

## Algorithms of Adaptive Beam Forming for Smart Antenna: A Comparative Study

Chintan S. Jethva and Dr. R.G. Karandikar

Electronics & Telecommunication Department K.J. Somaiya college of engineering, vidyavihar Mumbai-77, India

---

**Abstract:** For enhancing data rate Smart Antennas have been gaining prominence in the recent times where the main beam is steered towards direction of interest via phase shifter. The sense behind this development is the availability of high-end processors to handle the complex computations involved. The phase shifting and array weighing can be performed on digital data rather than in hardware is the major advantage of digital beam former. This paper gives comparative study of five adaptive algorithms –Least Mean Square (LMS), Normalized Least Mean Square (NLMS), Sample Matrix Inversion (SMI), Recursive Least Square (RLS) and Conjugate Gradient Method (CGM) for computing the array weights. This adaptive algorithm assures very high rate of convergence & highly reduced mean square error. The weights obtained by the above algorithms are then used to steer the antenna array beam in the direction of interest, thereby enhancing SNR.

**Keywords:** Least Mean Square (LMS), Normalized Least Mean Square (NLMS), Sample Matrix Inversion (SMI), Recursive Least Square (RLS) and Conjugate Gradient Method (CGM).

---

### I. Introduction

The video conferencing, video on demand and many more features are possible with the arrival of 3G and because of 3G the data rates available for mobile communication have acutely increased. This tendency has continued with the current research on Long Term Evolution that promises even higher speeds and better quality of service. For achieving such high data rates we have to overcome constraints like interference, noise, multipath, poor channel conditions etc and for that highly advanced technology is required. So in order to accentuate signals of interest and to minimize the interfering signals we need an intelligent. To achieve high SNR thereby enhancing the data rate as well as in communication, radar systems and biomedical engineering techniques like multiple antenna schemes and adaptive beam forming can be used. There are several kinds of adaptive beam forming algorithms, such as least mean squares algorithm, sample matrix inversion algorithm, recursive least squares algorithm, conjugate gradient method, and so on. This paper gives comparative study of algorithms of adaptive beam forming. The main factors to be measured before choosing an adaptive beam forming algorithm are computational burden and the precision for convergence and robustness of the algorithm.

### II. Adaptive Beamforming Algorithm

According to whether a training signal is used or not, most of the adaptive beam forming algorithms can be classified into Non-Blind Adaptive algorithm and Blind Adaptive algorithm [1]. Non-blind adaptive algorithms uses reference signal to modify the array weights repetitively, so that at the end of each & every iteration the output of the weights is compared to the reference signal and the generated error signal is used in the algorithms to modify the weights. The examples are Least Mean Square Algorithm (LMS), Recursive Least Square algorithm (RLS), Sample Matrix Inversion (SMI) and Conjugate Gradient (CG). Blind adaptive algorithms do not make use of the reference signal and hence no array weight adjustment is required. The examples are Constant Modulus algorithm (CMA) and Least Square Constant Modulus (LS-CMA) [2]. By adaptively changing the antenna array pattern, nulls are formed in the angular locations of the interference sources so that Adaptive beam forming technique is able to operate in an interference environment. The digital signal processor is the heart of adaptive beam forming, which interprets the incoming data, determines the complicated weights (amplification and phase info) and multiplies the weights to each element output and corrects the array pattern. The array thus minimizes the effect of noise & interference and produces maximum gain in the desired direction. Thus the smart antenna's efficiency and performance is dependent on the adaptive algorithms used for digital beam forming.

#### 1.1 Least Mean Square (Lms)

Least Mean Square (LMS) algorithm is relatively simple compared to other adaptive beam forming algorithms because it does not require correlation function calculation nor does it require matrix inversions. Fig 1 shows a generic adaptive beam forming system which requires a reference signal. As shown in Fig 1, the outputs of the individual sensors are linearly combined after being scaled using corresponding weights such that the

antenna array pattern is optimized to have maximum possible gain in the direction of the desired signal and nulls in direction of interferers [1]. Figure 1 shows the minimum mean-squares error system, where the error is (1). So, the squared error can be given by (2). To make it simpler, the time connected information 'n' is abandoned. Then, the cost function is given by (3) then can be computed by (4) (5). With the help of the gradient method, the gradient of performance surface of the cost function is obtained (6). The Optimum Wiener solution is (7) when (6) = 0. The gradient of cost function can be approximated by the method of steepest descent and the directions of the steepest descent with the direction of the gradient vector are opposite. The steepest descent iterative is given by following formula (8) where  $\mu$  represents the step size. Refer to the formula (4) and (5), and the LMS solution is obtained by (9). The convergence of the LMS algorithm is in proportion to the step size  $\mu$ , if the step size is too big, the LMS algorithm may not reach an expected solution, and it is so called the under damped case. If the step size is too small, then it will lead to the over damped case, and the antenna array will have difficulty tracking the interesting signal, so the step size  $\mu$  should meet the conditions (10). As the array correlation matrix  $R_{XX}$  is positive definite, the Eigen values are positive. Suppose that all interfering signals are noise and there is only one interesting signal, then Formula (10) can have the form (11) [3].

### 1.2 NORMALIZED LEAST MEAN SQUARE (NLMS)

In practice, an improved LMS method i.e. Normalized LMS (NLMS) is used to achieve stable calculation and faster convergence. The final weight vector in NLMS can be updated by (12) where the NLMS algorithm reduces the step size  $\mu$  to make the large changes in the update weight vectors. This prevents the updated weight vectors from diverging and makes the algorithm more stable and faster converging than when a fixed step size is used like LMS. Equation (12) represents the normalized version of LMS (NLMS) where step size is divided by the norm of the input signal to avoid gradient noise amplification due to  $x(n)$ . NLMS algorithm overcomes the limitation of LMS algorithm and it gives faster convergence rate than LMS for limited number of array elements. But when large number of array elements used, the convergence rate is really poorer as it considers all the array elements as active and updates the weight vectors for all of them. Fig 2 Shows Error Convergence performed by LMS, NLMS [1].

### 1.3 SAMPLE MATRIX INVERSION (SMI)

Sample Matrix Inversion (SMI) which is also known as Block adaptive approach because it involves block implementation or block processing i.e. a block of samples of the filter input and the desired output are collected and processed together to obtain a block of output samples. Thus, the process involves serial to parallel conversion of the input data, parallel processing of the collected data and parallel to serial conversion of the generated output data. The computational complexity can be further reduced by the elegant parallel processing of the data samples. Thus we can say that in this type of algorithm we are adapting the weights block by block thus increasing the convergence rate of the algorithm and reducing the computational complexity further [2]. The sample matrix is defined as the time average estimate of the array correlation, which uses  $N$  samples, and if the random process is ergodic in correlation, then the time average estimate is equal the real correlation matrix (13) (14). The matrix  $X_N(n)$  is (15). So,  $R_{XX}$  can be given by (16). Take a hypothesis that the desired signal is (17) then (18). Finally, the sample matrix inversion weights of the  $n$ th block can be computed as (19) [3].

### 1.4 RLS ALGORITHM (RLS)

In Recursive Least Square (RLS) algorithm [4], the weights are updated by the equation (20). Where,  $K(n)$  is referred to as the gain vector and  $\xi(n)$  is a priori estimation error which is given by the equation (21). The RLS algorithm does not require any matrix inversion computations as the inverse correlation matrix is computed directly. It requires reference signal and correlation matrix information [5].

### 1.5 CONJUGATE GRADIENT METHOD (CGM)

The problem with the steepest descent method has been the sensitivity of the convergence rates to the Eigen value spread of the correlation matrix. Greater spreads result in slower convergences. The convergence rate can be accelerated by use of the Conjugate Gradient Method (CGM) [4]. The aim of CGM is to iteratively search for the optimum solution by choosing conjugate (perpendicular) paths for each new iteration. The method of CGM produces orthogonal search directions resulting in the fastest convergence. The path taken at iteration  $n + 1$  is perpendicular to the path taken at the previous iteration  $n$ . CGM is an iterative method, whose goal is to minimize the quadratic cost function (22). The general weight update equation is given by (23) where the step size  $\mu(n)$  is determined by (24). The residual vector update is given by (25) and the direction vector update is given by (26). A linear search is used to determine  $\alpha(n)$  which minimizes  $J(w(n))$ . Thus, the procedure to use CGM is to find the residual and the corresponding weights and update until convergence is satisfied. It can be

shown that the true solution can be found in no more than  $K$  iterations. This condition is known as quadratic convergence [5].

### III. Indentations And Equations

The  $e(n)$  is error between a desired signal  $d(n)$  and the array output  $y(n)=w^H x(n)$ .

Signal  $x(n)$  received by multiple antenna elements which is given by,

$$e(n) = d(n) - w^H x(n) \tag{1}$$

Where  $H$  denotes Hermitian (complex conjugate) transpose. The weight vector  $w$  is a complex vectors.

$$|e(n)|^2 = |d(n) - w^H(n)x(n)|^2 \tag{2}$$

$$J(w) = E[|d|^2] - 2w^H r + w^H R_{xx} w \tag{3}$$

Where  $r$  is the signal correlation vector of the antenna array and  $R_{xx}$  represents the array correlation matrix &  $J(w)$  is the cost function.  $E[|d|^2]$  is the mean square of desired signal.

$$r(n) \approx d^*(n)x(n) \tag{4}$$

$$R(n) \approx x(n)x^H(n) \tag{5}$$

$$\Delta_w(J(w)) = 2R_{xx} w - 2r \tag{6}$$

Where  $\Delta_w J(n)$  is the instantaneous gradient vector ho update the weight vector.

$$w_{opt} = R_{xx}^{-1} r \tag{7}$$

$$w(n+1) = w(n) - 1/2\mu \Delta_w(J(w(n))) \tag{8}$$

Where  $w(n+1)$  is the update value of the weight vector at timen+1.

$$w(n+1) = w(n) - \mu[R_{xx} w - r] = w(n) + \mu e^*(n) x(n) \tag{9}$$

Where  $\mu$  is the step size.

$$0 \leq \mu \leq 1/2\lambda_{max} \tag{10}$$

Where  $\lambda_{max}$  is the largest Eigen value of the array correlation matrix.

$$0 \leq \mu \leq 1/2\text{Tr}[R_{xx}] \tag{11}$$

$$w(n+1) = w(n) + (\mu/x(n)^2) e^*(n) x(n) \tag{12}$$

Where  $w(n+1)$  is final weight vector in NLMS.

$$R_{xx} \approx (1/N) \sum_{n=1}^N x(n) x^H(n) \tag{13}$$

Where  $R_{xx}$  real correlation matrix &  $(1/N) \sum_{n=1}^N x(n) x^H(n)$  is time average estimate.

$$r \approx (1/N) \sum_{n=1}^N d^*(n) x(n) \tag{14}$$

Where  $r$  is the signal correlation vector.

$$X_N(n) = \begin{bmatrix} x_1(1+nN) & x_1(2+nN) & \dots & x_1(N+nN) \\ x_2(1+nN) & x_2(2+nN) & \dots & x_2(N+nN) \\ \vdots & \vdots & \ddots & \vdots \\ x_M(1+nN) & x_M(2+nN) & \dots & x_M(N+nN) \end{bmatrix} \tag{15}$$

Where the matrix  $X_N(n)$  is defined as the  $n$ th block of vectors  $x$  ranges over  $N$ -data snapshots. Where,  $n$  represents the block number, and  $N$  is the block length.

$$R_{xx}(n) = (1/N) X_N(n) X_N^H(n) \tag{16}$$

Where  $R_{xx}$  real correlation matrix.

$$d(n) = [d(1+nK) d(2+nK) \dots d(N+nN)] \tag{17}$$

Where  $d(n)$  desired signal.

$$r = (1/N) d^*(n) X_N(n) \tag{18}$$

Where  $r$  is the signal correlation vector.

$$W_{SMI} = R_{xx}^{-1}(n) r(n) = [X_N(n) X_N^H(n)]^{-1} d^*(n) X_N(n) \tag{19}$$

Where  $W_{SMI}$  is the sample matrix inversion weights.

$$w(n) = w(n-1) + K(n)\xi^*(n) , n=1,2,\dots \tag{20}$$

Where  $w(n)$  is updated weights &  $K(n)$  is referred to as the gain vector and  $\xi^*(n)$  is a priori estimation error which is given by the equation.

$$x(n) = d(n) - w(n-1) x(n) \tag{21}$$

Where,  $K(n)$  is referred to as the gain vector and  $\xi^*(n)$  is a priori estimation error which is given by the equation.

$$J(w) = (1/2) w^H A w - d^H \tag{22}$$

Where  $J(w)$  is the quadratic cost function & where  $A$  is the  $K \times M$  matrix of array snapshots ( $K$  – number of snapshots and  $M$  - number of array elements),  $w$  is the unknown weight vector and  $d$  is the desired signal vector of  $K$  snapshots.

$$w(n+1) = w(n) - \mu(n) D(n) \tag{23}$$

Where  $w(n+1)$  is the weight updates &  $\mu(n)$  is step size.

$$\mu(n) = (r^H(n) A A^H r(n)) / (D(n) A^H A D^H(n)) \tag{24}$$

Where  $\mu(n)$  is step size.

$$r(n+1) = r(n) + \mu(n)A D(n) \tag{25}$$

Where  $r(n+1)$  residual vector update.

$$D(n+1) = A^H r(n+1) - \alpha(n)D(n) \tag{26}$$

Where  $D(n+1)$  direction vector update &  $a(n)$  which minimizes  $J(w(n))$ .

#### IV. Figures And Tables

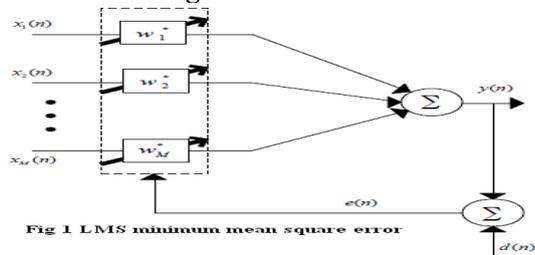


Fig 1 LMS minimum mean square error

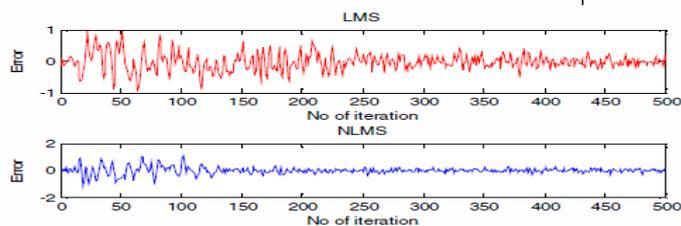


Fig 2. no. of Iterations of LMS & NLMS

#### V. Conclusion

When LMS and NLMS algorithms are compared it was noticed that the error convergence is more stable and shows quick convergence for NLMS algorithm, while the LMS algorithm shows output with more fluctuations and less stable and convergence takes more time in the case of LMS than NLMS because it depends mainly on the step size. The RLS algorithm shows high rate of convergence, but the side lobes are not completely cancelled. The recursive equations used in the RLS algorithm allow faster update of array weights. The Sample Matrix Inversion algorithm is a time average estimate of the array correlation matrix using  $K$  time samples i.e. dividing the input samples into  $k$  number of blocks and each number of blocks is of length  $K$ . Thus we can say that in SMI algorithm we are adapting the weights block by block thus increasing the convergence rate and reducing the computational complexity further. The CGM algorithm calculates the array weights by orthogonal search at each iteration. It shows good beam forming pattern and a high convergence rate. It also has a better resolution as compared to the previous algorithms because the main beam pointing towards the desired user is quite sharp with a high directivity and the side lobes are very less and have power level very low than what was achieved in the previous algorithms.

#### References

- [1] 'Performance Analysis of Adaptive Beam forming Algorithm for Smart Antenna System' S Razia, T Hossain M A Matin. Published in Informatics, Electronics & Vision (ICIEV) International Conference on May 2012 Dhaka, pp 946-949, doi : 10.1109/ICIEV.2012.6317510.
- [2] 'Study and Simulation of Smart antenna using SMI and CGM Algorithm' Neha Shrivastava, Sudeep Baudha, Rahul Singh Rathore, Bharti tiwari, International Journal of Engineering and Innovative Technology (IJEIT), vol 1, issue 6, June 2012.
- [3] 'Research on Adaptive Beamforming Algorithm' Sun Yong-jiang, Qiu Dong-dong, Rao Jia-ren, Zong Peng, and Darwin R. Becerra, China. Published in Image Analysis and Signal Processing (IASP), International Conference on Nov 2012 Hangzhou, pp 1-3, doi : 10.1109/IASP.2012.6425021
- [4] Frank Gross, "Smart Antenna for Wireless Communication" McGraw-hill, September 14, 2005.
- [5] 'Analysis of Adaptive Algorithms for Digital Beamforming in Smart Antennas' Anurag Shivam Prasad, Sandeep Vasudevan, Selvalakshmi R, Sree Ram K, Subhashini G, Sujitha S, Sabarish Narayanan B. Published in Recent Trends in Information Technology (ICRTIT), International Conference on June 2011 Chennai, Tamil Nadu, pp 64-68, doi : 10.1109/ICRTIT.2011.5972418
- [6] 'Adaptive Beamforming Algorithms for Smart Antenna Systems' Shahera Hossain, Mohammad Tariqul, Islam and Seichi S. Published in Control, Automation and Systems, ICCAS International Conference on Oct 2008 Seoul, pp 412-416, doi : 10.1109/ICCAS.2008.4694679
- [7] 'A modified multitarget adaptive array algorithm for wireless CDMA system' L. Yun-hui and Y. Yu-hang, Journal Zhejiang Univ SCI, Vol. 5, No. 11, pp. 1418-1423, 2004.
- [8] 'The adaptive algorithms of the smart antenna system in future mobile telecommunication systems' J. Zhang, Published in Antenna Technology: Small Antennas and Novel Meta materials, IWAT 2005. IEEE International Workshop on Mar 2005, pp 347-350, doi : 10.1109/IWAT.2005.1461088
- [9] 'Implementation of Smart Antenna System Using Genetic Algorithm and Artificial Immune System' Awan, P. H. Published in Microwaves, Radar and Wireless Communications, MIKON 2008. 17th International Conference on May 2008, Wroclaw.