AREA OPTIMIZED FPGA IMPLEMENTATION OF ADAPTIVE BEAMFORMER

Harpreet Kaur¹, Parminder Singh Jassal²

¹ Department of Electronics and Communication Engineering
² M.Tech Student, Yadavindra College of Engineering, Punjabi University Guru kashi Campus, Talwandi Sabo – 151 302, Punjab (India)
² Assistant Prof, Yadavindra College of Engineering, Punjabi University Guru kashi Campus, Talwandi Sabo – 151 302, Punjab (India)

¹ harpreet.virk363@gmail.com
² pammi_jassal@yahoo.co.in

Abstract— Quadratic Rotation decomposition (QRD) based recursive least squares (RLS) algorithm can be used in variety of communication applications and its low complexity implementation can be of interest. In this paper we have presented an application of QRD based RLS algorithm using Coordinate Rotation by Digital Computer (CORDIC) operator for implementing an adaptive beamformer. FPGA resource estimates along with actual implementation results have been presented and are being compared with its existing implementation.

Key words— Adaptive beamformer, Coordinate Rotation by Digital Computer, Field Programmable Gate Array, Quadratic Rotation decomposition, Recursive Least Squares.

I. INTRODUCTION

Spatially propagating signals encounter the presence of interfering signals and noise signals. If the desired signal and the interferers occupy the same temporal frequency band, then temporal filtering cannot be used to separate the signal from the interferers. However the desired and the interfering signals generally originate from different spatial locations. This spatial separation can be exploited to separate the signals from the interference using a beamformer. Some of the invaluable references that thoroughly outlined all the methods and algorithms include [1-4]. J. Litva and K. Y. Lo [1] explained in detail the basic concept of adaptive beamforming starting from the use of two elements array to suppress interference. The fundamental method in adaptive beamforming is to choose the weights of array elements in order to optimize the beamformer response to fulfill certain criterion. The criterion includes Minimum Mean-Square Error, Maximum Signal-to-Interference Ratio and Minimum Variance was discussed [1]. The choice of criteria is not critically important since they are closely related to each other. The more important part is the adaptive algorithms, which will determine the speed of convergence and hardware complexity required. The algorithms include Least Mean Squares algorithm (LMS), Direct Sample Covariance Matrix Inversion (SMI), Recursive Least Squares Algorithms (RLS) and Neural Networks. The notion of partially adaptivity then explained which the alternative technique is when the number of array elements becomes very large until it is difficult to implement full adaptivity. Two known types are temporal reference and spatial reference. B. D. V. Veen and K. M. Buckley [2] introduced beamforming as a versatile form of spatial filtering. Started with the basic concept, associated the explanation with FIR filtering. Beamformer was classified into data independent and statistically optimum beamformer. Independent of the received data, the first class of beamformer chose a fixed antenna arrays weights. The later class use statistical information of received data to select the weights. Adaptive beamforming comes into picture for the fact that the data statistics are often unknown and varying over time. Two basic adaptive approaches, block adaptation and continuous adaptation were discussed. In block adaptation, the statistics are estimated from temporal block of array data while continuous adaptation the weights are adjusted as the data is sampled. Two basics adaptive algorithms, LMS and RLS also introduced. Partial adaptivity was highlighted. The important comparison of different approach in beamforming may be compared based on type of adaptive algorithm it used. Many researches were done on each algorithm and some comparisons were highlighted in [1] [4]. Simplicity of Least Mean Square (LMS) algorithm makes it widely been used for tap coefficient adaptations of an adaptive processor in antenna array. However, this continuous adaptation approach algorithm causes signal acquisition and tracking problems due to its slow convergence in multipath fading channel. This is not suitable for mobile communication and some other measures need to be taken if this algorithm is to be used such as power control or normalized LMS algorithm. RLS can be seen as the solution for the slow convergence of LMS.
Simulation results show that RLS outperform LMS in flat fading channels [1]. But implementation of RLS algorithm requires large number of FPGA resources. This paper presents the FPGA implementation of adaptive beamformer using quadratic rotation decomposition based recursive least squares algorithm (QRD-RLS). The paper is organized as follows: Section I is introduction, section II presents preliminaries of adaptive beamforming, section III is QRD-RLS algorithm, systolic array based QR decomposition explained in section IV, FPGA implementation will be discussed in sections V, Concluding remarks will be in section VI.

II. PRELIMINARIES OF ADAPTIVE BEAMFORMING

The beamformer electronically steer a phased array by weighting the amplitude and phase of signal at each array element in response to changes in the propagation environment. Capacity improvement is achieved by effective co-channel interference cancellation and multipath fading mitigation. Figure 1 shows the concept of adaptive beamforming.

![Figure 1: Concept of Beamforming](image)

The beam pattern produced by a phased array antenna can be steered electronically to place the region of greatest sensitivity towards a source of interest, and nulls in the directions of interferers.

![Figure 2: Use of Adaptation Algorithm for Cancellation of Side Lobe](image)

The adaptive process (Figure 2) forms a set of filter coefficients that directs the spatial main lobe to the signal of interest while directing nulls to the undesired interfering signals. The adaptation process is performed subject to a constraint that the steering vector has unity gain in the signal direction. The steady state weights of such a beamformer form the minimum variance distortionless response (MVDR) from the array elements [2]. The beam forming process of extracting the time signal component from a selected direction while suppressing...
signals arriving from other directions is the same as the process performed by a tapped delay line FIR filter extracting a time series from a selected frequency band while cancelling signal components in other frequency bands contained in a single time series. The adaptive beamforming problem is called a sidelobe canceller while the tapped delay line filter is called a line canceller. For reasons of numerical robustness and computational complexity, a common method for computing the required weight vector without directly inverting the correlation matrix is based on QR decomposition [2], and this is the approach adopted here.

### III. QRD-RLS ALGORITHM

Adaptive signal processing algorithms are generally used for solving a specified set of equations as given below:

\[
\begin{align*}
  x_1(1) c_0 + x_2(1) c_1 + \ldots + x_N(1) c_N &= y(1) + \epsilon(1) \\
  x_1(2) c_0 + x_2(2) c_1 + \ldots + x_N(2) c_N &= y(2) + \epsilon(2) \\
  &\vdots \\
  x_1(m) c_0 + x_2(m) c_1 + \ldots + x_N(m) c_N &= y(m) + \epsilon(m)
\end{align*}
\]

where \( m > N \).

As mentioned earlier, among the different algorithms, the RLS algorithm is generally preferred for its fast convergence property. The least squares approach attempts to find the set of coefficients \( c_n \) that minimizes the sum of squares of the errors, i.e. representing \( \min \sum_m \epsilon(m)^2 \), equation 1 can be represented in matrix form as

\[
Xc = y + e
\]

Where \( X \) is a matrix \((m \times N, \text{with} \ m > N)\) of noisy observations, \( y \) is a known training sequence, and \( c \) is the coefficient vector to be computed such that the error vector \( e \) is minimized. Direct computation of the coefficient vector \( c \) involves matrix inversion, which is generally undesirable for hardware implementation. Matrix decomposition-based least squares schemes such as Cholesky, lower upper (LU), singular value (SV), and QR decompositions avoid explicit matrix inversions and are more robust and well suited for hardware implementation [5].

The least squares algorithm attempts to solve for the coefficient vector \( c \) from \( X \) and \( y \). To realize this, the QR decomposition algorithm is first used to transform the matrix \( X \) into an upper triangular matrix \( R \) (\( N x N \) matrix) and the vector \( y \) into another vector \( u \) such that \( Rc = u \). The coefficients vector \( c \) is then computed using a procedure called back substitution, which involves solving the equations shown in equation 3.

\[
C_N = \frac{u_N}{R_{NN}} \\
C_i = \frac{1}{R_{ii}} \left( N_i - \sum_{f=i+1}^{N} R_{if} c_f \right) \text{ for } i = N-1, \ldots, 1
\]

The QRD-RLS algorithm flow is depicted in Figure 3.
IV. SYSTOLIC ARRAY BASED QR DECOMPOSITION

The QR decomposition of the input matrix $X$ can be performed, as illustrated in Figure 4, using the well known systolic array architecture [6]. The rows of matrix $X$ are fed as inputs to the array from the top along with the corresponding element of the vector $y$. The $R$ and $u$ values held in each of the cells once all the inputs have been passed through the matrix are the outputs from QR decomposition. These values are subsequently used to derive the coefficients using back substitution technique.

![Figure 4: Systolic Array architecture for QR Decomposition](image)

Each of the cells in the array can be implemented as a CORDIC block. CORDIC describes a method to perform a number of functions, including trigonometric, hyperbolic and logarithmic functions [7]. The algorithm is iterative, and uses only additions, subtractions and shift operations. This makes it very attractive for hardware implementations.

V. IMPLEMENTATION OF ADAPTIVE BEAMFORMER

In the proposed work, the adaptive beamformer has been implemented on the Virtex 2Pxc2vp30-7f1896 FPGA device by using AccelDSP tool of Xilinx. This particular device has been chosen intentionally to compare the results with the implementation of the same adaptive beamformer by [8]. In [8] the System Generator design tool from Xilinx was used to implement and test the QR decomposition design on the Virtex 2Pxc2vp30-7f1896 FPGA.

Simulation results by [8] and our work have been shown in Figures 5 to 7. Figure 5 show the output of broadside array without using adaptive beamformer. Figure 6 show the performance of adaptive beamformer proposed by [8] while Figure 7 shows the performance of our design. From Figures 6 and 7, it is clearly visible that
our design shows better performance in terms of side lobe cancellation. Figure 8 shows the RTL diagram of proposed adaptive beamformer by using Xilinx Virtex-2 Pro.

![RTL diagram of proposed adaptive beamformer by using Xilinx Virtex-2 Pro.](image)

**Figure: 5 Broadside Array Output without Adaptive Beamforming**

![Beamformer Output - FFT](image)

**Figure: 6 Broadside Array Output with Adaptive Beamforming [8]**

![Beamformer Output - FFT](image)

**Figure: 7 Broadside Array Output with Adaptive Beamforming**

RTL diagram of proposed adaptive beamformer by using Xilinx Virtex-2 Pro.
AREA OPTIMIZED FPGA IMPLEMENTATION OF ADAPTIVE BEAMFORMER

![RTL diagram of proposed adaptive beamformer](image)

Table I shows the comparison of the area used by our design and the design proposed by [8].

<table>
<thead>
<tr>
<th>Resources</th>
<th>Our Work</th>
<th>Work by [8]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slices</td>
<td>1128</td>
<td>5175</td>
</tr>
<tr>
<td>Number of Slice Flip Flops</td>
<td>494</td>
<td>5052</td>
</tr>
<tr>
<td>Number of 4 input LUTs</td>
<td>2137</td>
<td>7316</td>
</tr>
<tr>
<td>Number of bonded IOBs</td>
<td>99</td>
<td>1304</td>
</tr>
<tr>
<td>Number of MULT18X18s</td>
<td>15</td>
<td>41</td>
</tr>
</tbody>
</table>

From Table I, it can be concluded that the resources utilized by our design are far better as compared to the design proposed by [8].

V. CONCLUSION

This paper describes the FPGA implementation of a QRD-RLS algorithm for implementation of an adaptive beamformer. The design uses CORDIC-based processing arithmetic that is suitable for FPGA like the Xilinx Virtex-2 Pro. We have used AccelDSP tool by Xilinx for implementation of the adaptive beamformer. Simulation results have been compared with the implementation by [8] where they have used the System Generator tool by Xilinx. Results show the superiority of our work in terms of area utilization.

VI. REFERENCES