Loughborough University

This item was submitted to Loughborough's Institutional Repository (https://dspace.lboro.ac.uk/) by the author and is made available under the following Creative Commons Licence conditions.

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/

Steady-State Performance of Incremental Learning over Distributed Networks for Non-Gaussian Data

Leilei Li [*]   **Yonggang Zhang** [*]   **Jonathon A. Chambers** [*]   **Ali H. Sayed** [+]

[*] Department of Electronic and Electrical Engineering, Loughborough University
Loughborough LE11 3TU, UK, E-mail: L.Li2@lboro.ac.uk

[+] Department of Electronic Engineering, University of California
Los Angeles, USA, CA 90095-1594

Abstract: In this paper, the steady-state performance of the distributed least mean-squares (dLMS) algorithm within an incremental network is evaluated without the restriction of Gaussian distributed inputs. Computer simulations are presented to verify the derived performance expressions.

Key words: Adaptive filters, distributed estimation, energy conservation, incremental algorithm

I. INTRODUCTION

A traditional centralized solution requires a powerful central processor and large amount of communications between nodes within a network. In order to reduce significantly the amount of processing and communications, distributed solutions, only exploiting local data exchanges and communications between immediate neighboring nodes, are desirable. The applications of such distributed adaptive networks would range from sensors networks to environmental monitoring, precision agriculture, military surveillance, transportation and factory instrumentation [1, 2]. Reference [3] developed the dLMS algorithm and studied its performance under Gaussian data.

In this paper, we evaluate the steady-state performance of dLMS for non-Gaussian data using weighted spatial-temporal energy conservation arguments [3]-[6]. In particular, we derive theoretical expressions for the mean-square deviation (MSD), the excess mean-squared-error (EMSE) and the mean-squared-error (MSE). Simulation studies are presented to confirm the convergence properties of the scheme and to verify the theoretical results.

The remainder of this paper is organized as follows. A network model is constructed and the dLMS algorithm is formulated in Section II. In Section III, we evaluate the steady-state performance of dLMS and derive the theoretical expressions. Section IV illustrates the comparison of theoretical and simulated results.

Throughout the paper, the following notations are adopted: boldface small and capital letters are used for random complex vectors or scalars and matrices; normal font is employed for deterministic quantities; $(\cdot)^T$ and $(\cdot)^*$ denote the transposition and complex-conjugate transposition respectively; $|\cdot|^2$ and $\|\cdot\|^2$ denote the absolute squared operation and squared Euclidean norm operation.

II. NETWORK MODEL AND THE dLMS ALGORITHM

As formulated in [3], an N-node network is considered, where each node has a separate zero-mean random complex valued desired response d_k and a $1 \times M$ spatially distinct row input vector u_k with zero-mean and random elements. Both d_k and u_k are assumed to be jointly wide-sense stationary, and k is used to denote the node index. At each node time

realizations $\{d_k(i), u_{k,i}\}$ of $\{d_k, u_k\}$ are observed, where i indicates the discrete time index. We formulate the linear least-mean-squares estimation problem:

$$\min_w J(w) \quad \text{and} \quad J(w) = E\|d - U w\|^2 \qquad (1)$$

where the $N \times 1$ global desired response vector and $N \times M$ input matrix are respectively:

$$d = \text{col}\{d_1, d_2, \dots, d_N\} \qquad (2)$$
$$U = \text{col}\{u_1, u_2, \dots, u_N\}. \qquad (3)$$

The solution w^o satisfies the normal equations, [7]:

$$R_{du} = R_u w^o \qquad (4)$$

where $R_{du} = E U^* d$ and $R_u = E U^* U$.

Note that the cost function can be decomposed into, as [3]

$$J(w) = \sum_{k=1}^{N} J_k(w) \quad \text{and} \quad J_k(w) = |d_k - u_k w|^2 \qquad (5)$$

Using this property, references [3, 4] proposed an incremental dLMS strategy with a cyclic estimation structure, as follows:

For each time instant $i \geq 0$:
$k = 1, 2, \dots N$
$\Psi_0^{(i)} = w_{i-1}$
$\Psi_k^{(i)} = \Psi_{k-1}^{(i)} + \mu_k u_{k,i}^* (d_k(i) - u_{k,i} \Psi_{k-1}^{(i)})$
end
$w_i = \Psi_N^{(i)}$

where $\Psi_k^{(i)}$ indicates a local estimate of w^o at node k and time i, and w_i indicates the estimate at the same instant. For each time i, each node utilizes the local data $\{d_k(i), u_{k,i}\}$ and $\Psi_{k-1}^{(i)}$ received from node $k-1$ in the ring to obtain $\Psi_k^{(i)}$. At the end of this cycle, $\Psi_N^{(i)}$ is employed as both the global estimate w_i and the initial condition for the next time instant.

III. PERFORMANCE ANALYSIS

To evaluate the performance of dLMS, the following assumptions are utilized,

A1) The relationship between the unknown system vector w^o and $\{d_k(i), u_{k,i}\}$ is:

227

$$d_k(i) = u_{k,i} w^o + v_k(i) \qquad (6)$$

where $v_k(i)$ is a temporally and spatially white noise sequence with variance $\sigma_{v,k}^2$ independent of $u_{l,j}$ and $d_l(j)$ for all l, j.

A2) $u_{k,i}$ has spatial and temporal independence, namely $u_{k,i}$ is independent of $u_{l,i}$ for $k \neq l$, and $u_{k,j}$ for $i \neq j$.

Introduce the following local error signals defined as in [3] to carry out the evaluation:

$$\tilde{\boldsymbol{\Psi}}_{k-1}^{(i)} = w^o - \boldsymbol{\Psi}_{k-1}^{(i)}, \quad \tilde{\boldsymbol{\Psi}}_k^{(i)} = w^o - \boldsymbol{\Psi}_k^{(i)} \qquad (7)$$

$$e_{a,k}(i) = u_{k,i} \tilde{\boldsymbol{\Psi}}_{k-1}^{(i)}, \quad e_{p,k}(i) = u_{k,i} \tilde{\boldsymbol{\Psi}}_k^{(i)} \qquad (8)$$

Note that the output error is given by

$$e_k(i) = e_{a,k}(i) + v_k(i) \qquad (9)$$

We are interested in evaluating the following performance measures at each node k :

$$\eta_k = E\left\|\tilde{\boldsymbol{\Psi}}_{k-1}^{(i)}\right\|^2 \qquad \text{(MSD)} \qquad (10)$$

$$\varsigma_k = E\left|u_{k,i} \tilde{\boldsymbol{\Psi}}_{k-1}^{(i)}\right|^2 = E\left|e_{a,k}(i)\right|^2 \qquad \text{(EMSE)} \qquad (11)$$

$$\xi_k = \varsigma_k + \sigma_{v,k}^2 \qquad \text{(MSE).} \qquad (12)$$

Introduce further weighted error signals:

$$e_{a,k}^{\Sigma_k}(i) = u_{k,i} \Sigma_k \tilde{\boldsymbol{\Psi}}_{k-1}^{(i)}, \quad e_{p,k}^{\Sigma_k}(i) = u_{k,i} \Sigma_k \tilde{\boldsymbol{\Psi}}_k^{(i)} \qquad (13)$$

where Σ_k is a Hermitian positive-definite weighting matrix that we are free to choose at each node k. Introduce also the weighted norm notation $\|x\|_\Sigma^2 = x \Sigma x$ for a vector x and Hermitian positive-definite $\Sigma > 0$. After the same manipulations as in [3], we obtain

$$\tilde{\boldsymbol{\Psi}}_k^{(i)} + \frac{u_{k,i}^* e_{a,k}^{\Sigma_k}(i)}{\left\|u_{k,i}\right\|_{\Sigma_k}^2} = \tilde{\boldsymbol{\Psi}}_{k-1}^{(i)} + \frac{u_{k,i}^* e_{p,k}^{\Sigma_k}(i)}{\left\|u_{k,i}\right\|_{\Sigma_k}^2} \qquad (14)$$

which reveals a relationship between two neighbouring nodes. By calculating the energies of both sides of (14), a spatial-temporal energy relation is obtained as

$$\left\|\tilde{\boldsymbol{\Psi}}_k^{(i)}\right\|_{\Sigma_k}^2 + \frac{\left|e_{a,k}^{\Sigma_k}(i)\right|^2}{\left\|u_{k,i}\right\|_{\Sigma_k}^2} = \left\|\tilde{\boldsymbol{\Psi}}_{k-1}^{(i)}\right\|_{\Sigma_k}^2 + \frac{\left|e_{p,k}^{\Sigma_k}(i)\right|^2}{\left\|u_{k,i}\right\|_{\Sigma_k}^2} \qquad (15)$$

which is an exact energy relation between two adjacent nodes in space and time, and is derived without any approximations For simplicity, we drop the time index i. Applying the expectation operation to both sides of (15), we get:

$$E\left\|\tilde{\boldsymbol{\Psi}}_k\right\|_{\Sigma_k}^2 = E\left\|\tilde{\boldsymbol{\Psi}}_{k-1}\right\|_{\Sigma_k'}^2 + \mu_k^2 \sigma_{v,k}^2 E\left\|u_k\right\|_{\Sigma_k}^2 \qquad (16)$$

where Σ_k' is given by

$$\Sigma_k' = \Sigma_k + \mu_k E\left(u_k^* u_k \Sigma_k + \Sigma_k u_k^* u_k\right) + \mu_k^2 E\left(\left\|u_k\right\|_{\Sigma_k}^2 u_k^* u_k\right). \qquad (17)$$

In order to evaluate the performance of the learning algorithm, we need to examine the following three moments:

$$Eu_k^* u_k = R_{u,kl}, E\left\|u_k\right\|_{\Sigma_k}^2 = \text{Tr}\left(R_{u,k}\Sigma_k\right) \text{ and } E\left(\left\|u_k\right\|_{\Sigma_k}^2 u_k^* u_k\right). \qquad (18)$$

where $\text{Tr}(A)$ denote the trace of a matrix A.

In [3, 4], the assumption of Gaussian regressors are used to evaluate the last item in (18). In the non-Gaussian case, we proceed as in [5,6] by introducing the $M^2 \times 1$ vectors:

$$\sigma_k = \text{vec}\{\Sigma_k\} \quad \text{and} \quad \delta_k' = \text{vec}\{\Sigma_k'\} \qquad (19)$$

where we use the $\text{vec}\{\cdot\}$ notation in two ways: $\delta = \text{vec}\{\Sigma\}$ denotes an $M^2 \times 1$ column vector whose entries are formed by stacking the successive columns of an $M \times M$ matrix on top of each other, and $\Sigma = \text{vec}\{\delta\}$ indicates a matrix whose entries are recovered from δ. We exploit the following useful property for the $\text{vec}\{\cdot\}$ notation when working with Kronecker products: for any matrices $\{P, \Sigma, Q\}$ of compatible dimensions, it holds that

$$\text{vec}\{P \Sigma Q\} = \left(Q^T \otimes P\right)\text{vec}\{\Sigma\}. \qquad (20)$$

Using (20) to express some items in (17), we find

$$\text{vec}\left\{E\left(u_k^* u_k\right)\Sigma_k\right\} = \left(I \otimes R_{u,k}\right)\delta_k \qquad (21)$$

$$\text{vec}\left\{\Sigma_k E\left(u_k^* u_k\right)\right\} = \left(R_{u,k}^T \otimes I\right)\delta_k \qquad (22)$$

$$\text{vec}\left\{E\left[\left\|u_k\right\|_{\Sigma_k}^2 u_k^* u_k\right]\right\} = E\left[\left(u_k^* u_k\right)^T \otimes \left(u_k^* u_k\right)\right]\delta_k. \qquad (23)$$

Applying the $\text{vec}\{\cdot\}$ operation to both sides of (17), a linear relation between the corresponding vectors $\{\delta_k', \delta_k\}$ is obtained, namely

$$\delta_k' = F_k \delta_k \qquad (24)$$

where F_k is an $M^2 \times M^2$ matrix and given by

$$F_k = I - \mu_k\left(R_{u,k}^T \otimes I\right) - \mu_k\left(I \otimes R_{u,k}\right) + \mu_k^2 E\left[\left(u_k^* u_k\right)^T \otimes \left(u_k^* u_k\right)\right]. \qquad (25)$$

For clarity, we recall the time index i. Therefore, expression (16) becomes

$$E\left\|\tilde{\boldsymbol{\Psi}}_k^{(i)}\right\|_{\text{vec}\{\delta_k\}}^2 = E\left\|\tilde{\boldsymbol{\Psi}}_{k-1}^{(i)}\right\|_{\text{vec}\{F_k \delta_k\}}^2 + \mu_k^2 \sigma_{v,k}^2\left(r_k'^T \delta_k\right) \qquad (26)$$

where we use the fact that $E\left\|u_k\right\|_{\Sigma_k}^2 = \text{Tr}\left(R_{u,k}\Sigma_k\right) = r_k'^T \delta_k$ with $r_k' = \text{vec}\{R_{u,k}^T\}$. For simplicity of notation, we drop the $\text{vec}\{\cdot\}$

228

notation from the subscripts in (26):

$$E\left\|\widetilde{\boldsymbol{\Psi}}_k^{(i)}\right\|_{\delta_k}^2 = E\left\|\widetilde{\boldsymbol{\Psi}}_{k-1}^{(i)}\right\|_{F_k\,\delta_k}^2 + \mu_k^2\sigma_{v,k}^2\left(r_k'^{\mathrm{T}}\,\delta_k\right) \qquad (27)$$

Let $\boldsymbol{p}_k = \widetilde{\boldsymbol{\Psi}}_k^{(\infty)}$. Then

$$E\left\|\boldsymbol{p}_k\right\|_{\delta_k}^2 = E\left\|\boldsymbol{p}_{k-1}\right\|_{F_k\,\delta_k}^2 + \mu_k^2\sigma_{v,k}^2\left(r_k'^{\mathrm{T}}\,\delta_k\right) \qquad (28)$$

By iterating (28) over one cycle, N coupled equations are obtained:

$$E\left\|\boldsymbol{p}_1\right\|_{\delta_1}^2 = E\left\|\boldsymbol{p}_N\right\|_{F_1\,\delta_1}^2 + g_1\,\delta_1 \qquad (29)$$

$$\vdots$$

$$E\left\|\boldsymbol{p}_{k-1}\right\|_{\delta_{k-1}}^2 = E\left\|\boldsymbol{p}_{k-2}\right\|_{F_{k-1}\,\delta_{k-1}}^2 + g_{k-1}\,\delta_{k-1} \qquad (30)$$

$$E\left\|\boldsymbol{p}_k\right\|_{\delta_k}^2 = E\left\|\boldsymbol{p}_{k-1}\right\|_{F_k\,\delta_k}^2 + g_k\,\delta_k \qquad (31)$$

$$\vdots$$

$$E\left\|\boldsymbol{p}_N\right\|_{\delta_N}^2 = E\left\|\boldsymbol{p}_{N-1}\right\|_{F_N\,\delta_N}^2 + g_N\,\delta_N \qquad (32)$$

with $g_k = \mu_k^2\sigma_{v,k}^2\,r_k'^{\mathrm{T}}$. By choosing the free parameters $\{\delta_k, \delta_{k-1}\}$, such that $\delta_{k-1} = F_k\,\delta_k$, we combine (30) and (31) to obtain

$$\begin{aligned} E\left\|\boldsymbol{p}_k\right\|_{\delta_k}^2 &= E\left\|\boldsymbol{p}_{k-1}\right\|_{\delta_{k-1}}^2 + g_k\,\delta_k \\ &= E\left\|\boldsymbol{p}_{k-2}\right\|_{F_{k-1}F_k\,\delta_k}^2 + g_{k-1}F_k\,\delta_k + g_k\,\delta_k. \end{aligned} \qquad (33)$$

Iterating across the cycle we arrive at

$$\begin{aligned} E\left\|\boldsymbol{p}_{k-1}\right\|_{\sigma_{k-1}}^2 &= E\left\|\boldsymbol{p}_{k-1}\right\|_{F_k\cdots F_N F_1\cdots F_{k-1}\,\delta_{k-1}}^2 \\ &\quad + g_k F_{k+1}\cdots F_N F_1 \cdots F_{k-1}\,\delta_{k-1} \\ &\quad + g_{k+1}F_{k+2}\cdots F_N F_1\cdots F_{k-1}\,\delta_{k-1} \\ &\quad \cdots + g_{k-2}F_{k-1}\,\delta_{k-1} + g_{k-1}\,\delta_{k-1}. \end{aligned} \qquad (34)$$

Let

$$\Pi_{k-1,l} = F_{k+l-1}F_{k+l}\cdots F_N F_1\cdots F_{k-1}, \quad \text{for } l = 1,\ldots N \qquad (35)$$

$$a_{k-1} = g_k\,\Pi_{k-1,2} + g_{k-1}\Pi_{k-1,3} + \ldots + g_{k-2}\Pi_{k-1,N} + g_{k-1}. \qquad (36)$$

Then

$$E\left\|\boldsymbol{p}_{k-1}\right\|_{(I-\Pi_{k-1,1})\delta_{k-1}}^2 = a_{k-1}\,\delta_{k-1} \qquad (37)$$

Expression (37) can be exploited to evaluate the performance measures at node k, as follows:

$$\eta_k = E\left\|\boldsymbol{p}_{k-1}\right\|_q^2, \quad q = \mathrm{vec}\{I\} \qquad \text{(MSD)} \qquad (38)$$

$$\varsigma_k = E\left\|\boldsymbol{p}_{k-1}\right\|_{r_k}^2, \quad r_k = \mathrm{vec}\{R_{u,k}\} \qquad \text{(EMSE)} \qquad (39)$$

$$\xi_k = \varsigma_k + \sigma_{v,k}^2 \qquad \text{(MSE).} \qquad (40)$$

Since we are free to select the weight vector δ_{k-1}, choosing $\delta_{k-1} = \left(I - \Pi_{k-1,1}\right)^{-1} q$ or $\delta_{k-1} = \left(I - \Pi_{k-1,1}\right)^{-1} r_k$ results in expressions for the steady-state MSD, EMSE and MSE at node k

$$\eta_k = a_{k-1}\left(I - \Pi_{k-1,1}\right)^{-1} q \qquad \text{(MSD)} \qquad (42)$$

$$\varsigma_k = a_{k-1}\left(I - \Pi_{k-1,1}\right)^{-1} r_k \qquad \text{(EMSE)} \qquad (42)$$

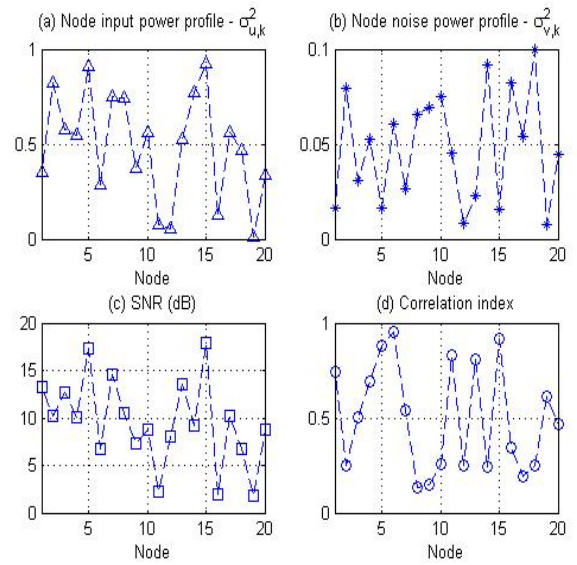$$\xi_k = \varsigma_k + \sigma_{v,k}^2 \qquad \text{(MSE)} \qquad (43)$$

Fig.1 Statistical profile of the Gaussian data input to a 20-node network
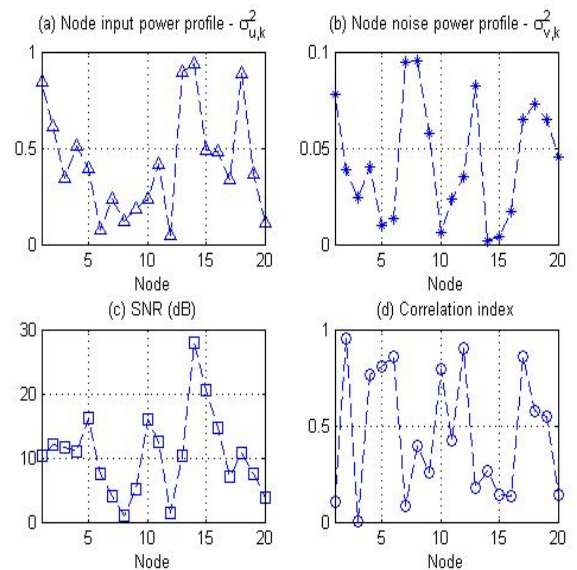

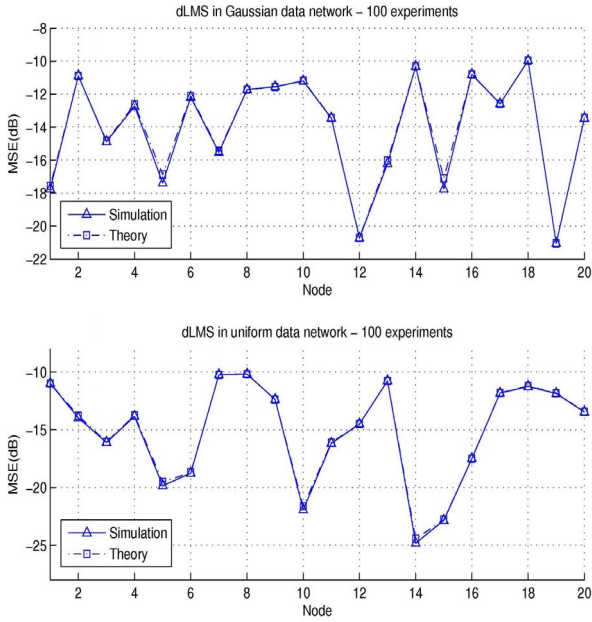Fig.2 Statistical profile of the uniform data input to a 20-node network

Fig.3   Steady-state MSE using $\mu_k = 0.008$ per node for Gaussian distributed inputs and uniformly distributed inputs.

IV. SIMULATIONS

In this section, we compare the theoretical performance with the computer simulations in a system identification scenario. All simulation results are averaged over 100 independent runs. The steady-state curves are obtained by averaging the last 2000 instantaneous samples of 20000 iterations. Consider a network with 20 nodes seeking an unknown filter with $M = 10$ taps. Two types of signals, Gaussian and uniformly distributed signals, are used to generated the inputs at each node k

$$u_k(i) = \alpha_k u_k(i-1) + \beta_k \cdot \rho_k(i) \qquad (44)$$

which is a first-order autoregressive (AR) process with a pole at α_k; $\rho_k(i)$ is a white, zero-mean, Gaussian random sequence with unit variance or a uniform random sequence between -1.0 and 1.0, $\alpha_k \in [0,1)$ and $\beta_k = \sqrt{\sigma_{u,k}^2 \cdot (1 - \alpha_k^2)}$. In this way, the covariance matrix $R_{u,k}$ of the regressor $\boldsymbol{u}_{k,i}$ is a 10×10 Toeplitz matrix with entries $r_k(m) = \sigma_{u,k}^2 \alpha_k^{|m|}$, $m = 0, \ldots, M-1$ with $\sigma_{u,k}^2 \in (0,1]$. The background noise is a white Gaussian process with variance $\sigma_{v,k}^2 \in [0,0.1)$. The node profiles for both Gaussian and uniformly distributed inputs are illustrated in Fig.1-2. Although there are some vary small discrepancies between simulation and practice at certain nodes, Fig 3 shows that the theoretical results for MSE match well with the simulated results. In addition, such discrepancies can be reduced by decreasing the step-size, as can be seen in Fig4. Similar results for EMSE and MSD have been obtained but are not included due to space limitations.

VII. CONCLUSIONS

In this paper, we carried out a steady-state mean square performance evaluation of dLMS under the assumptions A1 and A2. Using weighted spatial-temporal energy conservation arguments, we derived expressions for the steady-state MSD,

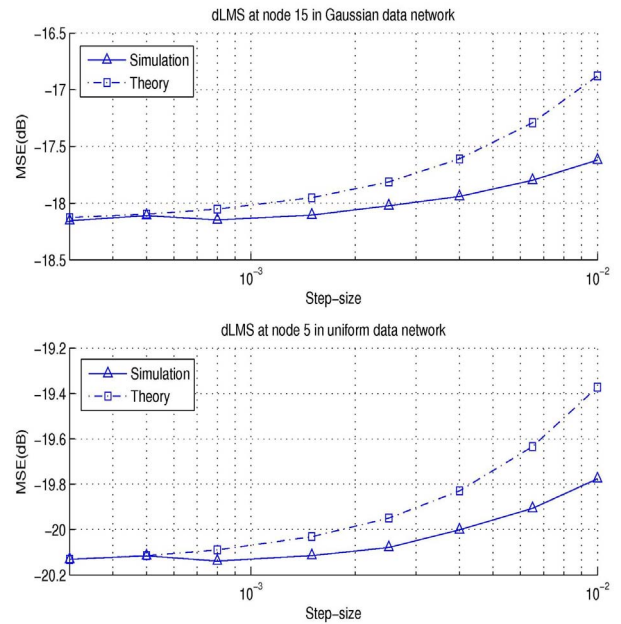EMSE and MSE without restricting the distribution of the inputs.

Fig.4   Steady-state MSE versus μ_k for Gaussian distribution data at node 15 and for uniformly distribution data at node 5

REFERENCES

[1]   D. Estrin, L. Girod, G. Pottie and M. Srivastava, "Instrumenting the world with wireless sensor networks", *Proc. ICASSP*, Salt Lake City, UT, May 2001, pp. 2033-2036.

[2]   M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for sistributed optimization", *IEEE Journal on Selected Areas in Communications,* vol. 23, no. 4, Apr. 2007, pp. 798-808.

[3]   C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks", *IEEE Trans. Signal Processing*, vol. 55, no. 8, Aug. 2007, pp. 4064-4077.

[4]   A. H. Sayed and C. G. Lopes, "Adaptive processing over distributed networks," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences,* vol. E90-A, no. 8, 2007, pp. 1504-1510.

[5]   A. H. Sayed, Fundamentals of Adaptive Filtering, John Wiley & Sons, Inc., Hoboken, NJ, 2003.

[6]   L. Li, J. A. Chambers, C. G Lopes. and A. H. Sayed, "Distributed estimation over an adaptive incremental network based on the affine projection algorithm", submitted to *IEEE Trans. Signal Processing* in revew.

[7]   B. Widrow and S. D. Stearns, Adaptive Signal Processing, Englewood Cliffs, Prentice-Hall, 1985.