PARTIAL UPDATE ADAPTIVE STRATEGIES FOR DISTRIBUTED WIRELESS NETWORKS

Vijay Barfa[1], R. P. Narwaria [2]

[1,2] *Electronics Department, Madhav Institute of Technology &Science, Gwalior M.P. (India)*

ABSTRACT

A partial update adaptive distributed strategy is developed based on incremental techniques.The proposed scheme apply on the problem of linear estimation with less number of computation in a co-operative manner.The proposed algorithm responds in real time to change in environment. It isefficient and power having learning mechanism for solving distributed estimation and optimization problem over wireless networks. In sensor networks there are various application that involves physical phenomenon featuring space varying parameter like battle field, surveillance, environment monitoring, precision agriculture and medical application.In incremental partial update algorithm, complexity in computation and communication are reduced. Computational complexity analysis is evaluated and performance characteristics of each algorithm are given with computer simulations.

Keyword: Incremental Network, Max Partial Update, Sequential Partial Update, Stochastic Partial Update

I. INTRODUCTION

A wireless sensor network consists of an arrangement of sensor nodes distributed over a geographical area to cooperatively observe physical phenomena through noisy observing processes. The nodes, which we also interchangeably call agents, consist of at least three maincomponents: process units, sensing devices and a wireless transmit-receive unit. In more advance sensor networks, nodes are also equipped with actuators to take action according to the command issued from a control unit. Initially, Wireless sensor networks were developed for military applications, including localization and battle field surveillance. Nowadays they are several areas in industrial monitoring and consumer applications, including intelligent transportation, precision agriculture, and smart spaces. Adaptive strategies with incremental mode of communication described in [2] focus on reduction in communication among the nodes by restricting a particular node receiving and transmitting to the immediate nodes only instead of every node of the network. The drawback of this technique is that it involves high computational complexity.

In a wireless sensor network, two different approaches are implemented to perform signal processing, namely centralized and distributed techniques. In former nodes send their information to a central unit for further processing and storage, whereas in latter the measured information or data are locally exchanged and processed within distributed network. In a centralized approach, transmitting a measured information to a fusion center may cause network congestion and result in a waste of power and communication resources. In addition fusion center require relatively high computation power to process the large collected data. In a distributed approach, the network computational load is divided among nodes no centralized structure is required. In a distributed approach data are exchange locally i.e. single hop or multi- hop data transmission also reduce the network energy consumption. These advantages encourage the use of distributed signal processing(DAP) approaches for various application in sensor network.

Over the past few years, there has been covering a large research on distributed signal processing, as it support the assurance of overcoming the issue of bandwidth limitation and limited energy budget in denser sensor network. Within this underlying structure, in distributed adaptive signal processing is apparent as a key technology to support the implementation of flexible co-operative learning and information processing scheme across a set of distributed nodes, with communication, computing and sensing capabilities.Distributed adaptive algorithm are particularly useful for parameter estimation and for the solution of optimization problem, where the underlying signal statics are time –varying or unknown. Clearly adaptivity helps the networks to track variation of the desired signal parameter as new measurement becomes available. More importantly, as a result of DAP, a sensor network becomes robust against changes in the environment condition and network topology.

In this paper we study and develop distributed adaptive strategies for monitoring time – varying physical phenomenon in sensor network under real – world limitation and change in environment condition.

In this paper, incremental partial update strategies are proposed for weight update which reduce computational complexity to a considerable amount. Incremental partial update algorithm choose a subset of coefficient to be updated in every iteration based on some criterion instead of updating all the coefficient. These algorithm are simple, less complex, adaptive and inherit robustness of distributed incremental LMS algorithm[1].

II. INCREMENTAL LMS FOR DISTRIBUTED SOLUTION

For distributed optimization problem there have been cover a lots of work for incremental solution[2,6]. Consider a network with P nodes as shown in figure.1

Let $\{d_k(i), v_{k,i}\}$, $k = 1, 2,3 ...P$be the data available for a particular node k at a time instant i from environment . At time i, the sensor at node k collects a measurement $d_k(i)$, where i denotes the discrete time index and k indicates the node index, and assuming an autoregressive(AR) model is adopted to represent these measurement as follows :

$d_k(i) = \sum_{m=1}^{M} \alpha_m d_k(i-m) + n_k(i)$ (1)

where $n_k(i)$ is additive zero – mean noise

Coefficients $\{\alpha_m\}$are the parameter of the underlying model.

Define parameter w^0 which is the desired optimum solution for the network which is

M×1 parametr vector

$w^0 = col\{\alpha_1, \alpha_2, \dots, \alpha_M\}$ (2)

and regressor vector

$u_{k,i} = [\, d_k(i-1) d_k(i-2) \dots d_k(i-m)]$ (3)

then (1) at each node k can be given as

$d_k(i) = u_{k,i} w^0 + n_k(i)$ (4)

Here, the objective is to estimate the model parameter vector w^0 from the measurement d_k (i) and $u_{k,i}$ over the network. Thus in order to find the M× 1 vectorw^0 , we formulate the linear space – time LMS estimation problem as min_w J (w) and J(w) = $E||\boldsymbol{d} - \boldsymbol{U}w||^2$ (5)

Where {d_k (i),$u_{k,i}$} are realization of {$\boldsymbol{d}_k$,$\boldsymbol{u}_k$}. Thus optimum minimum mean – square error (MMSE) solution $w^0$is calculated, for which the normal equation (5) are satisfied

$$R_{du} = R_u w^0 \quad (6)$$

Where R_u= E**U*U**and R_{du}= E**U*d**

When nodes in the network has access to data in order to take advantage of node cooperation, we can introduce a distributed network with incremental learning, where at least one cyclic path can be established across the network.in this type of network, information should be transferred from one node to its immediate node in cyclic manner to return to the initial node(see Fig. 1)

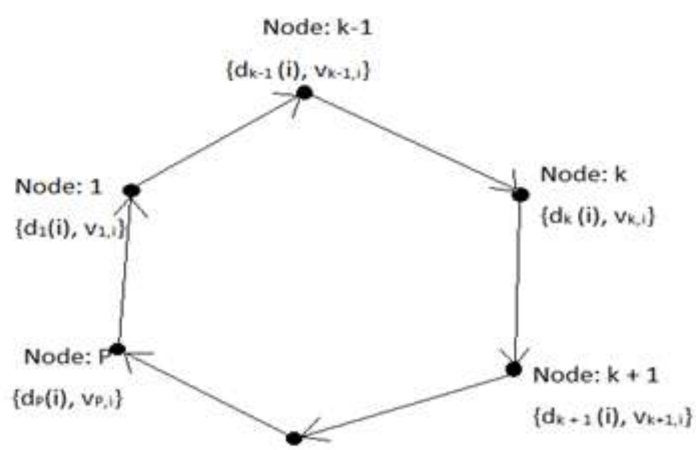

Fig. 1 Distributed network accessing data with P nodes

The incremental LMS solution for distributed network can given [1] by

$\gamma_0^{(i)} = w_{i-1}$ (7)

$\gamma_k^{(i)} = \gamma_{k-1}^{(i)} + \mu_k v_{k,j}^{*}(d_k(i) - v_{k,i}\gamma_{k-1}^{(i)})$ (8)

K = 1, 2…P

$w_i = \gamma_n^{(i)}$

Where

μ_k= step size parameter at node *k*

$\gamma_k^{(i)}$ = local estimate at node *k* at time *i*

w_i =estimate of w_i at node *k*

$v_{k,i}$ = input at node *k* at i^{th}iteration.

$\gamma_{k-1}^{(i)}$= local estimate of immediate node *k* -1

$v_{k,i}^{*}$is the hermitian of $v_{k,i}$ the above mentionedalgorithm uses local data realizations d_k(i),$v_{k,i}$and $\gamma_{k-1}^{(i)}$weight estimate of immediate node. This incremental procedure purely relies on local data estimation and gives truly distributed solution.

III. PARTIAL UPDATE INCREMENTAL SOLUTIONS

In spite of the incremental adaptive solutions reduce the number of communication at each iteration are equal to LMS. We can reduce computational complexity by partial update algorithm[3,4,6]. In few application adaptive filter have large number of coefficients. Updating the whole coefficient vector is costly in term of memory, power consumption and computation. Generally more hardware implies more power. Here we proposed incremental partial update techniques which reduce computational complexity to a considerable amount.

3.1 Sequential Partial Update Incremental LMS

Sequential partial update(SEPU) method updates a subset of the adaptive filter coefficient so as to reduce the computational complexity associated with adaptation process [3,6] at each iteration for every node in the network. The coefficient subset to be updated is selected in a deterministic fashion.

The update equation is given by

$\gamma_k^{(i)} = \gamma_{k-1}^{(i)} + \mu_k I_{N,k}^{(i)} e_k^{(i)} v_{k,i}^{*}$ (9)

Where $e_k^{(i)} = d_k(i) - v_{k,i}\gamma_{k-1}^{(i)}$ (10) and

$$I_{N,k}^{(i)} = \begin{bmatrix} b1(i) & 0 & 0 & . & . & 0 \\ 0 & b2(i) & 0 & . & . & 0 \\ . & . & b3(i) & . & . & . \\ . & . & . & b4(i) & . & . \\ . & . & . & . & . & . \\ 0 & . & . & . & . & bM\ (i) \end{bmatrix}$$

$\sum_{j=1}^{M} b_j(i) = N, \quad a_j(i) \in \{0,1\}$

$I_{N,k}^{(i)}$ is the coefficient selection matrix to select a subset of N coefficient out of M total coefficient at node k at i^{th} iteration.

Let the coefficient index set be Q = {1,2,3…M} i.e. there are M coefficient totally out of which N coefficient are to be updated. Then Q is divided into S number of subset$L_1, L_2, \ldots, L_s$ with each subset having N coefficient where S $= c_N^M$. Let R = M/N be an integer then R coefficient subsets are arranged in periodic sequences with respective coefficient selection matrix $I_{N,k}^{(i)}$

$$I_{N,k}^{(i)} = \begin{bmatrix} b1(i) & 0 & 0 & . & . & 0 \\ 0 & b2(i) & 0 & . & . & 0 \\ . & . & b3(i) & . & . & . \\ . & . & . & b4(i) & . & . \\ . & . & . & . & . & . \\ 0 & . & . & . & . & bM\ (i) \end{bmatrix} b_j(i)$$

= 1 if j $\in J_{(i \bmod R)}$+ 1 and zero otherwise.

For a given M and N, $I_{N,k}^{(i)}$ is not unique. Updating N out of M coefficient reduces the complexity of adaptation process by a factor R.

3.2 Stochatic Partial Update Incremental LMS

Stochastic partial update improves the performance of the network over the sequential partial update algorithm with same amount of computational complexity reduction. In this method coefficient subsets to be updated are chosen randomly instead of deterministic fashion as in SEPU algorithm.

The update equation is given by

$$\gamma_k^{(i)} = \gamma_{k-1}^{(i)} + \mu_k I_{N,k}^{(i)} e_k^{(i)} v_{k,i}^* \quad (11)$$

The coefficient selection matrix is given by

$$I_{N,k}^{(i)} = \begin{bmatrix} b1(i) & 0 & 0 & . & . & 0 \\ 0 & b2(i) & 0 & . & . & 0 \\ . & . & b3(i) & . & . & . \\ . & . & . & b4(i) & . & . \\ . & . & . & . & . & . \\ 0 & . & . & . & . & bM(i) \end{bmatrix}$$

a_j(i) = 1if j ∈ $J_{m(i)}$ and zero otherwise.

Where m(i) is an independent random process with probability mass function

Pr(m(i) = c) = π_c , c = 1…R

$\sum_{c=1}^{R} \pi_c = 1$

The computational complexity of stochastic algorithm(STPU) is same as that of the SEPU and slower than incremental LMS algorithm by a factor R because of the decimation of the adaptive filter coefficient.

3.3 Max - Partial Update Incremental LMS

In this algorithm at each iteration largest magnitude vector entries are updated.

This is a data dependent partial update technique which is based on finding N largest magnitude entries from M total coefficient [5].

The update equation is given by

$$\gamma_k^{(i)} = \gamma_{k-1}^{(i)} + \mu_k v_{k,i}^* I_{N,k}^{(i)} e_k^{(i)} \quad (12)$$

Where $e_k^{(i)} = d_k(i) - v_{k,i}\gamma_{k-1}^{(i)}$

The coefficient selection matrix $I_{N,k}^{(i)}$ is given by

$$I_{N,k}^{(i)} = \begin{bmatrix} b1(i) & 0 & 0 & . & . & 0 \\ 0 & b2(i) & 0 & . & . & 0 \\ . & . & b3(i) & . & . & . \\ . & . & . & b4(i) & . & . \\ . & . & . & . & . & . \\ 0 & . & . & . & . & bM\ (i) \end{bmatrix}$$

$$a_j(i) = 1\ If\ |v(i-j+1)| \in max_{1 \le l \le M}(|v(i-j+1)|, N)$$

$$a_j(i) = 0\ otherwise$$

Max partial update is similar to the sequential update in decimating the coefficient update vector, but magnitude of the update vector entries to be ranked before updating instead of deterministic fashion in sequential update method. The coefficient selection scheme determine the convergence of the algorithm

This reduce complexity by factor R = M/N

IV. SIMULATION

In this part we compare the simulation result of each technique. Number of node in network P = 20. The regressor vector or data vector $v_{k,i}$ is $1\times M$ and collects the data as follows.

$$v_{k,i} = col\{v_k(i), v_k(i-1), \dots \dots v_k(i-M+1)\} \quad (13)$$

In this network every node k depends on local statistics and influenced by immediate neighbors. 350 independent experiments were performed and averaged.In all experiment step size parameter should be small as possible and constant. The curve are generated for 100 iterations. MSE gives how far the local estimate from the optimum weight w^0. The performances of proposed algorithm are compared with that of incremental algorithm.

V SETTING OF PARAMETER

For all proposed algorithms ring topology is considered as shown in fig. 1

Table: Proposed incremental algorithms MSE result

% of coefficient	Number of coefficient M	Step size	Updated coefficient N	Simulation MSE Result	Simulation	Simulation MSE Result

update		μ		SEIA	MSE Result (STIA)	(MAIA)
70	10	0.03	7	0.1215	0.1379	0.0583
50	10	0.03	5	0.2015	0.1810	0.1004
30	10	0.03	3	0.2245	0.1854	0.1406

Above MSE result are compared with incremental algorithm in which all the coefficient are updated whose MSE s 0.0078

SEIA – sequential incremental algorithm

STIA – stochastic incremental algorithm

MAIA – max increment algorithm

The simulation results for performance estimation are compared with incremental LMS in which all the coefficient are updated in each iteration. From simulation result and figure, we say that max – partial outperform sequential partial and stochastic partial update in performance. Stochastic technique give better performance over sequential for same computational complexity. But sequential partial update converge with faster convergence rate compared to other to algorithm. Stochastic partial update converge at a fast rate compared to max partial update.

The advantage of proposed algorithm over incremental algorithm is achieved at the cost of degradation in performance.

From observing simulation result it is obvious –

1. It is more sensitive to local statistics.
2. Mean – square error depends on number of coefficient updated.
3. Because incremental mode of communication is considered every node k is influenced by its immediates neighbors.

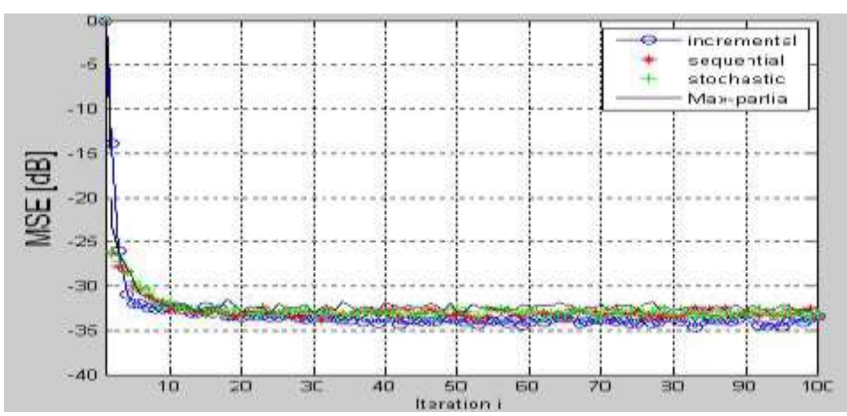

Fig. 2 Comparison of each techniques with incremental LMS for 70 % Coefficient update

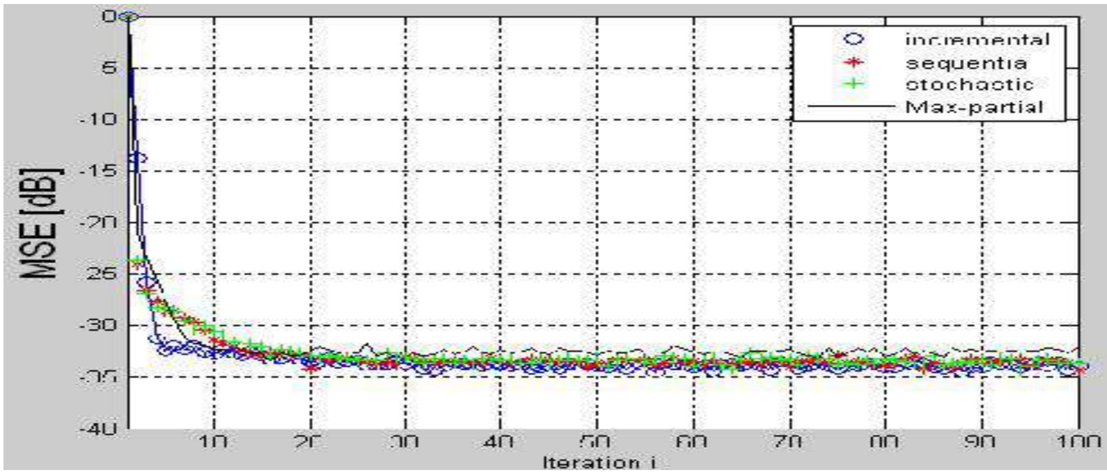

Fig. 3 Comparison of each techniques with incremental LMS for 50 % Coefficient update

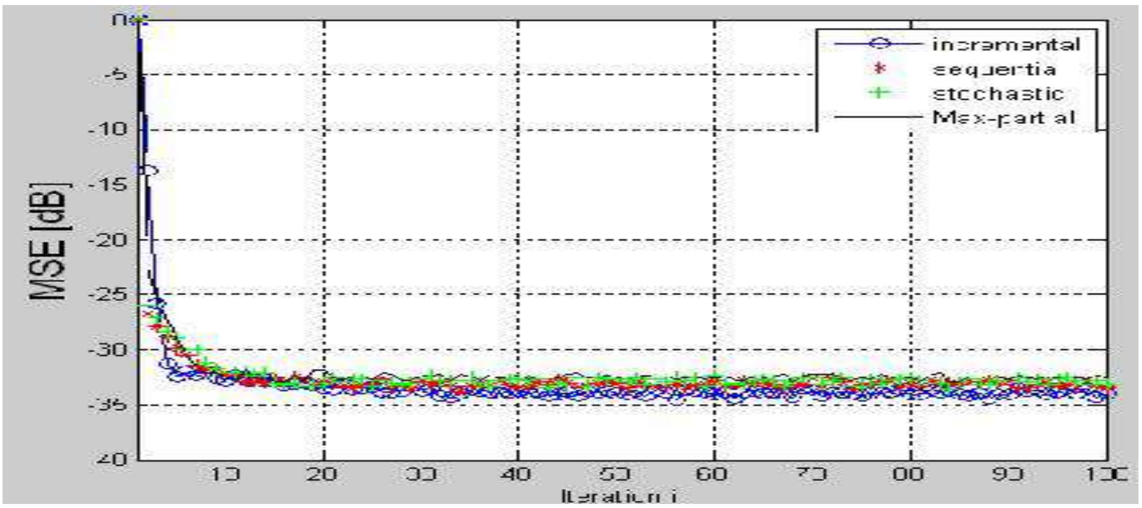

Fig.4 Comparison of each techniques with incremental LMS for 30 % Coefficient update

V. CONCLUSION

It is clear from the result and analysis that sequential and stochastic partial update algorithm the computational complexity but stochastic partial update algorithm gives better performance compared to sequential. Max – partial algorithm converge quickly and has consistent steady state performances and reduce computational complexity as the same amount as other two technique. So with little worse in the performance the computational complexity can be reduce to a considerable amount. This reduce power consumption and suitable for low energy budget i.e. low energy sources.

REFERENCES

[1] Cassio G.Lopes and Ali H.Sayed. "Incremental adaptive strategies over distributed Networks" IEEE Trans. on signal processing, Vol. 55, No 8, August 2007.

[2] Leilei Li, Jonathon A. Chambers and Cassio G. Lopes. "Distributed estimation over an adaptive incremental network based on the affine projection algorithm", IEEE Trans. On signal processing, Vol. 58, No. 1, January 2010.

[3] Mahesh.Godavarti and Alfred O.Hero, "Sequential partial update LMS algorithm" IEEE Trans. on signal processing, October 2001

[4] Mahesh.Godavarti and Alfred O.Hero, "Stochastic partial update LMS algorithm" IEEE Trans. on signal processing, October 2001.

[5] Kutluyil Dogancay," partial-update adaptive signal processing Design, analysis and implementation"

[6] Mahesh.Godavarti and Alfred O.Hero, "Partial Update LMS algorithm", journal of latex class files, vol.1, No.11, November 2002

[7] M.G Rabbat and R.D.Nowak "quantized incremental algorithms for distributed optimization",IEEE J.Sel.areas Commun.,vol.23,no.4.pp798-808,Apr.2005.