb{R}^N : (1-\lambda)\|\mathbf{w} - \boldsymbol{\phi}_m(k)\|^2\\
&\quad + \lambda\|\mathbf{w} - \boldsymbol{\phi}_l(k)\|^2 \leq (1-\lambda)\sigma_m^2 + \lambda\sigma_l^2\}
\end{aligned} \tag{8}$$

The resulting bounding sphere and its center $\mathbf{w}_m^*$, which is taken as the point estimate, are given by

$$\begin{aligned}
&\mathbf{w}_m^*(k) = (1-\lambda)\boldsymbol{\phi}_m(k) + \lambda\boldsymbol{\phi}_l(k)\\
&\sigma_m^{2*}(k) = (1-\lambda)\sigma_m^2(k) + \lambda\sigma_l^2(k) - \lambda(1-\lambda)\|\boldsymbol{\phi}_m(k) - \boldsymbol{\phi}_l(k)\|^2
\end{aligned} \tag{9}$$

Minimizing $\sigma_m^{2*}(k)$ with respect to λ yields

$$\lambda(k) = \begin{cases} \frac{1}{2}\left[1 - \frac{\sigma_m^2(k) - \sigma_l^2(k)}{\|\phi_m(k) - \phi_l(k)\|^2}\right] & \text{if } \lambda(k) \in (0,\ 1) \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

4. SIMULATIONS

In this section we demonstrate the features of the SMAF diffusion schemes described in Section 3. For comparison purposes we also present the results obtained with non-cooperative implementations using SM-NLMS and NLMS algorithms, i.e., the parameter estimation is independently performed at each node. The network topology used in the simulations is the same as the one in [9, Fig. 6]. The network considered has $M = 12$ nodes and the adaptive filter at each node has $N = 10$ coefficients. The coefficients of the unknown plant $\mathbf{w}_o$ were generated randomly. The SNR was set to 30 dB and the additive noise at each node was AWGN with the same variance σ_n^2. The input signal at each node m was taken as colored noise generated by filtering white Gaussian noise through a filter with a pole at β_m. The values $\{\beta_m\}_{m=1}^{M}$ were taken as independent and identically distributed random variables uniformly distributed in (0,1). For the simulation experiment, we have used the following parameters: $\mu = 0.2$ for the NLMS without cooperation, $\mu = 0.24$ for the NLMS with cooperation, and $\gamma = \sqrt{5\sigma_n^2}$ for the SMAF strategies. The parameters were set in order to have fair comparison in terms of final steady-state error.

The curves shown in Fig. 2 are the results of 100 independent runs. SM-NLMS (cooperation) refers to the algorithm presented in the first part of Section 3.1. Cooperation clearly improves the convergence speed substantially. For this particular example, the consensus estimate at a node was obtained by taking the average of the parameter vectors in its neighborhood, which turns out to render comparable results as do convex combinations using hyper-spheres. Employing the spatial innovation check, namely, SM-NLMS (SIC) yields speedy convergence and reduces the amount of feedforward traffic, i.e., the number of data pairs exchanged among network nodes, see Table 1. The reduced feedback solution, i.e., SM-NLMS (SIC-RFB), converges marginally slower, as expected. However, the number of local estimates that are diffused after the local update is now considerably lower, Table 1. The *diffusion only strategy*, i.e., SM-NLMS (NFF), which shares estimates but not data pairs, slows down convergence even more. On the other hand, the number of diffused parameter vectors is still very low. Note that all SMAF strategies provide low average computational complexity when compared to a conventional approach.

5. CONCLUSIONS

This paper introduces diffusion strategies that feature selective update of parameter estimates and selective cooperation among the nodes in a distributed adaptive sensor network. The core of the proposed strategies is an SM-NLMS adaptive algorithm which offers benefits to three key components: reduction of node computation complexity, reduction of communication traffic (both feedforward and feedback), and a systematic way of obtaining consensus estimates. Simulation results showed significant improvement over conventional schemes, e.g., the NLMS algorithm, that update parameter estimates continually regardless of the benefits of such updates.

6. REFERENCES

[1] M.G. Rabbat and R.D. Nowak, "Decentralized source localization and tracking," *Proc. IEEE Int'l Conf. Acoust., Speech, and Signal Processing*, pp. 921-924, Montreal, Canada, May 2004.

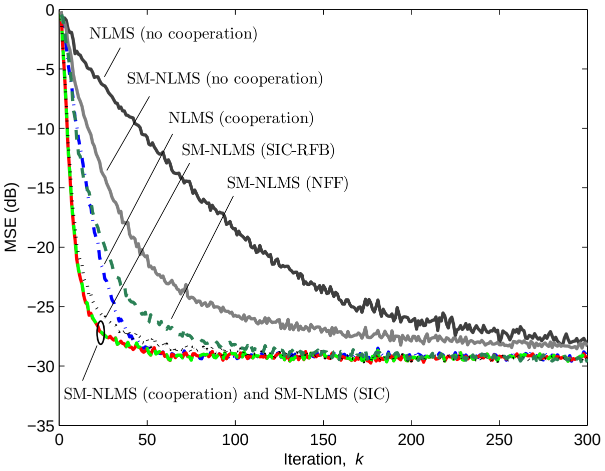

Fig. 2. MSE versus iteration k for all diffusion SM-NLMS algorithms proposed in this paper and the diffusion NLMS algorithm of [4]. Results for non-cooperative SM-NLMS and non-cooperative NLMS are included for comparison.

Table 1. Percentage of updates, feedforward (FF) and feedback (FB) traffic required by all proposed diffusion SM-NLMS algorithms.

Strategy	Updates	FF	FB
SM-NLMS (cooperation)	6.9%	100%	21.1%
SM-NLMS (SIC)	6.7%	6.9%	21.3%
SM-NLMS (SIC-RFB)	3.8%	7.6%	7.8%
SM-NLMS (NFF)	3.2%	0%	11.8%

[2] D. Spanos, R. Olfati-Saber, and R. Murray, "Distributed sensor fusion using dynamic consensus," *Proc. Int'l Fed. Automat. Contr. World Congr.*, Prague, Czech Republic, July 2005.

[3] L. Xiao, S. Boyd and S. Lall, "A space-time scheme for peer-to-peer least-squares estimation," *Proc. 5th Int'l Conf. Inform. Process. Sensor Networks*, pp. 168-176, Nashville, TN, U.S.A., April 2006.

[4] F. Cattivelli, C. G. Lopes and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Processing*, vol. 56, pp. 1865–1877, May 2008.

[5] S. Gollamudi, S. Nagaraj, S. Kapoor, and Y.-F. Huang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *IEEE Signal Processing Lett.*, vol. 5, pp. 111–114, May 1998.

[6] S. Nagaraj and S. Gollamudi and S. Kapoor and Y. F. Huang, "BEACON: An adaptive set-membership filtering technique with sparse updates," *IEEE Trans. Signal Processing*, vol. 47, pp. 2928–2941, Nov. 1999.

[7] P. S. R. Diniz and S. Werner, "Set-membership binormalized LMS data-reusing algorithms," *IEEE Trans. Signal Processing*, vol. 51, pp. 124–134, Jan. 2003.

[8] S. Werner, M. Mohammed, Y. F. Huang, and V. Koivunen, "Decentralized set-membership adaptive estimation for clustered sensor networks," *Proc. IEEE Int'l. Conf. Acoustics, Speech, Signal Processing*, pp. 3573–3576, Las Vegas, U.S.A., April 2008.

[9] C. G. Lopes and A. H. Sayed, "Diffusion least-mean-squares over adaptive networks," *Proc. IEEE Int'l. Conf. Acoustics, Speech, Signal Processing*, vol. 3, pp. 917-920, Honolulu, Hawaii, U.S.A., April 2007.