Smart Antennas Adaptive Beamforming through Statistical Signal Processing Techniques

Neelapu G R Reddy*, K.Rama Devi**
*(Department of Electronics and communication Engineering, University College of Engineering, Kakinada, JNTUK A.P., Email: ngrreddy@gmail.com)
** (Department of Electronics and communication Engineering, University College of Engineering, Kakinada, JNTUK A.P. Email: kolisettyramadevi@gmail.com)

ABSTRACT

The smart antenna improves the performance of wireless communication systems by increasing channel capacity and spectrum efficiency, extending range of coverage, steering multiple beams to track several mobiles. Smart antenna may direct main lobe, with increased gain, in the direction of angle of arrival (AOA) of the user (or desired signal), and direct nulls in direction away from the main lobe or in the AOA of interfering signals. For this purpose different adaptive algorithms are available to calculate the optimum weight vectors, to form phased beam, and to minimize the Mean Square Error (MSE). Least Mean Square (LMS) adaptive algorithm is preferred for Beam forming, because it is simple and easy to implement, but the error convergence is slow. In this paper Kernel Least Mean Square (KLMS) Algorithm is proposed to overcome the shortcomings (like MSE) of existing algorithms for a robust smart antenna system.

Keywords - Adaptive Beamforming, LMS, KLMS.

1. INTRODUCTION

Digital signal processing (DSP) has been a major role in the current technical advancements such as noise filtering, system identification, and voice prediction. Standard DSP techniques, however, are not enough to solve these problems quickly and obtain acceptable results. Adaptive filtering techniques must be implemented to promote accurate solutions and a timely convergence to that solution.

Smart antenna consists of an antenna array, combined with signal processing in both space and time. It overcomes the problem of limited channel bandwidth, satisfying a growing demand for a large number of mobiles on communications channels. Smart antenna help in improving the system performance by increasing channel capacity and spectrum efficiency, extending range coverage, steering multiple beams to track many mobiles, and compensating electronically for aperture distortion. To steer the main beam, according to the direction of arrival of desired signal, in adaptive beam forming is control by the values of weight vector. To find the optimum values of these weight vectors adaptive algorithms are used.

Wireless communication systems are limited in performance and capacity by three major impairments. The first of these is multipath fading, which is caused by the multiple paths that the transmitted signal can take to the receive antenna. The second impairment is delay spread, which is the difference in propagation delays among the multiple paths. The third impairment is co-channel interference, which is due to frequency reuse in the cell. Smart antenna which introduce space divide multiple access can distinguish the signals with the same frequency, the same code and the same interval on the basis of different signal with different transmission paths. The sensor array collects spatial samples of propagating wave fields, which are processed by the beam former. The objective is to estimate the signal arriving from a desired direction in the presence of noise and interfering signals. A beam former performs spatial filtering to separate signals that have overlapping frequency content but originate from different spatial locations [4].

The total radiated field of the array is equal to the field of a single element’s field positioned at the reference point (first element) multiplied by a factor, called as array factor. For the uniform linear array of \( N \) element with spacing \( d \) the array factor will be represented as

\[
AF = \sum_{n=1}^{N} w_n e^{j(n-1)(kd\cos \theta + \beta)} \quad (1)
\]
Where $\beta$ is the progressive phase shift, $\theta$ is the angle of arrival (AOA) of desired signal and $w$ represents the weight of the antenna element which is estimated by using adaptive algorithms [2].

II. ALGORITHMS

Least Mean Square (LMS)

Least Mean Square (LMS) is simple to understand and easy to implement. It computes the values of weight by subtracting the received signal by the reference signal. The reference signal is a signal which is highly correlated with the desired signal.

The signal $x(n)$ received by multiple antenna elements is multiplied with the coefficients in a weight vector $w$ (series of amplitude and phase coefficients) which adjusted the phase and the amplitude of the incoming signal accordingly. This weighted signal is summed up, resulted in the array output. An adaptive algorithm is then employed to minimize the error $e(t)$ between a desired signal $d(n)$ and the array output $y(n)$. For the beam former, the output at time is given by a linear combination of the data can be expressed as

$$\text{filter output } y(n) = w^H(n) \ldots (2)$$

$$\text{Estimation error } e(n) = d(n) - y(n) \ldots (3)$$

$$\text{Weight update } w(n+1) = w(n) + \mu x(n)e^*(n) \ldots (4)$$

where $H$ denotes Hermitian (complex conjugate) transpose.

The weight vector $w$ is a complex vectors. The process of weighting these complex weights $w_1...w_k$ adjusted their amplitudes and phases such that when added together forms the desired beam.

![LMS adaptive filter](image)

Fig.1. LMS adaptive filter

Currently LMS is most used adaptive technique for the adaptive Beam forming. By knowledge of angle of arrival of desired signal it shifts the main beam in desired signal arriving angle and giving null at interfering signals. It also needs the knowledge of reference signal. The rate of convergence is comparatively slow and the Mean Square Error (MSE) is also high. When the user moving rapidly, its angle is also moves and to move the beam with it, it’s important that the processing speed will be fast. The processing speed is nothing else convergence rate. So for good performance the rate of convergence should be as high as possible. For accurate result the estimated error should be minimum. Thus MSE should be least for good performance.

It is applied for the Smart antenna Beam forming to calculate the optimum value of the weights. It steer the antenna gain in the direction of the desired signal and null in interfering signal direction. Also mention its slow convergence presents an acquisition and tracking problem for cellular systems.

Kernel Least Mean Square (KLMS)

The Kernel method is a nonparametric modelling technique in which it transforms the input data into a high dimensional feature space via a reproducing kernel such that the inner product operation in the feature space can be computed efficiently through the kernel evaluations. After an appropriate linear methods are subsequently applied on the transformed data. As long as an algorithm can be formulated in terms of inner products (or equivalent kernel evaluation), there is no need to perform computations in the high dimensional feature space. This is the main advantage when compared to the traditional methods. Successful examples of this methodology include support vector machines (SVM's), kernel principal component analysis, etc. The SVM's have already shown good performance in the increasing the accuracy in the speech recognition activity [11]. The kernel adaptive filtering technique used in this work is an adaptive filtering technique for general nonlinear problems. It is a natural generalization of linear adaptive filtering in Reproducing Kernel Hilbert Spaces (RKHS). Kernel adaptive filters are online kernel methods, closely related to some artificial neural networks such as radial basis function networks and regularization networks. The KLMS algorithm is a stochastic gradient methodology to solve least squares problems in RKHS. Because the update equation can be written in terms of inner product, KLMS can be computed efficiently in the input space. The good approximation ability of KLMS stems from the fact that the transformed data include possibly infinite different features of the original data [11], [12].

III. INDENTATIONS AND EQUATIONS

KLMS algorithm formulation

The kernel induced mapping is employed to transform the input $x(n)$ dimensional feature space $F$ as $\Phi(x(n))$ shown if figure $w^T\varphi$ because of the difference in dimensionality of $x$ and $\Phi(x)$. So finding $w$ through stochastic descent may prove as an effective way of nonlinear filtering as LMS does for linear problems[11] and [12].

A kernel ($k$) is a continuous symmetric, positive definite function. Mercer Kernel theorem [13] states that kernel can be expanded as

$$\kappa(x,y)=\sum_{i=1}^{\infty} \zeta_i \phi_i(x) \phi_i(y) \quad \ldots (5)$$

where $\zeta_i$ are the Eigen value and $\phi_i$ are the eigen functions.

The mapping $\varphi$ can be constructed as

$$\varphi(x)=[\sqrt{\zeta_1},\phi_1(x),\sqrt{\zeta_2},\phi_2(x),\ldots] \quad \ldots (6)$$

where $\varphi(x)$ is the transformed feature vector. The relation between kernel and transformed feature space is

$$\phi(x)^T\phi(y)=\kappa(x,y) \quad \ldots (7)$$

By the help of kernel function we can calculate the value of transformed feature vector.

Using the LMS algorithm on the new example sequence $\{(i),d(i)\}$ yields

$$w(0) = 0$$

$$e(n) = d(n) - w(n-1)^T\varphi(n) \quad \ldots (8)$$

$$w(n) = w(n-1)+\eta e(n) \varphi(n) \quad \ldots (9)$$

where $w(n)$ denotes the estimate (at iteration $n$) of the weight vector in $F$. As the dimensionality of $\varphi(n)$ is high, the repeated application of the weight update equation through iterations yields

$$w(n)=w(n-1)+\eta e(n)\varphi(x(n)) \quad \ldots (10)$$

$$w(n)=[w(n-2)+\eta e(n-1)\varphi((x(n-1))] + \eta e(n)\varphi(x(n)) \quad \ldots (11)$$

$$w(n)=[w(n-2)+\eta e(n-1)\varphi((x(n-1))] + \eta e(n)\varphi(x(n))] \quad \ldots (12)$$

.$$w(n) = w(0)+ \eta\sum_{i=1}^{\infty} e_i \varphi(x(i)) \quad \ldots (13)$$

Initially $w(0) = 0$

That is, after i-step training, the weight estimate is expressed as a linear combination of all the previous and present (transformed) inputs, weighted by the prediction errors (and scaled by $\eta$).

Thus the equation for estimate the error will become

$$e(n) = d(n) - [\eta\sum_{i=1}^{\infty} e_i \varphi(x(i))\varphi(x(n))] \quad \ldots (15)$$

$$e(n) = d(n) - \eta\sum_{i=1}^{\infty} e_i [\kappa(x(i),x(n))] \quad \ldots (16)$$

equation for output

![Fig 2. Kernel filter structure](image-url)

**Kernel functions [14]**

By the help of kernel function we can calculate the value of transformed feature vector.

- **Linear Kernel**

  Kernel algorithms using a linear kernel are often equivalent to their non kernel counter parts.

  $$\kappa(x,y)=x^T y + 1 \quad \ldots (21)$$
- Polynomial kernel

The Polynomial kernel is a non-stationary kernel. Polynomial kernels are well suited for problems where all the training data is normalized.

\[
\kappa(x,y) = (x^T y + 1)^d
\]

\[
\kappa(x,y) = \exp\left(-\frac{||x-y||^2}{2h^2}\right)
\]

\[
\kappa(x,y) = \exp\left(-\frac{||x-y||^2}{2h^2}\right)
\]

Where \( h \) is kernel bandwidth. The adjustable parameter \( h \) plays a major role in the performance of the kernel, and should be carefully tuned to the problem at hand. If overestimated, the exponential will behave almost linearly and the higher-dimensional projection will start to lose its non-linear power. In the other hand, if underestimated, the function will lack regularization and the decision boundary will be highly sensitive to noise in training data.

The Gaussian kernel creates a reproducing kernel Hilbert space with universal approximating capability, whereas the polynomial kernel of finite order does not. The approximating capability of the polynomial kernel with order \( d \) is limited to any polynomial function with its degree less than or equal to \( d \). Unless it is clear from the problem domain that the target function is a polynomial function or can be well approximated by a polynomial function, the Gaussian kernel is usually a default choice. The Gaussian kernel has the universal approximating capability, is numerically stable, and usually gives reasonable results [18].

Convergence Rate

KLMS is the LMS algorithm in Reproducing Kernel Hilbert Spaces (RKHS), the role of the step size parameter remains in principle the same and the results from the adaptive filtering literature can be used. In particular, the step size parameter is the compromise between convergence time and misadjustment (i.e., increasing the step - size parameter decreases convergence time but increases misadjustment). Moreover, the step size parameter is upper bounded by the reciprocal of the largest eigen value of the transformed data autocorrelation matrix [13].

The step - size parameter is required to satisfy the following condition for the algorithm to stay stable

\[
0 < \eta < \frac{1}{\lambda_{\text{max}}}
\]

Where \( \lambda_{\text{max}} \) is the largest eigen value of \( R_\phi = (1/N)\phi \phi^T \).

The dimensionality of \( R_\phi \) could be high, and it is unfeasible to compute it directly.

Advantages and disadvantages

- Stable and robust performance against different signal conditions
- Fast convergence
- Good performance for both linear and non-linear signals
- MSE is comparatively less
- Comparatively, complex to implement

Adaptive Beamforming Problem Setup

Consider a Uniform Linear Array (ULA) with \( N \) isotropic elements and the distance between two elements is \( d \) which forms the integral part of the adaptive beamforming system as shown in fig.3.

The output of the array \( y(n) \) with variable element weights is the weighted sum of the received signals \( s_i(n) \) at the array elements and the noise \( n(m) \) at the receivers connected to each element. The weights \( w_m \) are iteratively computed based on the array output \( y(n) \), a reference signal \( d(n) \) that approximates the desired signal, and previous weights. The reference signal is approximated to the desired signal using a training sequence or a spreading code, which is known at the receiver. The format of the reference signal varies and depends upon the system where adaptive beamforming is implemented. The reference signal usually has a good correlation with the desired signal and the degree of correlation influences the accuracy and the convergence of the algorithm [15], [16].

In linear channel model the signal is received without fading effect. The addition of desired signal with interfering signal and noise makes the received signal, which is received at the receiver by the antenna elements

\[
X(t) = s(t) + a(\theta_{d1}) + a(\theta_{d2}) + I(t) + a(\theta_{i1}) + a(\theta_{i2})
\]

where \( \theta_{d1} \) and \( \theta_{d2} \) are angle of arrival for desired signal, \( \theta_{i1} \) and \( \theta_{i2} \) are angle of arrival for interfering signal.
Fig. 4. Shows the plot of MSE for LMS and KLMS algorithms. The above Plot shows that for KLMS signal converges in less (around 7) iterations where as for LMS it takes comparatively more (around 20) iterations. Fast convergence rate helps for quick response. To steer the beam with the rapidly moving user angle fast convergence rate is needed.

Array factor plotted by the help of KLMS and LMS algorithms. For beamforming the angle of arrival of desired signals are $10^\circ$and $40^\circ$ and for interfering signals the angle of arrival are $20^\circ$ and $50^\circ$.

Fig 5. shows semi-log plot of MSE for LMS and KLMS algorithms. Plot shows that the minimum error for KLMS is very less (0.0002) and for LMS it is (0.03). Thus KLMS will gives more accurate output compared to LMS. If angle of desired signal and interfering signal is close to each other then more accurate beamforming is needed.

Conclusion

Smart antenna plays an important role in 3G mobile communication system. Smart antenna uses the concept of adaptive beamforming to optimize the performance. In this report the performance of adaptive algorithm Kernel LMS (KLMS) and LMS is analysed for the linear phased array beamforming for both linear and non-linear channel. The simulation results show that for KLMS, the rate of convergence is faster and MSE is also very less compared to LMS, in less number of iterations. The rate of convergence plays an important role, when angle of arrival (AOA) of desired signal (user) is moving rapidly, to steer the beam with the desired signal. KLMS also gives higher gain than LMS, so that beam can cover longer distance and more users can be acknowledged. By high gain the requirement of the base station is decreases and less number of base station decreases the system complexity.

REFERENCES