ADAPTIVE ESTIMATION OVER DISTRIBUTED SENSOR NETWORKS WITH A HYBRID ALGORITHM

Elmira Mohyedinbonab, Omid Ghasemi, Mo Jamshidi, Yu-Fang Jin

Department of Electrical and Computer Engineering
University of Texas at San Antonio
San Antonio, TX, 78249
Email: {dgw319@my.utsa.edu, qys128@my.utsa.edu, moj@wacong.org, Yufang.jin@utsa.edu}

***Abstract* —Estimation of unknown parameters associated with a distributed sensor network using its noisy measurements has been an active research area recently. Several estimation algorithms, such as the incremental and diffusion algorithms, have been proposed to address this problem. Incremental algorithms require less communication among nodes of the networks while diffusion algorithms are more robust and require large amounts of energy for communication. In this study, we have proposed a hybrid methodology that combines incremental and diffusion algorithms based on the property of *a priori* error, where is the difference of output error and noise variance of each sensor. The proposed network started with an incremental communication scheme and switched to diffusion scheme to complete the rest of the estimation. Simulation results showed that the proposed algorithm largely improved the convergence rate as well as the estimation accuracy.**

Keywords-component; Distributed Estimation, incremental algorithm, diffusion algoorithm, cooperation, sensor network.

I. INTRODUCTION

Sensor networks play an important role in many applications nowadays such as transportation, precision agriculture, environmental monitoring, factory instrumentation, etc. In these applications, the exact location of the signal to be collected is generally unknown and sensors are distributed close to the phenomena of interest [1]. Each sensor located at a node in the network provides local measurements on time-variant parameters and then communicates with other nodes in a collaborative manner. The application of such cooperative processing, demands adaptive processing capabilities of the nodes to deal with all spatial and temporal changes in the environment and in the network [2]. The general goal is to accurately estimate the unknown parameter associated with the network by sharing information between neighboring nodes in the network. To be specific, let us consider a sensor network in which sensors are distributed over a geographic area and we need to estimate the average temperature based on the measurements from the sensors. In such a case, measurements from a given sensor *k* can reach an individual decision about the temperature locally for a period of time by communicating with other sensors.

The communications between nodes can be categorized into different groups based on their cooperative manners: diffusion, and incremental modes. In the incremental mode, there is a cyclic communication path through the nodes of the network. Each node provides its measurement to a single node in its neighborhood. In diffusion mode, a node broadcasts its measurements to all nodes in its neighborhood. Correspondingly, adaptation is executed at the node of interest by accessing the data from all communicated neighboring nodes. Properly weighted estimates are then combined to obtain an estimate of the parameters. Currently, both incremental and diffusion approaches have their benefits and drawbacks [2 - 5]. The incremental approach is susceptible to communication failure between nodes but has a high speed of convergence. In contrast, the diffusion algorithm is robust and its global estimation is reached through the network at the cost of high energy consumption and extensive amount of communication.

In this study, we proposed a hybrid algorithm by combining two incremental and diffusion approaches to obtain a robust scheme which has less communication load at the beginning of the estimation process while having a fast convergence rate. In the proposed hybrid scheme, we smoothly switch from incremental to diffusion algorithm to provide a higher rate of precise estimations in a limited time. The suitable switching time is calculated based on the value of the error in the last iteration of the estimation process. When the error reaches a threshold pre-defined by the total expected error in the incremental algorithm, the diffusion algorithm takes over.

The structure of this paper is described as follows. In the following section, the estimation problem of distributed sensor networks is described. The third section introduces the hybrid estimation algorithm. In addition, the convergence-rate of the diffusion algorithm is derived and compared to that of the incremental algorithm. In the fourth section, simulation results for the proposed hybrid algorithm are presented and compared to the results obtained with incremental and diffusion approaches. We presented our conclusion of this study at the end of the paper.

II. PROBLEM FORMULATION

Consider a network consisting of N nodes. At every time instant *i* sensors at each node measure two sets of correlated data, $\{d_k(i), u_{k,i}\}$, where $d_k(i)$ is a scalar measurement corresponding to the realization of random process $\boldsymbol{d}_k(i)$ at node *k*, and $u_{k,i}$ is an M dimensional row regression vector corresponding to a realization of the random process $\boldsymbol{u}_{k,i}$ with a mean of zero at node k. Assuming there is a linear relationship among $d_k(i), u_{k,i}$, the objective is to estimate the optimal solution ω^0 that satisfies equation (1).

$$d_k(i) = u_{k,i}\,\omega^0 + v_k(i) \tag{1}$$

where $v_k(i)$ is a zero-mean Gaussian white noise independent of $u_{k,i}$ with the variance denoted as $\sigma^2_{v_k}$. The objective is to estimate the unknown vector ω that minimizes the global cost function $J^{global}(\omega)$as:

$$J^{global}(\omega) \triangleq \sum_{k=1}^{N} E\,|\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\omega|^2 \tag{2}$$

where E denotes the expectation operator. In [2], the optimal solution to equation (2) satisfies equation (3).

$$R_{du} = R_u\,\omega^0 \tag{3}$$

where $R_{du} = \sum_{k=1}^{N} R_{du,k}$ and $R_u = \sum_{k=1}^{N} R_{u,k}$. The Cross-correlation and covariance matrices $R_{du,k}$ and $R_{u,k}$ are given as $R_{u,k} = E\,(\boldsymbol{u}^*_{k,i}\,\boldsymbol{u}_{k,i}) > 0, R_{du,k} = E\,(\boldsymbol{d}_k(i)\boldsymbol{u}^*_{k,i})$. "*" denotes complex conjugate-transposition operator.

In [3], starting with gradient-descent implementation, a recursive method has been developed to estimate ω in an iterative manner as shown in equation (4), which:

$$\omega_i = \omega_{i-1} + \mu \textstyle\sum_{k=1}^{N}(R_{du,k} - R_{u,k}\,\omega_{i-1}) \tag{4}$$

To compute cross-correlation and covariance matrices $R_{du,k}$and $R_{u,k}$ respectively, it is required to access all time realization information $\{d_k(i), u_{k,i}\}$at each node, otherwise the global estimation ω^0 cannot be computed. To address this problem, a distributed solution has been developed in [3] by replacing$R_{du,k}$and $R_{u,k}$ with their instantaneous approximations using LMS method as shown in equation (5).

$$R_{u,k} \approx u^*_{k,i}u_{k,i}\,, \qquad R_{du,k} \approx d_k(i)u^*_{k,i} \tag{5}$$

The derived LMS method is formulated as equation (6).

$$\omega_i = \omega_{i-1} + \mu \textstyle\sum_{k=1}^{N} u^*_{k,i}(d_k(i) - u_{k,i}\omega_{i-1}) \tag{6}$$

In equation (6), ω_i is defined globally and is updated at each time instant i. On the other hand, the estimation can be obtained locally at each node and gradually approaches to a global estimation of ω. In such a case, estimate of ω will be dependent on the communication scheme among different nodes. Two prominent distributed solutions that locally estimate the unknown parameter $\omega$are incremental and diffusion algorithm, which have been proposed in [2-7].

Lopez and Sayed proposed a Distributed Incremental Least Mean Square (DILMS) algorithm in [4] to estimate the parameter ω as shown in equation (7).

$$\begin{cases} \psi_0^{(i)} \leftarrow \omega_{i-1} \\ \psi_k^{(i)} = \psi_{k-1}^{(i)} + \mu_k u^*_{k,i}[d_k - u_{k,i}\,\psi_{k-1}^{(i)}] \\ \omega_i \leftarrow \psi_N^{(i)} \end{cases} \tag{7}$$

where $k = 1, 2, \ldots, N$, and $\mu_k$is the step size.

In the diffusion algorithm, each node communicates with a set of nodes in its neighborhood. Consequently, there is more communication among nodes. Also, more information is exchanged among nodes. As an example, node k exchanges its measurement $\{d_k(i), u_{k,i}\}$ and the intermediate estimates $\psi_k^{(i)}$ within nodes in its neighborhood. There are two approaches to compute estimates in diffusion algorithm known as adapt-then-combine (ATC) and combine-then-adapt (CTA) solution [3, 5]. Implementations of ATC algorithm for N nodes can be written as equation (8).

$$\begin{cases} \psi_k^{(i)} = \omega_k^{(i-1)} + \mu_k \sum_{l\,\in\,N_k} c_{l,k}\,u^*_{l,i}[d_l(i) - u_{l,i}\,\omega_k^{(i-1)}] \\ \omega_k^{(i)} = \sum_{l\,\in\,N_k} a_{l,k}\,\psi_l^{(i)} \end{cases} \tag{8}$$

where the weighting coefficients $\{, c_{l,k}, a_{l,k}\}$ are real nonnegative values. Comprehensive details on updating rules of incremental and diffusion algorithms can be found in [2-4].

Incremental algorithm is easy to implement, it also converges fast to the optimal solution and requires less exchanges of communication among nodes. Due to the higher communication exchanges among nodes in diffusion algorithms, diffusion is more robust and stable compared to its incremental counterpart. To achieve a fast convergent estimation for a sensor network with a better EMSE performance, a more complicated algorithm is needed.

III. Proposed Algorithm

In this study, we proposed a hybrid algorithm by integrating both incremental and diffusion algorithms. This algorithm has a better performance compared to incremental and diffusion algorithm. Different benchmarks are investigated to analyze the performance of the proposed hybrid algorithm.

A. *Hybrid Algorithm based on Error analysis*

To quantify the performance of different algorithms, we used the defined local error signals in [4] by equations (9), (10), (11) and (12),

$$\tilde{\psi}_k^{(i)} = \omega^0 - \psi_k^{(i)} \text{ (Intermediate weight error vector)} \tag{9}$$

$$\tilde{\omega}_k^{(i)} \triangleq \omega^0 - \omega_k^{(i)} \text{ (weight error vector)} \tag{10}$$

$$e_{a,k}(i) = u_{k,i}\tilde{\psi}_{k-1}^{(i)} \text{ (\textit{a priori} error)} \tag{11}$$

$$e_k(i) = d_k(i) - u_{k,i}\,\psi_k^{(i)} \text{ (output error)} \tag{12}$$

$\tilde{\psi}_k^{(i)}$ and $\tilde{\omega}_k^{(i)}$ measure the difference between optimal solution with intermediate estimate and weight estimate at node *k*, respectively. Substituting (1) in (12), we have:

$$e_k(i) = d_k(i) - u_{k,i}\,\psi_k^{(i)} = u_{k,i}\omega^0 + v_k(i) - u_{k,i}\,\psi_k^{(i)}$$

$$e_k(i) = e_{a,k}(i) + v_k(i) \qquad (13)$$

According to equation (13), we expect the output error $e_k(i)$ to converge to $v_k(i)$ if $\psi_k^{(i)}$ approaches to ω^0, since the *a priori* error goes to zero. Further on, $E|e_k(i)|^2 = E|e_{a,k}(i)|^2 + \sigma_{v_k}^2$, therefore we need to evaluate $E|e_{a,k}(i)|^2$ to derive the $E|e_k(i)|^2$. Consequently, these quantities are defined as follows [4]:

$$\xi_k \triangleq E|e_k(\infty)|^2 = \zeta_k + \sigma_{v_k}^2 \quad \text{(MSE)} \qquad (14)$$

$$\zeta_k \triangleq E|e_{a,k}(\infty)|^2 \quad \text{(EMSE)} \qquad (15)$$

$$\eta_k \triangleq E|\tilde{\psi}_{k-1}^{(\infty)}|^2 \quad \text{(MSD)} \qquad (16)$$

Mean square error (MSE) analyses evaluate the performance of the algorithm in steady state. If the estimation converges to the optimal solution ω^0 then the ideal MSE should converge to the noise variance at each node. The excess mean square error (EMSE) is defined as the distance between MSE and the noise variance to evaluate the performance of each node. Mean-square deviation (MSD) is defined as the expected value of square of weight error at each time. MSD evaluates the parameter convergence in the estimation problem. By measuring the variance of the *a priori* error at each cycle, we can monitor the changes of error .The variance of *a priori* error at each time instant *i* can be calculated using the following expression:

$$E|e_{a,k}(i)|^2 = e_{a,k}^*(i)e_{a,k}(i) = (e_{a,k}(i))^2 \qquad (17)$$

Due to the numerous communication exchanges among nodes in diffusion algorithms, the spatial diversity in data can be beneficially utilized to achieve a better EMSE performance compared to incremental algorithms with similar convergence rates [3]. As a result, the first nodes operate incrementally followed by diffusion communication.

In the hybrid algorithm, we evaluated the measure of closeness of EMSE to its optimal value in the incremental stage. When the error in the incremental solution reaches a factor of the *a priori* error value, the rate of convergence slows down to a point where more communication can help reduce the EMSE even further.

In the case of a link failure during the incremental process, a fallback time condition is applied, meaning that if the *a priori* threshold is not met after a certain amount of time the incremental algorithm switches to the diffusion algorithm automatically.

The steady state performance of each node in the mean-square sense for the incremental algorithm is given in [4], which includes the steady state EMSE, MSE and MSD calculations. The threshold for switching from incremental to diffusion algorithm is set when EMSE at time instant *i* goes below 145 percent of the value of EMSE at steady state. Therefore error at switching time would be in the 90% confidence interval of the steady state EMSE.. That happens when the EMSE is so close to its steady-state value in the incremental algorithm and the estimation process with incremental algorithm is not as fast as before.

By exploiting the advantages of both methods, the hybrid algorithm inherits the decrease in EMSE characteristics of diffusion algorithms and the fast convergence rate of the incremental solution.

B. *Convergence rate analysis*

In this section, the speed of convergence of diffusion (ATC type) and incremental algorithms are compared initially. To simplify the derivations, we assumed statistical profiles throughout the network are similar, i.e. $R_{u,k} = R_u > 0$, $R_{du,k} = R_{du}$ and $\mu_k = \mu$. Thereby, the ATC changes to equations (18) and (19):

$$\psi_k^{(i)} = \omega_k^{(i-1)} + \mu \sum_{l \in N_k} c_{l,k}(R_{du} - R_u\omega_k^{(i-1)}) \qquad (18)$$

$$\omega_k^{(i)} = \sum_{l \in N_k} a_{l,k}\psi_l^{(i)} \qquad (19)$$

Subtracting both sides of (18) and (19) from ω^0, substituting R_{du} with equation (3) and using the equations in (9-10) results in the following expressions (20) and (21):

$$\tilde{\psi}_k^{(i)} = \tilde{\omega}_k^{(i-1)} - \mu \sum_{l \in N_k} c_{l,k}R_u\tilde{\omega}_k^{(i-1)} \qquad (20)$$

$$\tilde{\omega}_k^{(i)} = \omega^0 - \sum_{m \in N_k} a_{m,k}\,(\omega^0 - \tilde{\psi}_m^{(i)}) \qquad (21)$$

Extending equation (21) yields (22):

$$\tilde{\omega}_k^{(i)} = (\omega^0 - \sum_{m \in N_k} a_{m,k}\,\omega^0) + \sum_{m \in N_k} a_{m,k}\,\tilde{\psi}_m^{(i)} \qquad (22)$$

Where the first term of this equation is equal to zero because $\sum_{m \in N_k} a_{m,k} = 1$. This simplifies to equation (23):

$$\tilde{\omega}_k^{(i)} = \sum_{m \in N_k} a_{m,k}\,\tilde{\psi}_m^{(i)} \qquad (23)$$

Now, substituting $\tilde{\psi}_m^{(i)}$ with equation (20), and considering that $\sum_{l \in N_m} c_{l,m} = 1$, we have

$$\tilde{\omega}_k^{(i)} = \sum_{m \in N_k} a_{m,k}\left(\tilde{\omega}_m^{(i-1)} - \mu \sum_{l \in N_m} c_{l,m} R_u \tilde{\omega}_m^{(i-1)}\right)$$
$$= \sum_{m \in N_k} a_{m,k}\left(I - \mu \sum_{l \in N_m} c_{l,m} R_u\right)\tilde{\omega}_m^{(i-1)} \qquad (24)$$
$$= \sum_{m \in N_k} a_{m,k}(I - \mu R_u)\,\tilde{\omega}_m^{(i-1)}$$
$$\tilde{\omega}_k^{(i)} = (I - \mu R_u) \sum_{m \in N_k} a_{m,k}\,\tilde{\omega}_m^{(i-1)}$$

The eigen-decomposing of $R_u = U\Lambda U^*$, where $UU^* = I$ and where we define $\bar{\omega}_k^{(i)} = U^* \tilde{\omega}_k^{(i)}$, we derive (25)

$$\bar{\omega}_k^{(i)} = (I - \mu\Lambda) \sum_{m \in N_k} a_{m,k}\,\bar{\omega}_m^{(i-1)} \qquad (25)$$

In [4], the authors derived a similar equation for incremental algorithm and rate of convergence, expressed by (26)

$$\bar{\omega}^{(i)} = (I - \mu\Lambda)^N \bar{\omega}^{(i-1)} \qquad (26)$$

As a matter of fact, each node of network in the diffusion algorithm has a slower convergence rate compared to that of incremental because the term $(I - \mu\Lambda)$ decreases exponentially with a rate of N in incremental algorithm. Consequently, $\bar{\omega}$ in incremental algorithm declines faster than that in diffusion algorithm.

In first part of the this section, the error analysis showed that steady state EMSE in Diffusion algorithm is smaller than that in incremental method. In the second part, we proved the speed of convergence of incremental solution is faster than that of diffusion algorithm. Therefore, the hybrid algorithm starts with incremental estimation method to rapidly converge to the mentioned value close to the steady state EMSE in incremental algorithm, then the algorithm switches to a diffusion algorithm to reduce EMSE more with diffusion communication scheme.

IV. Simulation Results

In our simulations we considered a network of 5 nodes. Fig.1. represents our network topology with both communication patterns. Nodes communicate in a circular way during incremental solution. When the diffusion algorithm is running, only connected nodes are able to communicate and exchange information.

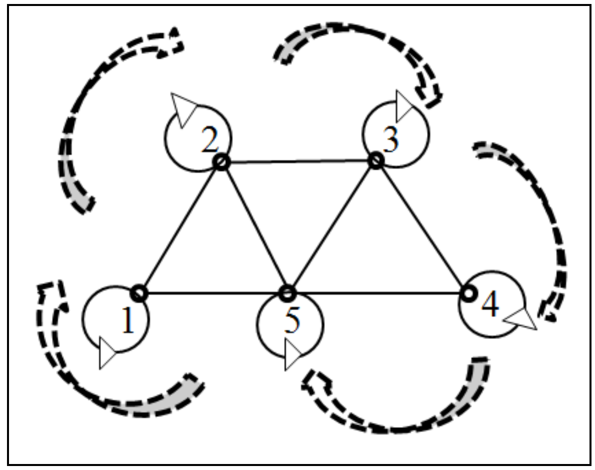

Fig.1. Network topology with five nodes. Dashed arrows represent communication path in incremental solution and solid lines illustrate that in diffusion solution.

The varying noise power levels across the nodes are shown in Fig.2. These noises are white noises with variances less than 0.002.

The system was modeled for 3,000 iterations and the results were averaged over 100 independent experiments. The step size was a constant value of 0.01.

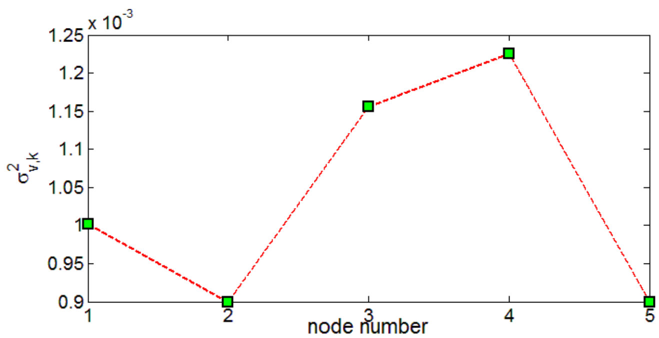

Fig. 2. Noise power profile across the network

According to [4], theoretically, EMSE in steady state for incremental algorithm with the same noise profile should be about 0.25e-5 (-55 dB). As time passes, EMSE in the incremental algorithm approaches to a very small value and the estimation process slows down. With 90% confidence interval, when EMSE in incremental solution goes below 145 percent of 0.25e-5, which is equal to 0.36e-5, the diffusion algorithm is triggered to perform the rest of the estimation to reduce the EMSE even further. By using this condition, the EMSE of a node for the proposed hybrid algorithm, along with incremental and diffusion algorithms, are shown in Fig.3.

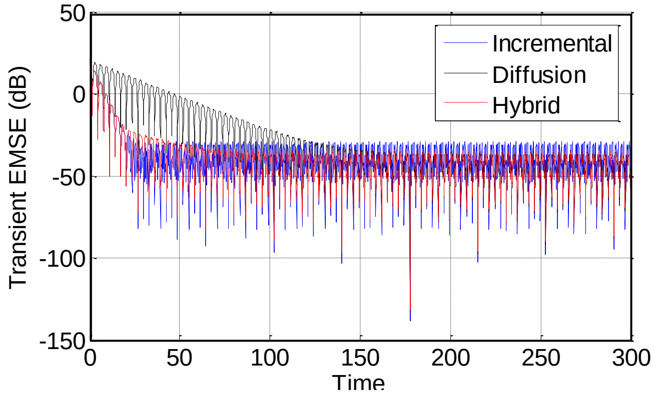

Fig. 3. Transient EMSE of one node with three algorithms

Because of the fluctuations along the estimation process, the maximum value of EMSE computed in the last 20 seconds was -29 dB for the incremental solution, -35.9 dB for the diffusion solution and -35.5 dB for the hybrid solution. As expected, the EMSE for the diffusion and hybrid algorithm are close, however the hybrid solution converges faster compared to diffusion algorithm performing individually. In the early stages of the process, all nodes cooperate in a cyclic manner and after a specified period, they change their behavior and communicate with more nodes in a diffusion manner. In Fig.4, the transient EMSE results are shown for the last 50 seconds during the steady state.

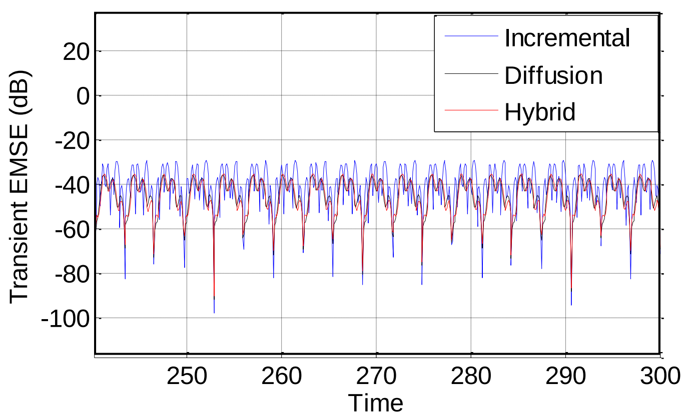

Fig. 4. Transient EMSE of one node with three algorithms in the last 50 iterations

Fig.5 illustrates the first 150 seconds of fig. 3. It compares the speed of convergence of incremental, diffusion and hybrid algorithms. As mentioned earlier, the convergence rate of incremental algorithm is faster than that in diffusion algorithm. Since hybrid algorithm starts with incremental algorithm, it has the similar convergence property as incremental one. As seen in fig. 5, the switching point of the incremental algorithm occurs at the right instant of time. Afterwards, the incremental algorithm's rate of convergence considerably slows down.

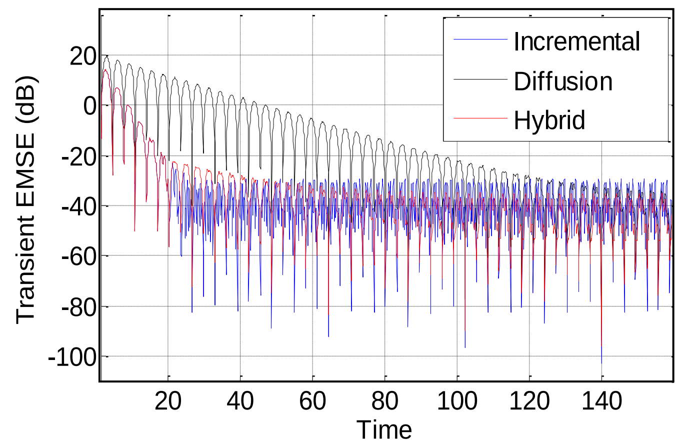

Fig. 5. Transient EMSE of one node with three algorithms in the beginning of the estimation process

To show the parameter convergence of all three algorithms, MSD of one node is demonstrated in Fig.6. As seen, the MSD for the proposed hybrid algorithm is less than the former two algorithms. This means all parameters converge to the optimal value in small amount of time with minimal error.

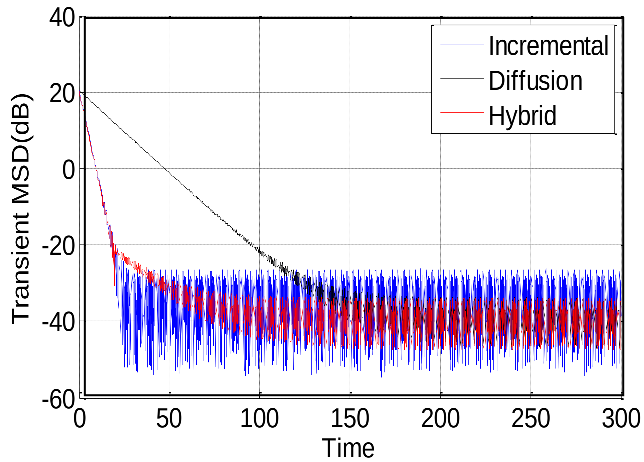

Fig. 6. Transient MSD of one node with three algorithms

The steady state performance of MSD for one node is also provided in Fig.7 for the final 30 seconds. It is clear that the MSD for the Hybrid algorithm is less than the other two algorithms. The maximum value of MSD in steady state for the incremental, diffusion and hybrid algorithms are -26.41 dB, -34.57 dB, -34.62 dB, respectively.

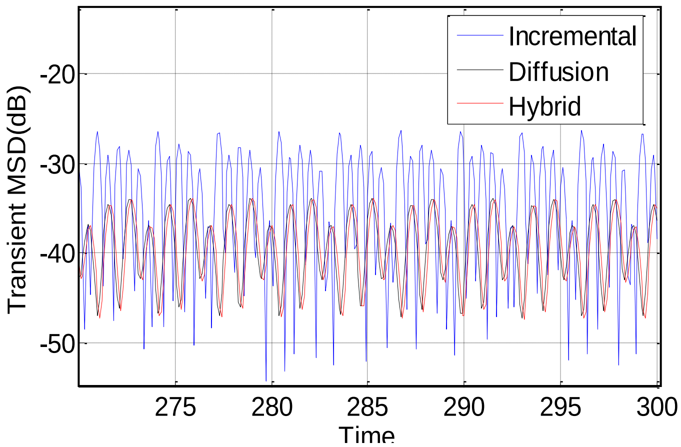

Fig. 7. Transient MSD of one node with three algorithms in steady state

When the threshold is not adjusted properly, the switching behavior is not beneficial. If the threshold value is very small, the proposed hybrid solution is similar to the incremental algorithm. On the contrary, a bigger value of threshold leads to slower convergence of the proposed hybrid algorithm. In this case, the proposed algorithm is as slow as the diffusion one. Using optimal threshold of -54.43 dB, the proposed hybrid algorithm reaches an optimal EMSE sooner than the others and has a lower EMSE at the end, compared to non-hybrid algorithm.

In Fig.8, the effects of various thresholds in hybrid algorithm are simulated.

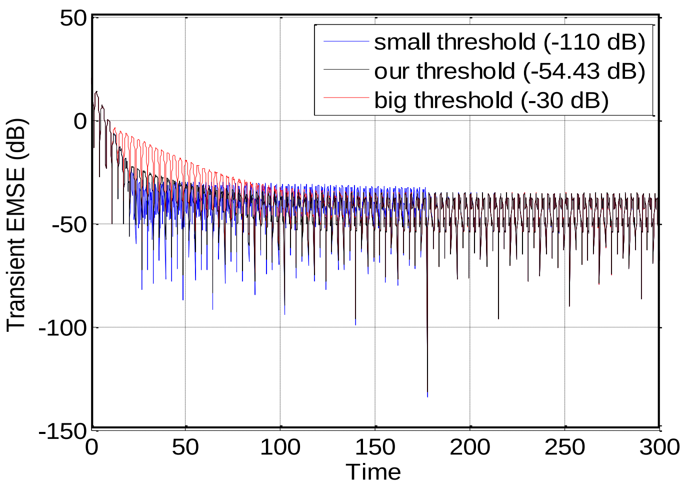

Fig. 8. Transient EMSE of one node with three different threshold with proposed algorithm

V. Conclusion

In this paper, we proposed a hybrid algorithm that employs a high convergence speed using the incremental algorithm and exceptional EMSE performance with diffusion algorithms. In our proposed hybrid algorithm, the order of switching can occur in two ways, an incremental followed by a diffusion stage or a diffusion succeeded by an incremental stage. Since we are interested in a low EMSE in steady state and a fast rate of convergence in the beginning of process, the former sequence is more appealing than the latter. This means that at first, nodes in the network communicate in an incremental way. When the *a priori* error meets a specified threshold, as discussed in this paper, diffusion algorithm is triggered to perform the rest of the estimation until it reaches the optimal EMSE. In comparison with incremental and diffusion algorithms, this method performs better with improved EMSE and high speed of convergence. Our proposed method is supported by derivations of the EMSE and rate of convergence in incremental and diffusion algorithm in great detail. The rate of convergence for the diffusion algorithm was derived, proving that each node of network in diffusion algorithms have slower convergence rates compared to that of incremental.

References

[1] D. Estrin, L. Girod, G. Pottie, M. Srivastava, Instrumenting the World with Wireless Sensor Networks, International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001), Salt Lake City, Utah, May 2001.

[2] C. G. Lopez, A. H. Sayed, Distributed processing over adaptive networks, in Proc. Adaptive Sensor Array Processing Workshop, MIT Lincoln Lab., Lexington, MA, June 2006.

[3] Cattivelli, F.S.; Sayed, A.H.; , "Diffusion LMS Strategies for Distributed Estimation," Signal Processing, IEEE Transactions on , vol.58, no.3, pp.1035-1048, March 2010.

[4] C. G. Lopez, A. H. Sayed, Incremental adaptive strategies over distributed networks, IEEE Trans. Signal Processing, vol 55, no. 8, pp. 4064-4077, August 2007.

[5] X. Zhao, A. H. Sayed, Performance limits of LMS-based adaptive networks, 2011 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3768-3771, May 2011.

[6] A. H. Sayed, C. G. Lopez, Adaptive processing over distributed networks, IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences, vol 90, no. 8, pp. 1504-10, August 2007.

[7] F. Cattivelli, A. H. Sayed, Self-organization in Bird Flight Formations using different adaptation, 2009 3rd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, pp. 49-52, December 2009.

[8] A. Khalili, M. A. Tinati, A. Rastegarnia, An adaptive Estimation Algorithm for distributed networks with noisy links, 2011 Symposium on Artificial Intelligence and Signal Processing (AISP), pp. 97-100, June 2011.