NOVATEUR PUBLICATIONS
INTERNATIONAL JOURNAL OF INNOVATIONS IN ENGINEERING RESEARCH AND TECHNOLOGY [IJIERT]
ISSN: 2394-3696
VOLUME 2, ISSUE 6, JUNE-2015

SIMULATIONS OF ADAPTIVE ALGORITHMS FOR SPATIAL BEAMFORMING

Ms Juslin F
Department of Electronics and Communication,
VVIET, Mysuru, India.

ABSTRACT

The main aim of this paper is to simulate different types of Adaptive Algorithms for Spatial Beam forming, which is achieved by combining elements of a phased array in such a way that signals at particular angles experience constructive interference while others experience destructive interference. Here, simulations are done on different types of Adaptive Algorithms in MATLAB and Simulink to determine the desired signal from clutter/noise by updating its weight value for better execution speed and computational complexity and the characteristics of individual algorithms are compared and their area of applications. Adaptive filter is a filter that self-adjusts its transfer function according to an optimization algorithm driven by an error signal. The adaptive beamforming algorithms are used to update the weight vectors periodically to track the signal source in time varying environment by adaptively modifying the system's antenna pattern so that nulls are generated in the directions of the interference sources.

INDEX TERMS - Adaptive Algorithms, Spatial Beamforming, weight vectors, antenna pattern.

INTRODUCTION

When a real time signal is transmitted or received by a system, they are interfered with other signals of the environment of same frequency which is called as noise/clutter. This leads to co-channel interference and fading effects of the desired signal. In order to optimize the signal received by the system Adaptive filters are used. The adaptive filters are one which self-adjusts its transfer function by updating weights. Smart antennas are antenna arrays with smart signal processing algorithms used to identify spatial signal signature such as the direction of arrival of the signal, angle of arrival and use it to calculate beam forming vectors, to track and locate the antenna beam on the mobile/target. The system dynamically reacts to its environment to provide better signals and frequency usage for wireless communications. This is a new and promising technology in the field of wireless and mobile communications in which capacity and performance are usually limited by two major impairments like multipath and co-channel interference. Multipath is a condition that arises when a transmitted signal undergoes reflection from various obstacles in the environment. This gives rise to multiple signals arriving from different directions at the receiver. The receiver of the smart antenna identifies the spatial signal signature and calculates beam forming vectors, to track and locate the antenna beam on the mobile targets. The antenna could

ISSN: 2394-3696
VOLUME 2, ISSUE 6, JUNE-2015

optionally be any sensor. Smart antenna enables a higher capacity in wireless networks by effectively reducing multipath and co-channel interference. This is achieved by focusing the radiation only in the desired direction and adjusting itself to changing traffic conditions or signal environments. Smart antenna system is capable of efficiently utilizing the radio spectrum and to provide effective solution to the present wireless systems problems while achieving reliable and robust high speed high data rate transmission. The dual purpose of a smart antenna system is to augment the signal quality of the radio-based system through more focused transmission of radio signals while enhancing capacity through increased frequency reuse. From the frequency domain point of view, each spectral component of frequency corresponds to a different pointing direction. The signal that is being transmitted by the antenna is inculcated by some noise caused due to environmental effects or signal from other sources or the receiver noise itself. Hence at the receiver a decision has to be made whether the incoming signal is target or noise. This is done based on are three different ways:

i. Based on the magnitude of the received signal. If the received signal magnitude crosses predetermined threshold then we say that target is present else only noise is present.
ii. Receiver noise: In the absence of signal, if the noise is crosses threshold below some tolerable threshold value then a false alarm is triggered.
iii. External noise: If the noise is due to external source and much larger than the receiver noise then we consider different Adaptive filtering methods.

PROPOSED METHODOLOGY

Block diagram of adaptive filter is as illustrated in Fig-1, where $x(k)$ denotes the input signal, $y(k)$ is the adaptive-filter output signal, and $d(k)$ defines the desired signal. The error signal $e(k)$ is calculated as $d(k) - y(k)$. The error signal is then used to form a performance (or objective) function that is required by the adaptation algorithm in order to determine the appropriate updating of the filter coefficients. The minimization of the objective function implies that the adaptive-filter output signal is matching the desired signal in some sense.

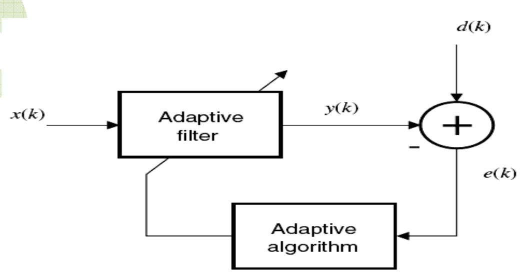

Fig-1: Block diagram of Adaptive filter

Fig-2 represents the classification of Adaptive Algorithms, an Adaptive Beamforming algorithm is mainly classified into two categories that is:

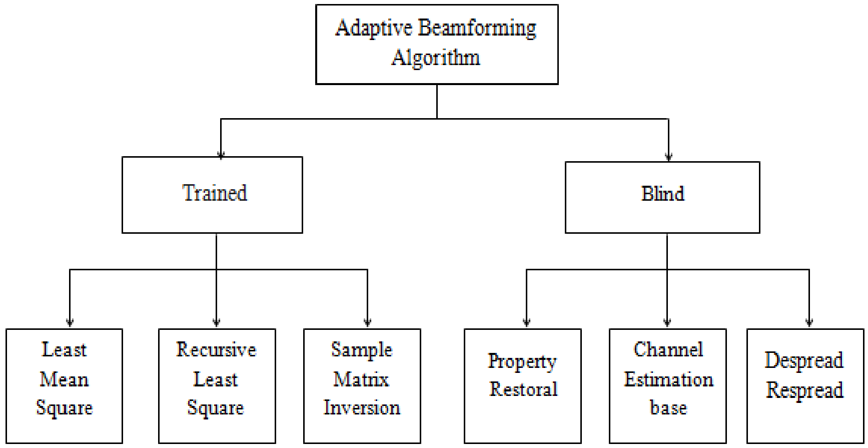

Fig-2: classification of Adaptive algorithms

Trained algorithm: Trained algorithms use training signal to adapt the weights of the array and minimize mean square error. The processor in the adaptive array has a pre-stored training signal and the array adapts its weights when the training signal is transmitted by the transmitter. This technique requires synchronization. These algorithms work very well, but the only cost paid is the excess transmission time or wastage of bandwidth. The trained algorithms are classified based on their adaptation criteria and they are least- mean squares method (LMS), sample matrix inversion (SMI) or least-squares method (LS) and recursive least-squares method (RLS). All these techniques minimize the squared error.
Blind algorithm: Unlike training algorithm blind algorithm do not require training signals to adapt their weights. Therefore these algorithms save transmission bandwidth. Blind algorithms can be classified as property restoral algorithms, channel estimation algorithms, and dispread and respread algorithms. Property restoral algorithms restore certain properties of the desired signal and hence enhance the SNR. The property that is being restored may be the modulus or the spectral coherence. Blind property restoral algorithms can be classified as Constant Modulus (CM) algorithm, Spectral self-Coherence Restoral (SCORE) algorithms, and decision directed (DD) algorithms. Channel estimation techniques use the knowledge of the special code properties of the spread spectrum signals to obtain estimates of the channel parameters. These techniques first estimate the channel parameters and then use the channel estimates to form beams in the direction of the desired signals.

A. Least Mean Square(LMS) Algorithm

The LMS algorithm is the most widely used algorithm invented in 1960 by Stanford University professor Bernard Widrow and his first Ph.D. student, Ted Hoff. The main features that attracted the use of the LMS algorithm are low computational complexity, proof of convergence in stationary environment, unbiased convergence, and stable behaviour when implemented with finite-precision arithmetic. LMS algorithms are a class of adaptive filter used to mimic a desired filter by finding the filter coefficients that relate to producing the least mean squares of the error signal. It is a stochastic gradient descent method in that the filter is only adapted based on the error at the current time. Therefore, the LMS algorithms require fewer computational resources and memory. The implementation of the LMS algorithms also is less complicated. Before the LMS algorithm reaches the steady-state behaviour, a number of iterations are spent in the transient part. During this time, the adaptive-filter coefficients and the output error change from their initial values to values close to that of the corresponding optimal solution. Applications of this algorithm are in linear prediction, System Identification. Given the input signal x(n) and output signal y(n), we can estimate the impulse response of the system or plant using the LMS algorithm, Interference Cancellation, Estimation of the time delay between two measured signals is a problem which occurs in a diverse range of applications including radar, sonar, geophysics and biomedical signal analysis.

The steps and procedure for LMS algorithm is discussed here with appropriate equations. Determining the learning step μ is an essential problem in LMS algorithm. If μ is selected too small the convergent speed of the algorithm will be very slow, or severe maladjustment can easily cause instability if μ is too large. The LMS algorithm requires only 2M complex multiplications per iteration, where M is the number of weights used in the adaptive array.

LMS algorithm is described by the following equations

Parameters: μ= convergence factor can vary from 0 to 1.
Initialization: 1. Direction of arrival of desired and noise signals
2. w(0) = initial value of weight
Computation: $y(n) = w(n)*x^t(n)$; y(n) is the system response.
$e(n) = d(n) - y(n)$
$w(n+1) = w(n) + (\mu * e(n) * w(n))$
w(n+1) is the updated weight.

x(n) is the input vector at sampling time n, w(n) is the coefficient vector of the adaptive filter, in first iteration w(0) is initialized. d(n) is the expected output value, e(n) is the deviation error, The dimension of w(n) is the length of the adaptive filter and superscript t represents transpose of matrix. The signal x(n) received by multiple antenna elements is multiplied with the coefficients in a weight vector which adjusted the phase and the amplitude of the incoming signal accordingly. The weighted signal is summed up, resulted in the array output y(n). An adaptive algorithm is then employed to minimize the error e(n) between a desired signal d(n) and the array output y(n).Convergent speed and stable miss-adjustment are two important factors for evaluating the performance of an adaptive filter. In generally, decreasing the learning step mu can reduce the stable miss-adjustment of the filter, but slows down the convergent progress. When increasing mu, the convergent speed is enhanced, but the miss-adjustment of the filter output is increased with undesirable effects. The value of mu can vary from 0 to 1 depending on required convergence speed.

B. Recursive Least Square(RLS) Algorithm

RLS algorithm was rediscovered by Plackett in1950, the original work of Gauss from 1821. In general, RLS can be used to solve any problem that can be solved by adaptive filters The RLS adaptive filter is an algorithm which recursively finds the filter coefficients that minimize a Weighted linear least squares cost function relating to the input signals. This is in contrast to other algorithms such as the least mean squares (LMS) that aim to reduce the mean square error. In the derivation of the RLS, the input signals are considered deterministic a deterministic system is a system in which no randomness is involved in the development of future states of the system. A deterministic model will thus always produce the same output from a given starting condition or initial state. Compared to most of its competitors, the RLS exhibits extremely fast convergence. However, this benefit comes at the cost of high computational complexity.

RLS algorithm can be explained by following equations

Parameters: λ=forgetting factor

Initialization: $w(0)$ = initial value of weight

Computation: $r(n) = \sum^{k}_{i=0}\lambda^{k-I}\, x(n)x^{t}(n)$; correlation of input signal

$p(n) = \sum^{k}_{i=0}\lambda^{k-I}\, x(n)d(n)$; correlation of input and desired signal

$e(n) = d(n)-y(n)$; error between desired and actual response

$w(n) = r^{-1}(n)p(n)$; updated new weight

$y(n) = w^{t}(n)x(n)$; new response of system

When the RLS algorithm operates in a time-varying environment, the suggested value of λ. is usually less than unity. This gives the RLS algorithm a finite memory where slow statistical changes in its environment can be tracked. However, changing the value of λ to less than unity modifies the behaviour of the algorithm by introducing maladjustment noise and delay in the formulation of the least squares estimate. Choosing λ the smaller λ is the smaller contribution of previous samples. This makes the filter more sensitive to recent samples, which means more fluctuations in the filter coefficients. The λ=1 case is referred to as the growing window RLS algorithm. In practice, λ is usually chosen between 0.98 and 1. It requires $4M^2 + 4M + 2$ complex multiplications per iterations, where M is the number of weights used in the adaptive array.

C. Sample Matrix Inversion (SMI) Algorithm

In 1974, Reed proposed SMI algorithm for adaptively adjusting the array weights. SMI algorithm uses block adaptation. The statistics are estimated from a temporal block of array data and used in an optimum weight equation. This algorithm also called as direct matrix inversion algorithm. The Direct Matrix Inversion algorithm provides good performance in a discontinuous traffic when the number of interferers and their positions remain constant during the duration of the block acquisition. Weight adaptation in the DMI algorithm can be achieved by using block adaptation technique where the adaptation is carried over disjoint intervals of time is the most common type. The overlapping block adaptation technique is computational intensive as adaptation intervals are not disjoint but overlapping. This technique gives better performance but number of inversions required is more when compared to block adaptation method. Another block adaptation technique is the block adaptation technique with memory.

This method utilizes the matrix estimates computed in the previous blocks. This approach provides faster convergence for spatial channels that are highly time correlated. This technique works better when the signal environment is stationary. The DMI algorithm employs direct inversion of the covariance matrix R and therefore it has faster convergence rate. The stability of the SMI algorithm depends on the ability to invert the large covariance matrix. Since SMI employs direct matrix inversion the convergence of this algorithm is much faster compared to the LMS algorithm. However, huge matrix inversions lead to computational complexities that cannot be easily overcome. Weight adaptation in the SMI algorithm can be achieved in different ways:

1. Block Adaptation: The adaptation is carried over disjoint intervals of time, is the most common type. This is well suited for a highly time varying signal environment as in mobile communications.
2. Overlapping block adaptation approach is computational intensive as adaptation intervals are not disjoint but overlapping. It provides better performance but has an increased number of inversions when compared to the above method.
3. Block adaptation with memory: This method utilizes the matrix estimates computed in the previous blocks. This approach provides faster convergence for spatial channels that are highly time correlated. It works better when the signal environment is stationary.

SMI algorithm equation can be explained as follows where d is interference signal and x is desired signal. The SMI algorithm using matrix inversion lemma requires $3.5M^2+M$ complex multiplications per iteration. Where M is the number of weights used in the adaptive array.

Initialization: $w(0)$ = weight vector initialization

Computation: $R = d(k)\, x^H(k)$ = co-variance matrix

$R = d(k)x(k)$ = correlation matrix

$E = RW-r$ = error

$W = R^{-1}\, r$=updated weight

D. Constant Modulus Algorithm(CMA)

CMA is a blind algorithm based on the idea to reduce systems overhead and maintain gain on the signal while minimizing the total output energy. As a result, number of bits for transmitting information is increased which leads to the increased capacity. This algorithm seeks for a signal with a constant magnitude like modulus within the received data vector and is only applicable for modulation scheme, which uses signal of equal power that includes phase and frequency modulated signals. The received data vector consists of desired signal plus interference and noise. Therefore, it can identify only one signal usually. During transmission, corruption from the channel and noise can distort this envelope. Using the constant modulus algorithm (CMA), the envelope of the adaptive array output can be restored to a constant by measuring the variation in the signal's modulus and minimizing it by using the cost function. μ represents the rate of adaptation, controlled by the processing gain of the antenna array.

If a large value μ of is taken then convergence becomes faster but makes the array system unstable/noisy. Conversely if a small value μ is taken then convergence becomes slow that is also not desirable. Therefore, value of is taken in between that satisfy the following conditions for good convergence and to avoid instability. The constant modulus cost function is a positive definite measure of how much the array output's envelope varies from the unity modulus used to minimize the result. The value of $p = 1$, $q = 2$ provides the deepest nulls of the four configurations and provides the best signal to interference noise ratio flow chart and equations of this algorithm is given below.

Initialization: 1. Direction of arrival of noise and desired signal

2. Initial weight $w(0)$

Computation: $J(K) = [||y(k)|^p - 1|^q]$ = cost function

$(J(k)) = x(k)(y(k) - y(k)/|y(k)|)$ = gradient of cost function

$w(k+1) = w(k) - \mu x(k)(y(k) - y(k)/|y(k)|)$ = new weight

$e(k) = y(k) - y(k)/|y(k)|$ = error

RESULTS

In this paper the main focus is made on different adaptive algorithms for the formation of beam in the desired direction and eliminating the noise received. To do so Adaptive Algorithms code is written in MATLAB. The output of various parameters like phase, amplitude, rate of error change and the Beam formation of different algorithm is compared at different angle. The error value of the same code is obtained by simulating Adaptive Algorithm in Simulink. Here for the simplicity of the execution we have considered three different angles i.e., (1) angle of desired signal (2) two angles at which noise is generated.

The complete procedure of executing the program is given below:

1. Input the angle of desired signal and noise angle values and select the Algorithm to be executed.
2. Beam formation at an angle of 100^0 for all the algorithms is illustrated in Figures below.
3. In LMS Algorithm the phase of noise and the desired signal are in phase, the rate of convergence depends on the μ factor. The output of LMS Algorithm is as shown in Fig-3.

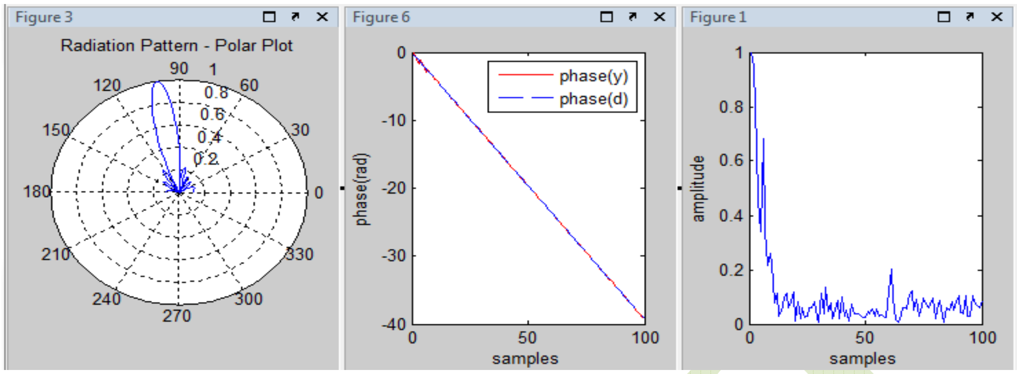

Fig-3: Output of LMS Algorithm

4. RLS Algorithm is the efficient algorithm as shown in Fig-4, where the rate of convergence is high and error is less.

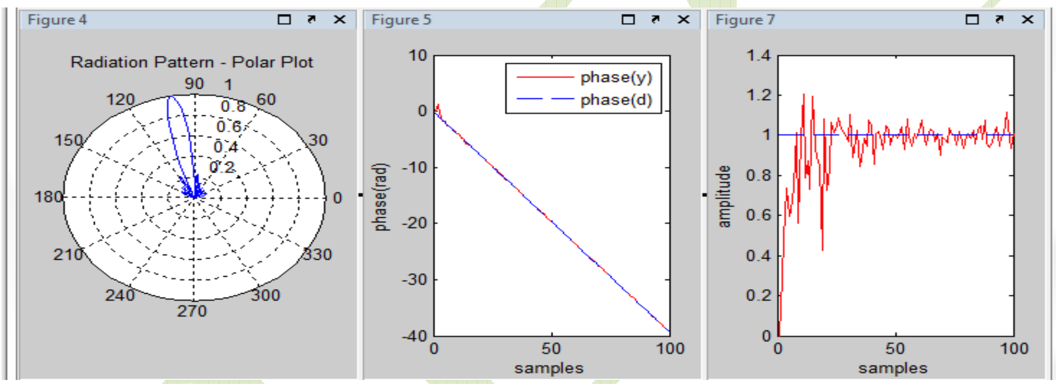

Fig-4: output of RLS Algorithm

5. SMI Algorithm involves only matrix operation, so the computation time and the error will be more. The Fig-5 illustrates the simulation results of SMI Algorithm.

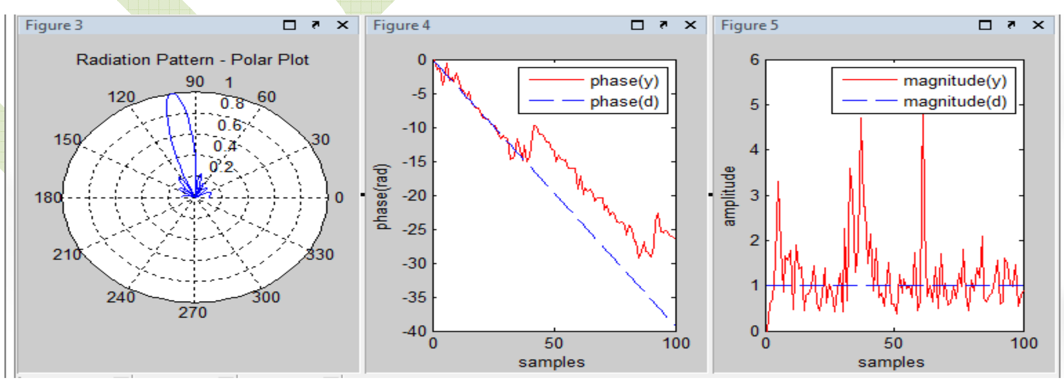

Fig-5: Results of SMI algorithm

ISSN: 2394-3696
VOLUME 2, ISSUE 6, JUNE-2015

6. CMA Algorithm is a Blind Adaptive Algorithm which is independent of training sequence. Hence the overhead will be less. The output of CMA Algorithm is as shown in Fig-6.

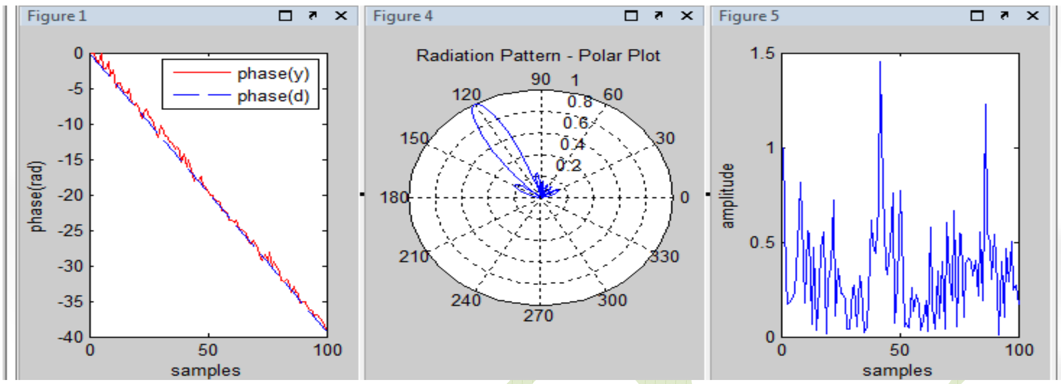

Fig-6: Results of CMA Algorithm

COMPARISON TABLE:

Algorithms	LMS	RLS	SMI	CMA
CONVERGENCE RATE	LOW	VERY HIGH	VERY LOW	HIGH
MAGNITDE OF ERROE	0.1769	6.27*10^-4	1.176	1.009
COMPLEXITY	HIGH	VERY HIGH	LOW	HIGH
NO. OF COMPUTATION	2M	$4M^2+4M+2$	$2M^2-1$	2M

Table-1: Comparison of Algorithms

CONCLUSION

Four Adaptive Algorithms are simulated for Spatial Beam forming both in MATLAB and Simulink; these tools have efficiency for signal processing. The results of simulated algorithms are compared with respect to various parameters like computation complexity, convergence rate, and magnitude of error. Accordingly the results showed that RLS to be the best algorithm in spite of computational complexity followed by LMS, CMA, SMI based on their magnitude of error. By trading off these parameters one choose the better algorithms for the requirement.

REFERENCES

[1] B.Widrow, P.E.Mantely,L.J.Griffiths, and B.B.Goode, *"Adaptive Antenna Systems",Proc. IEEE, Vol.55,No.12, pp.2143-21559,Aug. 1967.*

[2] A.Alexiou and M. Haardt, "*Smart Antenna technologies for future wireless systems: trends and challenges", Communications Magzine, IEEE, vol.42, pp. 90-97, 2004.*

[3] Ch. Santhi Rani, Dr. P.V. Subbaia, Dr. K. Chennakesava Reddy, "*LMS and RLS algorithm for Smart Antennas in a CDMA Mobile Communication Environment"*.

[4] M.Chryssomallis, "*Smart Antennas", IEEE Antennas Propagation Magzine Vol. 42, no. 3, June 2000.*

[5] P.J. Burt. *Smart sensing within a pyramid vision. Proceedings of the IEEE, 76(8):1006- 1015, 1988.*