SPARSE DIFFUSION LMS FOR DISTRIBUTED ADAPTIVE ESTIMATION

†*Paolo Di Lorenzo,* †*Sergio Barbarossa and* ‡*Ali H. Sayed*

†Sapienza University of Rome, DIET, Via Eudossiana 18, 00184 Rome, Italy
‡Electrical Engineering Department, University of California, Los Angeles, CA 90095
E-mail: dilorenzo,sergio@infocom.uniroma1.it, sayed@ee.ucla.edu

ABSTRACT

The goal of this paper is to propose diffusion LMS techniques for distributed estimation over adaptive networks, which are able to exploit sparsity in the underlying system model. The approach relies on convex regularization, common in compressive sensing, to improve the performance of the diffusion strategies. We provide convergence and performance analysis of the proposed method, showing under what conditions it outperforms the unregularized diffusion version. Simulation results illustrate the advantage of the proposed filter under the sparsity assumption on the true coefficient vector.

Index Terms— Diffusion LMS, adaptive networks, compressive sensing, distributed estimation, sparse vector.

1. INTRODUCTION

We consider the problem of distributed estimation, where a set of nodes is required to collectively estimate some parameter of interest from noisy measurements by relying solely on in-network processing. Thus, consider a set of N nodes distributed over some geographic region. At every time instant i, every node k takes a scalar measurement $d_k(i)$ of some random process $\boldsymbol{d}_k(i)$ and a $1 \times M$ regression vector, $u_{k,i}$, of some random process $\boldsymbol{u}_{k,i}$ with covariance matrix $R_{u,k} = \mathbb{E}\boldsymbol{u}_{k,i}^*\boldsymbol{u}_{k,i} > 0$. The objective is for every node in the network to use the collected data $\{d_k(i), u_{k,i}\}$ to estimate some parameter vector w_0 in a distributed manner. For such purposes, several diffusion adaptation techniques were proposed and studied in [1, 2], where the nodes exchange information locally and cooperate in order to estimate w_0 without the need for a central processor. The resulting adaptive networks exploit both the time- and spatial-diversity of the data, endowing the networks with powerful learning and tracking abilities. In many scenarios, the vector parameter w_0 can be sparse, containing only a few large coefficients among many negligible ones. The prior information about the sparsity of w_0 can help improve estimation performance and this fact has already been investigated in other contexts in the literature. For example, motivated by LASSO [3] and works on compressive sensing [4, 5], several algorithms have been proposed before for sparse adaptive filtering using LMS [6], RLS [7, 8], and projection-based methods [9]. A distributed algorithm implementing LASSO over an ad-hoc network has also been proposed for sparse linear regression [10]. The basic idea of these techniques is to introduce a convex penalty, i.e., an ℓ_1-norm term, into the cost function to favor sparsity. However, none of these earlier works considered the design of *adaptive* distributed solutions that are able to process data online and exploit sparsity at the same time. Doing so would endow networks with learning abilities and would allow them to learn the sparse structure from incoming data recursively and also to track variations in the sparsity of the underlying vector.

In this work, we consider adaptive networks running diffusion techniques under general constraints enforcing sparsity. In particular, we consider two convex regularization functions. First, we consider the ℓ_1-norm, which acts as a uniform zero-attractor. Then, to improve the estimation performance, we employ a reweighted regularization to selectively promote sparsity on the zero elements of w_0, rather than uniformly on all the elements. We provide convergence analysis of the proposed methods, giving a closed form expression for the bias on the estimate due to the regularization. We also provide a mean-square analysis, showing the conditions under which the sparse diffusion filter outperforms its unregularized version in terms of steady-state performance. Interestingly enough, it turns out that, if the system model is sufficiently sparse, it is possible to tune a single parameter to achieve better performance than the standard diffusion algorithm.

The basic contribution of this paper is twofold: (a) the exploitation of sparsity for distributed estimation over adaptive networks; and (b) the derivation of the mean square properties of the sparse diffusion adaptive filter.

Notation: we use bold face letters to denote random variables and normal font letters to denote their realizations. Matrices and vectors are respectively denoted by capital and small letters.

2. SPARSE DISTRIBUTED ESTIMATION

We assume the presence of a linear measurement model where, at every time instant i, every node k takes a measurement according to the model:

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w_0 + \boldsymbol{v}_k(i) \tag{1}$$

where $\boldsymbol{v}_k(i)$ is a zero mean random variable with variance $\sigma_{v,k}^2$, independent of $\boldsymbol{u}_{k,i}$ for all k and i, and independent of $\boldsymbol{v}_j(l)$ for $l \neq k$ and $i \neq j$. Linear models as in (1) arise frequently in applications and are able to capture many cases of interest. The cooperative sparse estimation problem can be cast as the distributed minimization of the following cost function:

$$J_w(w) = \sum_{k=1}^{M} \mathbb{E}|\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w|^2 + \rho f(w) \tag{2}$$

where $\mathbb{E}(\cdot)$ denotes the expectation operator, and $f(w)$ is a convex regularization term weighted by the parameter $\rho > 0$, which is used to enforce sparsity. Proceeding as in [2], it is possible to develop several diffusion adaptation schemes for such purpose. In this paper,

This work has been supported by FREEDOM Project, Nr. ICT-248891. The work of A.H. Sayed was supported in part by NSF grants CCF-1011918, and CCF-0942936.

we consider the Adapt-then-Combine (ATC) strategy and refer to the following algorithm as the ATC-sparse diffusion (or ATC-SD) version:

$$\begin{cases}\psi_{k,i} = w_{k,i-1} + \mu_k \displaystyle\sum_{l\in\mathcal{N}_k} c_{l,k} u_{l,i}^*[d_l(i) - u_{l,i}w_{k,i-1}] \\ \qquad -\mu_k\rho\partial f(w_{k,i-1}) \qquad\qquad \text{(adaptation step)} \\ \\ w_{k,i} = \displaystyle\sum_{l\in\mathcal{N}_k} a_{l,k}\psi_{l,i} \qquad\qquad \text{(diffusion step)}\end{cases} \tag{3}$$

$k = 1, \ldots, N$, where μ_k is a positive step-size chosen by node k, the operator * denotes complex conjugate transposition, and $\partial f(w)$ is the sub-gradient of the convex function $f(w)$. The first step in (3) is an adaptation step, where the coefficients $c_{l,k}$ determine which nodes $l \in \mathcal{N}_k$ should share their measurements $\{d_l(i), u_{l,i}\}$ with node k. The second step is a diffusion step where the intermediate estimates $\psi_{l,i}$, from the neighborhood $l \in \mathcal{N}_k$, are combined through the coefficients $\{a_{l,k}\}$. The non-negative combination matrices $C = \{c_{l,k}\} \in \mathbb{R}^{M\times M}$ and $A = \{a_{l,k}\} \in \mathbb{R}^{M\times M}$ satisfy $c_{l,k} > 0, a_{l,k} > 0$ if $l \in \mathcal{N}_k$, $\mathbb{1}^T C = \mathbb{1}^T$, $C\mathbb{1} = \mathbb{1}$, and $\mathbb{1}^T A = \mathbb{1}^T$. In this paper we consider two different convex regularization terms. Motivated by LASSO [3] and work on compressive sensing [4], we first use the following ℓ_1-norm as regularization function, i.e.,

$$f_1(w) = \|w\|_1 \tag{4}$$

in the global cost function (2). This choice leads to an algorithm update in (3) where the subgradient vector is given by $\partial f_1(w) = \text{sign}(w)$, where $\text{sign}(x)$ is a component-wise function defined as

$$\text{sign}(x) = \begin{cases} x/|x| & x \neq 0 \\ 0 & x = 0 \end{cases} \tag{5}$$

This update leads to what we shall refer to as the zero-attracting (ZA) ATC diffusion algorithm. The ZA update uniformly shrinks all components of the vector, and does not distinguish between zero and non-zero elements. Since all the elements are forced toward zero uniformly, the performance would deteriorate for systems that are not sufficiently sparse. Motivated by the idea of reweighting in compressive sampling [5], we also consider the following regularization function:

$$f_2(w) = \sum_{m=1}^{M} \log(1 + \varepsilon|w_m|) \tag{6}$$

which behaves more similarly to the l_0-norm than the l_1-norm [5], thus enhancing the sparsity recovery of the algorithm. The algorithm in (3) is then updated by using

$$\partial f_2(w) = \varepsilon\frac{\text{sign}(w)}{1 + \varepsilon|w|} \tag{7}$$

leading to what we shall refer to as the reweighted zero-attracting (RZA) ATC diffusion algorithm. The update in (7) selectively shrinks only the components whose magnitudes are comparable to $1/\varepsilon$, and there is little effect on components satisfying $|w_m| \gg 1/\varepsilon$.

3. PERFORMANCE ANALYSIS

In what follows we view the estimates $w_{k,i}$ as realizations of a random process $\boldsymbol{w}_{k,i}$ and analyze the performance of the algorithm in terms of its mean square behavior. Using (3), we define the error quantities $\tilde{\boldsymbol{w}}_{k,i} = w_0 - \boldsymbol{w}_{k,i}$, $\tilde{\boldsymbol{\psi}}_{k,i} = w_0 - \boldsymbol{\psi}_{k,i}$, and the global vectors:

$$\boldsymbol{w}_i = \begin{bmatrix}\boldsymbol{w}_{1,i}\\ \vdots \\ \boldsymbol{w}_{N,i}\end{bmatrix}, \quad \tilde{\boldsymbol{w}}_i = \begin{bmatrix}\tilde{\boldsymbol{w}}_{1,i}\\ \vdots \\ \tilde{\boldsymbol{w}}_{N,i}\end{bmatrix}, \quad \tilde{\boldsymbol{\psi}}_i = \begin{bmatrix}\tilde{\boldsymbol{\psi}}_{1,i}\\ \vdots \\ \tilde{\boldsymbol{\psi}}_{N,i}\end{bmatrix} \tag{8}$$

We also introduce the diagonal matrix

$$\mathcal{M} = \text{diag}\{\mu_1 I_M, \ldots, \mu_N I_M\} \tag{9}$$

and the extended weighting matrices

$$\mathcal{C} = C \otimes I_M, \qquad \mathcal{A} = A \otimes I_M \tag{10}$$

where $\otimes$ denotes the Kronecker product operation. We further introduce the following random quantities:

$$\begin{aligned}\boldsymbol{D}_i &= \text{diag}\left\{\sum_{l=1}^{N} c_{l,1}\boldsymbol{u}_{l,i}^*\boldsymbol{u}_{l,i}, \ldots, \sum_{l=1}^{N} c_{l,N}\boldsymbol{u}_{l,i}^*\boldsymbol{u}_{l,i}\right\} &(11)\\ \boldsymbol{g}_i &= \mathcal{C}^T \text{col}\{\boldsymbol{u}_{1,i}^*\boldsymbol{v}_1(i), \ldots, \boldsymbol{u}_{N,i}^*\boldsymbol{v}_N(i)\} &(12)\end{aligned}$$

Then, we can write (3) in compact form as

$$\begin{aligned}\tilde{\boldsymbol{\psi}}_i &= \tilde{\boldsymbol{w}}_{i-1} - \mathcal{M}[\boldsymbol{D}_i\tilde{\boldsymbol{w}}_{i-1} + \boldsymbol{g}_i] + \rho\mathcal{M}\partial f(\boldsymbol{w}_{i-1})\\ \tilde{\boldsymbol{w}}_i &= \mathcal{A}^T\tilde{\boldsymbol{\psi}}_i \end{aligned} \tag{13}$$

where $\partial f(\boldsymbol{w}_{i-1}) = \text{col}[\partial f(\boldsymbol{w}_{1,i-1}), \ldots, \partial f(\boldsymbol{w}_{N,i-1})]$, or, equivalently,

$$\boxed{\tilde{\boldsymbol{w}}_i = \mathcal{A}^T[I - \mathcal{M}\boldsymbol{D}_i]\tilde{\boldsymbol{w}}_{i-1} - \mathcal{A}^T\mathcal{M}\boldsymbol{g}_i + \rho\mathcal{A}^T\mathcal{M}\partial f(\boldsymbol{w}_{i-1})} \tag{14}$$

3.1. Mean stability

Assuming all regressors $\boldsymbol{u}_{k,i}$ are spatially and temporally independent and taking the expectation of (14), we get

$$\mathbb{E}\tilde{\boldsymbol{w}}_i = \mathcal{A}^T[I - \mathcal{M}\mathbb{E}\boldsymbol{D}_i]\mathbb{E}\tilde{\boldsymbol{w}}_{i-1} + \rho\mathcal{A}^T\mathcal{M}\mathbb{E}\partial f(\boldsymbol{w}_{i-1}) \tag{15}$$

Since the subgradient vector $\partial f(\boldsymbol{w}_{i-1})$ has bounded entries, the algorithm (14) converges in the mean if the matrix $\mathcal{A}^T[I - \mathcal{M}\mathbb{E}\boldsymbol{D}_i]$ is a stable matrix. Since the entries on the columns of $\mathcal{A}^T$ add up to one, and since $\mathcal{M}$ is diagonal, we can show that the previous condition holds if the matrix $I - \mathcal{M}\mathcal{D}$ is stable, where $\mathcal{D} = \mathbb{E}\boldsymbol{D}_i$. Using (11) we conclude that the algorithm converges in the mean for any step-size satisfying:

$$0 < \mu_k < \frac{2}{\lambda_{\max}\left(\sum_{l=1}^{N} c_{l,k}R_{u,l}\right)} \qquad k = 1, \ldots, N \tag{16}$$

where $\lambda_{\max}(X)$ denotes the maximum eigenvalue of a Hermitian matrix X. Furthermore, taking the limit of equation (15) as $i \to \infty$, we get

$$\lim_{i\to\infty}\mathbb{E}\boldsymbol{w}_i = w_0 - \rho\mathcal{B}\lim_{i\to\infty}\mathbb{E}\partial f(\boldsymbol{w}_i) \tag{17}$$

where $\mathcal{B} = \left[I - \mathcal{A}^T\left[I - \mathcal{M}\mathcal{D}\right]\right]^{-1}\mathcal{A}^T\mathcal{M}$. Thus, the estimate $\boldsymbol{w}_i$ is asymptotically biased; moreover, the smaller the value of ρ, the smaller the bias.

3.2. Mean-Square Performance

Following the energy conservation framework of [1,2], we can evaluate the mean of a square weighted norm of $\tilde{\boldsymbol{w}}_i$, obtaining:

$$\mathbb{E}\|\tilde{\boldsymbol{w}}_i\|^2_{\Sigma} = \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^2_{\Sigma'} + \mathbb{E}[\boldsymbol{g}_i^* \mathcal{M}\mathcal{A}\Sigma\mathcal{A}^T\mathcal{M}\boldsymbol{g}_i] + \phi_i(\rho) \qquad (18)$$

where Σ is a Hermitian nonnegative-definite matrix that we are free to choose, and

$$\begin{aligned} \Sigma' &= \mathbb{E}(I - \boldsymbol{D}_i\mathcal{M})^T\mathcal{A}\Sigma\mathcal{A}^T(I - \mathcal{M}\boldsymbol{D}_i) && (19) \\ \phi_i(\rho) &= \rho\beta_i\left(\rho - \frac{\alpha_i}{\beta_i}\right) && (20) \end{aligned}$$

where

$$\begin{aligned} \beta_i &= \mathbb{E}\|\partial f(\boldsymbol{w}_{i-1})\|^2_{\mathcal{M}\mathcal{A}\Sigma\mathcal{A}^T\mathcal{M}} \geq 0 && (21) \\ \alpha_i &= -2\mathbb{E}\partial f(\boldsymbol{w}_{i-1})^T\mathcal{M}\mathcal{A}\Sigma\mathcal{A}^T\left[I - \mathcal{M}\mathcal{D}\right]\tilde{\boldsymbol{w}}_{i-1} && (22) \end{aligned}$$

Moreover, setting

$$G = \mathbb{E}[\boldsymbol{g}_i\boldsymbol{g}_i^*] = \mathcal{C}^T\mathrm{diag}\{\sigma^2_{v,1}R_{u,1}, \ldots, \sigma^2_{v,N}R_{u,N}\}\mathcal{C} \qquad (23)$$

we can rewrite (18) in the form

$$\mathbb{E}\|\tilde{\boldsymbol{w}}_i\|^2_{\Sigma} = \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|^2_{\Sigma'} + \mathrm{Tr}[\Sigma\mathcal{A}^T\mathcal{M}G\mathcal{M}\mathcal{A}] + \phi_i(\rho) \qquad (24)$$

where $\mathrm{Tr}(\cdot)$ denotes the trace operator. Let $\sigma = \mathrm{vec}(\Sigma)$ denote the vector that is obtained by stacking the columns of Σ on top of each other. Using the Kronecker product property $\mathrm{vec}(U\Sigma V) = (V^T \otimes U)\mathrm{vec}(\Sigma)$, we can vectorize Σ' in (19) as $\sigma' = \mathrm{vec}(\Sigma') = F\sigma$, where the matrix F is given by

$$\begin{aligned} F &= (I \otimes I)\{I - I \otimes (\mathcal{D}\mathcal{M}) - (\mathcal{D}\mathcal{M}) \otimes I \\ &+ \mathbb{E}(\boldsymbol{D}_i\mathcal{M}) \otimes (\boldsymbol{D}_i\mathcal{M})\}(\mathcal{A} \otimes \mathcal{A}). && (25) \end{aligned}$$

Then, using the property $\mathrm{Tr}(\Sigma X) = \mathrm{vec}(X^T)^T\sigma$ and taking the limit as $i \to \infty$ (assuming the step-sizes are small enough to ensure convergence to steady-state), we deduce from (24) that:

$$\lim_{i\to\infty}\mathbb{E}\|\tilde{\boldsymbol{w}}_i\|^2_{\Sigma-\Sigma'} = [\mathrm{vec}(\mathcal{A}^T\mathcal{M}G\mathcal{M}\mathcal{A})]^T\sigma + \rho\beta_\infty\left(\rho - \frac{\alpha_\infty}{\beta_\infty}\right)$$

where $\alpha_\infty = \lim_{i\to\infty}\alpha_i$ and $\beta_\infty = \lim_{i\to\infty}\beta_i$. The network steady-state mean square deviation (MSD) is given by:

$$\mathrm{MSD}_{net} = \lim_{i\to\infty}\frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 \qquad (26)$$

Then, if the step sizes $\{\mu_k\}$ are small enough so that the matrix $(I - F)$ is invertible, and choosing $\sigma = (I - F)^{-1}\mathrm{vec}(I \otimes I)$, the network MSD is given by:

$$\begin{aligned} \mathrm{MSD}_{net} &= \frac{1}{N}[\mathrm{vec}(\mathcal{A}^T\mathcal{M}G^T\mathcal{M}\mathcal{A})]^T(I - F)^{-1}\mathrm{vec}(I \otimes I) \\ &+ \frac{1}{N}\rho\beta_\infty\left(\rho - \frac{\alpha_\infty}{\beta_\infty}\right) && (27) \end{aligned}$$

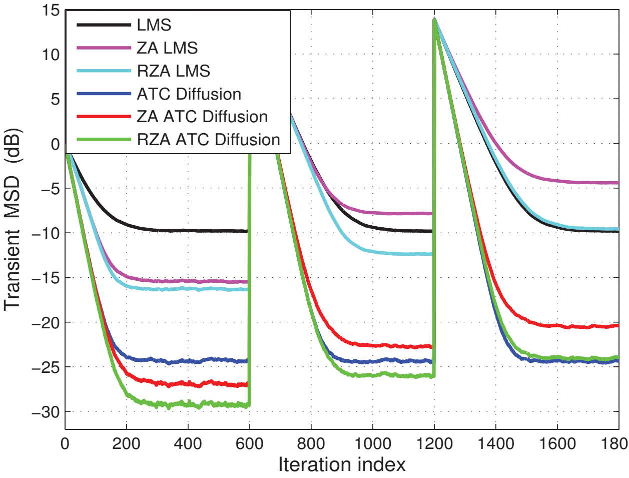

Fig. 1. Transient network MSD for the non-cooperative approaches LMS, ZA-LMS [6], RZA-LMS [6], and the diffusion techniques ATC [2], ZA-ATC (eq.(3)-(4)), RZA-ATC (eq.(3)-(6)).

The first term on the right-hand side of (27) is the network MSD of the standard diffusion algorithm (compare with (48) in [2]), whereas the second term is due to the regularization. Then, if

$$\alpha_\infty > 0 \quad \text{and} \quad 0 < \rho < \frac{\alpha_\infty}{\beta_\infty} \qquad (28)$$

the ATC-SD algorithm would perform better than the standard diffusion [2]. Let us examine the interpretation of the condition $\alpha_\infty > 0$, where α_i is given by (22), relating this condition to the sparsity of the vector w_0. Indeed, since $f(\cdot)$ is a convex regularization function, it holds that $f(x + y) - f(x) \geq \partial f(x)^T y$. Then, choosing $x = \boldsymbol{w}_i$ and $y = B_\Sigma(w_0 - \boldsymbol{w}_i)$, where $B_\Sigma = 2\mathcal{M}\mathcal{A}\Sigma\mathcal{A}^T\left[I - \mathcal{M}\mathcal{D}\right]$, the first condition in (28) can be recast as

$$\alpha_\infty \geq \lim_{i\to\infty}\mathbb{E}[f(\boldsymbol{w}_i) - f(\boldsymbol{w}_i + B_\Sigma(w_0 - \boldsymbol{w}_i))] > 0 \qquad (29)$$

If the step-sizes are sufficiently small, we can approximate $B_\Sigma \backsimeq 2\mathcal{M}\mathcal{A}\Sigma\mathcal{A}^T$, neglecting the second term that depends on μ^2. Then, we have $\bar{\boldsymbol{w}}_i = \boldsymbol{w}_i + B_\Sigma(w_0 - \boldsymbol{w}_i) \backsimeq \boldsymbol{w}_i - 2\mathcal{M}\mathcal{A}\Sigma\mathcal{A}^T(\boldsymbol{w}_i - w_0)$. This expression can be interpreted as a gradient descent update minimizing the function $\|\boldsymbol{w}_i - w_0\|^2_{\mathcal{A}\Sigma\mathcal{A}^T}$, yielding $\bar{\boldsymbol{w}}_i$ closer to w_0 than $\boldsymbol{w}_i$. As a consequence, if w_0 is sparse, $\bar{\boldsymbol{w}}_i$ will be more sparse than $\boldsymbol{w}_i$. Thus, since this is true for all i, considering the expectation and taking the limit as $i \to \infty$, the condition in (29) will likely be true. Then, by selecting properly the sparsity coefficient ρ, the ATC-SD algorithm will have better MSD than the standard ATC diffusion algorithm. On the other hand, if w_0 is not sparse, condition (29) in general would not be true, thus leading the ATC-SD algorithm to perform worse than standard ATC diffusion.

4. NUMERICAL RESULTS

In this section, we provide some numerical examples to illustrate the performance of the ATC-SD algorithm. We consider a connected network composed of 20 nodes. The regressors have size $M = 50$ and are zero-mean white Gaussian distributed with covariance matrices $R_{u,k} = \sigma_u^2 I$, with $\sigma_u^2 = 0.1$, for all k. The background white noise power is set to $\sigma_v^2 = 0.01$. The first example aims to show the tracking and steady-state performance for the ATC-SD algorithm. In Fig. 1, we report the learning curves in terms of network MSD of 6 different adaptive filters: ATC diffusion LMS [2], ZA-ATC (eq.(3)-(4)) and RZA-ATC diffusion (eq.(3)-(6)), and the corresponding non cooperative approaches from [6]. The simulations use a value of $\mu = 0.2$ and the results are averaged over 100 independent experiments. The sparsity parameters are set equal to

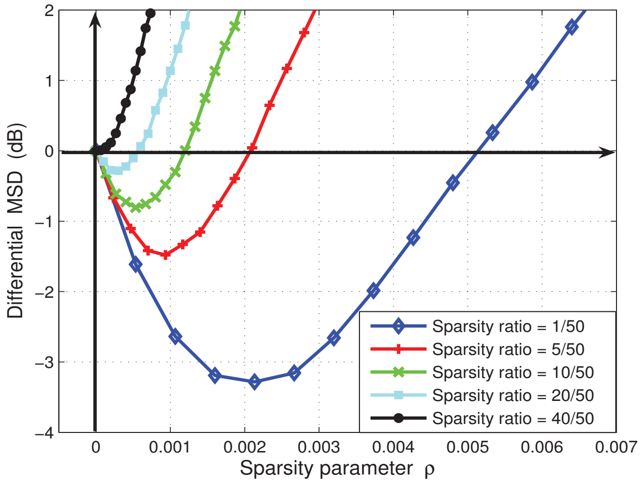

Fig. 2. Differential MSD versus sparsity parameter ρ for ZA-ATC Diffusion LMS, for different degrees of system sparsity.

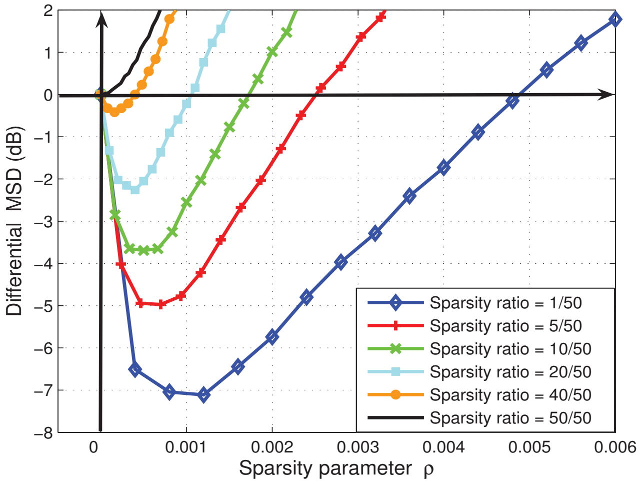

Fig. 3. Differential MSD versus sparsity parameter ρ for RZA-ATC Diffusion LMS, for different degrees of system sparsity.

$\rho_{LMS} = 5\times10^{-3}$ for the non cooperative approaches, $\rho_{ZA} = 10^{-3}$ for ZA-ATC, $\rho_{RZA} = 0.25 \times 10^{-3}$ for RZA-ATC, and $\epsilon = 10$. In this simulation, we consider diffusion algorithms without measurement exchange, i.e., $C = I$, and a combination matrix A that simply averages the estimates from the neighborhood, hence, such that $a_{l,k} = 1/|\mathcal{N}_k|$ for all l. Initially, only one of the 50 elements of w_0 is set equal to one while the others are equal to zero, making the system very sparse. After 600 iterations, 25 elements are randomly selected and set equal to 1, making the system have a sparsity ratio of $25/50$. After 1200 iterations, all the elements are set equal to 1, leaving a completely non-sparse system. As we see from Fig. 1, when the system is very sparse both ZA-ATC and RZA-ATC yield better steady-state performance than standard diffusion. The RZA-ATC outperforms ZA-ATC thanks to the reweighted regularization. When the vector w_0 is only half sparse, the performance of ZA-ATC deteriorates, performing worse than standard diffusion, while RZA-ATC has the best performance among the three diffusion filters. When the system is completely non-sparse, the RZA-ATC still performs comparably to the standard diffusion filter. Finally, we can also notice the gain of the diffusion schemes with respect to the non-cooperative approaches from [6]. To quantify the effect of the sparsity parameter ρ on the performance of the ATC-SD filters, we consider two additional examples. In Fig. 2, we show the behavior of the difference (in dB) between the network MSD of ATC-ZA and standard diffusion, versus ρ, for different sparsity degrees of w_0. The results are averaged over 100 independent experiments and over 100 samples after convergence. As we can see from Fig. 2, reducing the sparsity of w_0, the interval of ρ values that yield a gain for ATC-ZA with respect to standard diffusion becomes smaller, until it reduces to zero when the system is not sparse enough. In Fig. 3, we repeat the same experiment considering the ATC-RZA algorithm. As wee can see, ATC-RZA gives better performance than ZA-ATC and yields a performance loss with respect to standard diffusion, for any ρ, only when the vector w_0 is completely non-sparse.

5. CONCLUSION

In this paper we proposed a class of diffusion LMS strategies, regularized by convex sparsifying penalties, for distributed estimation over adaptive networks. Convergence and mean square analysis of the sparse adaptive diffusion filter show under what conditions we have dominance of the proposed method with respect to its unregularized counterpart in terms of steady-state performance. Two different penalty functions have been employed, the ℓ_1-norm, which uniformly attracts to zero all the vector elements, and a reweighted function, which selectively shrinks only the elements with small magnitude. Numerical results show the potential benefits of using such strategies. Other penalty functions can also be useful. Adaptive diffusion strategies for the distributed optimization of convex cost functions are further considered in [11].

6. REFERENCES

[1] C. G. Lopes and A. H. Sayed, “Diffusion least-mean squares over adaptive networks: Formulation and performance analysis,” *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3122–3136, July 2008.

[2] F. S. Cattivelli and A. H. Sayed, “Diffusion LMS strategies for distributed estimation,” *IEEE Trans. on Sig. Proc.*, vol. 58, pp. 1035–1048, March 2010.

[3] R. Tibshirani, “Regression shrinkage and selection via the LASSO,” *J. Royal Statist. Soc B.*, vol. 58, pp. 267–288, 1996.

[4] R. Baraniuk, “Compressive sensing,”, *IEEE Signal Proc. Mag.*, vol. 25, pp. 21–30, March 2007.

[5] E.J. Candes, M. Wakin, and S. Boyd, “Enhancing sparsity by reweighted ℓ_1 minimization,” *J. Fourier Anal. Appl.*, vol. 14, pp.877–905, 2007.

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

[7] D. Angelosante, J.A. Bazerque, and G.B. Giannakis, “Online adaptive estimation of sparse signals: where RLS meets the ℓ_1-norm,” *IEEE Trans. on Sig. Proc.*, vol. 58, no. 7, pp. 3436–3447, 2010.

[8] B. Babadi, N. Kalouptsidis, and V. Tarokh, “SPARLS: The sparse RLS algorithm,” *IEEE Trans. Sig. Proc.*, vol. 58, no. 8, pp. 4013–4025, 2010.

[9] Y. Kopsinis, K. Slavakis, and S. Theodoridis, “Online sparse system identification and signal reconstruction using projections onto weighted ℓ_1 balls,” *IEEE Trans. on Sig. Proc.*, vol. 59, no. 3, pp. 936–952, 2010.

[10] G. Mateos, J. A. Bazerque, and G. B. Giannakis, “Distributed sparse linear regression,” *IEEE Trans. on Sig. Proc.*, vol 58, No. 10, pp. 5262–5276, Oct. 2010.

[11] J. Chen and A. H. Sayed, “Distributed optimization via diffusion adaptation,” *Proc. IEEE CAMSAP*, Puerto Rico, Dec. 2011.