Asian Journal of Information Technology 15 (16): 3108-3124, 2016
ISSN: 1682-3915

Efficient FPGA Architecture for RLS Algorithm Based Adaptive Beam Forming for Smart Antenna System

C. Thiripurasundari and V. Sumathy
Department of ECE, Madha Engineering College, 600069 Chennai, Tamil Nadu, India

Abstract: Smart antennas are an array of antennas used for directional transmission/reception signal or beams toward each user in the system. Beamforming is a general signal processing technique used to control the directionality of the reception or transmission of a signal on a transducer phased array. The magnitude and phase of the signal to and from each antenna is adjusted by multiplying each user signal by complex weights. Adaptive beamforming can be used as a method for estimating an unknown waveform from a source impinging on an array of sensors. In the adaptive beamforming the weights are computed and adaptively updated in real time. Different beamforming algorithms like side-lobe cancellers, Linearly constrained minimum variance (Lcmv), Least Mean Squares (LMS), Recursive Least Square (RLS) and Direction of Arrival (DOA) exist in literature. Among the Direction of Arrival (DOA) algorithms, MUSIC and ESPRIT play the most important role. This study presents study and analyzes an FPGA architecture Adaptive Beamforming of Smart Antenna system based on MUSIC algorithm. This study focuses on adaptive beam forming approach used in smart antennas and Recursive Least Square (RLS) algorithm used to compute the complex weights. Designed model is simulated using Matlab/Simulink software and the Synthesis report for the designed system is generated from XSG tool.

Key words: Smart antenna system, MUSIC algorithm, Beamforming, LMS, RLS algorithm, FPGA, Xilinx system generator, CORDIC processor

INTRODUCTION

One of the most promising technologies or algorithms used to identify spatial signal signature such as the Direction of Arrival (DOA) of the signal is Smart antenna. DOA is the procedure of identifying the angle of the incoming signal and can be achieved by litigating the relative phases of the incident signal detected by each of the antenna elements. Thus for the beamforming DOA is used (Ardi *et al*., 2003a, b). With use of MUSIC algorithms smart antennas add a new possibility of user separation by space through Space Division Multiple Access (SDMA). Since, FPGA-based processing has flexibility of the hardware plus the ability to modify the implementation of the algorithm between hardware and software, it is becoming most appealing technology. As there is no need to invert matrices in RLS algorithm as the inverse correlation matrix is computed directly and another benefit, i.e., extremely fast convergence.

This system takes advantage of diversity effect at the source and the destination i.e. transmitter or receiver or both. By focusing the radiation only in the desired direction and making it adjustable itself for changing traffic conditions or signal environments and reducing multipath and co-channel interference, we can achieve higher data rates and higher capabilities in wireless networks. Smart antenna consists of radiating elements arranged in the form of an array (Shubair *et al*., 2007; Shubair and Jessmi, 2005). These elements are processed adaptively to exploit the spatial domain of the mobile radio channel. The signals from these elements are combined to form a mobile beam pattern that follows the desired user. In a Smart antenna system the arrays by themselves are not smart, it is the digital signal processing that makes them smart. There are basically two approaches, Switched beam and Adaptive Arrays to implement antennas that dynamically change their antenna pattern to mitigate interference and multipath affects while increasing coverage and range. The simpler approach is Switched beam. It provides a considerable increase in network carrying capacity as compared to traditional Omni directional antenna systems or sector-based systems. This system forms multiple fixed beams with heightened sensitivity in particular directions. These beams are overlapping beams that cover the surrounding area. The base station determines the beam that is best aligned in the signal-of-interest direction and switches to that beam to communicate with the user. In the other hand, Adaptive system is the most advanced technique. This system tracks the mobile user continuously by steering the main

Corresponding Author: Thiripurasundari, Department of ECE, Madha Engineering College, 600069 Chennai, Tamil Nadu, India

beam towards the user. At the same time this system forms nulls in the directions of the interfering signal. Like switched beam systems, gthey also incorporate arrays. The received signal from each of the distributed antenna elements in the array is multiplied by a weight. This computation is done by digital processing unit in which adaptive algorithm is preprogrammed. The weights are complex in nature to adjust the amplitude and phase. The output of the array is formed by combining these signals.

There are many techniques of beamforming such as iterative and non-iterative. Direct matrix inversion in non-iterative technique and RLS is of iterative one. Now unique interests in beam forming for smart antennas are the Least Mean Squares (LMS) and the Recursive Least Squares (RLS) algorithms. There are various ways of estimating the direction like correlation, Maximum Likelihood, MUSIC, ESPRIT and Matrix Pencil. A common a estimation algorithm is MUSIC (Multiple Signal Classification), (Godara, 1997). Study by Lounici *et al.* (2013) give study about implementations for multiuser detection for the base station.

Proposed research work focuses the estimation of Adaptive weights for Beamforming of smart antenna system for RLS algorithm by designing architecture on SIMULINK/MATLAB environment. The MUSIC spectrum is obtained by designing FPGA architecture in XILINX ISE and synthesizes results is obtained by XILINX SYSTEM GENERATOR and results are compared with existing results. The beamforming polar plot is executed using SIMULINK /MATLAB tool.

Literature review: Gu *et al.* (2013) presented a nontraditional approach for estimating and tracking signal Direction-of-Arrival (DOA) using an array of sensors. The presented method consists of two stages: in the first stage, the sources modeled by AutoRegressive (AR) processes are estimated by the celebrated Kalman filter and in the second stage, the efficient QR-Decomposition-based Recursive Least Square (QRD-RLS) technique was employed to estimate the DOAs and AR coefficients in each observed time interval.

Dikmese *et al.* (2011) have studied implementations of some CDMA compatible beamforming algorithms, namely Least Mean Square (LMS), Constant Modulus (CM) and Space Code Correlator (SCC) algorithms, using Xilinx's Virtex family FPGAs. This study exhibited feasibility of implementing even simple, practical and computationally small algorithms based on today's most powerful FPGA technologies. 16 and 32 bits floating point implementations of the algorithms were investigated using both Virtex II and Virtex IV FPGAs. CDMA2000 reverse link baseband signal format was used in the signal modeling. Randomly changing fading and Direction-of-Arrivals (DOAs) of multipath were considered as a channel condition. The implementation results in terms of beamforming accuracy, FPGA resource utilization, weight vector computation time and DOA estimation error were presented.

Yang and Bohme (1992) presented a computationally simple and efficient subspace-based Adaptive Method for Estimating Directions-of-arrival (AMEND) for multiple coherent narrowband signals impinging on a Uniform Linear Array (ULA) where the previously proposed QR-based method was modied for the number determination, a new Recursive Least-Squares (RLS) algorithm was proposed for null space updating and a dynamic model and the Luen berger state observer were employed to solve the estimate association of directions automatically. Furthermore, an analytical study of the RLS algorithm was carried out to quantitatively compare the performance between the RLS and Least-Mean-Square (LMS) algorithms in the steady-state. The theoretical analyses and effectiveness of the proposed RLS algorithm were substantiated through numerical examples.

Wang and Lamare (2010) and Rader (1992) presented constrained adaptive algorithms based on the Conjugate Gradient (CG) method for adaptive beamforming. The presented algorithms were derived for the implementation of the beamformer according to the minimum variance and constant modulus criteria subject to a constraint on the array response. A CG-based weight vector strategy is created for enforcing the constraint and computing the weight expressions. The devised algorithms avoid the covariance matrix inversion and exhibit fast convergence with low complexity. A complexity analysis compared the presented algorithms with the existing ones. The convergence properties of the CCM criterion were studied, conditions for convexity were established and a convergence analysis for the presented algorithms was derived.

Shaukat *et al.* (2009) and Haykin presented a systematic comparison of the performance of different Adaptive algorithms for beamforming for Smart Antenna System. Simulation results revealed that training sequence algorithms like Recursive Least Squares (RLS) and Least Mean Squares (LMS) are best for beamforming (to form main lobes) towards desired user but they have limitations towards interference rejection. While Constant Modulus Algorithm (CMA) has satisfactory response towards beamforming and it gives better outcome for interference rejection, but Bit Error Rate (BER) is maximum in case of single antenna element in CMA. It was verified that convergence rate of RLS is faster than LMS so RLS is proved the best choice. The effect of changing step size for LMS algorithm had also been studied.

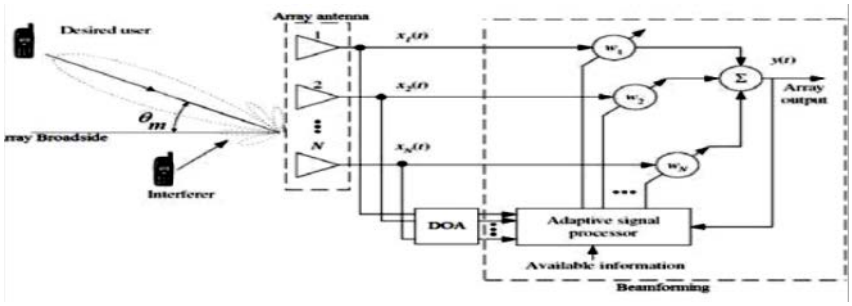

Fig. 1: A functional block diagram of a smart antenna system

MATERIALS AND METHODS

Recently, several beamforming techniques have been utilized by the experts. Above literature survey presents some basic beamforming algorithms. A beamformer is a processor used in conjunction with an array of antennas to provide a versatile form of spatial filtering. The motivation behind the research is to develop the less complex, FPGA realization-based architecture for DOA using MUSIC and beam forming using RLS in wireless communication. With FPGAs, this system can be customized for the designated task and for this reason it is highly efficient. In this research, we have considered MUSIC algorithm that is used to estimate the Direction of Arrival (DOA) of the input signals. MUSIC deals with the decomposition of covariance matrix into two orthogonal matrices, i.e., signal-subspace and noise-subspace. And the estimation of DOA is performed from one of these subspaces, assuming that noise in each channel is highly uncorrelated. The well-known LMS and NLMS are the most widely used adaptive algorithms due to their simplicity, robustness and low computational complexity but cannot meet convergence criteria. Recursive Least Squares algorithms are known to exhibit better performances. Hence the best choice is the Recursive Least Squares (RLS) algorithm. In this study recursive least square algorithm is used for the adaptive weight estimation. It finds the filter coefficients that minimize a weighted linear least squares cost function relating to the input signals. The RLS algorithm performs at each instant an exact minimization of the sum of the squares of the desired signal estimation errors. For developing the VLSI architecture of MUSIC algorithm, the separate modules will be developed for this algorithm and it will be modeled using MATLAB\SIMULINK model. Then, each module will be separately simulated with XILINX system generator to find the FPGA resource utilization summary of architecture .The RLS Simulink model can provide the beam forming graphs in a clear way.

Theory of smart antenna system: A smart antenna system at the base station of a cellular mobile system is depicted in Fig.1.It consists of a uniform linear antenna array for which the current amplitudes are adjusted by a set of complex weights using an adaptive beamforming algorithm. The adaptive beamforming algorithm optimizes the array output beam pattern such that maximum radiated power is produced in the directions of desired mobile users and deep nulls are generated in the directions of undesired signals representing co-channel interference from mobile users in adjacent cells. Prior to adaptive beam forming, the directions of users and interferes must be obtained using a Direction-of-Arrival (DOA) estimation algorithm (McWhirter, 1983).

High accuracy, precision and data rates are required in many wireless communication applications. There are many attempts in implementing DOA estimation, beamforming using FPGA-based hardware. It strikes a balance between ASICs and DSPs because they have programmability of software with performance capacity approaching that of a custom hardware implementation. Signal processing is nothing but taking the estimation from measurements a set of constant parameters upon which the received signals depend.

Schmidt introduced a new technique by deriving a complete geometric solution in the absence of noise, then extending the geometric concepts to obtain a reasonable approximate solution in the presence of noise. This algorithm is called as MUSIC (Lounici *et al*., 2013; Rader, 1992). This algorithm detects frequencies in a signal by performing Eigen decomposition on the covariance matrix of a data vector Y of M samples obtained from the samples of the received signal. If a signal y(n), consists of P complex exponentials in the presence of Gaussian white noise and R_x is an autocorrelation matrix of size (M×M), if the eigenvalues are sorted in decreasing order, the eigenvectors corresponding to the P largest eigenvalues, (i.e., directions of largest variability) span the signal subspace. The remaining (M-P) eigenvectors span the orthogonal space where there is only noise.

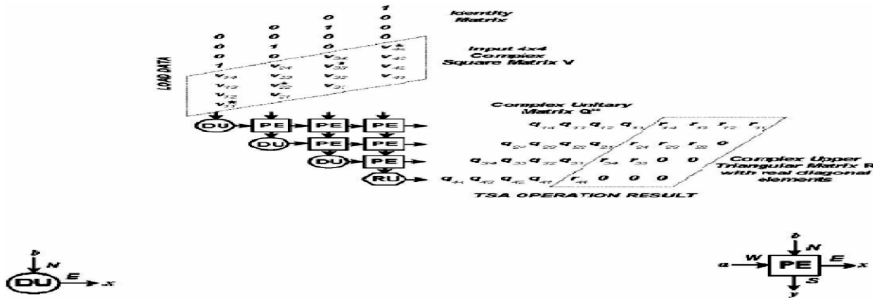

Fig. 2: Block schematic for triangular systolic archetecture (4×4) matrix

Signal model: Let a uniform linear array be composed of N sensors and let it receive M narrowband source signals $s_m(t)$ from desired users arriving at directions θ1,θ2 …. θm,as shown in Fig. 2. The array also receives I narrowband source signals $s_i(t)$ from undesired (or interference) users arriving at directions θ1,θ2 …. θm, At a particular instant of time t = 1, 2, ..., K,where K is the total number of snapshots taken. The desired users signal vector $x_i(t)$ can be defined as:

$$X_M(t) = \sum_{m-1}^{M} a(\theta m)\, S_m(t) \tag{1}$$

Where a(θm)is the N x M matrix of the desired users signal direction vectors and is given by:

$$a(\theta m) = \left[\exp\left[j(n-1)\right]\psi m\right]^T; \quad 1 \le n \le N \tag{2}$$

Where [(.)] is the transposition operator and ψm represents the electrical phase shift from element to element along the array. This can be defined by:

$$\psi m = 2\pi(d/\lambda)\sin(\theta m) \tag{3}$$

Where d is the inter-element spacing and λ is the wavelength of the received signal.The desired users signal vector $x_m(t)$ of (1) can be written as:

$$Xm(t) = Am\, S(t) \tag{4}$$

Where Am is the N x M matrix of the desired users signal direction vectors and is given by:

$$AM = [a(\theta 1), a(\theta 2) \ldots\ldots a(\theta m)] \tag{5}$$

And s(t) is the M×1 desired users source waveform vector defined as:

$$s(t) = [s_1(t)s_2(t)\ldots s_M(t)]^T \tag{6}$$

We also define the undesired (or interference) users signal vector $x_I(t)$ as:

$$X_I(t) = A_I I(t) \tag{7}$$

Where A_I is the N x I matrix of the undesired users signal direction vectors and is given by:

$$A_I = [a(\theta_1), .\ a(\theta_2)\ldots\ldots a(\theta_1)] \tag{8}$$

And I (t) is the I x 1 undesired (or interference) users source waveform vector defined as:

$$i(t) = [i_1(t)i_2(t)\ldots\ldots i_1(t)^T \tag{9}$$

The overall received signal vector x(t) is given by the superposition of the desired users signal vector $x_m(t)$, undesired (or interference) users signal vector $X_I(t)$ and an N x 1 vector n(t) which represents white sensor noise. Hence, x(t) can be written as:

$$x(t) = x_M(t) + n(t) + x_I(t) \tag{10}$$

Where, n(t) represents w hite Gaussian noise. The conventional (forward-only) estimate of the covariance matrix defined as:

$$R = E\left\{x(t)x^H(t)\right\} \tag{11}$$

Where, E{.} represents the ensemble average and $(.)^H$ is the Hermitian transposition operator. Equation 11

can be approximated by applying temporal averaging over K snapshots (or samples) taken from the signals incident on the sensor array. This averaging process leads to forming a spatial correlation (or covariance) matrix R given by Shubair *et al.* (2007), Seskar and Mandayam (1999):

$$R=\frac{1}{K}\sum_{k=1}^{k} x(k)X^{H}(k) \quad (12)$$

DOA estimation using MUSIC algorithm: A common a subspace-based DOA estimation algorithm is MUSIC (Multiple Signal Classification), (McWhirter, 1983). It starts by expressing the covariance matrix R obtained in (Godara, 1997) as:

$$R = JR \times J \quad (13)$$

Where, J is the exchange matrix with ones on its anti-diagonal and zeros elsewhere and (.)* stands for complex conjugate. The covariance matrix R in (Godara, 1997) is known to be Centro-Hermitian if and only if S is a diagonal matrix, i.e., when the signal sources are uncorrected.It can be shown (Xin *et al.*, 2011; Wang and Lamare, 2010) that the covariance matrix R obtained in (Godara, 1997) has M signal eigenvalues with corresponding eigenvectors $v_1, v_2, ..., v_m$, i.e.:

$$Vs = [v_1, v_2 .....v_m] \quad (14)$$

The remaining N-M eigenvalues of the covariance matrix R represent noise eigenvalues with corresponding eigenvectors $v_{M+1}, v_{m+2...} v_N$ i.e.,

$$v_n = [v_{M+1}, v_{m+2...}v_N] \quad (15)$$

Hence, the eigen-decomposition of the covariance matrix in (Godara, 1997) be defined in a standard way as (Seskar and Mandayam, 1996):

$$R = VIIV = V_s II_s V_s + \sigma s_2 V_n^{H} \quad (16)$$

where the subscripts s and n stand for signal-subspace, respectively. In Eq. 16, II_S is defined as:

$$II_s = diag\{\pi_1, \pi_2....M\}$$

The normalized MUSIC angular spectrum is defined as Eq. 17:

$$P(\theta) = \frac{A^{H}A^{H}}{A^{H}V_n V_N^{H}A} \quad (17)$$

By examining the denominator in (17) it is evident that peaks in the MUSIC angular spectrum occur at angles θ for which the array manifold matrix A is orthogonal to the noise subspace matrix E_n. Those angles θ define the desired directions-of-arrival of the signals impinging on the sensor array. The number of signals that can be detected is restricted by the number of elements in the sensor array. Lounici *et al.* (2013) it was verified that an N element sensor array can detect up to N-1 uncorrelated signal. This number reduces to N/2 signals if they are correlated. A comprehensive performance evaluation of the MUSIC algorithm for DOA estimation can be found in (Lounici *et al.*, 2013; Correal *et al.*, 1999).

Eigen value decomposition Based on CORDIC algorithm: Eigenvectors and eigenvalues are numbers and vectors associated to the square matrices. Both together provide the eigen-decomposition of a matrix which analyzes the structure of this matrix (Lounici *et al.*, 2013). Eigen decomposition (spectral decomposition) is a particularly simple expression for a class of matrices often used in multivariate analysis such as correlation, covariance or cross-product matrices. If:

$$E \times Z = \lambda Z \quad (18)$$

Where:

E = $[E(t) \times E^{H}(t)]$

λ = The eigenvalue corresponding to an eigenvector Z of the matrix E

Z = Eigenvector of E

The EVD has complex logic and heavy computational load for complex valued covariance matrix. MUSIC based processors cannot compute this EVD. The EVD of the covariance matrix can be solved with real number only (Wang and Lamare, 2010). The QR decomposition is nothing else than the Gram-Schmidt procedure applied to the columns of the matrix and with the result expressed in matrix form. It has faster convergence as compared to the other matrix factorization methods. And hence, to compute the EVD of the covariance matrix, QRD based on CORDIC (Coordinate Rotation Digital Computer) can successfully be implemented on FPGA device.

QRD background: QR-factorization of a matrix is a decomposition of a matrix X into a product Y of an orthogonal matrix QR and an upper triangular matrix Y. This is used to solve linear least suare problems. The

Gram-Schmidt process, givens rotations and Householder transformations are various methods for the decomposition. In this study, we have considered Givens rotations for computing the QR decomposition. Givens each rotation introduces zeros in the subdiagonal of the Matrices, forming the upper triangular matrix X and, the orthogonal matrix X is formed by concatenation of the givens rotation. This is not actually performed by building a whole matrix and doing a matrix multiplication. A Givens rotation procedure is used instead which does the equivalent of the sparse Givens matrix multiplication, without the extra work of handling the sparse elements. The Givens rotation procedure is useful in situations where only a relatively few off diagonal elements need to be zeroed and is more easily parallelized than Householder transformations. X is a square matrix, where its column vectors represent the (N-O) signal eigenvectors and O noise Eigen vectors. Assume that, we have a matrix Q(M×M). Decomposition of matrix Q can be written as Q = XU.

Where, U is an upper triangular matrix, X is an orthogonal matrix (which will zero in the lower triangular entry of the matrix) The rotation angle is suitably chosen as E = XY:

$$\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}\begin{bmatrix} Q_{11} & Q_{12} & \cdots & Q_{1m} \\ Q_{21} & Q_{22} & \cdots & Q_{2m} \end{bmatrix} = \begin{bmatrix} U_{11} & U_{12} & \ldots & U_{1m} \\ 0 & U_{22} & \ldots & U_{2m} \end{bmatrix} \quad (19)$$

Where, $\theta = \tan^{-1}(b/a)$ since Coordinate Rotation by Digital Computer (CORDIC) is capable of vector rotation by angle, operators can be efficiently time-shared to perform the QR decomposition while consuming minimal resources.

CORDIC Algorithm: CORDIC is a trigonometric algorithm (Andraka, 1998) which is also known as digit-by-digit method and Volder's algorithm, stands for Coordinate Rotation Digital Computer. It is a simple and efficient algorithm to calculate hyperbolic and trigonometric functions, used where no hardware multipliers are available. It minimizes number of gates required to implement the functions in FPGA. The CORDIC algorithm provides an iterative method of performing vector rotations by arbitrary angles using only shifts and adds. It generally produces one additional bit of accuracy for every iteration.CORDIC is efficient when fixed point implementations of signal processing algorithms on hardware are considered. CORDIC is a good choice for hardware solutions such as FPGA in which cost (gate count) minimization is more important than throughput maximization. In QR decomposition; the CORDIC algorithm is operated in two modes, rotation mode and vectoring mode. The vector rotations are computed in the rotation mode while the rotation angles are computed in the vectoring mode. The CORDIC equations are given below

$$\begin{aligned} X_{i+1} &= X_i - Y_i \times d_i * 2^{-i} \\ Y_{i+1} &= Y_i - X_i \times d_i * 2^{-i} \\ Z_{i+1} &= Z_i - d_i \times \tan^{-1}(2^{-i}) \end{aligned} \quad (20)$$

In rotation mode; $d_i = -1$ for $Z_i<0$, $d_i = +1$ for $Z_i = 0$. Where, $d_i = \pm 1$ (direction of rotation) and z is the angle accumulator. After a finite number of iterations, the CORDIC algorithm in rotation mode from above equations provides the following results; Where; A_n is a computational gain. In vectoring mode:

$$\begin{aligned} X_n &= A_n\left(X_o\cos(Z_o) - Y_o\sin(Z_o)\right) \\ Y_n &= A_n\left(Y_o\cos(Z_o) - X_o\sin(Z_o)\right) \\ Z_n &= 0 \\ A_n &= \prod_{i=0}^{n}\sqrt{1+2^{-2i}} \end{aligned} \quad (21)$$

$$d_i = +1 \text{ for } Y_i < 0,\ d_i = -1 \text{ for } Y_i \geq 0 \quad (22)$$

After a finite number of iterations, the CORDIC algorithm in vectoring mode from above equations provides the following results. The scaled magnitude of the original vector (x component of the result) and a rotation angle:

$$\begin{aligned} X_n &= A_n\sqrt{X_0^2 + Y_0^2};\ Y_n = 0 \\ Z_n &= Z_0 + \tan^{-1}\left(\frac{Y_0}{X_0}\right) \\ A_n &= \prod_{i=0}^{n}\sqrt{1+12^{-2i}} \end{aligned} \quad (23)$$

After the final step, scaling operation must be performed elsewhere in the system. From Eqs the scaling factor is obtained by:

$$K_n = \frac{1}{A_n} = \prod_{i=0}^{n}\frac{1}{\sqrt{1+2^{-2i}}} \quad (24)$$

Hardware design of EVD processor: Eigen decomposition (Lounici *et al*., 2013) or spectral decomposition is the factorization of a matrix into a canonical form whereby

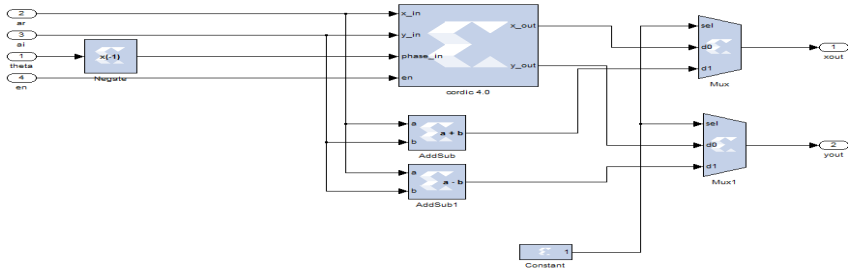

Fig. 3: TSA with XSG blocks

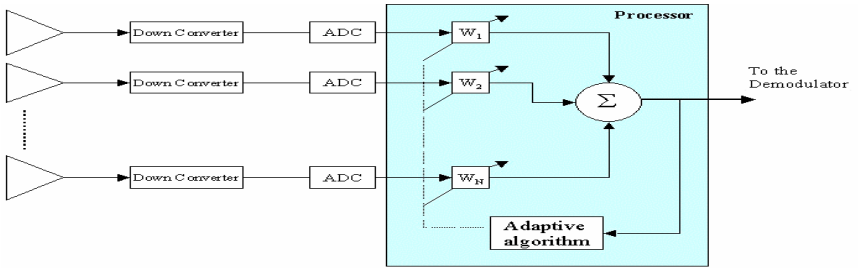

Fig. 4: Processing element in rotation mode

the matrix is represented in terms of its eigen values and eigenvectors. The Triangular Systolic Array (TSA) is used for Eigenvector decomposition. Here, the rows of the complex covariance matrix M(4×4) are fed to the Triangular Systolic Array (TSA) architecture from the top followed by the identity matrix. The 16-bit fixed point Xilinx System Generator (XSG) is used to realize this decomposition.Three main processing cells with different functionality used are: Delay Units (DU), Processing Elements (PE) and one Rotational Unit (RU). The input matrix V is loaded in the temporally (skewed) triangular shape and followed by the identity matrix to produce unitary matrix QR at the output. The diagonal elements of matrix are denoted with flag (*) which represents the control signal propagated together with the data element of PE and RU cells. The DU reads the elements from the North input N and delays them (by a period equal to PE operation time) and then passes to the right output 0. PE is the heart of the TSA architecture, it is a signal processing cell based on the CORDIC processor. The same is represented in fig.2. Each PE may operate in two different modes: Vectoring and Rotation. The PE mode is controlled by flag *. If the data sample carries flag * and enters PE from the west port, then PE operates in Vectoring mode. In all cases the Rotation mode is active.

Vectoring mode: The PE is said to be operating in vectoring mode if the data value carries flag (*) and enters PE from the west port W and flows through to the output port E. These CORDIC modules of PE are established to calculate the angle θ which is stored into the internal register. It delivers the amplitude of complex elements in input matrix as $(\sqrt{|a|^2+|b|^2})$ at the output port O and a zero is obtained at the south output S,when the flagged value enters from port W and value from port N, respectively.

Rotation mode: (CORDIC) modules of PE are configured to perform vector rotation in polar coordinates. If the data value doesn't carry flag (*) and enters PE from the west port W and flows through to the output port O then PE operates in Rotation mode. In this mode the elements a from left and b taken from north port, flows through to the output ports i.e., right and bottom port respectively. And in this mode, the angle accumulator is initiated to the desired rotation angle and hence after finite iterations angle obtained is zero as in the equation above.

EVD processor using XSG: Figure 3 shows the Proposed design Using XSG blocks given by Matlab-Simulink library.The first block is the preprocessing unit where the covariance matrix is loaded in a triangular shape, followed by the identity matrix and delayed matrix element for period equal to '▲' the second block is the TSA describedabove.

Figure 4 describes the TSA units for 4×4 matrix we have four Processing elements (Pe) and four delay units. Pe architecture which may operate in two different modes:

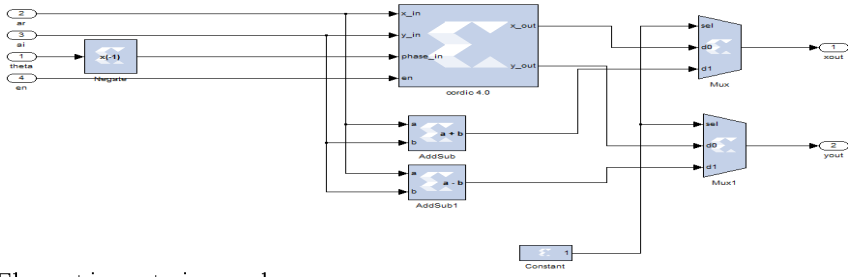

Fig. 5: Processing Element in vectoring mode

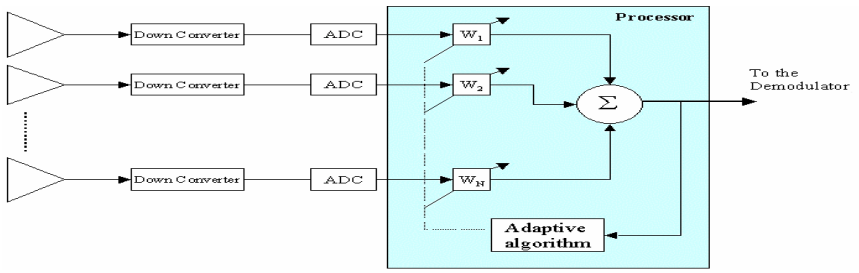

Fig. 6: Block diagram of an adaptive beamformer

Vectoring and Rotation mode. Figure 4 shows the Rotation mode and vectoring mode is described in Fig. 5 where both of modes are based on CORDIC block given by XSG library.

First we simulate the model fig for 4x4 real covariance matrixes comparing with MATLAB based model, successful results are obtained for both of Eigen values and vectors using 16 bit fixed point XSG based model. After simulating the model for 4×4 matrix, we redesign the system for different cases of possible covariance matrix size for unitary MUSIC with different array element, the resource estimator block is used to estimate the logic device utilization values. The results are obtained for 16-bit fixed point operations resumed on Tabulation.

Adaptive beamforming: Figure 6 shows the smart antenna system of a cellular mobile system. The current amplitudes of a uniform linear antenna array are so adjusted by a set of complex weights such that maximum radiated power is produced in the directions of desired. This is done by using an adaptive beamforming algorithm. This algorithm generates deep nulls in the directions of undesired signals representing co-channel interference from mobile users in adjacent cells.

An adaptive beam former provides a means for separating the desired signal from interfering signals. It automatically optimizes the array pattern by adjusting the elemental control weights until a prescribed objective function is satisfied. An algorithm designed for that purpose specifies the means by which the optimization is achieved. It is the technique in which an array of antennas is exploited to achieve maximum reception in desired direction and jamming signals from other directions are rejected. It consists of multiple antennas arranged in an array. The main function adjusts the amplitude of the received signals from these arrays and processes these signals to enhance the signal of interest. The signal from each element is multiplied by a variable complex weight and the weighted signals are then summed to form the array output. The array output due to the desired signal is:

$$y(i) = w(i)^{H} \times X(i) \tag{25}$$

Where, w is the complex weights vector and X is the received signal vector given in (36).H is Hermitian (complex conjugate) transpose.w is a complex vectors. The amplitudes and phases of the complex weights? are added together to form the beam in desired direction. Typically, these weights are computed in order to optimize the performance in terms of a certain criterion. In order to compute the Optimum weights the array response vector or steering vector from the sampled data of the array output has to be known. The array output y (t), a reference signal d(t) and previous weights are considered while computation of weights to get the desired signal. The approximation of the reference signal with the desired signal is done using a training sequence or a spreading code which is known at the receiver.

RLS Algorithm for beamforming: From the above signal model information, adaptive algorithms are required to estimate Weight vector w (i) from X (i) while assuming

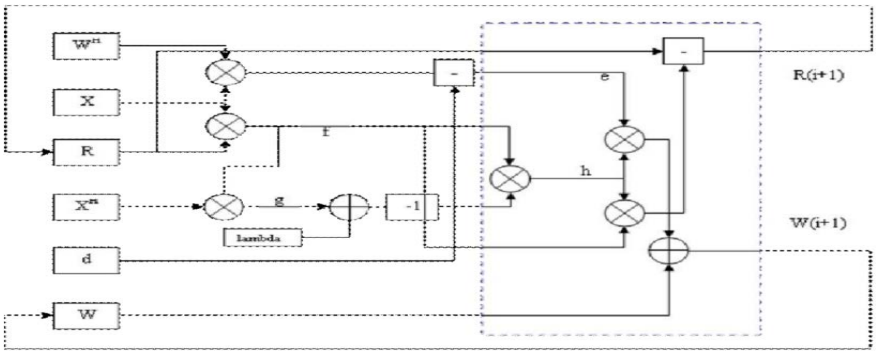

Fig. 7: SIMULINK architecture for the RLS algorithm

d(i) as the reference signal. Here RLS algorithm is used for the adaptive beamforming. The benefit of the RLS algorithm is that there is no need to invert matrices as the inverse correlation matrix is computed directly. It requires reference signal and correlation matrix information. The RLS exhibits another benefit, i.e., extremely fast convergence. However, this benefit comes at the cost of high computational complexity. Let, N be length of input sequence:

$$R(i) = \delta X1 \tag{26}$$

Where; W is a small positive constant. Be the identity matrix of size N×N. Then, for I = 1 to N

$$\left.\begin{aligned} f(i) &= (R(i) \times X(i)) \\ f(i) &= (R(i) \times X(i)) \end{aligned}\right\} \tag{27}$$

$$\left.\begin{aligned} g(i) &= (X(i)^H \times R \times X(i)) \\ &= (X(i)^H \times f(i)) \end{aligned}\right\} \tag{28}$$

$$\left.\begin{aligned} h(i) &= \frac{(R(i) \times X(i))}{\lambda + (X(i)^H \times R(i) \times X(i))} \\ &= \frac{f(i)}{\lambda + (X(i)^H \times f(i))} \\ &= \frac{f(i)}{\lambda + g(i)} \end{aligned}\right\} \tag{29}$$

Where; λis known as forgetting factor or exponential weight factor which is in range ($0 \le \delta \le 1$).

$$R(i+1) = R(i) - \left(\frac{(R(i) \times X(i))}{\lambda + (X(i)^H \times R(i) \times X(i))} \times R \times X(i) \right) \tag{30}$$

But, substituting equations (27) and (29) in equation (30) above we get,

$$R(i+1) = R(i) - (h(i) \times f(i)) \tag{31}$$

And, from Eq. 25, researchers can update (minimize) the error between estimated signal and original signal:

$$e(i) = d(i) - (w(i)^H \times X(i)) \tag{32}$$

Substituting Eq. 25 in Eq. 32 we get:

$$e(i) = d(i) - y(i) \tag{33}$$

Thus, the desired weight vectors of the smart antenna adaptive beamforming system are estimated by:

$$w(i+1) = w(i) + (e(i) \times h(i)) \tag{34}$$

Therefore, the desired output of the smart antenna system is obtained by the sum of weight vector corresponding to the input signal vector. It is represented in Fig. 7 as:

$$y_d(i) = \sum_{i=1}^{N} (w(i)^H \times X(i)) \tag{35}$$

RESULTS AND DISCUSSION

Experimental Setup and Results for DOA Estimation Using MUSIC Algorithm: In order to study the performance of the Uniform Linear Array (ULA) Antenna system with respect to the system parameters like number of elements (N), Inter-element spacing (d) and Number of samples (K); graphs are plotted by taking related power along y-axis and Estimated Direction of arrival by MUSIC algorithm along x-axis and the effect of a particular parameter can be determined by keeping the two of three system parameters (N, K, d) as constants and plotting the

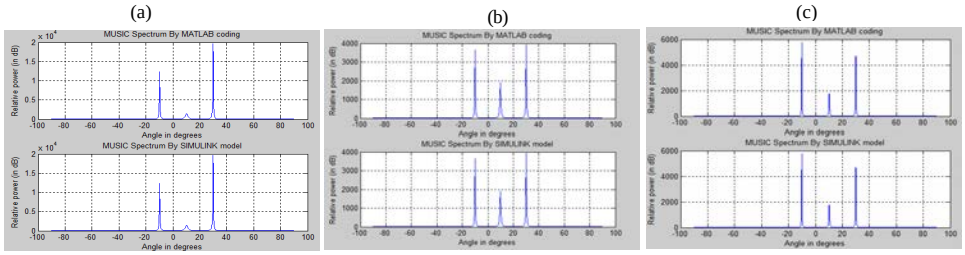

Fig. 8: MUSIC angular spectrum for different values of number of elements: a) N = 5; b) N = 6; c) N = 7

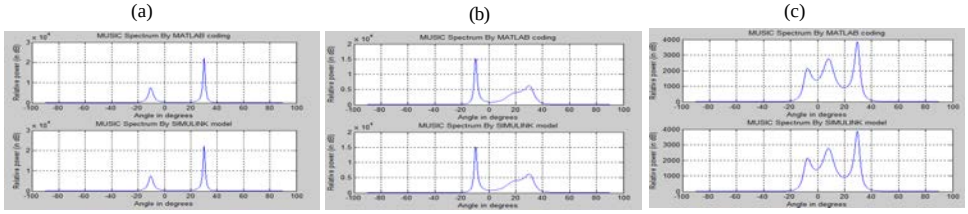

Fig. 9: MUSIC angular spectrum for different values of number of elements: a) N = 5; b) N = 6; c) N = 7

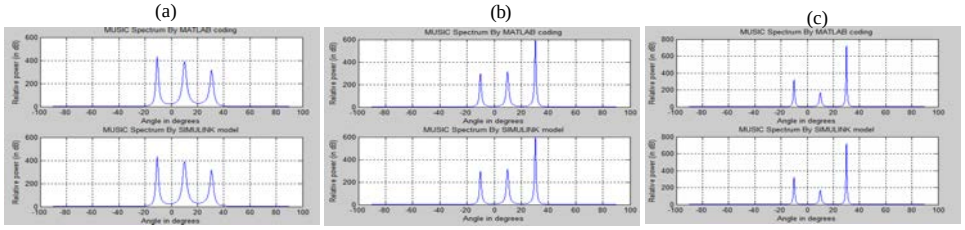

Fig. 10: MUSIC angular spectrum for different values of Number of Elements: a) N = 5; b) N = 6; c) N = 7

graph for the third parameter value with respect to the constants. There are three cases for our experiment and is discussed as follows.

Case (1); music angular spectrum for different values of no of elements: In this case, performance of the ULA antenna system can be determined from the MUSIC angular spectrum. The variation in related power with respect to Direction of Arrival for the distinct number of elements of ULA antenna system at constant number of samples and inter-element spacing is plotted to analyze the system performance shown in Fig. 8.

For K = 100 and d = 0.5, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 9.

For K = 100 and d = 0.2, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 10.

For K = 50 and d = 0.5, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 11.

For K = 10 and d = 0.5, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 12.

For K = 10 and d = 0.2, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 13.

From the results of case (1), it is observed that the proposed MUSIC DOA algorithm performs best when the number of sensors or elements of uniform linear array (N) is having higher values. For example, if N = 7 for K = 100 and d = 0.5 the MUSIC spectrum will produce best DOA plot than the N< = 6.

Case (2): music angular spectrum for different values of interelement spacing: The impact of the inter element spacing between the antenna elements over the performance of the system can be studied from the MUSIC angular spectrum obtained by plotting the graph below,

For K = 10 and N = 5, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at (Fig. 15).

For K = 10 and N = 6, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at Shown in Fig. 15.

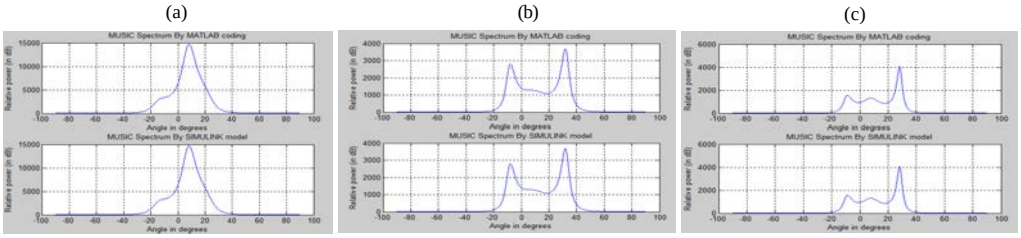

Fig. 11: MUSIC angular spectrum for different values of number of elements: a) N = 5; b) N = 6; c) N = 7

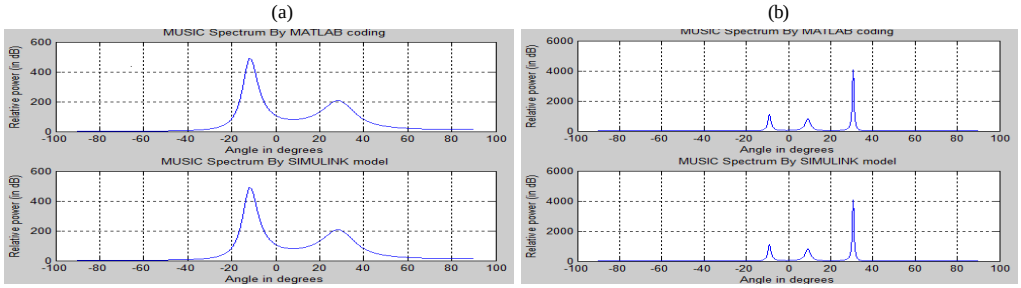

Fig. 12: MUSIC angular spectrum for different values of number of elements: a) N = 5; b) N = 6; c) N = 7

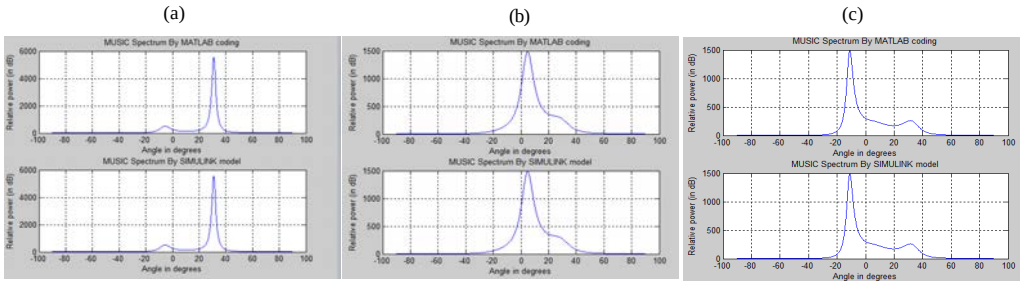

Fig. 13: MUSIC angular spectrum for different values of number of elements: a) N = 5; b) N = 6; c) N = 7

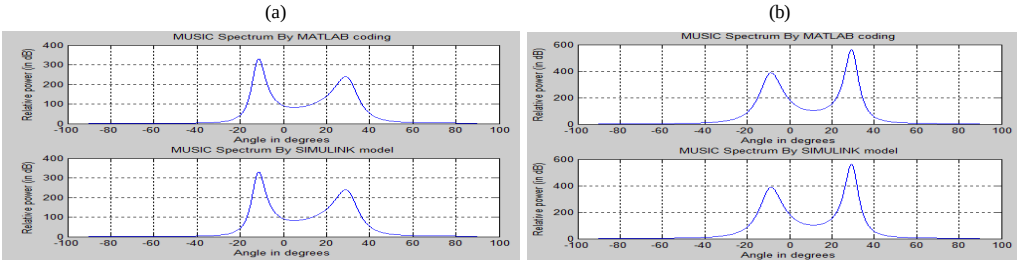

Fig. 14: MUSIC angular spectrum for different values of Inter element spacing: a) d = 0.5; b) d = 0.2

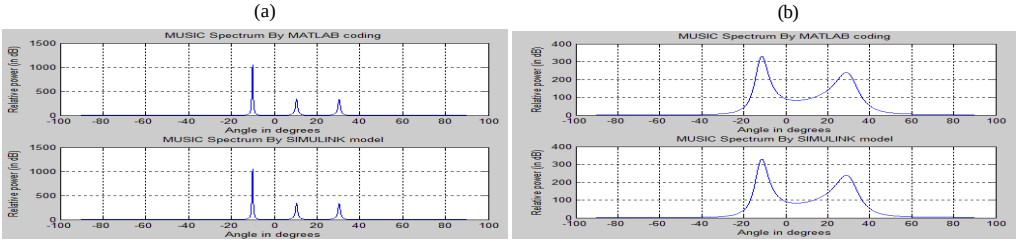

Fig. 15: MUSIC angular spectrum for different values of Inter element spacing: a) d = 0.5; b) d = 0.2

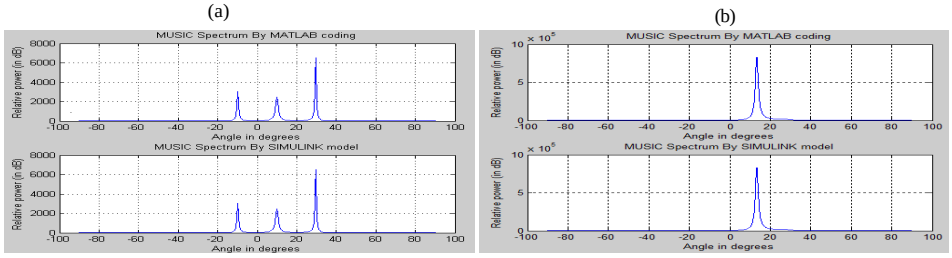

Fig. 16: MUSIC angular spectrum for different values of Inter element spacing: a) d = 0.5; b) d = 0.2

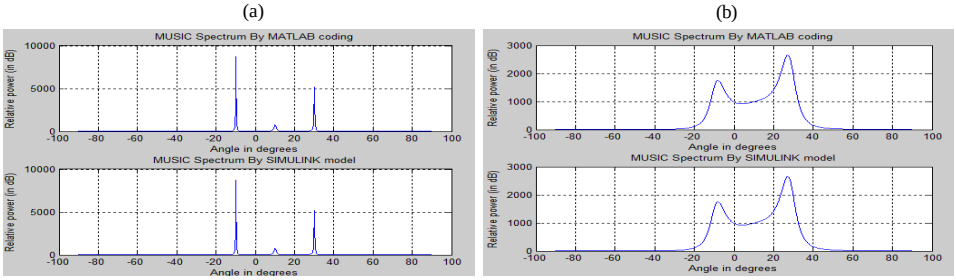

Fig. 17: MUSIC angular spectrum for different values of Inter element spacing: a) d = 0.5; b) d = 0.2

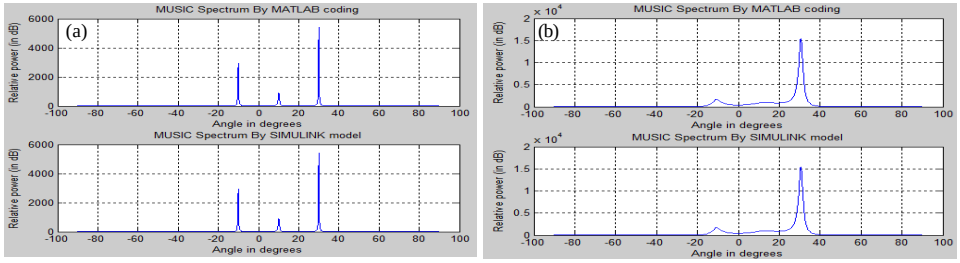

Fig. 18: MUSIC angular spectrum for different values of inter element spacing: a) d = 0.5; b) d = 0.2

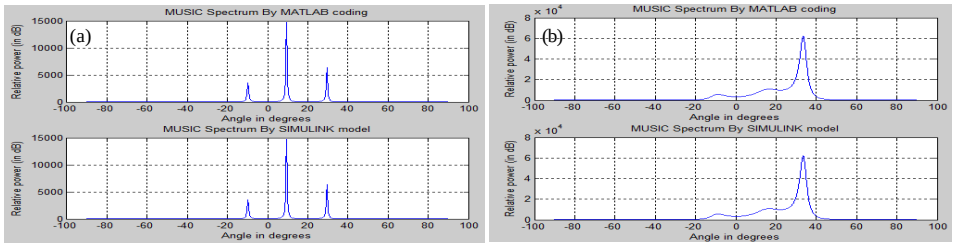

Fig. 19: MUSIC angular spectrum for different values of inter element spacing: a) d = 0.5; b) d = 0.2

For K = 10 and N = 7, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 16.

For K = 50 and N = 5, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 17.

For K = 50 and N = 6, we plot MUSICspectrum plot for the proposed Simulinmodel and the Matlab model at shown in Fig. 18.

For K = 50 and N = 7, we plot MUSICspectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 19.

For K = 100 and N = 5, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 20.

For K = 100 and N = 6, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at d = 0.5 shown in Fig. 21.

For K = 100 and N = 7, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at d = 0.2 shown in Fig.22.

From the results of case (2), it is observed that the proposed MUSIC DOA algorithm performs best when the

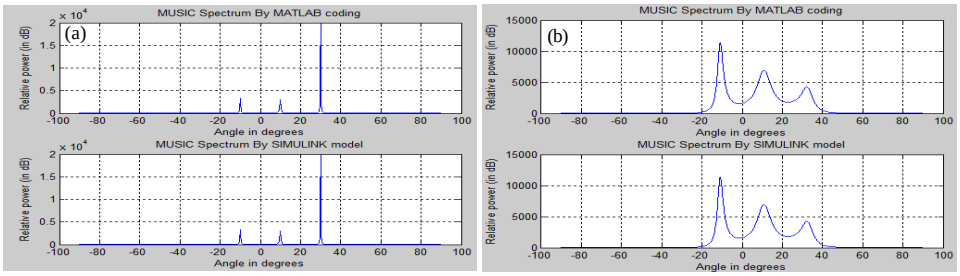

Fig. 20: MUSIC angular spectrum for different values of inter element spacing: a) d = 0.5; b) d = 0.2

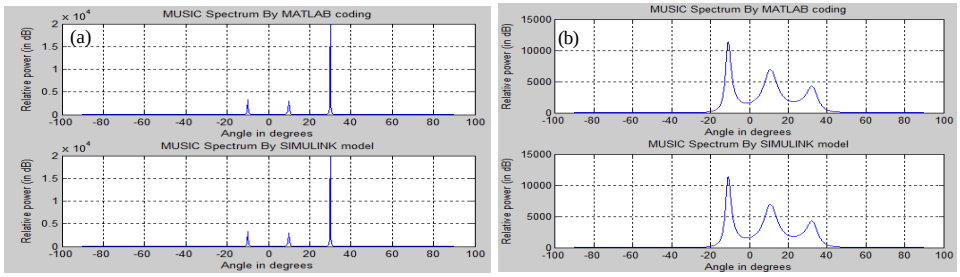

Fig. 21: MUSIC angular spectrum for different values of inter element spacing: a) d = 0.5; b) d = 0.2

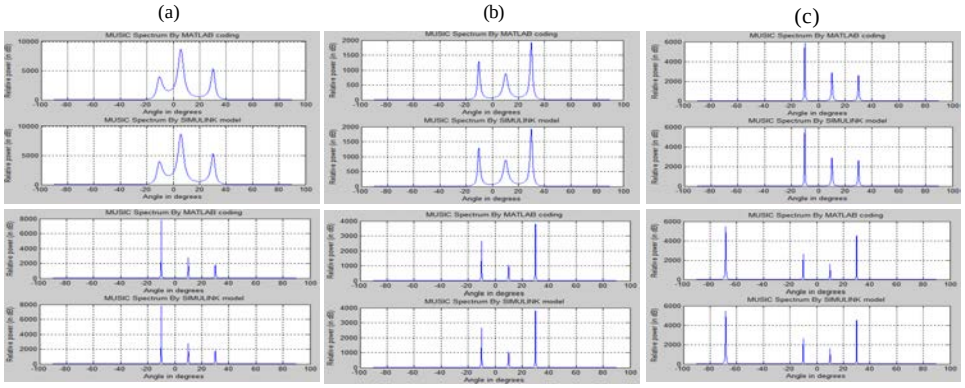

Fig. 22: MUSIC angular spectrum for different values of inter element spacing: a) d = 0.2; b) d = 0.3; c) d = 0.4; d) d = 0.5; e) d = 0.6; f) d = 0.7

inter-element spacing (d) is having values in the range d = 0.3-0.6. For example, if K = 100 and N = 7 the MUSIC spectrum will produce best DOA, otherwise if d = 0.7 and above the spectrum will generate malicious results. Also if d = 0.2 the spectrum will be generated for K>100 this is discussed in case (3) below.

Case (3); music angular spectrum for different values of number of samples: In this case the relation between the number of samples and the performance of the system can be determined by obtaining a graph with the basic axis as mentioned earlier and plotting it by following the below steps to determine the MUSIC angular spectrum.

For d = 0.5 and N = 5, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 23.

For d = 0.5 and N = 6, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 24

For d = 0.5 and N = 7, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 25.

For d = 0.2 and N = 5, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig.26.

For d = 0.2 and N = 6, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 27.

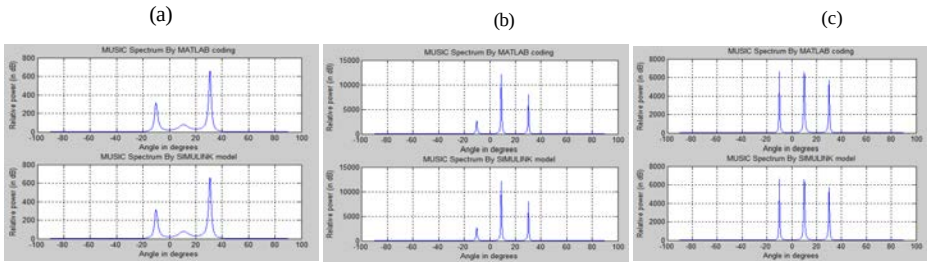

Fig. 23: MUSIC angular spectrum for different values of number of samples: a) k = 10; b) = 50; c) k = 100

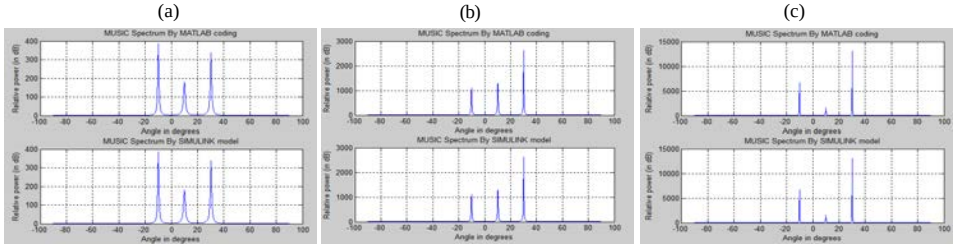

Fig. 24: MUSIC angular spectrum for different values of number of samples: a) k = 10; b) k = 50; c) k = 100

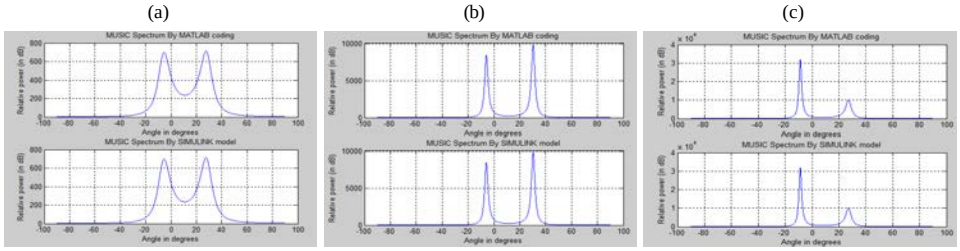

Fig. 25: MUSIC angular spectrum for different values of number of samples: a) k = 10; b) k = 50; c) k = 100

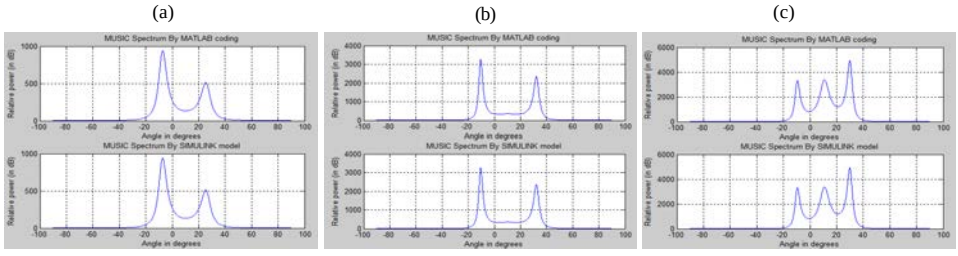

Fig. 26: MUSIC angular spectrum for different values of number of samples: a) k = 10; b) k = 50; c) k = 100

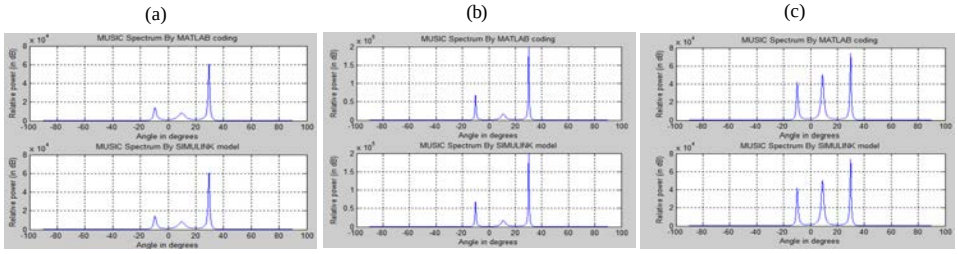

Fig. 27: MUSIC angular spectrum for different values of number of samples: a) k = 10 (b) k = 50; c) k = 100

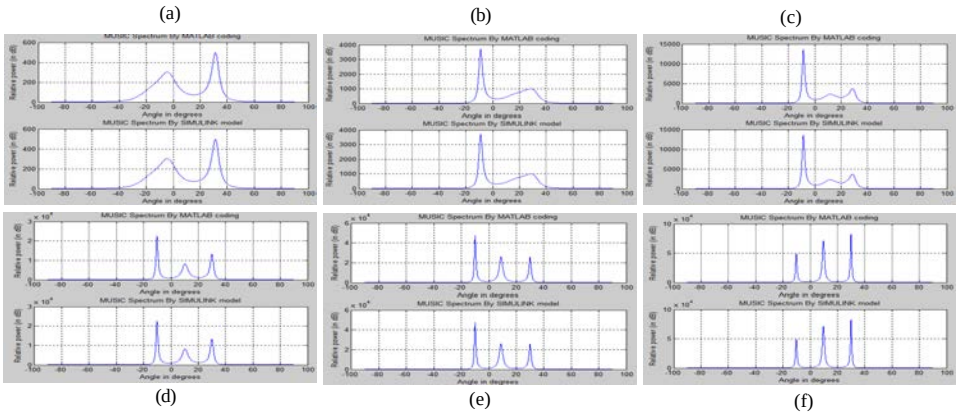

Fig. 28: MUSIC angular spectrum for different values of number of samples: a) k = 10; b) k = 50; c) k = 100; d) k = 500; e) k = 1000; f) k = 2000

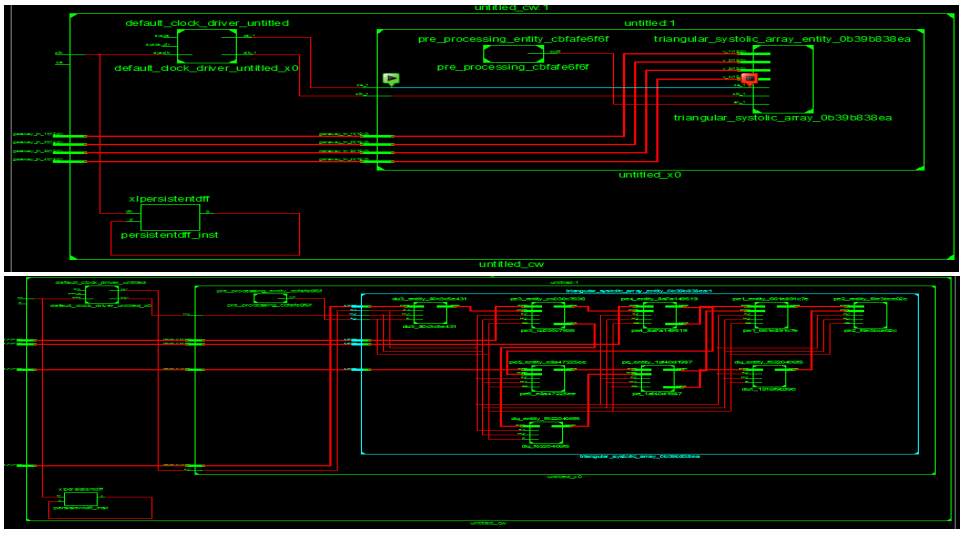

Fig. 29: Internal structure of MUSIC in RTL view represents the XSG based FPGA model

For d = 0.2 and N = 7, we plot MUSIC spectrum plot for the proposed Simulink model and the Matlab model at shown in Fig. 28.

From the results of case (3), it is observed that the proposed MUSIC DOA algorithm produces best results when the total number of snapshots taken (K) is having values greater than 100. Also, in the above section (case (2)) we found that if d = 0.2 the DOA produced by the spectrum are malicious. This problem is solved by operating the proposed system for higher snapshots or samples. The results for K = 100, 500, 1000, 2000. And it is observed that the higher the value of K, better is the DOA spectrum for all range of the inter-element spacing (d = 0.2 to 0.6) shown in Fig. 29.

RTL view

Device Utilization Table (DUT) For Doa model: The Device or logic utilization by the proposed DOA estimation system for different N values on Virtex-4 FPGA device (xc4vfx12-12sf363) family is compared in the Table 1;

Table 1: Device utilization table comparison

Comparison of logic device utilization

Existing evd using xilinx system generator (Lounici *et al.*, 2013)				
Size of matrix	No. of slices	No. of LUTs	No. of FF's	No. of Bonded IOBs
(3×3)	2424	7986	1185	528
Proposed evd using xilinx system generator				
(4×4)	76	97	33	134
(5×5)	118	160	49	161
(6×6)	167	240	65	193
(7×7)	224	336	81	225

Experimental setup and simulation results for RLS algorithm: For experimenting Beamforming RLS algorithm, Polar spectrums are plotted using the estimated weight vectors. We observe the beam forming

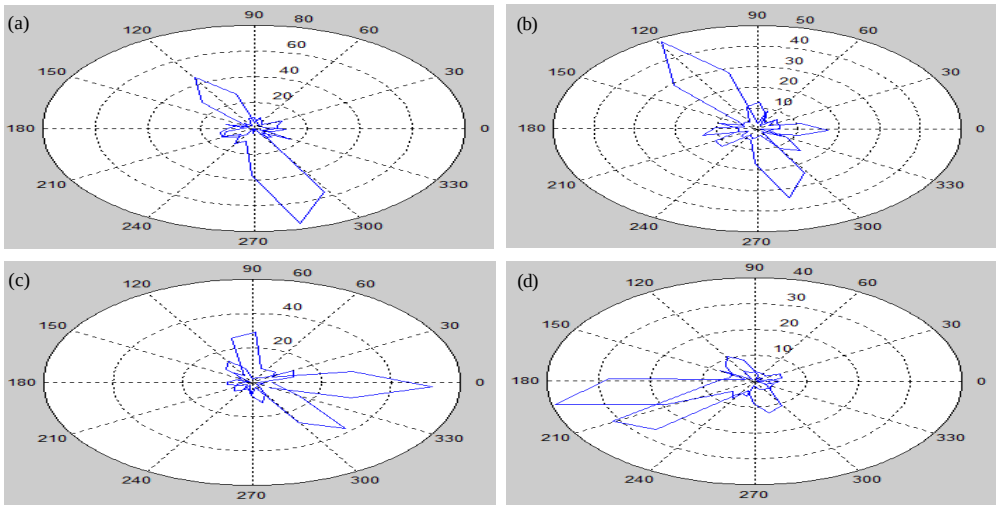

Fig. 30: Polar plat for Beamforming Using MATLAB/SIMULINK model Using Different Lambda values: a) 3000000; b) 4000000; c) 5000000; d) 6000000

for different lambda values. Lambda = c/fc, we change fc values. The plots for beam forming are as follows in Fig. 30.

CONCLUSION

In this study, the proposed FPGA design of an EVD system for DOA estimation with MUSIC Algorithm is described. The EVD architecture is built using the QR decomposition method which utilizes CORDIC algorithm. The proposed FPGA based design is realized using Xilinx system generator tool in Matlab/Simulink library is presented in detail and the approached logic device values are produced, this make the selection of the FPGA device more comfortable compared to the existing MUSIC implementation in (Lounici *et al*., 2013) and also proposed RLS Beamforming algorithm using MUSIC algorithm based DOA estimation is designed and implemented on Matlab/simulink tool. The proposed design was synthesized on Virtex-4 FPGA family device. The device utilization results for the N element array is compared which concludes that the proposed system area will increase with increase in N elements, but the system estimates better DOA's. Also the proposed system will produce best DOA results for greater sample values K and for the inter element spacing (d) in range between 0.2-0.6.

REFERENCES

Andraka, R., 1998. A survey of CORDIC algorithms for FPGA based computers. Proceedings of the ACM/ SIGDA 6th International Symposium on Field Programmable Gate Arrays, February 22-25, 1998, Monterey, USA., pp: 191-200.

Ardi, E.M.A., R.M. Shubai and M.E.A. Mualla, 2003a. Investigation of high-resolution DOA estimation algorithms for optimal performance of smart antenna systems. Proceedings of the 4th International Conference on 3G Mobile Communication Technologies, June 25-27, 2003, IEEE, New York, USA., ISBN:0-85296-756-X, pp: 460-464.

Ardi, E.M.A., R.M. Shubair and M.E.A. Mualla, 2003b. Performance evaluation of direction finding algorithms for adapative antenna arrays. Proceedings of the 2003 10th IEEE International Conference on Electronics Circuits and Systems (ICECS03.), December 14-17, 2003, IEEE, New York, USA., ISBN: 0-7803-8163-7, pp: 735-738.

Correal, N.S., R.M. Buehrer and B.D. Woerner, 1999. A DSP-based DS-CDMA multiuser receiver employing partial parallel interference cancellation. IEEE. J. Sel. Areas Commun., 17: 613-630.

Dikmese, S., A. Kavak, K. Kucuk, S. Sahin and A. Tangel, 2011. FPGA based implementation and comparison of beamformers for CDMA2000. Wirel. Personal Commun., 57: 233-253.

Godara, L.C., 1997. Application of antenna arrays to mobile communications II. Beam-forming and direction-of-arrival considerations. Proc. IEEE., 85: 1195-1245.

Gu, J.F., S.C. Chan, W.P. Zhu and M.N.S. Swamy, 2013. Joint DOA estimation and source signal tracking with Kalman filtering and regularized QRD RLS algorithm. IEEE. Trans. Circuits Syst. Express Briefs, 60: 46-50.

Lounici, M., X. Luan and S.A.A.D. Wahab, 2013. Implementation of QR-decomposition based on CORDIC for unitary MUSIC algorithm. Proceedings of the Fifth International Conference on Digital Image Processing (ICDIP 2013), July 19, 2013, IPCSIT, Beijing, China, pp: 88784-88784.

McWhirter, J.G., 1983. Recursive least-squares minimization using a systolic array. Proceedings of the 27th Annual Techincal Symposium on International Society for Optics and Photonics, November 28, 1983, San Diego, California, pp: 105-113.

Rader, C.M., 1992. MUSE-A systolic array for adaptive nulling with 64 degrees of freedom, using Givens transformations and wafer scale integration. Proceedings of the International Conference on Application Specific Array Processors, August 4-7, 1992, IEEE, New York, USA., ISBN: 0-8186-2967-3, pp: 277-291.

Seskar, I.P. and N.B. Mandayam, 1999. A software radio architecture for linear multiuser detection. IEEE. J. Sel. Areas Commun., 17: 814-823.

Shaukat, S.F., M. Hassan, R. Farooq, H.U. Saeed and Z. Saleem, 2009. Sequential studies of beamforming algorithms for smart antenna systems. World Appl. Sci. J., 6: 754-758.

Shubair, R.M. and W. Jessmi, 2005. Performance analysis of SMI adaptive beamforming arrays for smart antenna systems. Proceedings of the 2005 IEEE International Symposium on Antennas and Propagation Society, July 3-8, 2005, IEEE, New York, USA., ISBN:0-7803-8883-6, pp: 311-314.

Shubair, R.M., M.A.A. Qutayri and J.M. Samhan, 2007. A setup for the evaluation of MUSIC and LMS algorithms for a smart antenna system. J. Commun., 2: 71-77.

Wang, L. and R.C.D. Lamare, 2010. Constrained adaptive filtering algorithms based on conjugate gradient techniques for beamforming. IET. Signal Process., 4: 686-697.

Xin, J., N. Zheng and A. Sano, 2011. Subspace-based adaptive method for estimating direction-of-arrival with Luenberger observer. IEEE. Trans. Signal Process., 59: 145-159.

Yang, B. and J.F. Bohme, 1992. Rotation-based RLS algorithms: Unified derivations, numerical properties and parallel implementations. IEEE. Trans. Signal Process., 40: 1151-1167.