Frequency Domain Incremental Strategies over Distributed Network

Siba Prasad Mishra and Ajit Kumar Sahoo

Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela
Rourkela-769008
Odisha

212ec6191@nitrkl.ac.in,
ajitsahoo@nitrkl.ac.in,

Abstract. This paper presents a frequency domain adaptive strategy based on incremental techniques. The proposed scheme represents the problem of linear estimation using frequency domain transformation methods like DCT, DFT in a cooperative manner, where nodes are having the computing ability to find the local estimation in frequency domain and sharing them among the predefined neighbours. This algorithm is distributed and cooperative in nature. In addition to this it also responds to real time environments and produces a better result than that of the incremental time domain adaptive method under colored input. Each node share information with its immediate neighbours to fully exploit the spatial dimension there by lowering the communication burden. Computer simulation result illustrates the performance of the new algorithm.

Keywords: incremental algorithm, distributed processing, incremental DCTLMS, incremental DFTLMS.

1 introduction

Distributed processing is the technique of extracting information from data collected From different nodes spread over a geographical area. In distributed process Nodes collect noisy information, performs local estimation then share it with the neighbour node, followed by some defined topology to estimate the parameter of interest. As compare with the centralized solution distributed solution has better advantage, since it does not require a powerful central processor and extensive amount of communication between node and processor, it only depends upon their local data and the interaction with its immediate neighbours [1]. The Distributed processing reduces the communication burden and number of processing [2-3].

The convergence rate of LMS (Least mean square) type filter is dependent on the Autocorrelation matrix of the input data and on the eigen value spread of the covariance matrix of the regressor data. The mean square error (MSE) of an adaptive filter

adfa, p. 1, 2011.

using LMS algorithm decreases with time as sum of the exponentials, whose time constants are inversely proportional to the eigen value of the auto correlation matrix of input data[4]. The smaller eigen value of autocorrelation matrix of the input results slower convergence mode and larger eigen values limit on the maximum learning rate that can be chosen without encountering stability problem. Best convergence and learning rate results when all the eigen values of the input autocorrelation matrix are equal i.e. Autocorrelation matrix should be represent in the form of some constant multiplication with the identity matrix [5].

Practically the input data's are colored and the eigen values of autocorrelation matrix vary from smallest to the largest. The filter response can be improved by prewhitening the data, but for this the autocorrelation of the input data should be known. It is difficult to know the autocorrelation of the input data .It can be achievable by using unitary transformation , such as discrete cosine transform(DCT) , discrete Fourier transform(DFT) etc. These transformation have de-correlation properties that improves the convergence performance of LMS for correlated input data [5].
Transform domain (which is also called frequency domain) can be applied in two ways one is block wise frequency domain algorithm other is non-block wise frequency domain algorithm. In block wise frequency domain algorithm a block of input data is first transformed then input to the incremental LMS algorithm and in non-block or real time algorithm the data are continuously transformed by a fixed data-independent transform to de-correlate the input data [5]. DFT-LMS algorithm was first introduced by Narayan [6] belongs to a simplest algorithm family because of the exponential nature. But in many practical situation it was found that DCT-LMS performs better than that of DFT-LMS and other transform domain [7].In this paper we interpret the incremental LMS using DCT/DFT algorithm and found that it produce better convergence and performance than previous.

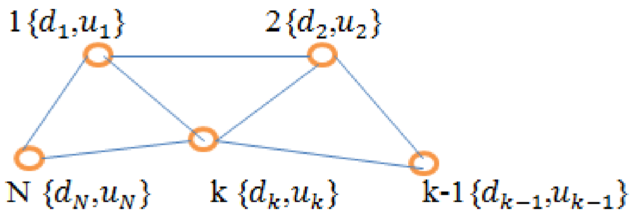

Fig. 1. Distributed network with N nodes accessing space time data

2 Estimation Problem and the Adaptive Distributed Solution

We are interested to estimate the unknown vector w^0 by using the incremental adaptive LMS in frequency domain method. Let consider there are N number of nodes having its local zero mean spatial desired data and regressor data d_k and u_k respectively, distributed in a geographical area as shown in the Fig.1. k=1, 2…, N, d_k is a scalar and u_k is regressor vector of size 1×M.

$$U \triangleq col\{u_1, u_2, \dots, u_N\}(N \times M) \quad (1)$$

$$d \triangleq col\{d_1, d_2, \dots, d_N\}(N \times 1) \quad (2)$$

These quantities collected data across all nodes; the objective is to estimate the M×1 vector w that solves the distributed solution [1].the cost function can be decomposes for each node[1] given by

$$\mathrm{J}\ (w) = \sum_{k=1}^{N} J_k(w) \quad (3)$$

Now let $\phi_k{}^{(i)}$ be the local estimate of w^0 at node k and time i and let the initial weight assign at node 1 is $\phi_0{}^{(i)} \leftarrow w_{i-1}$, and after complete one cycle across node, at the last node i.e. at node N it will coincide with w_i , according to steepest descent solution [1]

$$\phi_k{}^{(i)} = \phi_{k-1}{}^{(i)} - \mu[\nabla j_k(w_{i-1})]^* \quad (4)$$

Still it is not purely distributed solution, since in whole updating process it uses global weight information w_{i-1} in order to find $\nabla J_k(w_{i-1})$, hence to make it distributed perfectly we can use the incremental gradient algorithms which uses the local estimate $\psi_{k-1}{}^{(i)}$ at each node instead of global information w_{i-1}i.e. the (4) can be written as

$$\phi_k{}^{(i)} = \phi_{k-1}{}^{(i)} - \mu\left[\nabla j_k\left(\phi_{k-1}{}^{(i)}\right)\right]^* \quad (5)$$

The distributed incremental LMS algorithm summarize as

$$\begin{cases} \phi_0{}^{(i)} \leftarrow w_{i-1} \\ \phi_k{}^{(i)} = \phi_{k-1}{}^{(i)} + \mu_k u_{k,i}{}^*\left(d_k(i) - u_{k,i}\phi_{k-1}{}^{(i)}\right) \\ w_i \leftarrow \psi_N{}^{(i)} \end{cases} \quad (6)$$

3 Frequency Domain Adaptive Distributed Solution

Transform domain adaptive filter refer to LMS filter whose inputs are pre-processed with a unitary data independent transformation. The frequency domain transformations are discrete Fourier transform (DFT), discrete cosine transform (DCT). This pre-processing improves the eigen value distribution of input autocorrelation matrix of the LMS filter, as a result its convergence speed increases. In this paper we use DCT-LMS and DFT-LMS frequency domain approach. In an incremental mode of cooperation each node uses its spatial data to estimate the local weight then share it to the neighboring node. But the proposed algorithm pre-processed with a unitary process the input regressor prior to processing, then estimates the weight in frequency domain and advances it to the next node for future estimation. It is found that the frequency domain approach yields better performance than that of previous algorithm, since this approach transform the input data to white form and make eigen spread equal to unity results improving convergence and learning ability [5]. Fig. 2 represents the data processing structure of incremental transform domain adaptive filter.

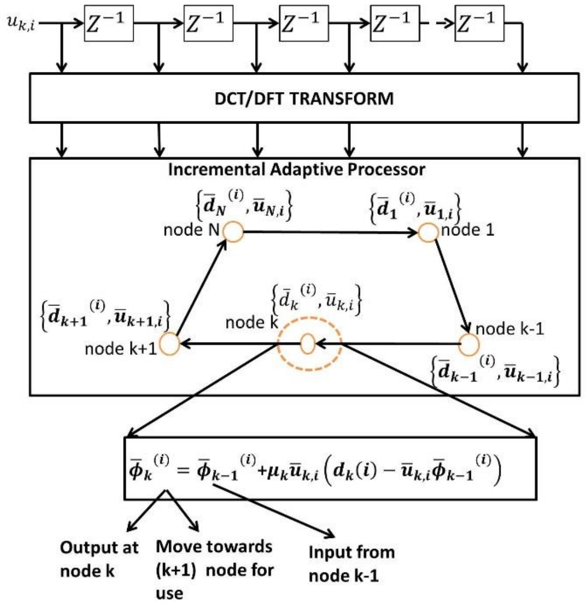

Fig. 2. Data processing structure for transform domain incremental adaptive LMS Algorithm

3.1 DCT-LMS AND DFT-LMS ALGORITHM

The Transform domain algorithm such as DCT-LMS (discrete cosine transform LMS) and DFT-LMS (discrete Fourier transform LMS) [5] are described below:

Table 1. Discrete Fourier Transforms Algorithm

The optimal weight vector w^0 that solves $\min_w E\,\lvert d - uw\rvert^2$ can be approximated iteratively via $w_i = T\overline{w_\iota}$, T is the unitary DFT matrix $[F_{mk}]=\frac{1}{\sqrt{M}}e^{-\frac{j2\pi mk}{M}}$, m, k=0, 1,...,M-1, S=diag$\{1,e^{-\frac{-j2\pi}{M}},\ldots,e^{-\frac{j2\pi(M-1)}{M}}\}$, $\rho_k(-1)=\epsilon,\overline{w_{-1}}=0,\overline{u_{-1}}=0$, For i ≥0 $\overline{u_\iota}=\overline{u_{\iota-1}}S+\frac{1}{\sqrt{M}}\{u(i)-u(i-M)\}[1,1\ldots,1]$, $\rho_k(i) = \beta\,\rho_k(i-1) + (1-\beta)\left\lvert\overline{u_\iota(k)}\right\rvert^2$, k=0, 1,…, M-1 D_i=\diag $(\rho_k(i))$, e (i) =d (i) $-\overline{u_\iota}\,\overline{w_{\iota-1}}$, $\overline{w_\iota}=\overline{w_{\iota-1}}+\mu D_i^{-1}\overline{u_\iota}^*$, μ is positive step size (usually small) and 0<<β<1

Table 2. Discrete Cosine Transforms Algorithm

The optimal weight vector w^0 that solves $\min_w E\,\|d-uw\|^2$ can be approximated iteratively via $w_i = T\overline{w_\iota}$, Q is the unitary DCT trix$[Q_{mk}] = \chi(k)\cos(\frac{k(2M+1)\pi}{2M})$,$\chi(0) = \frac{1}{\sqrt{M}}$,$\chi(k) = \frac{\sqrt{2}}{\sqrt{M}}$ S=diag{1,$e^{-\frac{-j2\pi}{M}}$,…., $e^{-\frac{j2\pi(M-1)}{M}}$ },$\rho_k(-1)$=ϵ(a small value),$\overline{w}_{-1}$ =0,$\overline{u}_{-1}$ =0,and repeat for i$\geq$0
δ(k)= [u (i)-u (i-1)] cos ($\frac{k\pi}{2M}$), c (k)= $(-1)^k$[u(i-M)-u(i-M-1)]cos($\frac{k\pi}{2M}$), $\gamma(k) = \alpha(k)[\delta(k) - c(k)]$,$\overline{u_\iota}$=$\overline{u}_{i-1}Q$ – $\overline{u}_{i-2}$+ [$\gamma(0)\ \gamma(1)$ ...$\gamma(M-1)$],
$\rho_k(i) = \beta\rho_k(i-1)$ +(1–β)$\left\|\overline{u_\iota(k)}\right\|^2$, k=0, 1… m-1
D_i=diag ($\rho_k(i)$), e (i) =d (i)-$\overline{u_\iota}\overline{w_{\iota-1}}$, $\overline{w_\iota}$=$\overline{w_{\iota-1}}$+$\mu D_i^{-1}\overline{u_\iota}^*$, μ is step size (usually small) and 0<<β<1

4 Simulation

A Network of consisting N = 20 nodes and each local filter has M=10 taps consider for simulation study. We take 1000 iterations and perform 500 independent experiment to get the simulation result. The measurement data $d_k^{(i)}$ can be generated by using the data model $d_k^{(i)}$=$u_{k,i}w^0$+$v_k^{(i)}$ at each node and the vector w^0=col {1, 1,…,1}/$\sqrt{M}$, of size M×1.the background noise is white and Gaussian with σ_v^2=10^{-3}. The EMSE (Excess Mean square error), MSE (Mean square error) and MSD (Mean square deviation) can be plot by using $\left|u_{k,i}(\bar{\phi}_k^{(i)} - \bar{w}^0)\right|^2$,$\left|d_k(i) - \bar{u}_{k,i}\bar{\phi}_{k-1}^{(i)}\right|^2$, $\left|(\bar{\phi}_k^{(i)} - \bar{w}^0)\right|^2$.In this example , the network consists of N=20 nodes , with each regressor of size (1×M) collecting data by observing a time-correlated sequence $\{u_k^{(i)}\}$, generated as

$u_k^{(i)}$=$\propto_k u_k^{(i-1)}$+$\beta_k z_k^{(i)}$, i>-∞

Here $\propto_k\in[0,1)$, is the correlation index and $z_k^{(i)}$ is a spatially Gaussian independent process with unit variance and $\beta_k = \sqrt{\sigma_{u,k}^2(1-\alpha_k^2)}$ [1]. The resulting regressor have Toeplitz covariance matrices $R_{u,k}$, with correlation sequence r_k(i) = $\sigma_{u,k}^2(\propto_k)^{|i|}$, i=0,…., M-1. The input regressor power profile $\sigma_{u,k}^2$ ∈ (0,1] , the correlation index $\propto_k$∈ (0,1] and the Gaussian noise variance $\sigma_{v,k}^2$ ∈ (0,0.1] chosen at random. The algorithms such as DCT-LMS and DFT-LMS, which are described in section 3, are used to update the tap weights at each node. The step size used for all simulation is

0.03.the values of node power profile and correlation index and used in this simulation given by,
$\sigma_{u,k}{}^{2}$= [0.2 0.5 0.8 0.1 0.5 0.8 0.7 0.4 0.9 0.9 0.3 0 0.2 0.6 0.2 0.7 0.9 0.4 0.5 0.7]
α_k= [0.8 0 0.7 0.4 0.8 0.5 0.7 0.4 0.3 0.2 0.2 0.6 0.3 0.5 0.2 0.6 0.4 0.9 0.9 0.6]
The convergence rate and performance of MSE, EMSE and MSD simulation results are

Shown in Fig.3, Fig.4 and Fig.5.The simulation result clearly shows that the convergence rate and performance of DCT-LMS using incremental strategies is better than that of rest.

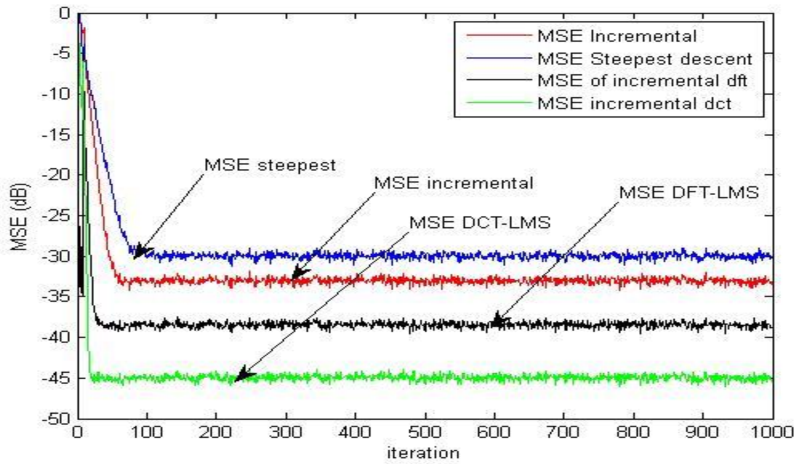

Fig. 3. Transient MSE performance at node 1

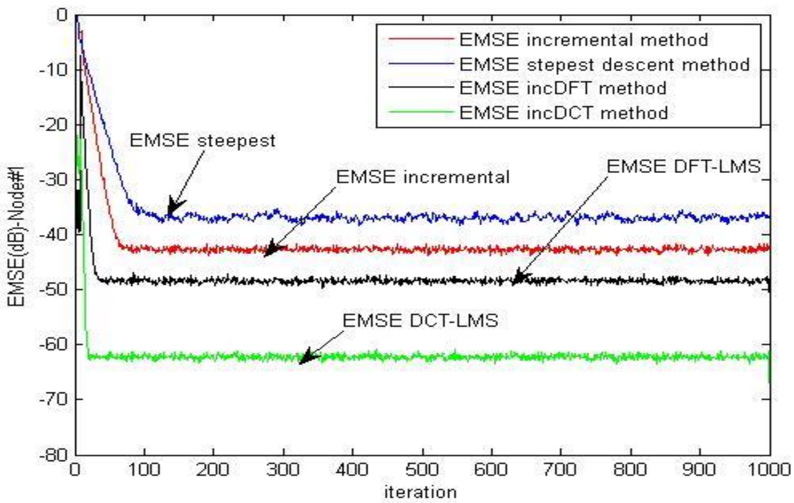

Fig. 4. Transient EMSE performance at node 1

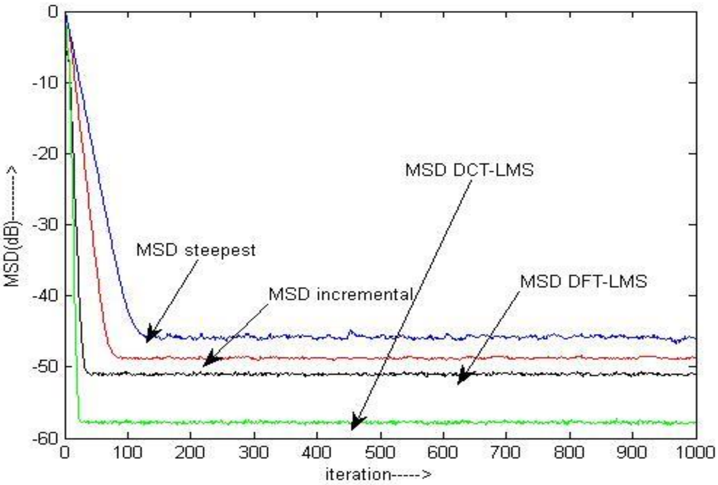

Fig. 5. Transient MSD performance at node 1

5 Conclusion

The simulation results of the proposed transform domain incremental adaptive algorithm not only gives better steady state performance but also improves the convergence rate under colored data. These algorithms are very useful for better convergence of adaptive filter, since it is really impossible to construct a prewhitening filter for produce unity Eigen spread.

6 References

1. Lopes, C. G. and Sayed, A. H.: Incremental Adaptive strategies over Distributed Networks, IEEE Trans. Signal processing, vol. 55, no. 8, pp.4064-4057(2007)
2. Estrin, D., Pottie, G. and Srivastava, M.: instrumenting the world with wireless sensor Networks, in proc. IEEE int.conf. Acoustics, Speech, Signal Processing (ICASSP), pp. 2033-2036, Salt Lake City, UT(2001)
3. Wax, M. and Kailath, T. :decentralized processing in sensor arrays, IEEE trans. Acoustic, speech signal process., vol.Assp-33, no. 4, pp.1123-1129,(1985)
4. Widrow, B. and Stearns, S. D.: Adaptive signal process. Englewood Cliffs, NJ: prentice-Hall (1985)
5. Sayed, A. H.: fundamentals of Adaptive Filtering. New York: Wiley,(2003)
6. Narayan, S. S., Peterson, A.M. and Narsimha, M.J. : Transform Domain LMS algorithm, IEEE Trans. Acoustic, Speech signal processing, vol. ASSP-31, pp.609- 615(1983)
7. Beaufays, Francoise :Transform Domain Adaptive Filters, IEEE Trans. Signal processing, vol. 43.no. 2, pp. 422-431(1995)