

Performance Comparison of LMS, SMI and RLS Adaptive Beamforming Algorithms for Smart Antennas

¹L. Surendra, ²Syed. Shameem, ³Dr. Habibullah Khan

^{1,2,3}Dept. of ECE, KL University, Vijayawada, AP, India

Abstract

Smart Antenna systems have become a practical reality after the advent of powerful, low cost and digital signal processing components. It is recognized as promising technologies for high user capacity in 3G wireless networks by effectively reducing multipath and co-channel interference. The core of smart antenna is the selection of smart algorithms in adaptive array. These algorithms use different criterion to adapt the system for better performance and steer the beam towards signal of Interest. An algorithm with less complexity, low computation costs, good convergence rates usually preferred. Smart essentially means computer control of the antenna performance. Smart antenna holds the promise for improved radar systems, improved system capacities with mobile wireless and wireless communications through the implementation of space division multiple access (SDMA). This research paper investigates the performance of adaptive beamforming algorithms such as LMS(Least Mean Square), SMI(Sample Matrix Inversion) and RLS(Recursive Least Square) are to be implemented in Matlab.

Keywords

Smart antennas, switched beamforming, Adaptive beamforming, LMS(Least Mean Square), SMI(Sample Matrix Inversion), RLS(Recursive Least Square).

I. Introduction

Smart antenna technology offers a significantly improved solution to reduce interference levels and improve the system capacity. Each user's signal is transmitted and received by the basestation only in the direction of that particular user. This drastically reduces the overall interference level in the system. A smart antenna system, consists of an array of antennas that together direct different transmission/reception beams towards each user in the system. This method of transmission and reception is called beamforming and is made possible through smart (advanced) signal processing at the baseband. Beamforming is a signal processing technique used in sensor arrays for directional signal transmission or reception. This is achieved by combining elements in the array in a way where signals at particular angles experience constructive interference and while others experience destructive interference. Beamforming can be used at both the transmitting and receiving ends in order to achieve spatial selectivity. The improvement compared with an omnidirectional reception/transmission is known as the receive/transmit gain (or loss). Beamforming can be used for both radio or sound waves. It has found numerous applications in radar, sonar, seismology, wireless communications, radio astronomy, speech, acoustics, and biomedicine. Adaptive beamforming is used to detect and estimate the signal-of-interest at the output of a sensor array by means of data-adaptive spatial filtering and interference rejection.

II. Beamforming Techniques

Beamforming techniques can be broadly divided into two categories

1. Conventional(switced beam) beamformers

2. Adaptive beamformers or phased array

- Desired signal maximization mode
- Interference signal minimization or cancellation mode

A. Switched Beamforming

Conventional beamformers use a fixed set of weightings and time-delays (or phasings) to combine the signals from the sensors in the array. It primarily using only information about the location of the sensors in space and the wave directions of interest. In contrast, adaptive beamforming techniques generally combine this information with properties of the signals actually received by the array to improve rejection of unwanted signals from other directions. This process may be carried out in either the time or the frequency domain.



Fig. 1: Switched Beam Strategy

B. Adaptive Beamforming

Adaptive beamforming is a technique in which an array of antennas is exploited to achieve maximum reception in a specified direction by estimating the signal arrival from a desired direction. It not only direct the beam in desired direction but also introduces nulls at interfering directions. Adaptive arrays are able to dynamically update their weights to the changing signal conditions. Thus the weights are usually computed according to the characteristics of the received signals, which are periodically sampled. The ability to self-update is extremely desirable in many applications where the signals change, such as in a mobile communications system, radar target tracking etc.

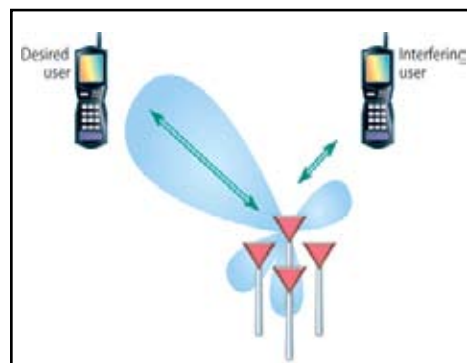


Fig. 2: Adaptive Beam Strategy

III. LMS (Least Mean Square) Algorithm

The Least Mean Square (LMS) algorithm is an adaptive algorithm, which uses a gradient-based method of steepest decent. It uses the estimates of the gradient vector from the available data. It incorporates an iterative procedure that makes successive

corrections to the weight vector in the direction of the negative of the gradient vector which eventually leads to the minimum mean square error compared to other algorithms. LMS algorithm is relatively simple it does not require correlation function calculation nor does it require matrix inversions. Consider a Uniform Linear Array (ULA) with N isotropic elements, which forms the integral part of the adaptive beamforming system as shown in the fig. 3. The output of the antenna array $x(t)$ is,

$$x(t) = s(t)a(\theta_0) + \sum_{i=1}^{N_u} u_i(t)a(\theta_i) + n(t) \quad (1)$$

$S(t)$ denotes the desired signal arriving at angle θ_0 and $u_i(t)$ denotes interfering signals arriving at angle of incidences θ_i respectively. $a(\theta_0)$ and $a(\theta_i)$ represents the steering vectors for the desired signal and interfering signals respectively. Therefore it is required to construct the desired signal from the received signal amid the interfering signal and additional noise $n(t)$. From the method of steepest descent, the weight vector equation is given by

$$w(n+1) = w(n) + 1/2\mu[-\Delta(E\{e^2(n)\})] \quad (2)$$

$$w(n+1) = w(n) + \mu x(n)[d^*(n) - x^h(n)w(n)] \quad (3)$$

$$w(n+1) = w(n) + \mu x(n)e^*(n) \quad (4)$$

Where μ is the step-size parameter and controls the convergence characteristics of the LMS algorithm $e^2(n)$ is the mean square error between the beamformer output $y(n)$ and the reference signal which is given by

$$e^2(n) = [d(n) - w^h x(n)]^2 \quad (5)$$

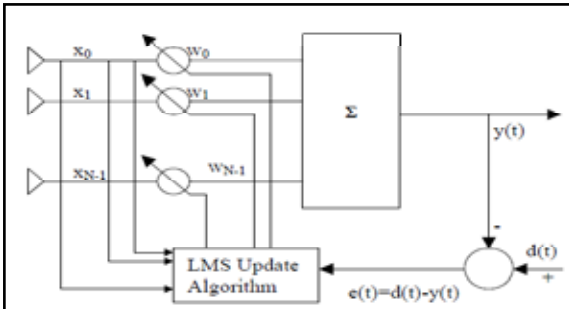


Fig. 3: Adaptive Beamforming System

The gradient vector in the above weight update equation can be computed as

$$\nabla_w(E\{e^2(n)\}) = -2r + 2Rw(n) \quad (6)$$

In the method of steepest descent the biggest problem is the computation involved in finding the values r and R matrices in real time. The LMS algorithm on the other hand simplifies this by using the instantaneous values of covariance matrices r and R instead of their actual values i.e.

$$R(n) = x(n)x^h(n) \quad (7)$$

$$r(n) = d^*(n)x(n) \quad (8)$$

Therefore the weight update can be given by the following equation,

$$w(n+1) = w(n) + \mu x(n)[d^*(n) - x^h(n)w(n)] \quad (9)$$

$$w(n+1) = w(n) + \mu x(n)e^*(n) \quad (10)$$

The LMS algorithm is initiated with an arbitrary value $w(0)$ for the weight vector at $n=0$. The successive corrections of the weight vector eventually leads to the minimum value of the mean

squared error. Therefore the LMS algorithm can be summarized in following equations

$$\text{Output: } y(n) = w^h x(n) \quad (11)$$

$$\text{Error: } d^*(n) - y(n) \quad (12)$$

$$\text{Weight: } w(n+1) = w(n) + \mu x(n)e^*(n) \quad (13)$$

The LMS algorithm initiated with some arbitrary value for the weight vector is seen to converge and stay stable for $0 < \mu < 1/\lambda_{\max}$. Where $\lambda_{\max}$ is the largest eigenvalue of the correlation matrix R . The convergence of the algorithm is inversely proportional to the eigenvalue spread of the correlation matrix R . When the eigenvalues of R are widespread, convergence may be slow. The eigenvalue spread of the correlation matrix is estimated by computing the ratio of the largest eigenvalue to the smallest eigen value of the matrix. If μ is chosen to be very small then the algorithm converges very slowly. A large value of μ may lead to a faster convergence but may be less stable around the minimum value. An upper bound for μ based on several approximations as $\mu \leq 1/(3\text{trace}(R))$.

IV. SMI (Sample Matrix Inversion) Algorithm

The Sample matrix inversion is also called as Direct Matrix Inversion Algorithm. Sample matrix is a Time average estimate of the array correlation matrix using k -time samples. If the random process is ergodic in the correlation, the time average estimate will equal the actual correlation matrix. The drawbacks of LMS adaptive beamforming algorithm must go through many iterations before satisfactory convergence is achieved. If the signal characteristics are rapidly changing, the LMS adaptive algorithm may not allow tracking of the desired signal in a satisfactory manner. The rate of convergence of the weights is dictated by the eigenvalue spread of the array correlation matrix. The LMS algorithm did not converge until after 70 iterations. 70 iterations corresponded to more than half of the period of the waveform of interest. One possible approach to circumventing the relatively slow convergence of the LMS scheme is by use of SMI method. The optimum Wiener solution as

$$w_{\text{opt}} = R_{xx}^{-1} r \quad (14)$$

The equation for correlation matrix r is given by equation

$$r = E[d^* x] \quad (15)$$

The equation for co-variance matrix R is given by

$$R_{xx} = E[xx^H] \quad (16)$$

we can estimate the correlation matrix by calculating the time average such that

$$R_{xx} = 1/K \sum_{k=1}^K x(k)x^H(k) \quad (17)$$

Where K is the observation interval.

The correlation vector r can be estimated by

$$r(k) = 1/K d^*(k)X_K(k) \quad (18)$$

The K -length block of data is called a block adaptive approach. Uses the weights block-by-block. It is easy in matlab to calculate the array correlation matrix and the correlation vector by the following procedure. Define the matrix $K(k)$ as the k th block of x vectors ranging over K -data snapshots. The SMI weights can then be calculated for the k th block of length K as

$$w_{SMI}(k) = R_{xx}^{-1}(k)r(k) \quad (19)$$

V. Recursive Least Square (RLS) Algorithm

The SMI technique has several drawbacks. Even though the SMI method is faster than the LMS algorithm, the computational burden and potential singularities can cause problems. However we can recursively calculate the required correlation matrix and the required correlation vector. The estimate of the correlation matrix and vector was taken as the sum of the terms divided by the block length K. When we calculate the weights, the division by K is cancelled by the product $R_{xx}^{-1}(k)r(k)$. Thus, we can rewrite the correlation matrix and the correlation vector omitting K as

$$R_{xx}(k) = \sum_{i=1}^k x(i)x^H(i) \quad (20)$$

$$r(k) = \sum_{i=1}^k d^*(i)x(i) \quad (21)$$

where k, is the block length and last time sample k and $R_{xx}(k), r(k)$ is the correlation estimates ending at time sample k. Both summations Equations use rectangular windows, thus they equally consider all previous time samples. Since the signal sources can change or slowly move with time, we might want to deemphasize the earliest data samples and emphasize the most recent ones. This can be accomplished by modifying Equations such that we forget the earliest time samples. This is called a weighted estimate. Thus

$$\hat{R}_{xx}(k) = \sum_{i=1}^k \alpha^{k-i} x(i)x^H(i) \quad (22)$$

$$\hat{r}_{xx}(k) = \sum_{i=1}^k \alpha^{k-i} d^*(i)x(i) \quad (23)$$

where α is the forgetting factor. The forgetting factor is also sometimes referred to as the exponential weighting factor. α is a positive constant such that $0 \leq \alpha \leq 1$. When $\alpha = 1$ also indicates infinite memory. Let us break up the summation in Equations into two terms, the summation for values up to $i = k-1$ and last term $i=k$.

$$\hat{R}(k) = \alpha \sum_{i=1}^{k-1} \alpha^{k-1-i} x(i)x^H(i) + x(k)x^H(k) \quad (24)$$

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. It is almost ten times faster compared to LMS.

VI. Results

In 6.1, 6.2, 6.3 performed comparison of adaptive algorithms such as LMS, SMI, RLS.

A. Results for LMS Algorithm

The LMS algorithm was simulated by using matlab. Linear array is used for simulation purpose that is 20 elements and distance separation between elements is 0.5λ . When the angle of arrival of the desired user is at 45° and interfering user at 60° . The fig. 4 & 5, shows at interfering user null and maximum radiation along desired user by increasing number of elements sharp beams will be formed. The optimum weight vector for N=20 elements is

Table 1: Weights for Each Element in the Linear Array

Element no	Weights
1	w1 = 1
2	w2 = -0.54681+0.72269i
3	w3 = -0.26776-0.78213i
4	w4 = 0.7693+0.16813i
5	w5 = -0.54804+0.58855i
6	w6 = -0.21553-0.84286i
7	w7 = 0.89401+0.35259i
8	w8 = -0.91231+0.52333i
9	w9 = 0.20811-1.1031i
10	w10 = 0.71059+0.91806i
11	w11 = -1.1579-0.084405i
12	w12 = 0.80401-0.78337i
13	w13 = 0.068438+1.0495i
14	w14 = -0.78814-0.54992i
15	w15 = 0.82145-0.2865i
16	w16 = -0.18733+0.78208i
17	w17 = -0.56548-0.54801i
18	w18 = 0.79974-0.2094i
19	w19 = -0.29978+0.85523i
20	w20 = -0.55297-0.8332i

$$\mu=0.0126, d=0.5\lambda$$

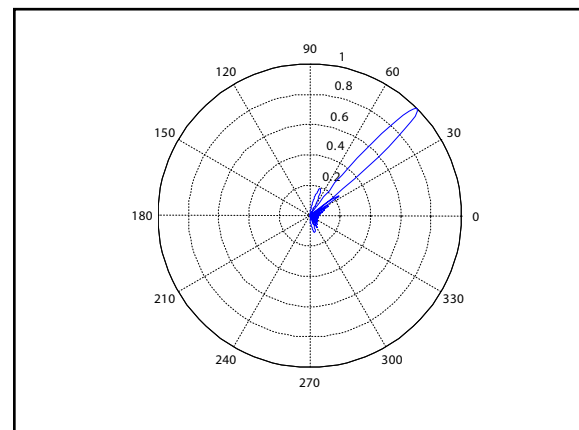


Fig. 4: Polar Plot Representing Main Beam Along Desired User at 45° and Null Along Interfering User at 60°

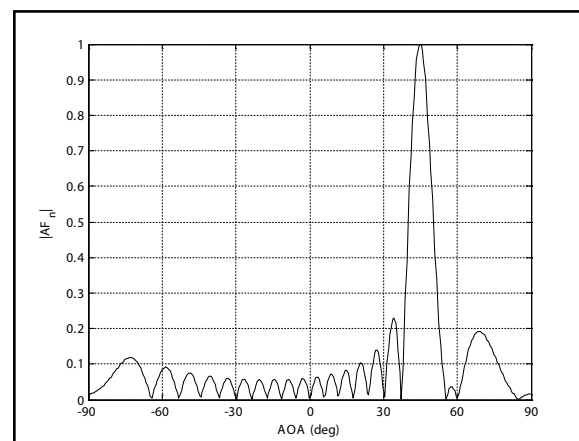


Fig. 5: Normalized Array Factor Plot for LMS Algorithm N=20 Elements. When the AOA of Desired User at 45° and Interfering user at 60° and $d=0.5\lambda$.

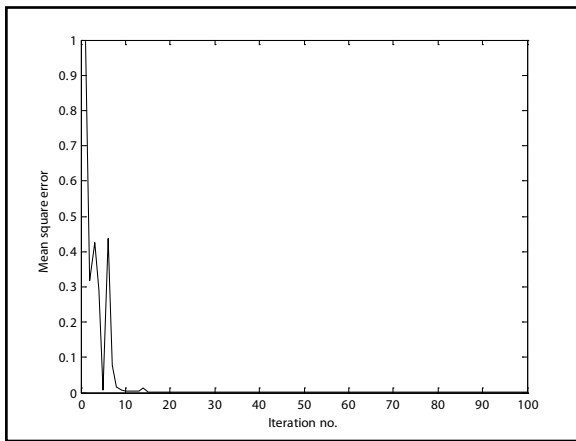


Fig. 6: Mean Square Error for N=20 Elements

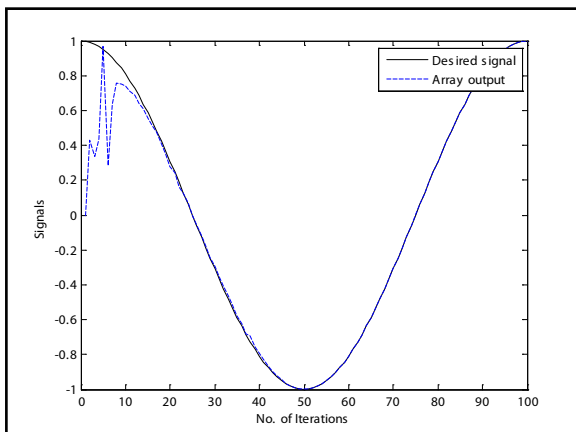


Fig. 7: Acquisition and Tracking of Desired Signal for N=20 Elements

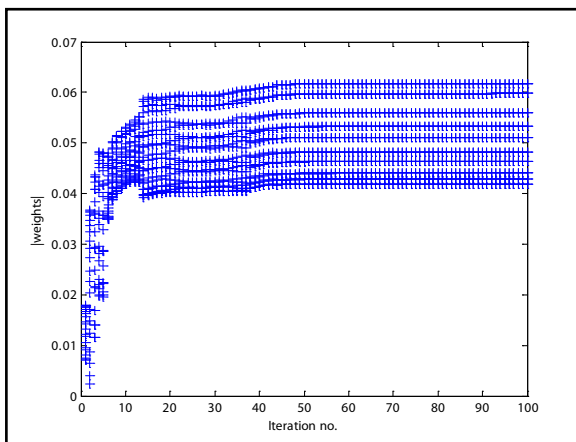


Fig. 8: Magnitude of Array Weights for N=20 Elements

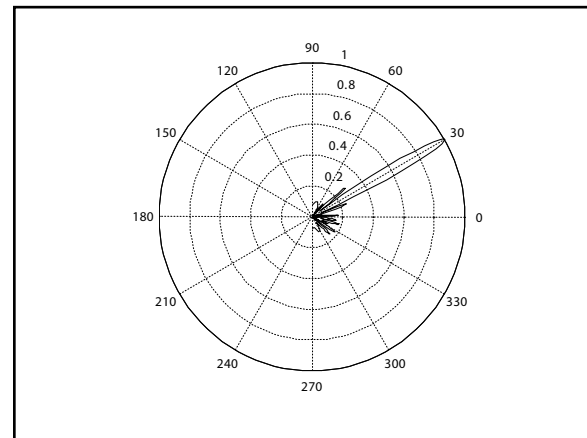
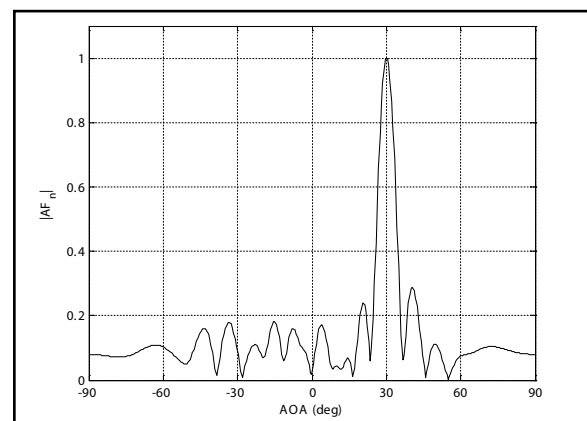
B. Results for SMI Algorithm

The SMI algorithm was simulated by using matlab. Linear array is used for simulation purpose that is 20 elements and distance separation between elements is 0.5λ . When the angle of arrival of the desired user is at 30° and interfering user at 55° . The figure 9&10 shows at interfering user nulls and maximum radiation along desired user by increasing number of elements sharp beams will be formed. The optimum weight vector for N=20 elements is

Table 2: Weights for Each Element in the Linear Array

Element no	Weights
1	$w1=1.0000$
2	$w2=0.5716-0.4195i$
3	$w3=-0.9084+ 0.0718i$
4	$w4=0.1730+0.5590i$
5	$w5=0.6855+0.2428$
6	$w6=0.1792-1.0187i$
7	$w7=0.6601+0.0279i$
8	$w8=0.0390+0.5469i$
9	$w9=0.5619+0.2631i$
10	$w10=-0.0763-0.6262i$
11	$w11=-0.9918-0.1182i$
12	$w12=-0.2338+0.5076i$
13	$w13=0.6703+0.0709i$
14	$w14=-0.2325-0.9776i$
15	$w15=-0.8954 - 0.5556i$
16	$w16=0.0671+1.0477i$
17	$w17=0.8839+0.2721i$
18	$w18=-0.0053 - 0.7281i$
19	$w19= -0.5897 + 0.0848i$
20	$w20= -0.0836 + 0.4757i$

$e=1.0021 - 0.0029i$.

Fig. 9: Polar Plot Representing Main Beam Along Desired User at 30° and Null Along Interfering User at 55° .Fig. 10: Normalized Array Factor Plot for SMI Algorithm N=20 Elements, when the AOA of Desired User at 30° and Interfering User at 55° and $d=0.5\lambda$.

C. Results for RLS Algorithm

The RLS algorithm was simulated using matlab. Linear array is used for simulation purpose that is 20 elements and distance separation between elements is 0.5λ . When the angle of arrival of the desired user is at 30° and interfering user at -30° . The fig. 11 & 12, shows at interfering user nulls and maximum radiation along desired user by increasing number of elements sharp beams will be formed.

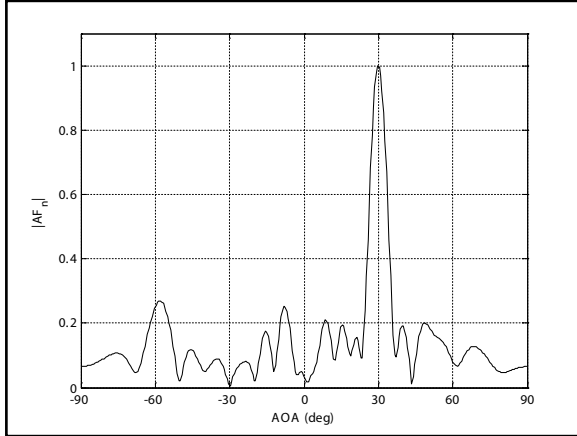


Fig. 11: Normalized Array Factor Plot for RLS Algorithm N=20 Elements, When the AOA of Desired user at 30° and Interfering User at -30° and $d=0.5\lambda$.

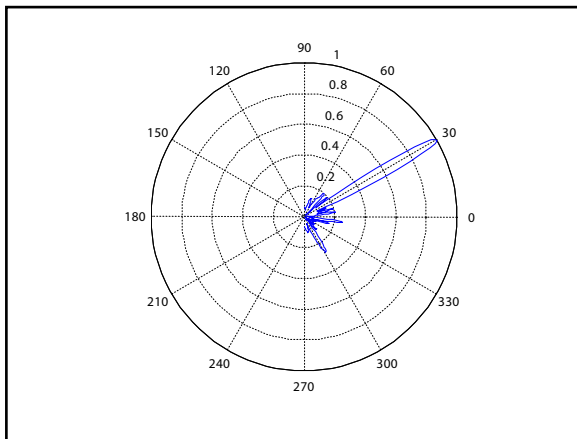


Fig. 12: Polar Plot Representing Main Beam Along Desired User at 30° and Null Along Interfering user at -30° .

VII. Conclusions

In this paper, LMS (Least Mean Square), SMI (Sample Matrix Inversion) and RLS (Recursive Least Square) adaptive beamforming algorithms are implemented in matlab for N=20 elements and $d=0.5\lambda$. From the results it is observed by increasing array elements improves the directivity in desired direction. The convergence of the LMS algorithm is directly proportional to the step size parameter μ . In comparison with LMS had more side lobes compared to SMI and RLS. RLS had more sidelobes compared to SMI.

References

- [1] L.C Godara, "Smart Antennas", published by CRC press in 2004.
- [2] Liberti, J., T. Rappaport, "Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications", Prentice Hall, New York, 1999.

- [3] Lal.C.Godara, "Applications of Antenna Arrays to Mobile Communications, part I. Performance Improvement, Feasibility, and System Considerations," Proceeding of the IEEE, Vol. 85, No. 7, pp. 1031-1060, July 1997.
- [4] Lal.C.Godara, "Applications of Antenna Arrays to Mobile Communications, part II. Beamforming and Direction-of-Arrival Considerations", Proceeding of the IEEE, Vol. 85, No. 8 August 1997.
- [5] Monzingo, R., T. Miller, "Introduction to Adaptive Arrays", Wiley Interscience, John Wiley & Sons, New York, 1980.



Mr. L Surendra was born in Vijayawada India, 1985. He obtained his B.Tech from JNTU Hyderabad. M.Tech from Gitam University, Visakhapatnam. He worked as Assistant professor in SRK Institute of Technology Vijayawada from 2010 to May 2012. Currently he is working as Assistant Professor in the department of ECM of K L University. His present research interests include Microwave Communications, Smart Antennas, Wireless Communications.



Syed Shameem was born in Vijayawada, Krishna District, Andhra Pradesh, India. He received his B.Tech degree in ECE from JNTU, Hyderabad and M. Tech from Acharya Nagarjuna University, Guntur. His research interests include Antennas, Radar Systems and Wireless Communications. Presently, he is working as an Associate Professor in KL University, Guntur.



Dr. Habibullah Khan born in India, 1962. He obtained his B.E. from VR Siddhartha Engineering College, Vijayawada during 1980-84. M.E from C.I.T, Coimbatore during 1985-87 and PhD from Andhra University in the area of antennas in the year 2007. He is having more than 20 years of teaching experience and having more than 20 international, national journals/conference papers in his credit. Dr. Habibullah Khan presently working as Head of the ECE department at K.L. University. He is Member Board of Studies in ECE and EIE of Acharya Nagarjuna University, Guntur. He is a fellow of I.E.T.E, Member IE, SEMCE and other bodies like ISTE. His research interested areas include Antenna system designing, microwave engineering, Electro magnetics and RF system designing.