EFFICIENT ADAPTIVE STRATEGIES OVER DISTRIBUTED NETWORKS

Kondalarao Jyothi and Ajit Kumar Sahoo
Department of Electronics and Communication
Engineering, National Institute of Technology
Rourkela-769008

***Abstract*— In this paper, we review some of the computationally efficient adaptive distributed strategies developed using *incremental partial update techniques*. The schemes mentioned here solve the problem of linear estimation with less number of computations in a cooperative fashion. In a distributed network each node contains local computing equipment which estimates and shares them with other nodes. The resulting algorithms are less complex in computations and in communication because of *Incremental partial update algorithms* and each node communicate with immediate node only. Computational complexity analysis is evaluated and performance characteristics of each algorithm are given with computer simulations. Simulation results show that with a small degradation in performance, considerable amount of computational complexity is reduced.**

***Index Terms*—Distributed strategies, Incremental partial update, sequential partial update, stochastic partial update, Max-partial update, computational complexity.**

1. Introduction

A distributed network consists of certain number of processing elements called nodes. These nodes are distributed over a geographical area which collects the information or data for particular phenomena. In distributed processing each node interacts with other nodes in the network to arrive at an estimate of the particular parameter.
Nodes in the network communicate in a certain manner as dictated by the topology of the network. Adaptive strategies with incremental mode of communication described in [1] 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 this paper, incremental partial update algorithms are proposed for weight update which reduces computational complexity to a considerable amount. Incremental partial update algorithms select a subset of coefficients to be updated in each iteration based on some criterion instead of updating all the coefficients. These algorithms are simple, adaptive, less complex and inherit robustness of distributed incremental Least Mean Square (LMS) algorithm of [1].

In Section 2, a review of incremental LMS algorithm is presented. In Section 3 partial incremental update techniques sequential, stochastic and Max-partial are presented. In Section 4, simulation results are presented for performance evaluation of each technique.

2. Incremental LMS for distributed solution

For distributed optimization problems there have been extensive works for incremental solution in literature [1, 6]. Consider a network with L nodes as shown the figure.

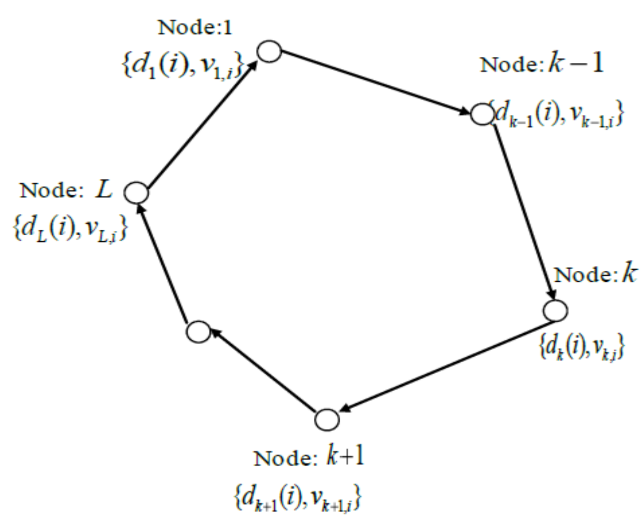

Fig1: Incremental mode of communication in the network.

Let $\{d_k(i), v_{k,i}\}$, $k = 1, 2....L$ be the data available for a particular node k at a time instant from environment. w^0 is the desired optimum solution for the network. v_k is the input regressor vector of size $1 \times M$ and d_k is the desired value at node k which is a scalar quantity. For incremental adaptive solution these regressor vectors can replaced by instantaneous approximations [1].

$R_{v,k} \approx v_{k,i}^* v_{k,i}$, $p_{v,k} = d_k(i) v_{k,i}^*$

Where $R_{v,k}$ is auto correlation vector and $p_{v,k}$ is cross correlation vector. The incremental LMS solution for distributed network can given [1] by

$$\varphi_0^{(i)} = w_{i-1}$$
$$\phi_k^{(i)} = \phi_{k-1}^{(i)} + \mu_k v_{k,i}^* (d_k(i) - v_{k,i}\phi_{k-1}^{(i)}) \rightarrow (1)$$
$$k = 1, 2 \ldots L$$
$$w_i = \varphi_n^{(i)}$$

Where

μ_k =step size parameter at node k

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

w_i =estimate of w^o at node k

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

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

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

3. Partial update incremental solutions

Even though the incremental adaptive solutions reduce the numbers of communications to considerable amount the numbers of calculations at each iteration are equal to LMS. We can reduce computational complexity by partial update techniques [2, 3, and 5]. In some applications adaptive filters have large number of coefficients. Updating the entire coefficient vector is costly in terms of memory, computations and power consumption. Generally more number of hardware multipliers implies to more power. Here we propose incremental partial update techniques which reduce computational complexity to a considerable amount.

I. Sequential partial update Incremental LMS

This method updates a subset of the adaptive filter coefficients so as to reduce the computational complexity associated with adaption process [2, 5] 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

$$\phi_k^{(i)} = \phi_{k-1}^{(i)} + \mu_k I_{N,k}^{(i)} e_k^{(i)} v_{k,i}^* --- \rightarrow (2)$$

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

$$I_{N,k}^{(i)} = \begin{bmatrix} a_1(i) & 0 & . & . & 0 \\ 0 & a_2(i) & . & . & . \\ . & . & a_3(i) & . & . \\ . & . & . & . & . \\ 0 & . & . & . & a_M(i) \end{bmatrix}$$

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

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

Let the coefficient index set be $Q = \{1, 2 \ldots M\}$ i.e. there are M coefficients totally out of which N coefficients are to be updated. Then Q is divided into S number of subsets $P_1, P_2, \ldots . P_s,$ with each subset having N coefficients where $S = {}^M c_N$. Let $R = M / N$ be an integer then R coefficient subsets are arranged in periodic sequences with respective coefficient selection matrix $I_{N,k}^i$.

$$I_{N,k}^{(i)} = \begin{bmatrix} a_1(i) & 0 & . & . & 0 \\ 0 & a_2(i) & . & . & . \\ . & . & a_3(i) & . & . \\ . & . & . & . & . \\ 0 & . & . & . & a_M(i) \end{bmatrix}$$

$a_j(i)$ =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 coefficients reduces the complexity of adaption process by a factor R.

II. Stochastic 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 sequential partial update algorithm [3, 4].

The update equation is given by

$$\phi_k^{(i)} = \phi_{k-1}^{(i)} + \mu_k v_{k,i}^* I_{N,k}^{(i)} e_k^{(i)} -- \rightarrow (3)$$

The coefficient selection matrix is given by

$$I_{N,k}^{(i)} = \begin{bmatrix} a_1(i) & 0 & . & . & 0 \\ 0 & a_2(i) & . & . & . \\ . & . & a_3(i) & . & . \\ . & . & . & . & . \\ 0 & . & . & . & a_M(i) \end{bmatrix},$$

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

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

$$\Pr\{\ m(i) = c\ \} = \pi_c,\ c=1\ldots R$$

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

The computational complexity of stochastic algorithm is same as that of the sequential algorithm and slower than Incremental LMS algorithm by a factor R because of the decimation of the adaptive filter coefficient.

III. Max- partial update incremental LMS

In this algorithm at 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 coefficients [4].

The update equation is given by

$$\phi_k^{(i)} = \phi_{k-1}^{(i)} + \mu_k v_{k,i}^{*} I_{N,k}^{(i)} e_k^{(i)} \rightarrow (4)$$

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

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

$$I_{N,k}^{(i)} = \begin{bmatrix} a_1(i) & 0 & . & . & 0 \\ 0 & a_2(i) & . & . & . \\ . & . & . & . & . \\ . & . & . & . & 0 \\ 0 & . & . & . & a_N(i) \end{bmatrix},$$

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

$a_j(i) = 0$ otherwise.

Max partial update is similar to the sequential update method 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 determines the convergence of the algorithm. This reduces the complexity by factor $R = M / N$.

4. Simulations

In this section we give the simulation results comparing the each technique. Number nodes in network L= 20.The regressor vector or data vector $v_{k,i}$ is $1 \times M$ and collects the data as follows.

$$v_{k,i} = col\{v_k(i), v_k(i-1), \ldots\ldots, v_k(i-M+1)\} - \rightarrow (5)$$

In the network each node k depends on local statistics and influenced by immediate neighbors. 300 independent experiments were performed and averaged. In all the experiments step-size parameter is chosen to be small and kept constant. The curves are generated for 100 iterations. Here Mean Square Error (MSE) is taken as the performance metric. MSE gives how far the local estimate from the optimum weight w^0. The performances of proposed algorithms are compared with that of incremental algorithm.

Parameter settings

For all the algorithms proposed here ring type topology is considered as shown in Fig.1

In sequential incremental algorithm number of coefficients M is 10. For 70% coefficient update, 7 coefficients to be updated in each iteration i.e. N =7. Step-size parameter is taken as 0.03.The MSE obtained from simulation is 0.1203. For 50% coefficient update 5 coefficient (N =5) are updated in each iteration and obtained MSE is 0.2005. For 30% coefficient update 3coefficients (N =3) are updated in each iteration, MSE obtained is 0.2255 and the results are compared with incremental algorithm in which all the coefficients are updated whose MSE is 0.0078.

In stochastic incremental algorithm the parameters are same as in sequential and MSE calculated from simulation results is 0.1389 for 70% update, 0.1910 for 50% update and 0.1954 for 30% coefficient update respectively.

In max-incremental algorithm also the parameters are same as in sequential algorithm and stochastic algorithms.MSE is 0.0593 for 70%, 0.1014 for 50% and 0.1416 for 30%coefficient update.

The simulation results for performance estimation are shown in Figs. 2, 3 and 4. Partial update techniques are compared with incremental LMS in which all the coefficients are updated at each iteration. In all the cases Max-partial outperforms sequential partial and stochastic partial incremental techniques in performance and Stochastic technique gives better performance over sequential for the same computational complexity. But sequential partial update technique converges with a faster convergence rate compared to stochastic and Max algorithms. Stochastic algorithm converges at a faster rate over Max algorithm. The advantage of proposed algorithms over incremental algorithm is achieved at the cost of degradation in performance.

From the simulation results it is obvious

1. MSE depends on number of coefficients updated.
2. It is more sensitive to local statistics.

3. Since incremental mode of communication is considered each node k is influenced by its immediate neighbors.

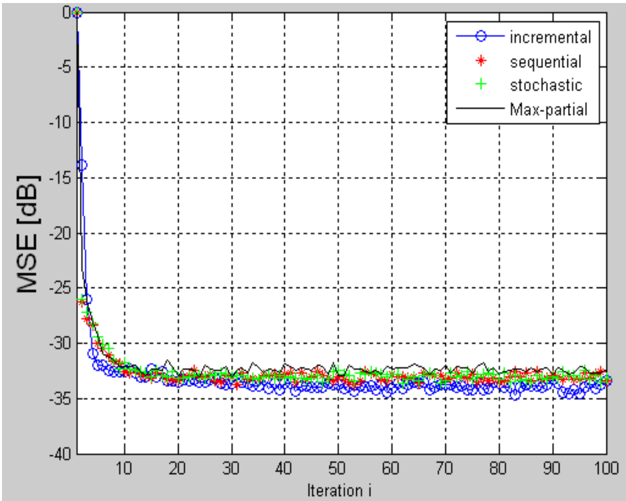

Fig2. Performance with 70% coefficient update

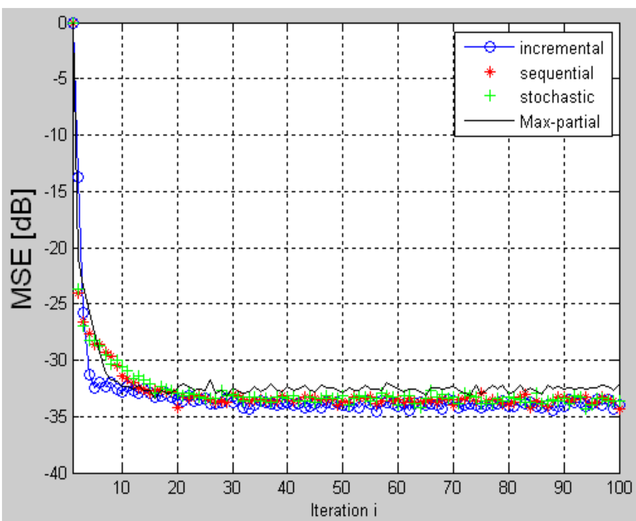

Fig4. Performance with 30% coefficient update

Conclusion

It is clear from the analysis that sequential and stochastic partial update algorithms reduces the computation complexity in equal manner but stochastic partial update algorithm gives-

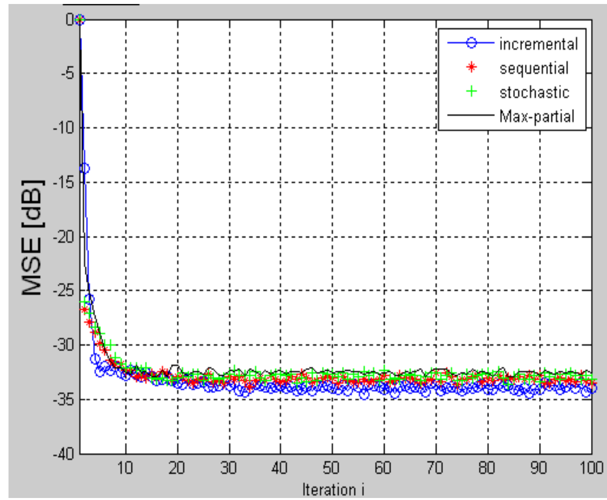

Fig3. Performance with 50% coefficient update

Better performance over sequential but converges slowly. Max-partial algorithm converges slowly but has consistent steady state performance and has minimum MSE over sequential and stochastic algorithms. It reduces the computational complexity in the same amount as of other two techniques. So with a little deterioration in the performance the computational complexity can be reduced to considerable amount. This in turn reduces the power consumption and is suitable for networks with 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]. Mahesh.Godavarti and Alfred O.Hero, “Sequential partial update LMS algorithm” IEEE Trans. on signal processing, October 2001.

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

[4]. Kutluyil Dogancay *Partial-update adaptive signal processing Design, analysis and implementation*

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

[6].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.