# PERFORMANCE ANALYSIS OF DISCRETE COSINE TRANSFORM IN MULTIBEAMFORMING

A Thesis Submitted to the College of

Graduate Studies and Research

In Partial Fulfillment of the Requirements

For the Degree of Master of Science

In the Department of Electrical and Computer Engineering

University of Saskatchewan

Saskatoon

By

ZIAD IBNE GIAS

© Ziad Ibne Gias, May/2015. All rights reserved.

# PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, it is agreed that the Libraries of this University may make it freely available for inspection. Permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professors who supervised this thesis work or, in their absence, by the Head of the Department of Electrical and Computer Engineering or the Dean of the College of Graduate Studies and Research at the University of Saskatchewan. Any copying, publication, or use of this thesis, or parts thereof, for financial gain without the written permission of the author is strictly prohibited. Proper recognition shall be given to the author and to the University of Saskatchewan in any scholarly use which may be made of any material in this thesis.

Request for permission to copy or to make any other use of material in this thesis in whole or in part should be addressed to:

Head of the Department of Electrical and Computer Engineering

57 Campus Drive

University of Saskatchewan

Saskatoon, Saskatchewan, Canada

S7N 5A9

# ABSTRACT

Aperture arrays are widely used in beamforming applications where element signals are steered to a particular direction of interest and a single beam is formed. Multibeamforming is an extension of single beamforming, which is desired in the fields where sources located in multiple directions are of interest. Discrete Fourier Transform (DFT) is usually used in these scenarios to segregate the received signals based on their direction of arrivals. In case of broadband signals, DFT of the data at each sensor of an array decomposes the signal into multiple narrowband signals. However, if hardware cost and implementation complexity are of concern while maintaining the desired performance, Discrete Cosine Transform (DCT) outperforms DFT.

In this work, instead of DFT, the Discrete Cosine Transform (DCT) is used to decompose the received signal into multiple beams into multiple directions. DCT offers simple and efficient hardware implementation. Also, while low frequency signals are of interest, DCT can process correlated data and perform close to the ideal Karhunen-Loeve Transform (KLT).

To further improve the accuracy and reduce the implementation cost, an efficient technique using Algebraic Integer Quantization (AIQ) of the DCT is presented. Both 8-point and 16-point versions of DCT using AIQ mapping have been presented and their performance is analyzed in terms of accuracy and hardware complexity. It has been shown that the proposed AIQ DCT offers considerable savings in hardware compared to DFT and classical DCT while maintaining the same accuracy of beam steering in multibeamforming application.

# ACKNOWLEDGEMENTS

I would like to express my heartiest gratitude to my supervisor, Professor Khan A. Wahid for his guidance and teaching throughout the pursuit of my masters degree. His guidance and encouragement from the very beginning of this program has always kept me on the right track and increased my confidence in doing research independently. I am privileged to have the opportunity to work under his supervision.

I would also like to thank all of my friends and colleagues for their support, Department of Electrical and Computer Engineering of University of Saskatchewan and Department of Applied Physics, Electronics and Communication Engineering of University of Dhaka for providing quality education and research environment.

My heartiest thanks to Dr. Nurul Chowdhury and Dr. Li Chen for being in my M.Sc. thesis advisory committee and Professor Akindele Odeshi for being an external examiner. Their constructive criticism, comments and suggestions have been and will be an asset.

Finally I would like to express my gratitude to my parents, Md. Gias Uddin and Late Mrs. Piar Jahan, my sister Tabas Ahmed and last but not the least my wife Samia Mahmud for their unconditional support, patience and inspiration all over.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

Figure Page

# LIST OF ABBREVIATIONS

| | |
|---|---|
| ADC | Analog to Digital Converter |
| AF | Array Factor |
| AIQ | Algebraic Integer Quantization |
| ATA | Allen Telescope Array |
| BAS | Bouguezel-Ahmad-Swamy approximation |
| CB | Cintra-Bayer approximation |
| DAC | Digital to Analog Converter |
| DCT | Discrete Cosine Transform |
| DDC | Digital Down Converter |
| DFT | Discrete Foutrier Transform |
| DST | Discrete Sine Transform |
| FP | Floating Point |
| FFT | Fast Fourier Transform |
| HEVC | High Efficiency Video Coding |
| JPEG | Joint Photographic Experts Group |
| LOFAR | Low Frequency Array |
| MPEG | Moving Picture Experts Group |
| PMCB | Potluri-Madanayake-Cintra-Bayer |
| SKA | Square Kilometer Array |
| ULA | Uniform Linear Array |
| WHT | Walsh Hadamard Transform |

# CHAPTER 1

# INTRODUCTION

Antennas employed in wireless communication and space-imaging systems range from one simple antenna to large groups of complex three-dimensional (3D) antennas [1]. Groups of antennas have traditionally been used to form beams to be transmitted to or received from a particular direction. This enables filtering of signals spatially. Signals having the same temporal frequency but generating from different locations require spatial filters, i.e, beamformers to retrieve the desired signal [2].

The size of antenna required to transmit or receive quality signals to or from a large distance is often not practical. Grouping a number of antennas to form beams serves the purpose in these situations. The factors that affect beamforming are: antenna types and their geographical arrangement while grouping, technology applied to combine the signals, and form beams.

## 1.1 Theory of Beamforming

### 1.1.1 Basic Antennas

The very basic element of a communication system is antenna. Transmission and reception of signals over a distance is made possible due to the application of this element. It can be a simple wire or a complicated radio telescope built with thousands of antennas. The basic antennas or single element antennas can also be of different kinds depending on their shape, functionality, directionality, performance, principle, application and architecture. However, all they do is convert the electrical signal into electromagnetic wave and vice versa.

### 1.1.2 Antenna Arrays and Smart Antennas

In many long distance communications, it is necessary to design antennas with high directivity [3]. High directivity can be achieved by increasing the electrical dimension of the antenna. Increasing geometric dimension is not always feasible. However, assembling multiple antenna elements is another way of achieving the same outcome, which can be defined as an array. Fig. 1.1 [36] shows an example of an aperture array. Different kinds of antennas are employed in forming arrays as required by the application. The response of the arrays can be made adaptive. Adaptive arrays are more popularly known as smart antennas. Smart antennas are named so due to their adaptive nature.

### 1.1.3 Array Types

Antenna arrays are classified primarily based on their geometric configuration. Another classifying factor is the antenna types that are used in the array. Selecting identical radiators is the rule of thumb in designing antenna arrays.

Fig. 1.1 [36]. Example of an aperture array

The response of the arrays depends on the kind of antennas used in forming the array. Both omnidirectional and directional antennas can be used in arrays. Omnidirectional antennas are those, which produce radiation in all directions having the same signal strength. Directional antennas are responsive or sensitive to particular directions only. Arrays in beamforming can be considered as a unit working as a directional antenna. Geometrically, the array can be linear, planar, circular or a combinational one. A linear array is one where the antenna elements are organized along a straight line. This would be a one-dimensional (1D) array. A planar array is a two-dimensional (2D) array where the elements are arranged in a rectangular shape. When the geometric shape of the array is circular, evidently, that would be a circular array.

Arrays can also be uniform or non-uniform depending on the inter-element spacing. When the elements are equidistant, the array is a uniform array.

The array response varies depending on these classifications. Also, the excitation phase and amplitude of each element along with the individual response determine the output pattern of an antenna array.

### 1.1.4 Beamforming concepts

Aperture arrays are designed so as to transmit or receive radiation of a particular pattern, which contain information to/from a particular direction. In radio communication, radar and space imaging applications, determining the direction of arrival is an important task. In those cases, aperture arrays are implemented to form beams in a particular direction of interest.

An introduction to beamforming concepts and algorithms is presented in [2][4]. Consider an 8-element aperture array arranged along the z-axis as in Fig. 1.2. However, an array can be planar or circular with uniform or non-uniform in terms of distance between them. In a Uniform

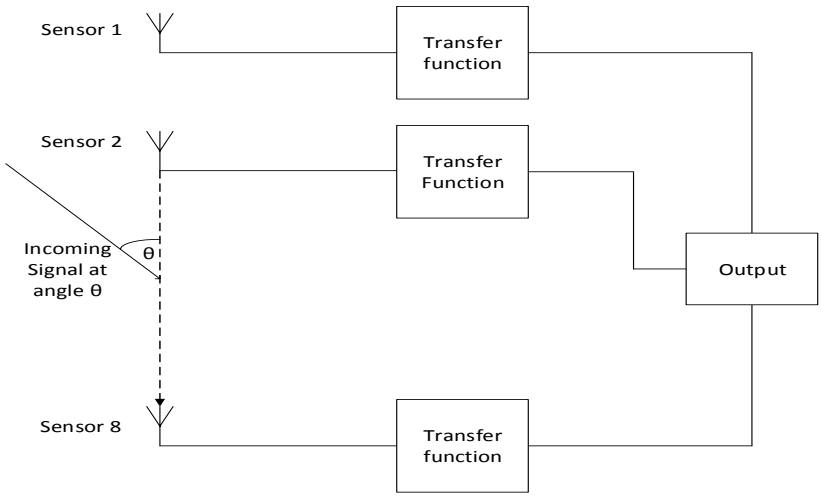



Fig. 1.2. 8-element Aperture Array

Linear Array (ULA), the same distance from each other separates each element. In this particular case, a half wavelength distance is considered between each pair of elements.

Let us assume a uniform plane wave arriving at the array at an angle $\theta$ with respect to the z-axis. Considering the elements to be isotropic, the element pattern would be proportional to the E-field and summing together the response of all the elements the array pattern, *re*, would be:

$$re = \sum_{n=1}^{8} e^{-Nj\pi\cos\theta} \tag{1.1}$$

If we take all other coordinates to be zero except that of the z-axis and consider the weighting such as to direct the beam to $\theta_e$ direction with shifting phases of each element, the array factor would be:

$$AF = \sum_{N=1}^{8} e^{-Nj\pi(\cos\theta_e - \cos\theta)} \tag{1.2}$$

Here, $\theta$ is from 0 to 2π, $\theta_e$ is the expected beam direction and $N$ is the number of sensors in the array. Beam generated towards the expected beam direction having the maximum power is termed as main beam or main lobe. All the other beams having lower power are called side lobes. Side lobes are signals to or from undesired directions. Attention is given to lower the number and the power of the side lobes as this is a major performance criterion for beamforming algorithms. Fig. 1.3 shows an example of main lobes and side lobes on polar plot.

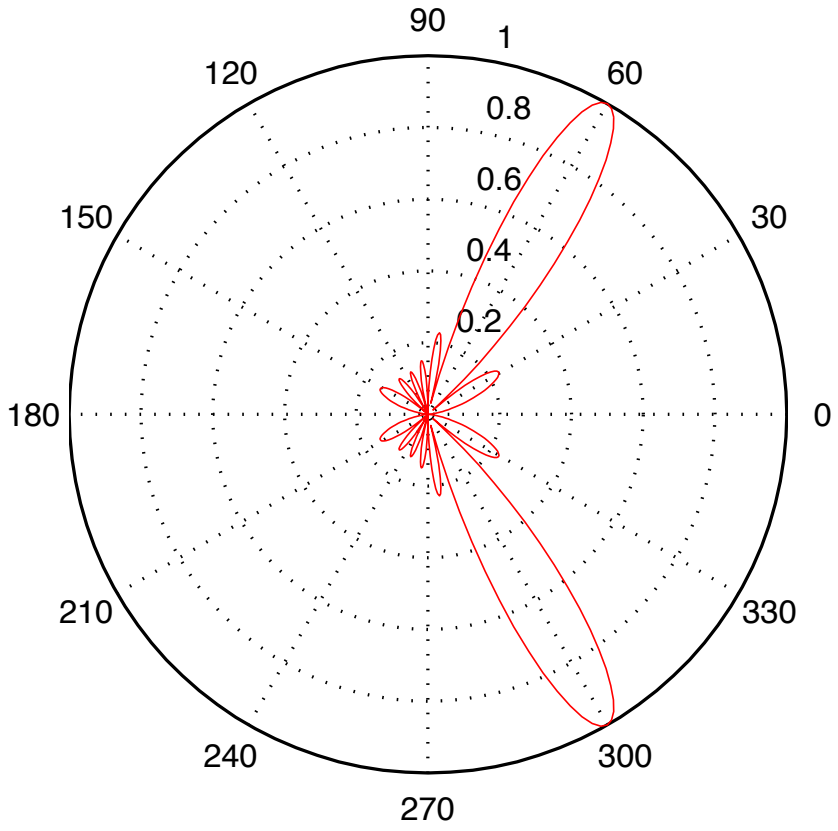



Fig. 1.3. Example of main lobe and side lobe on polar plot

Here, beams directed towards 60 degrees and 300 degrees are the main lobes. The beam directed towards 300 degree is sometimes termed as back lobe. All the other beams with distributed along the other directions having lower magnitude are the side lobes.

In multibeamforming applications, there would be a number of main lobes in the desired directions. Fig. 1.4 [34] shows an artists view of the generated beams from an antenna array. This is an example of multibeamormer as can be realized from the figure having multiple beams in multiple directions. Beams with maximum power and toward the desired directions are main beams. All the other beams are unwanted side lobes. In an ideal multibeamformer, there would be multiple main lobes seperated only by nulls instead of any side lobes in between. Nulls are, as the name implies, absence of any field strength. Nulls are produced in desired directions if it is of particular interest not to transmit or receive any signal to or from that directions.

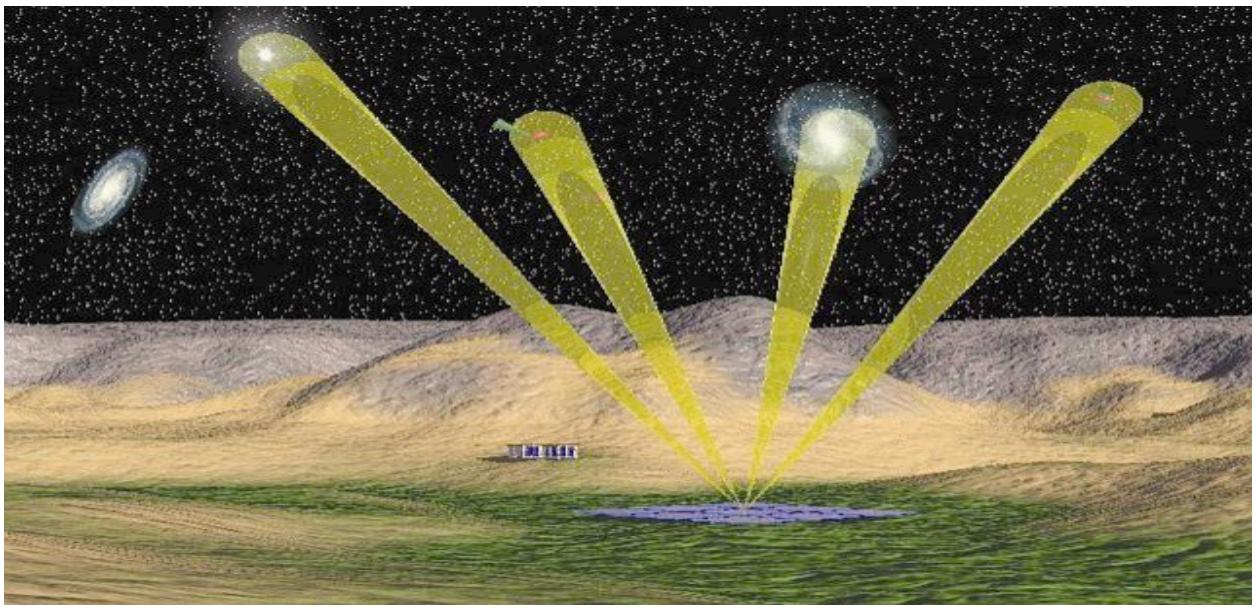


Fig. 1.4. [34] Visualization of beams to/from an array

Antenna theory dictates that the array response would be same for both reception and transmission, and the response of an array is characterized by the array factor and the types of antennas used to form the array. For simplicity and practical reasons, identical antennas are form arrays.

### 1.1.5 Major Classifications

In broad categorization, beamformers can be classified as data independent and data dependent [2][4]. Data independent beamformers are the conventional beamformers which steer the beam to the look direction irrespective of the properties of the signal data. This is similar to mechanically moving the antenna to the desired direction except it moves the antenna electronically instead of mechanically. Data dependent beamformers, on the other hand, respond accordingly to the received signal data. These are also termed as adaptive beamformers.

The most common of the conventional beamformers is the time delay beamformer where a delay in the time domain is introduced to produce in-phase signals and thereby a beam in the desired direction. However, producing time delay with cables is vulnerable to various physical constraints and errors [5].

Availability and advancement in digital technology enables the employment of digital processing techniques in beamforming which can overcome the erroneous analog time delay concerns in beamforming. Time delays can be achieved in digital beamforming by buffering the respective signals by multiple sample times as required. The high frequency signal has to be down-converted before processing in the digital processor. Fig. 1.5 shows a generalized system architecture employing digital beamforming.

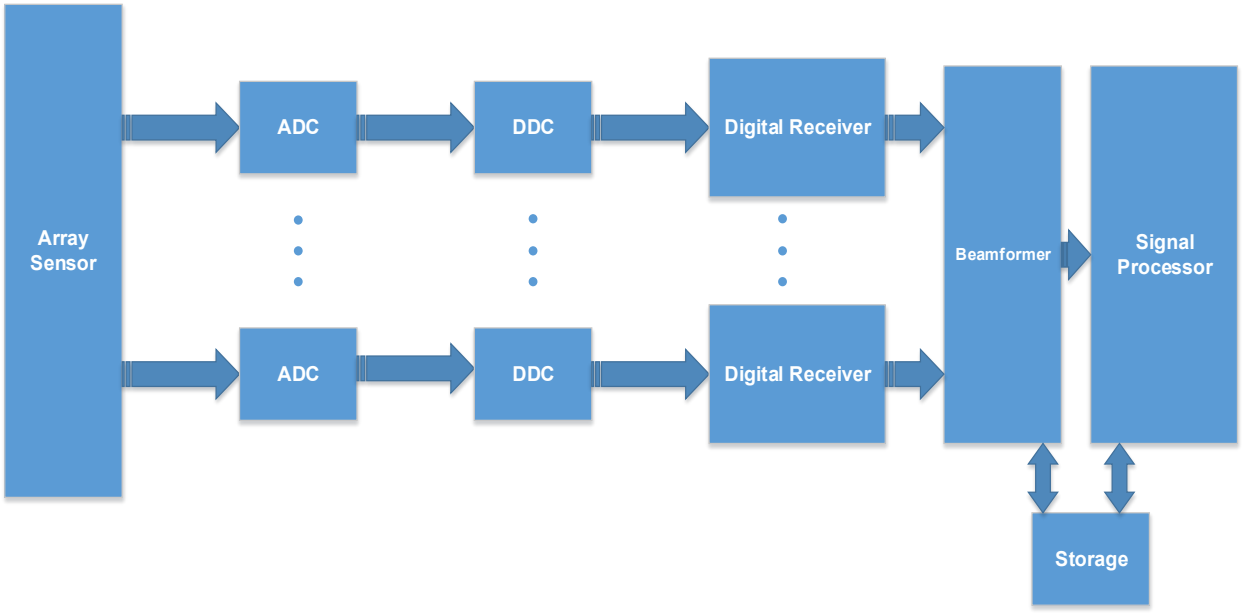



Fig. 1.5. Flow graph of a system with beamformer

The signals received from the sensor array are down-converted with Digital Down Converters (DDC) after doing the digitization using Analog to Digital Converters (ADC) and then received at the appropriate digital receivers before applying to the beamformer processor. The beamformed output is then sent to the main processing unit for performing the appropriate signal processing to produce the desired result.

Another way of producing time delay is to produce phase shifts and thus to do the beamforming in the frequency domain. But this also limits the processing only to narrowband signals. In the case of broadband signals, the signal has to be segregated into multiple narrowband signals first to implement the phase shift beamformer. Fast Fourier Transform is traditionally used in this case. Thus, beamformers in the case of narrowband and wideband signals have to be different to ensure optimal and accurate performance [6].

### 1.1.6 Applications

Beamformers find extensive importance and applications in various fields like RADAR, SONAR and other wireless communication systems [2]. Mobile communication systems largely employ beamformers to determine the direction of arrival of the signals [4]. In space imaging, astrophysical and geophysical explorations systems, arrays with thousands of antennas spreading over a large area are implemented to form beams to desired directions. Square Kilometer Array is the most talked about aperture array these days in the space imaging area where the apertures are arranged across an area of one square kilometer to build the beamformer. Besides these, biomedical technology also involve beamformers for various medical applications.

### 1.1.7 Multibeamforming

Multibeamforming, as the name implies, is a type of beamforming where multiple beams are produced from a single array or entity. Applications where the directions of interest are spread over a wide area find multibeamformers exclusively essential. More on multibeamforming will be discussed throughout the thesis.

## 1.2 Algebraic Integer

In number theory, an algebraic integer is defined as the root of a monic polynomial [17]. The leading coefficient in this case has to be 1 unlike algebraic numbers. Let us consider the following monic polynomial:

$$x^n + c_{n-1}x^{n-1} + c_{n-2}x^{n-2} + .....+ c_1x + c_0 = 0 \tag{1.3}$$

If $r$ is a root of the above equation, it would be an algebraic integer.

As an example, let us consider $\omega = e^{2\pi i/R}$ to be the $R^{th}$ root of unity. Here, $R = 2^v$ and $v \geq 2$. If we consider $v$=3, $R$ would be 8. In that case, $\omega$ would the root of: $x^4 + 1 = 0$ over $Z$ and $\{\omega, \omega^2, \omega^3\}$ would be the integral basis of $Z[\omega]$. The highest degree of $\omega$ would be restricted to $\{(R/2)-1\}$.

According to the property of algebraic integers, the sum and product of algebraic integers are also algebraic integers.

An algebraic integer in DSP has been considered since long back [16-17]. Algebraic integers have been proposed and implemented to realize the DCT and IDCT coefficients in the past [13]. Multidimensional algebraic integers have been considered to reduce the dynamic range of the transform coefficients. In the field of image compression, significant savings have been observed while using multidimensional algebraic integers. Also, an error free implementation of the classical DCT is possible with algebraic integer quantization.

In this thesis, we will use algebraic integer techniques to implement transform basis functions in beamforming applications. The algebraic integers used in the thesis will be denoted as, Algebraic Integer Quantiation or in short AIQ.

## 1.3 Thesis Motivation

Multibeamforming is drawing significant attention due to it's widespread applications in recent times. Demands of high-resolution images in astrophysics and other geophysical fields call for wide view angle having uncompromised image quality. Increasing demand in wireless communications is threatening the limited frequency spectrum to find an alternative before it

runs out of scopes. In addition, the advent of huge radio telescopes such as Square Kilometer Array (SKA) calls for efficient and accurate beamforming algorithms to cope with the tremendous expectation. One of the interesting facts of Square Kilometer Array is that an airport radar located tens of light years away would be detected at the SKA telescope [34]. In recent radio telescopes containing 100 to 1000 antennas to form the array are grouped following a hierarchical architecture, i.e, tile level and station level. Outputs from the tile levels can be grouped and correlated together in the station level for further processing. A multibeamforming algorithm can be considered for the tile level architectures to provide multiple beams to be further processed.

Discrete Fourier Transform (DFT) has been the algorithm explored and applied to form multiple beams. However, considering its implementation complexity and cost, it is time to consider an alternative algorithm that can replace DFT and reduce implementation complexity.

## 1.4 Thesis Objective

The main objectives of this thesis are summarized in steps as the followings:

- Apply multiple discrete transforms such as, Discrete Sine Transform, Walsh Hadamard Transform and Discrete Cosine Transform in multibeamforming and analyze their performance. The purpose is to determine a suitable alternative for DFT.

- DCT has reported the lowest cost among all discrete transforms with good performance. The next task is to apply Integer DCT approximation techniques and compare their performance. The purpose is to propose the best approximation DCT algorithm for multibeamforming application.

- The next objective is to apply an error-free implementation of classical 8-point DCT using AIQ mapping and assess its performance in terms of accuracy and hardware complexity.

- Lastly, propose both 1D and 3D AIQ mapping for the 16-point DCT and assess its performance for the particular application.

## 1.5 Thesis Organization

The remaining portion of the thesis is organized as follows. Chapter 2 contains the relevant works along with a comparison among different transforms in multibeamforming. Different benchmark DCT algorithms and their performances are compared and analyzed in chapter 3. The application of 8-point AIQ DCT in multibeamforming is proposed and its performance is analyzed in chapter 4. In chapter 5, the 16-point AIQ algorithm is given with its performance analysis in multibeamforming. Finally, chapter 6 contains the conclusion and the scope of future work.

# CHAPTER 2

## RELEVANT WORKS

Extensive research and application with array and beamforming technology has been possible with the advancement in digital technology. Expensive and unrealistic earlier, A/D converters, microprocessors, random access memories and other digital circuitry have become more available these days. Various radio telescope projects have been developed and are under development in space imaging and communication systems due the advancements in digital technology. Square Kilometer Array (SKA) [23-25] is the latest addition to the space imaging telescopes and is still under research and development. Some other examples include LOFAR [26-27], ATA [28] and Argus [29].

## 2.1 Beamforming Algorithms

The most basic kind of beamformer is the delay sum beamformer where all the received/transmitted signals are delayed or advanced with reference to one particular sensor in the array so that all the signals arrive or transmit at the same instant of time and thereby result in constructive interferences only. In analog domain, the delays can be achieved simply with the cabling, but it lacks accuracy and also comes with physical constraints.

An efficient alternative of this is to implement the beamformer in the digital domain. Sampling and buffering the samples in that case can implement the delay. However, the sampling has to be done at a rate greater than the Nyquist rate to avoid aliasing. Another criterion in beamforming that must be taken into account to avoid aliasing is the distance between the array sensors. Delay sum beamformers in digital domain also require a significant amount of

storage to produce buffering. The cable bandwidth has to be large enough to accomplish the required sampling rate. Analog to Digital Converters are required to convert the antenna signal to be processed by a digital processor. In the case of high frequency signals, Radio Frequency translators are used prior to ADCs to bring the signal frequency down before converting. After the conversion, the signals are passed through the digital down-conversion process to shift the center frequency to 0 Hz. This produces a quadrature output signal which is used as input in the beamformer.

Phase shift beamformer is another way to implement a delay sum beamformer in the frequency domain. Phase delays are implemented in the array sensors instead of time delay in this case. It is designed so as to produce the output signals from all the sensors in phase. The output signals are then summed to get the beam in the desired direction. To apply phase shift beamformers in broadband applications, the signal has to be segregated into multiple narrowband signals prior to applying the phase shifts to the corresponding narrowband signals. A comparative analysis of different beamforming algorithms is presented in [31].

## 2.2 Multibeamforming

The radio telescopes deployed in radio astronomy need to cover a wide viewing angle with high resolution. Multibeamforming provides multiple beams with higher resolution that can have a wide area of coverage. Apart from that, the increasing demand in wireless communication systems coupled with limited frequency spectrum calls for frequency reuse maintaining the quality of demand. Smart antennas and other multibeamforming algorithms can facilitate this case to a great extent.

Smart antennas with multibeamforming can produce multiple beams in different directions. These beams can reuse the same frequency in wireless communication systems which extensively increase the service capacity. Multibeamformers, when implemented with satellite communication systems, can increase the efficiency in terms of resolution and capacity.

Multibeamforming and smart antennas in wireless communication systems have been analyzed in [31]. Multibeamforming with FFT has been discussed in [32] with null forming in required directions in addition to forming multiple beams in multiple directions.

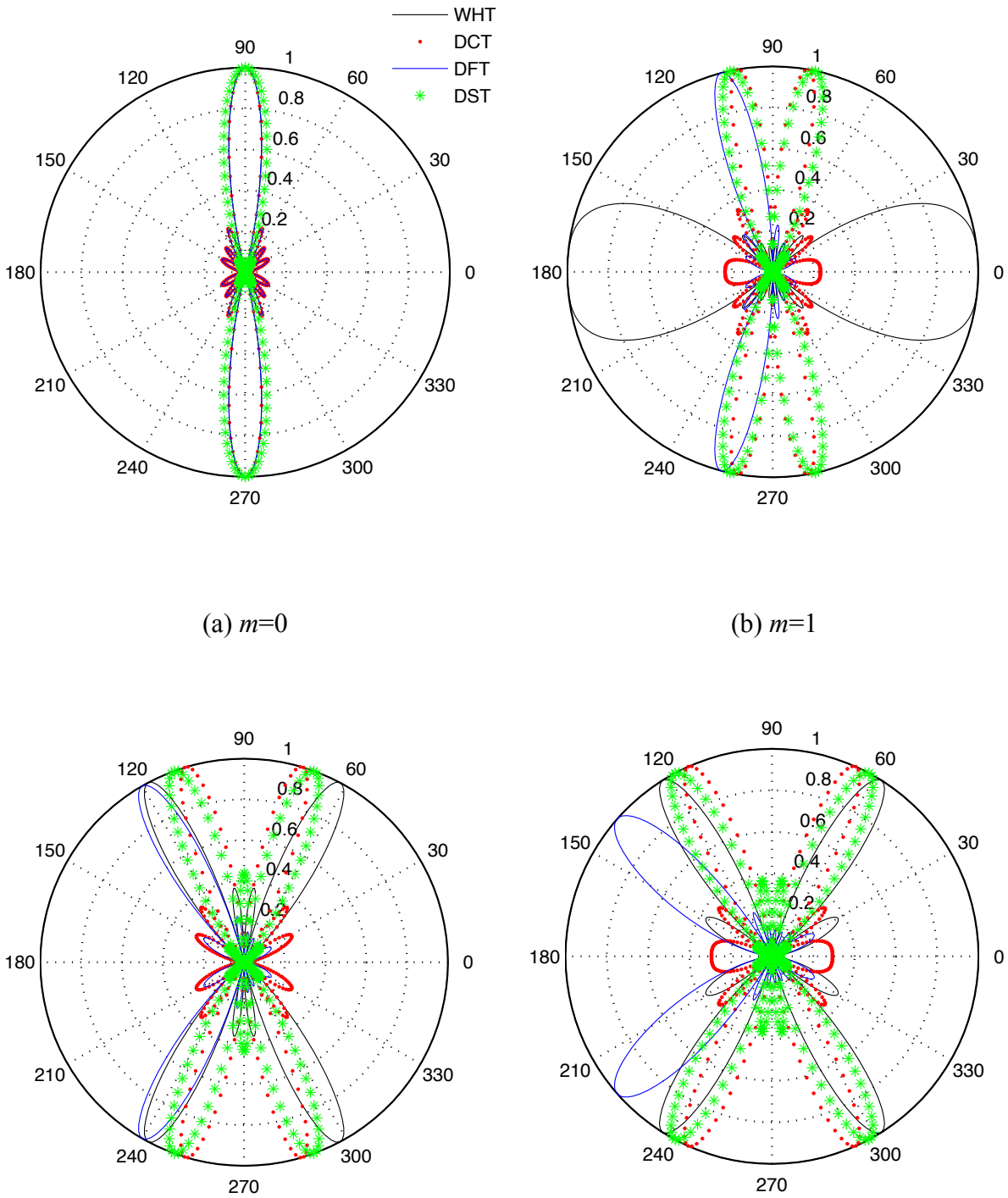
The cost of multibeamforming in $N$ directions with direct matrix multiplication is $O(N^2)$ where the same with FFT can achieved with $O(N \log_2 N)$.

## 2.3 Multibeamforming With Different Transforms

The array response vector for an eight element uniform linear array with impinging plane wave along the z-axis with different transformation matrices is the following:

$$\hat{Y} = \sum_{N=1}^{8} x(m,n).e^{-Nj\pi(\cos\theta_e - \cos\theta)} \qquad 2.1$$

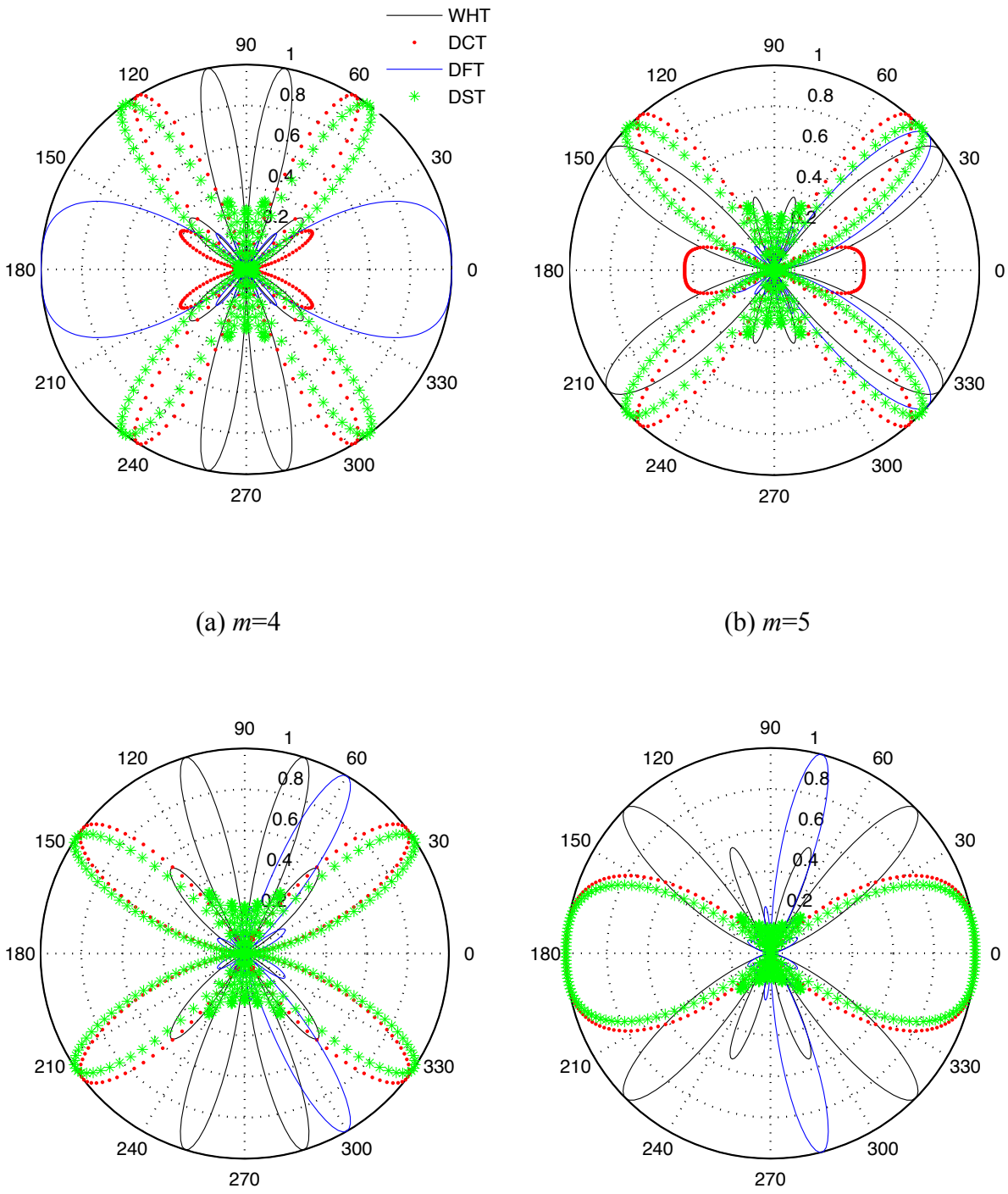
*x(m,n)* is the preferred transformation matrix employed to form the beams with *m* and *n* being the row and column of the matrix. The normalized polar plots of the array response of ULA using different transformation matrices are shown in Fig. 2.1 and Fig. 2.2. WHT, DST and DCT are considered here with DFT to provide us with a comparative view. As seen from the polar plots, if the beam is steered towards broadside, the same can be achieved with all the transforms for *m*=0. For all the other indices, a fan of beams is formed. With each row, a beam in a different direction is achieved. Multiple beams are formed in multiple directions with all the transforms.




(a) $m$=0 (b) $m$=1

(c) $m$=2 (d) $m$=3

Fig. 2.1. Polar Plot with DWT, DST, DCT and DFT for $m$=0,1,2 and 3 with the initial beam steered at 90 degrees.




(a) $m$=4 (b) $m$=5

(c) $m$=6 (d) $m$=7

Fig. 2.2. Polar Plot with DWT, DST, DCT and DFT for $m$=4,5,6 and 7 with the initial beam steered at 90 degrees.

However, a number of side lobes are also produced with all the transforms, which can be considered as noise and unwanted signals from unwanted directions. The normalized peak of a main beam is one. All the beams having peak lower than one are from or to unwanted directions.

The locations in degrees and levels in normalized form of the lobes for all the transforms for m=7 are presented in Table 2.1. It can be observed from the table that, for this particular index, transform with the lowest peak for a side lobe is DCT. Besides, transform that has the minimum highest peak for a side lobe is also DCT. The number of side lobes produced is also less in case of DCT compared to DFT.

The directions of main lobes are different with different transforms with the same indices. For *m*=7, the directions of main and back lobes having normalized unity peak are 60 and 300 degree with DFT. DCT produces main and back lobes along 37, 143, 217 and 323 degrees. The direction of main lobes with DST for *m*=7 are 34 and 146 degrees. The back lobes in this case are along 214 and 326 degrees. Main lobes along 73 and 107 degrees and back lobes along 253 and 287 degrees are observed for *m*=7 in case of WHT.

The directions of the main beams produced are closest in case of DCT and DST. Although they produce almost the same directional main beam, the number of side lobes with DST is greater than DCT. For all the directions, beams can be obtained for all the transforms. However, the index varies with transforms. If a beam in 60 degree is of interest, DFT produces a beam in this direction with normalized unity peak for *m*=6. A main beam along 60-degree direction is observed for the fifth row in case of DCT. DST, as marked with green asterisks on the polar plots, produces beam in the direction of interest for *m*=3. The third row in case of WHT provides the beam in the desired direction.

WHT offers the minimum number of side lobes but the peak of the side lobes produced in this case is more than 50 percent of the main lobe, which causes a performance concern. The normalized value as can be seen from Table 2.1 is 0.5485.

**Table 2.1 Beam locations (Degrees) and levels (Normalized) for different transforms for *m*=7**

| **DFT** | | **DST** | | **WHT** | | **DCT** | |
|---|---|---|---|---|---|---|---|
| **Location** | **Level** | **Location** | **Level** | **Location** | **Level** | **Location** | **Level** |
| 31 | 0.2291 | 34 | 1.0000 | 49 | 0.5485 | 37 | 1.0000 |
| 60 | 1.0000 | 59 | 0.3453 | 73 | 1.0000 | 66 | 0.1005 |
| 82 | 0.2291 | 75 | 0.2530 | 107 | 1.0000 | 81 | 0.0258 |
| 97 | 0.1507 | 90 | 0.2321 | 131 | 0.5485 | 99 | 0.0258 |
| 112 | 0.1275 | 105 | 0.2530 | 229 | 0.5485 | 114 | 0.1005 |
| 129 | 0.1274 | 121 | 0.3453 | 253 | 1.0000 | 143 | 1.0000 |
| 152 | 0.1509 | 146 | 1.0000 | 287 | 1.0000 | 217 | 1.0000 |
| 208 | 0.1509 | 214 | 1.0000 | 311 | 0.5485 | 246 | 0.1005 |
| 231 | 0.1274 | 239 | 0.3453 | | | 261 | 0.0258 |
| 248 | 0.1275 | 255 | 0.2530 | | | 279 | 0.0258 |
| 263 | 0.1507 | 270 | 0.2321 | | | 294 | 0.1005 |
| 278 | 0.2291 | 285 | 0.2530 | | | 323 | 1.0000 |
| 300 | 1.0000 | 301 | 0.3453 | | | | |
| 329 | 0.2291 | 326 | 1.0000 | | | | |

DFT has side lobes having minimum peak of 0.1274 and maximum of 0.2291. 0.3453 is the maximum peak side lobe with DST. 0.2321 is the minimum peak. In case of DCT, 0.1005 and 0.0258 are the maximum and minimum levels of the side lobes for $m$=7.

## 2.4 Summary

Beamforming and multibeamforming have been a research interest for some time. Various algorithms for different applications and kinds of beamforming have been proposed and analyzed over time. Each algorithm has its own advantages and disadvantages. Application specific algorithms have also been developed. The advent of advancement in digital electronics and signal processing has facilitated consideration and implementation of different beamforming algorithms and also extended its field of application.

# CHAPTER 3

## PERFORMANCE ANALYSIS OF BENCHMARK DCT ALGORITHMS IN MULTIBEAMFORMING

Recently, the authors in [7] showed that significant performance improvement can be achieved using Approximate Discrete Cosine Transform (DCT), which is also known as Integer DCT, in multibeamforming. The DCT, due to its efficiency in de-correlating data, is the preferred filter in video coding [8]. Challenges in optimal bit rate and hardware implementation cost have resulted in a number of variations in integer DCT approximations. The benchmark DCT approximations considered here are JPEG, MPEG-2/4, H.264/AVC, VC-1, AVS and HEVC. The performance of these approximations has been analyzed here.

### 3.1 Benchmark DCT Matrices

The coefficients of the $8 \times 8$ matrix of a DCT approximation have to fulfill the properties of a DCT matrix. After extensive research and analysis, the DCT approximation matrices standardized for image and video compression have been developed. The 8-point 1-D DCT matrices for JPEG, MPEG, H.264/AVC, VC-1, AVS and HEVC are presented below [9-12]:

$$JPEG_8 = \begin{bmatrix} 362 & 362 & 362 & 362 & 362 & 362 & 362 & 362 \\ 502 & 426 & 284 & 100 & -100 & -284 & -426 & -502 \\ 473 & 196 & -196 & -473 & -473 & -196 & 196 & 473 \\ 426 & -100 & -502 & -284 & 284 & 502 & 100 & -426 \\ 362 & -362 & -362 & 362 & 362 & -362 & -362 & 362 \\ 284 & -502 & 100 & 426 & -426 & -100 & 502 & -284 \\ 196 & -473 & 473 & -196 & -196 & 473 & -473 & 196 \\ 100 & -284 & 426 & -502 & 502 & -426 & 284 & -100 \end{bmatrix}$$

$$MPEG_8 = \begin{bmatrix} 362 & 362 & 362 & 362 & 362 & 362 & 362 & 362 \\ 502 & 426 & 284 & 100 & -100 & -284 & -426 & -502 \\ 473 & 196 & -196 & -473 & -473 & -196 & 196 & 473 \\ 426 & -100 & -502 & -284 & 284 & 502 & 100 & -426 \\ 362 & -362 & -362 & 362 & 362 & -362 & -362 & 362 \\ 284 & -502 & 100 & 426 & -426 & -100 & 502 & -284 \\ 196 & -473 & 473 & -196 & -196 & 473 & -473 & 196 \\ 100 & -284 & 426 & -502 & 502 & -426 & 284 & -100 \end{bmatrix}$$

$$H.264_8 = \begin{bmatrix} 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 12 & 10 & 6 & 3 & -3 & -6 & -10 & -12 \\ 8 & 4 & -4 & -8 & -8 & -4 & 4 & 8 \\ 10 & -3 & -12 & -6 & 6 & 12 & 3 & -10 \\ 8 & -8 & -8 & 8 & 8 & -8 & -8 & 8 \\ 6 & -12 & 3 & 10 & -10 & -3 & 12 & -6 \\ 4 & -8 & 8 & -4 & -4 & 8 & -8 & 4 \\ 3 & -6 & 10 & -12 & 12 & -10 & 6 & -3 \end{bmatrix}$$

$$VC-1_8 = \begin{bmatrix} 12 & 12 & 12 & 12 & 12 & 12 & 12 & 12 \\ 16 & 15 & 9 & 4 & -4 & -9 & -15 & -16 \\ 16 & 6 & -6 & -16 & -16 & -6 & 6 & 16 \\ 15 & -4 & -16 & -9 & 9 & 16 & 4 & -15 \\ 12 & -12 & -12 & 12 & 12 & -12 & -12 & 12 \\ 9 & -16 & 4 & 15 & -15 & -4 & 16 & -9 \\ 6 & -16 & 16 & -6 & -6 & 16 & -16 & 6 \\ 4 & -9 & 15 & -16 & 16 & -15 & 9 & -4 \end{bmatrix}$$

$$AVS_8 = \begin{bmatrix} 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 10 & 9 & 6 & 2 & -2 & -6 & -9 & -10 \\ 10 & 4 & -4 & -10 & -10 & -4 & 4 & 10 \\ 9 & -2 & -10 & -6 & 6 & 10 & 2 & 9 \\ 8 & -8 & -8 & 8 & 8 & -8 & -8 & 8 \\ 6 & -10 & 2 & 9 & -9 & -2 & 10 & -6 \\ 4 & -10 & 10 & -4 & -4 & 10 & -10 & 4 \\ 2 & -6 & 9 & -10 & 10 & -9 & 6 & -2 \end{bmatrix}$$

$$HEVC_8 = \begin{bmatrix} 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 \\ 89 & 75 & 50 & 18 & -18 & -50 & -75 & -89 \\ 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 \\ 75 & -18 & -89 & -50 & 50 & 89 & 18 & -75 \\ 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 \\ 50 & -89 & 18 & 75 & -75 & -18 & 89 & -50 \\ 36 & -83 & 83 & -36 & -36 & 83 & -83 & 36 \\ 18 & -50 & 75 & -89 & 89 & -75 & 50 & -18 \end{bmatrix}$$

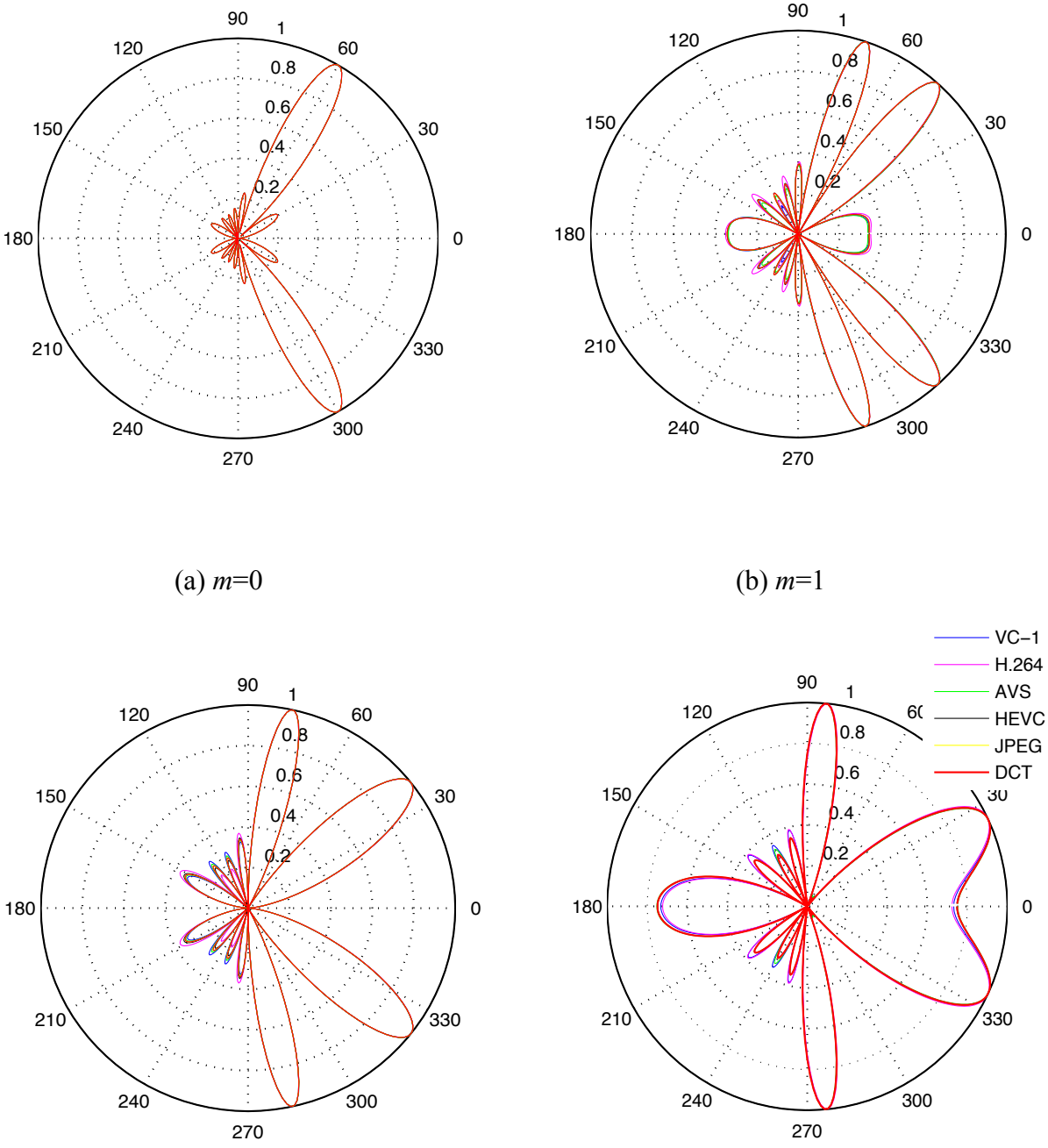
## 3.2 Comparison with FP DCT in Multibeamforming

Fig. 3.1 and Fig. 3.2 present the normalized polar plot of the following array factor with Uniform Linear Array with different transformation matrices.

$$\hat{Y} = \sum_{n=0}^{7} x(m,n).e^{(-nj2\pi fd(\cos\theta_e - \cos\theta)/c)} \qquad 3.1$$

As transformation matrices, AVC/H.264, AVS, VC-1, JPEG/MPEG-2/4, HEVC and FP DCT have been used for comparison. $\theta_e$ is taken to be 60 degrees in equation 3.1. For $m$=0, the main lobe is directed towards 60 degrees as the array has been steered towards that direction. This direction, as seen from Fig. 3.1 and Fig. 3.2, is the same for all transformation matrices applied here having the same beam level and direction for $m$=0. For the next indices, the beam pattern forms a fan of beams with reference to the initially steered direction and multiple beams in specific directions are achieved. In addition to the main beams, there are some small beams, which are the side lobes in unwanted directions. The preferred transformation matrix would be the one that shows the best performance in suppressing side lobes, i.e, that produces the minimum number of side lobes with the lowest peaks.

FP DCT is considered to be pivotal here to evaluate the performance of the other DCT approximations in multibeamforming.



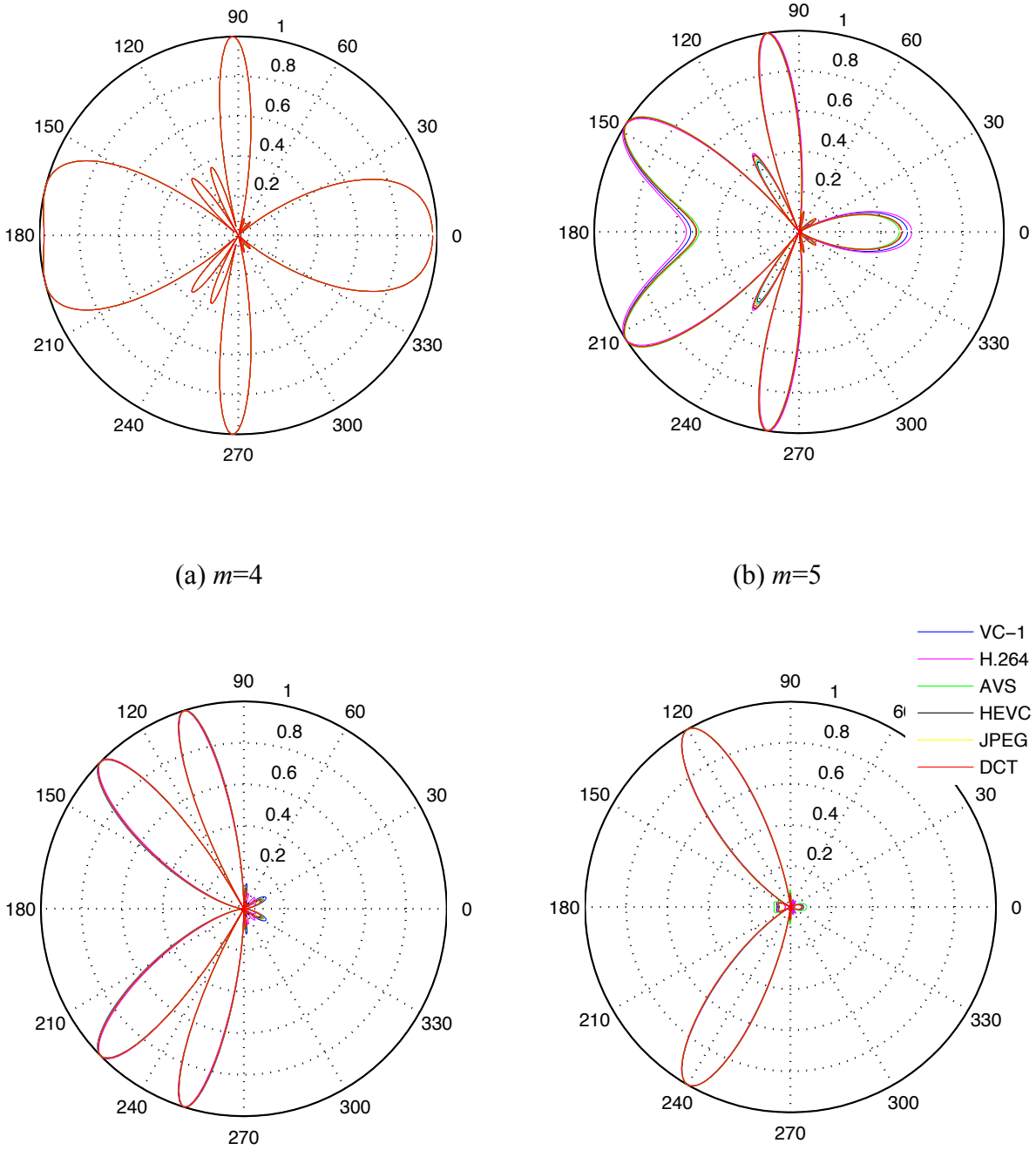

(a) *m*=0

(b) *m*=1

(c) *m*=2

(d) *m*=3

Fig. 3.1. Polar plots of the array factor of an eight-element aperture array steered at 60 degrees with different benchmark DCT algorithm for *m*=0,1,2 and 3




(a) *m*=4

(b) *m*=5

(c) *m*=6

(d) *m*=7

Fig. 3.2. Polar plots of the array factor of an eight-element aperture array steered at 60 degrees with different benchmark DCT algorithm for *m*=4,5,6 and 7

In most applications, the accuracy of the directions of main and side lobes have to be uncompromised.

Exact DCT and H.264/AVC have been plotted with solid red and magenta respectively. The side lobes with AVC compared to classical DCT are a bit of the higher order as can be seen from the polar plots. The main lobes are same with both the transforms. Similar performance is observed in case of VC-1. However, the main beams for all the standards have similar shape.

The worst performance is observed for the fourth and sixth rows where the widths and peaks of the side lobes are inseparable when compared to the main lobes. However, the same performance is observed with classical DCT for that particular index.

Approximate locations and normalized levels of all the lobes obtained with the main lobe being towards 60 degrees for all the standards are presented in Table 3.1.

**Table 3.1 Locations (Degrees) and levels (Normalized) of the lobes in classical DCT and H.264/AVC, AVS, HEVC, VC-1 and JPEG standards for *m*=0**

| FP DCT | | H.264/AVC [10] | | AVS and HEVC [10] | | VC-1 [11] and JPEG [9] | |
|---|---|---|---|---|---|---|---|
| **Location** | **Level** | **Location** | **Level** | **Location** | **Level** | **Location** | **Level** |
| 31 | 0.2291 | 31 | 0.2291 | 31 | 0.2291 | 31 | 0.2291 |
| 60 | 1.0000 | 60 | 1.0000 | 60 | 1.0000 | 60 | 1.0000 |
| 82 | 0.2291 | 82 | 0.2291 | 82 | 0.2291 | 82 | 0.2291 |
| 97 | 0.1507 | 97 | 0.1507 | 97 | 0.1507 | 97 | 0.1507 |
| 112 | 0.1275 | 112 | 0.1275 | 112 | 0.1275 | 112 | 0.1275 |
| 129 | 0.1274 | 129 | 0.1274 | 129 | 0.1274 | 129 | 0.1274 |
| 152 | 0.1509 | 152 | 0.1509 | 152 | 0.1509 | 152 | 0.1509 |
| 208 | 0.1509 | 208 | 0.1509 | 208 | 0.1509 | 208 | 0.1509 |
| 231 | 0.1274 | 231 | 0.1274 | 231 | 0.1274 | 231 | 0.1274 |
| 248 | 0.1275 | 248 | 0.1275 | 248 | 0.1275 | 248 | 0.1275 |
| 263 | 0.1507 | 263 | 0.1507 | 263 | 0.1507 | 263 | 0.1507 |
| 278 | 0.2291 | 278 | 0.2291 | 278 | 0.2291 | 278 | 0.2291 |
| 300 | 1.0000 | 300 | 1.0000 | 300 | 1.0000 | 300 | 1.0000 |
| 329 | 0.2291 | 329 | 0.2291 | 329 | 0.2291 | 329 | 0.2291 |

It is evident that all the columns and rows of the table have same values, which means the performance deviation is quite insignificant. The minimum peak for the side lobes is 0.1274, which is identical for all the considered transforms. The maximum level is 0.2291 for the side lobes here. Considering the normalized unity peak for the main beam, the maximum level of side lobe is 22 percent of the main lobe, which can be a concern in many applications.

As far as the number of side lobes observed is concerned, it is the same for all the benchmark DCT transforms considered here for multibeamforming.

## 3.3 Error Estimation

A measure of normalized deviation of all the Integer DCTs used here from exact DCT has been obtained and plotted in Fig. 3.3 and 3.4 using [18]:

$$\left|\hat{Y}(c)-\hat{Y}(p)\right|^2 \tag{3.2}$$

Here $c$ is for exact DCT and $p$ for approximations.

As it can be seen from the plots, $m$=0 and $m$=4 produce the minimum error which is negligible. For all the other indices, the deviation for different approximations varies from zero to 0.006, which makes it difficult to prefer one approximation to another. However, a closer look reveals that the maximum error, though apparently insignificant, can be observed in the case of AVC/H.264 with VC-1 following. HEVC is the next one that shows prominent deviation from FP DCT. Apparently, JPEG shows the best performance as can be seen from Fig. 3.3 and 3.4.

The error spikes, however, are not spread over the whole viewing region. They appear for particular angles only. Also for different indices, the performance is different. The overall performance can be evaluated considering the maximum normalized peak observed for the side lobes.

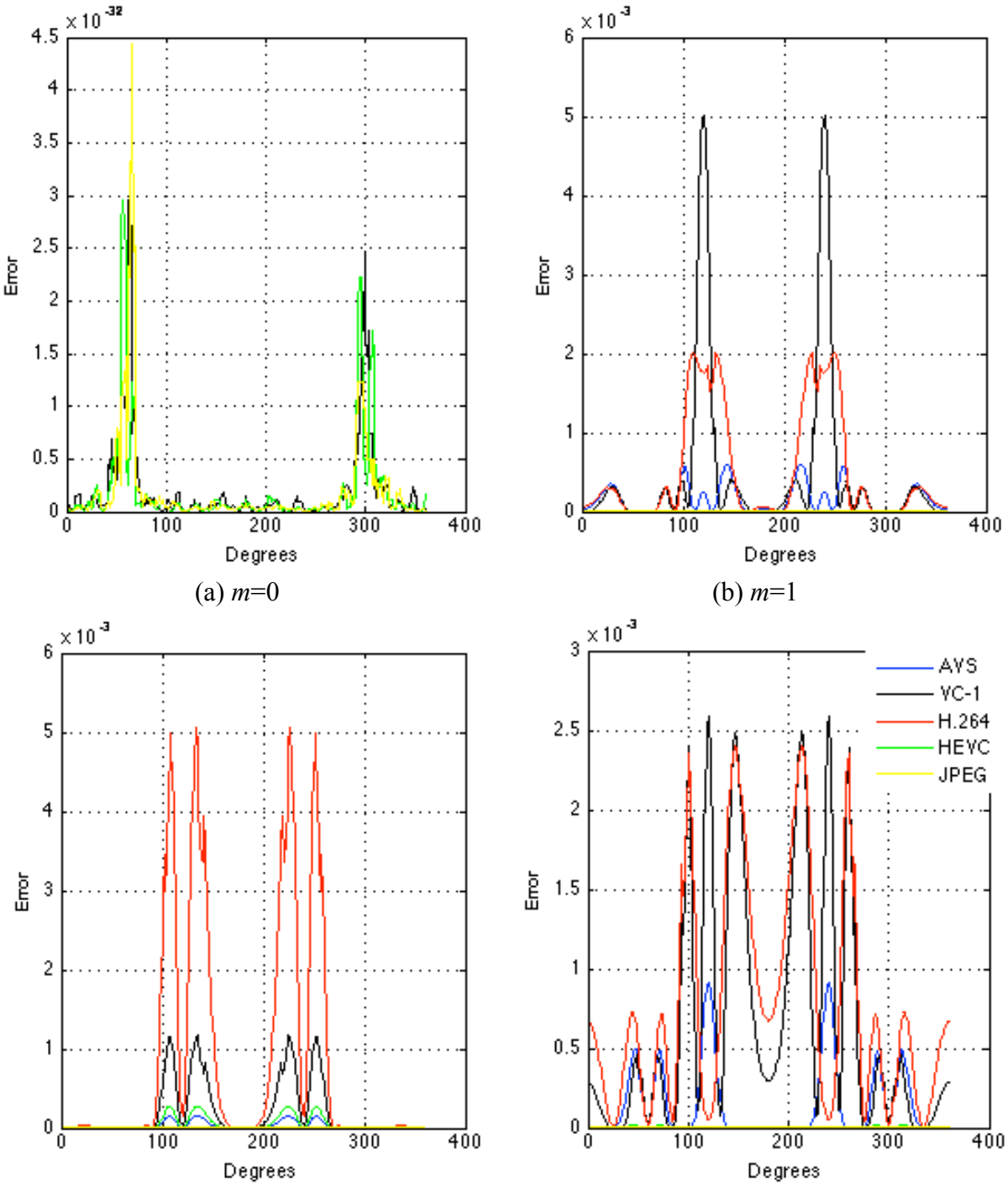
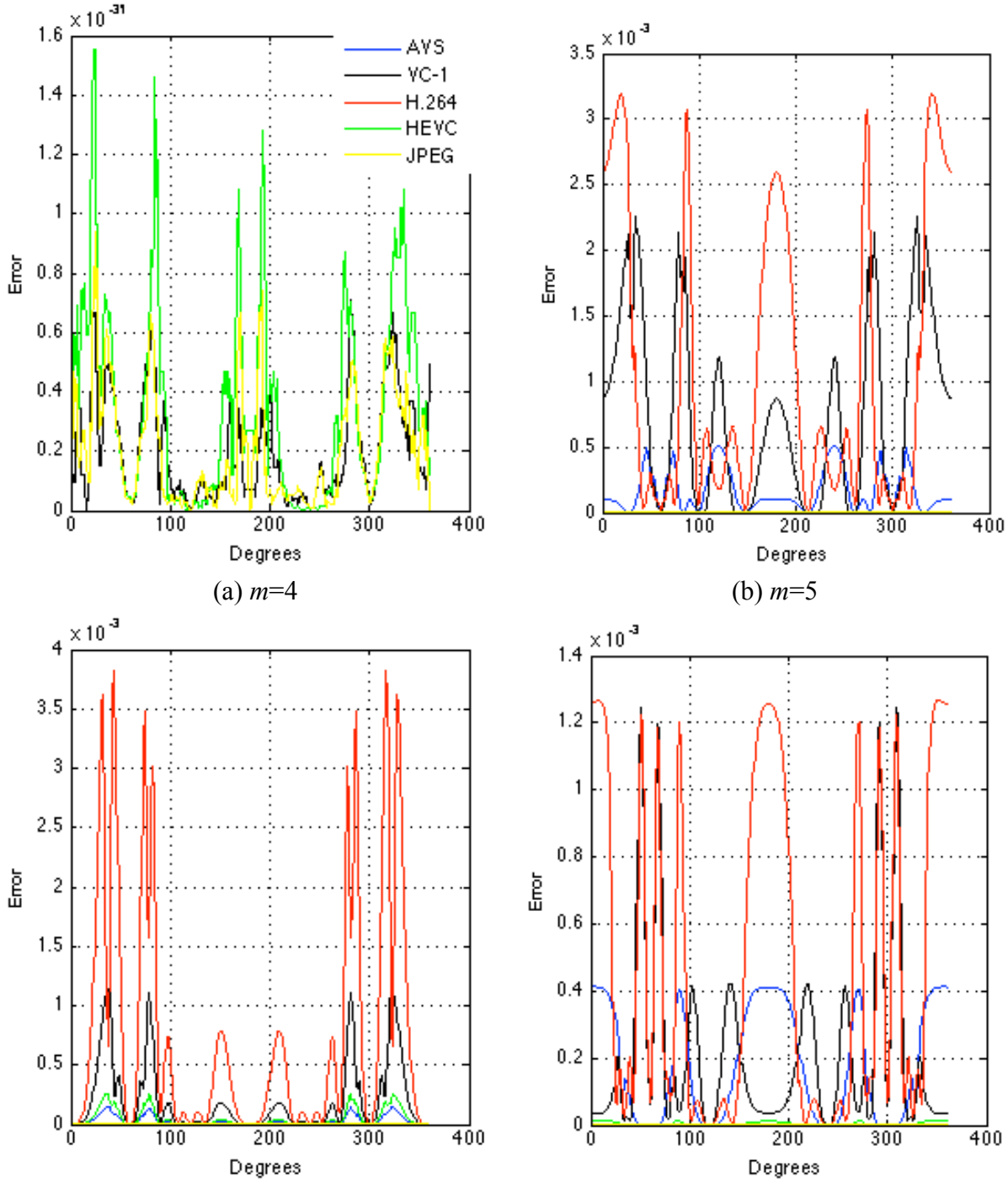



Fig. 3.3. Normalized error estimation for Int-DCT used in AVS, AVC, VC-1, JPEG and HEVC




(a) *m*=4 (b) *m*=5

(c) *m*=6 (d) *m*=7

Fig. 3.4. Normalized error estimation for Int-DCT used in AVS, AVC, VC-1, JPEG and HEVC

As mentioned earlier, the maximum error which is 0.006 appears for $m$=2. But this spike appears for a small region and specific angles only. This area is smaller for some other indices which contain lower peaks than $m$=2. If the area over which the deviation is spread is considered, the second, fourth, sixth and eighth rows show worse performance.

Although all the results stand for great performance for all the standards as the error and deviation is quite insignificant, this can be a deciding factor where even this much deviation can be misleading and cause unwanted signal from unwanted directions. Moreover, as these are the normalized results only, this seemingly negligible deviation would be considerable in practical situations where many other environmental factors and interferences would deteriorate the performance of the integer DCT approximations considered here.

## 3.4 Summary

Benchmark integer DCT algorithms standardized in the field of video and image compression have been considered here as the transformation matrices in a multibeamformer. Compared with classical DCT, the normalized outputs appear to be within a range of error, which is apparently negligible. Also the performance of all the transforms is quite similar to each other, which makes it difficult to prefer one over the other.

# CHAPTER 4

## PROPOSED 8-POINT AIQ DCT

Considering the requirement of an error free and efficient algorithm in multibeamforming applications, we propose the 8-pont Algebraic Integer Quantization implementation of DCT. Accuracy of data and consequently the analytical result based on that data is highly sensitive to the performance of the systems elements and implemented algorithms. An erroneous algorithm incrementally accumulates the error throughout the system. Again, implementation cost of an algorithm has also to be taken under consideration. Considering both accuracy and cost, an optimal algorithm is traditionally finalized which can be implemented with a feasible cost keeping the error margin within an acceptable range as required by the specific application. In image compression and video coding, 8-point AIQ is already a known algorithm due to its hardware efficiency in implementing classical DCT [13-15]. This efficiency accompanied with error free results would make AIQ a preferred algorithm in multibeamforming.

### 4.1 DCT Coefficients

The 8-point 1D DCT matrix $C_{8\times8}$ is shown below where,

$$c_k = \cos\frac{k\pi}{16} \tag{4.1}$$

where $k$=1,2,……,7

Instead of direct implementation of all the cosine angles, multidimensional algebraic integers are considered here which can be used to obtain all the cosine angles required for DCT implementation.

$$C_{8\times8} = \frac{1}{2}\cdot\begin{bmatrix} c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4 \\ c_1 & c_3 & c_5 & c_7 & -c_7 & -c_5 & -c_3 & -c_1 \\ c_2 & c_6 & -c_6 & -c_2 & -c_2 & -c_6 & c_6 & c_2 \\ c_3 & -c_7 & -c_1 & -c_5 & c_5 & c_1 & c_7 & -c_3 \\ c_4 & -c_4 & -c_4 & c_4 & c_4 & -c_4 & -c_4 & c_4 \\ c_5 & -c_1 & c_7 & c_3 & -c_3 & -c_7 & c_1 & -c_5 \\ c_6 & -c_2 & c_2 & -c_6 & -c_6 & c_2 & -c_2 & c_6 \\ c_7 & -c_5 & c_3 & -c_1 & c_1 & -c_3 & c_5 & -c_7 \end{bmatrix}$$

We consider here,

$$z_1 = 2\times c_1$$

$$\Rightarrow z_1 = 2\times\cos\frac{\pi}{16}$$

$$\Rightarrow z_1 = 2\times\frac{\sqrt{2+\sqrt{2+\sqrt{2}}}}{2}$$

$$\Rightarrow z_1 = \sqrt{2+\sqrt{2+\sqrt{2}}}$$

$$z_1 = 1.11110110 = 1.000\bar{1}10\bar{1}0 = 2 - 2^{-4} + 2^{-5} - 2^{-7} \qquad 4.2$$

and

$$z_2 = 2\times c_4$$

$$\Rightarrow z_2 = 2\times\cos\frac{4\pi}{16}$$

$$\Rightarrow z_2 = 2\times\frac{\sqrt{2}}{2}$$

$$\Rightarrow z_2 = \sqrt{2}$$

$$z_2 = 1.01101010 = 1 + 2^{-2} + 2^{-3} + 2^{-5} + 2^{-7} \qquad 4.3$$

This leads to expressions for all coefficients used in an 8-point DCT in terms of these two parameters only ($z_1$ and $z_2$). The expressions are presented in Table 4.1. All the cosine angles can be realized using these two parameters. The implementation cost, therefore, is greatly reduced as

re-usage of these two parameters can implement all the cosine angles. Also, they do not need any multipliers for their implementation. The final reconstruction stage is also error free.

The final reconstruction would be of the following symmetrical form:

$$y_{2n+1} = z_1[a_{2n+1} + z_1^2(b_{2n+1} + z_2 c_{2n+1}) + z_2 d_{2n+1}] \text{ and} \quad 4.4$$

$$y_{2n} = z_1^2(a_{2n} + z_2 b_{2n}) - 2z_2 c_{2n} + d_{2n} \quad 4.5$$

**Table 4.1 AIQ Representation of 8-point Classical DCT Coefficients**

| **Basis coefficients** | $a_{ij}$ | $f(z_1,z_2)$ |
|---|---|---|
| $2cos(\frac{0\pi}{16})$ | 2 0 0 0<br>0 0 0 0 | 2 |
| $2cos(\frac{1\pi}{16})$ | 0 1 0 0<br>0 0 0 0 | $z_1$ |
| $2cos(\frac{2\pi}{16})$ | −2 0 1 0<br>0 0 0 0 | $-2 + z_1^2$ |
| $2cos(\frac{3\pi}{16})$ | 0 −3 0 1<br>0 0 0 0 | $-3z_1 + z_1^3$ |
| $2cos(\frac{4\pi}{16})$ | 0 0 0 0<br>1 0 0 0 | $z_2$ |
| $2cos(\frac{5\pi}{16})$ | 0 3 0 −1<br>0 1 0 0 | $3z_1 - z_1^3 + z_1 z_2$ |
| $2cos(\frac{6\pi}{16})$ | 2 0 −1 0<br>−2 0 1 0 | $2 - z_1^2 - 2z_2 + z_1^2 z_2$ |
| $2cos(\frac{7\pi}{16})$ | 0 −1 0 0<br>0 −3 0 1 | $-z_1 - 3z_1 z_2 + z_1^3 z_2$ |

## 4.2 Matrix Decompositions

Decomposing the principal matrix into multiple matrices has been a proven way to reduce the number of operations for the matrix implementation. Here, the 8-point 1D DCT

matrix is decomposed into several matrices before applying the AIQ implementation, which would offer a great advantage in terms of hardware implementation cost.

The 8-point DCT matrix can be decomposed as:

$$C_{8\times 8} = p2 \times c2 \times p1 \tag{4.6}$$

Here,

$$c2 = \begin{bmatrix} c_4 & c_4 & c_4 & c_4 & 0 & 0 & 0 & 0 \\ c_2 & c_6 & -c_6 & -c_2 & 0 & 0 & 0 & 0 \\ c_4 & -c_4 & -c_4 & c_4 & 0 & 0 & 0 & 0 \\ c_6 & -c_2 & c_2 & -c_6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -c_7 & -c_5 & -c_3 & -c_1 \\ 0 & 0 & 0 & 0 & c_5 & c_1 & c_7 & -c_3 \\ 0 & 0 & 0 & 0 & -c_3 & -c_7 & c_1 & -c_5 \\ 0 & 0 & 0 & 0 & c_1 & -c_3 & c_5 & -c_7 \end{bmatrix}$$

$$p1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad p2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

*p1* is a matrix requiring eight additions for the eight rows only. *p2* is a permutation matrix which does not need any arithmetic operation for the implementation. It needs rewiring only for the implementation.

*c2* can be represented with two 4×4 matrices and implemented with a direct sum operation.

$$c2 = c21 \oplus c22 \tag{4.7}$$

Here, *c21* and *c22* can be presented as the below matrices. This further provides another decomposable matrix c21 to reduce the operations more.

$$c21 = \begin{bmatrix} c_4 & c_4 & c_4 & c_4 \\ c_2 & c_6 & -c_6 & -c_2 \\ c_4 & -c_4 & -c_4 & c_4 \\ c_6 & -c_2 & c_2 & -c_6 \end{bmatrix}, c22 = \begin{bmatrix} -c_7 & -c_5 & -c_3 & -c_1 \\ c_5 & c_1 & c_7 & -c_3 \\ -c_3 & -c_7 & c_1 & -c_5 \\ c_1 & -c_3 & c_5 & -c_7 \end{bmatrix}$$

*c21* can further be obtained with:

$$c21 = c211.c212 \tag{4.8}$$

Here,

$$c211 = \begin{bmatrix} c_4 & c_4 & 0 & 0 \\ 0 & 0 & -c_6 & -c_2 \\ c_4 & -c_4 & 0 & 0 \\ 0 & 0 & c_2 & -c_6 \end{bmatrix}, c212 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}$$

*c212* needs additions only for the implementation as it contains only 1's and 0's. This realization also reduces the number of non zero coefficients required to be implemented. This reduces hardware cost as expected.

The final equation would be:

$$C_{8\times8} = p2 \times \{(c211 \times c212) \oplus c22\} \times p1 \tag{4.9}$$

This can be realized with adders and shifters only.

## 4.3 Performance Analysis

Fig. 4.1 and Fig. 4.2 present the polar plots of the array factor with different transformation matrices. As transformation matrix, several algorithms such as, the exact DCT, BAS approximation [18], CB approximation [19], PMCB approximation [7] and AIQ implementation have been used for comparison. The exact DCT and AIQ have been plotted with

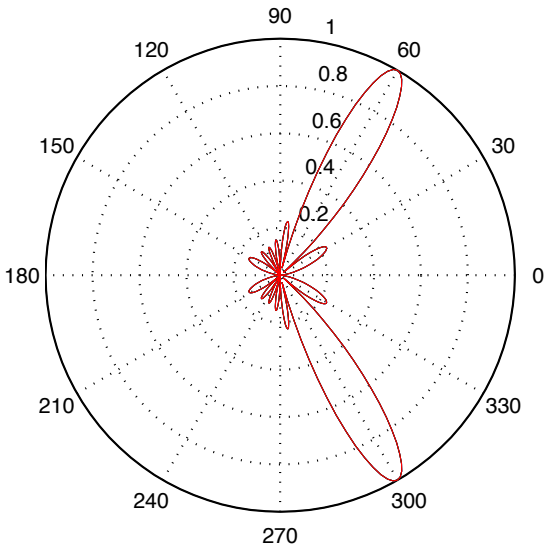
solid red and black respectively. It can be seen that the width of the side lobes and main lobes are the same for each other and narrower compared to the other approximations as expected. However, beams for $m$=0 and $m$=4 have similar shape and direction due to the same coefficients in all the approximations considered in the case of these two particular indices.

For all the other indices, deviation in beam shape is observed with all the transform matrices considered except AIQ. Apart from the main lobes, a number of side lobes are also observed with all the transforms and for all the indices, which make a difference in considering the preferred algorithm.

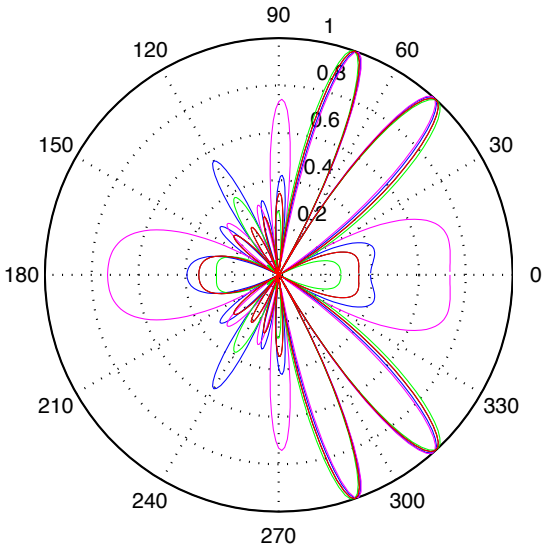
The same can also be realized from Table 4.1. The values of the peaks at different angles have not been normalized in this table. The minimum peak for the side lobes is observed in case of FP DCT and AIQ DCT. Though the number of lobes appears to be less in BAS and CB approximation compared to FP and AIQ DCT, it results in side lobes with larger peaks, which makes these approximations more vulnerable and unreliable. PMCB produces the same number of lobes but with different peaks.

A measure of deviation of the other approximations used here from exact DCT has been obtained and plotted in Fig. 4.3 and Fig. 4.4 using $\left|\hat{Y}(c)-\hat{Y}(p)\right|^2$. Here $c$ is for exact DCT and $p$ is for approximations. No deviation is observed in case of AIQ DCT, which offers an error free implementation of classical DCT in multibeamforming as expected.

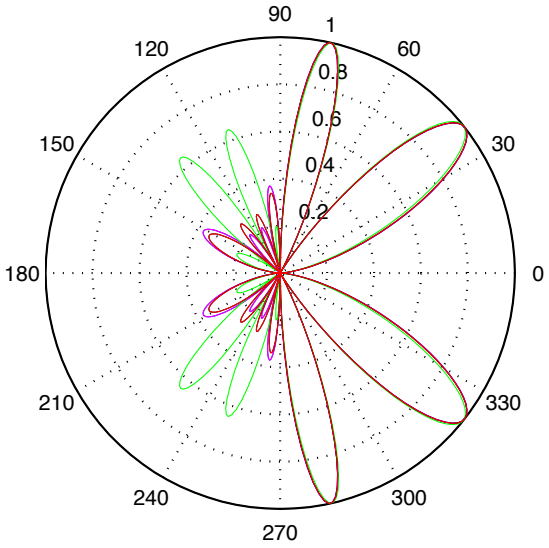
The deviations shown in this plots are the normalized measures of the error. Apart from AIQ, all the other approximations deviate from classical DCT as can be seen from the plots. These deviations, which appear to be insignificant in some cases due their normalization, would affect real time performance to a great extent.



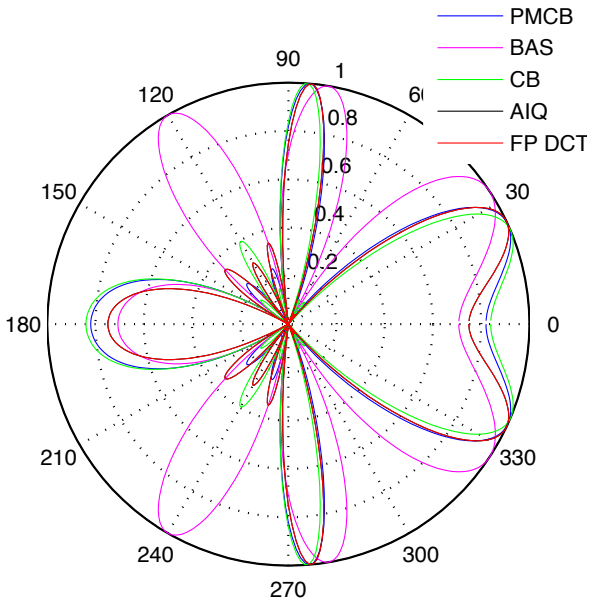

(a) *m*=0




(b) *m*=1

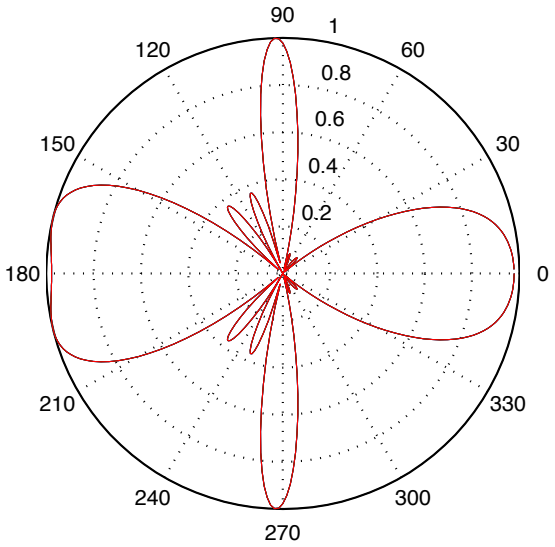



(c) *m*=2

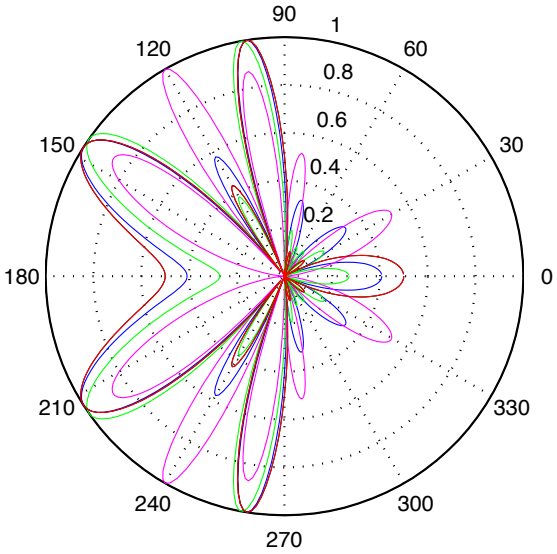



(d) *m*=3

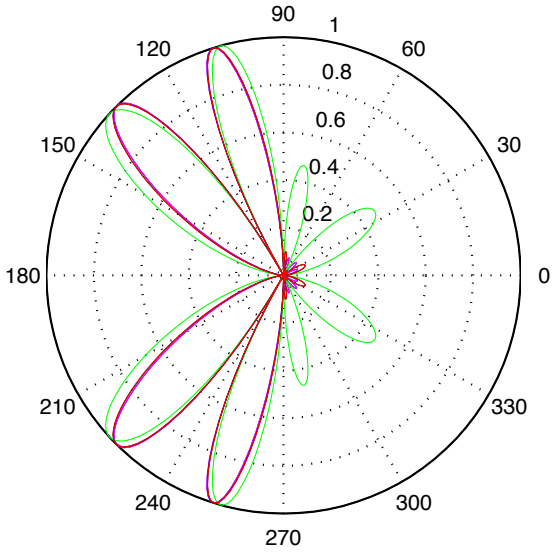
Fig. 4.1. Polar Plot of PMCB, BAS, CB, AIQ and FP DCT with steering angle 60 degrees for *m*=0,1,2 and 3



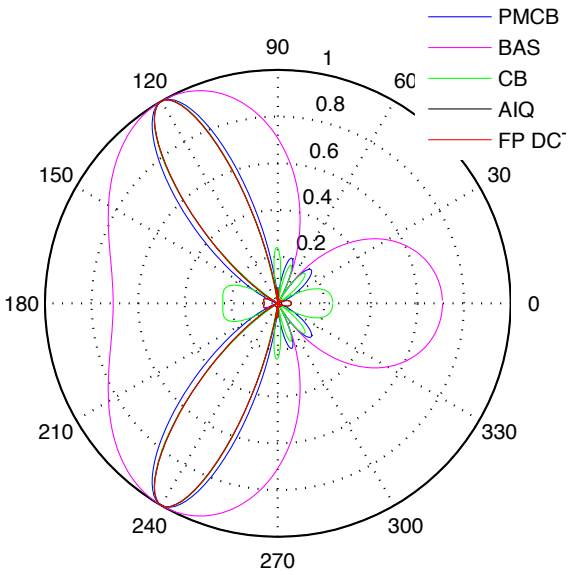

(a) *m*=4




(b) *m*=5




(c) *m*=6

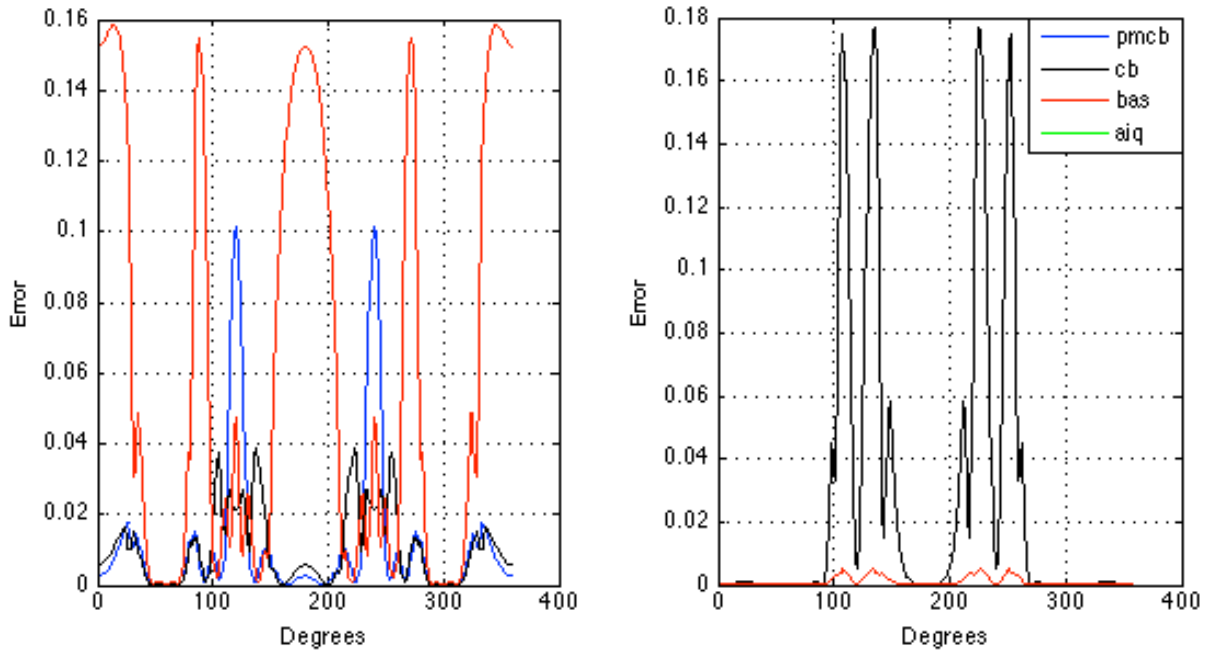



(d) *m*=7

Fig. 4.2. Polar Plot of PMCB, BAS, CB, AIQ and FP DCT with steering angle 60 degrees for *m*=4,5,6 and 7

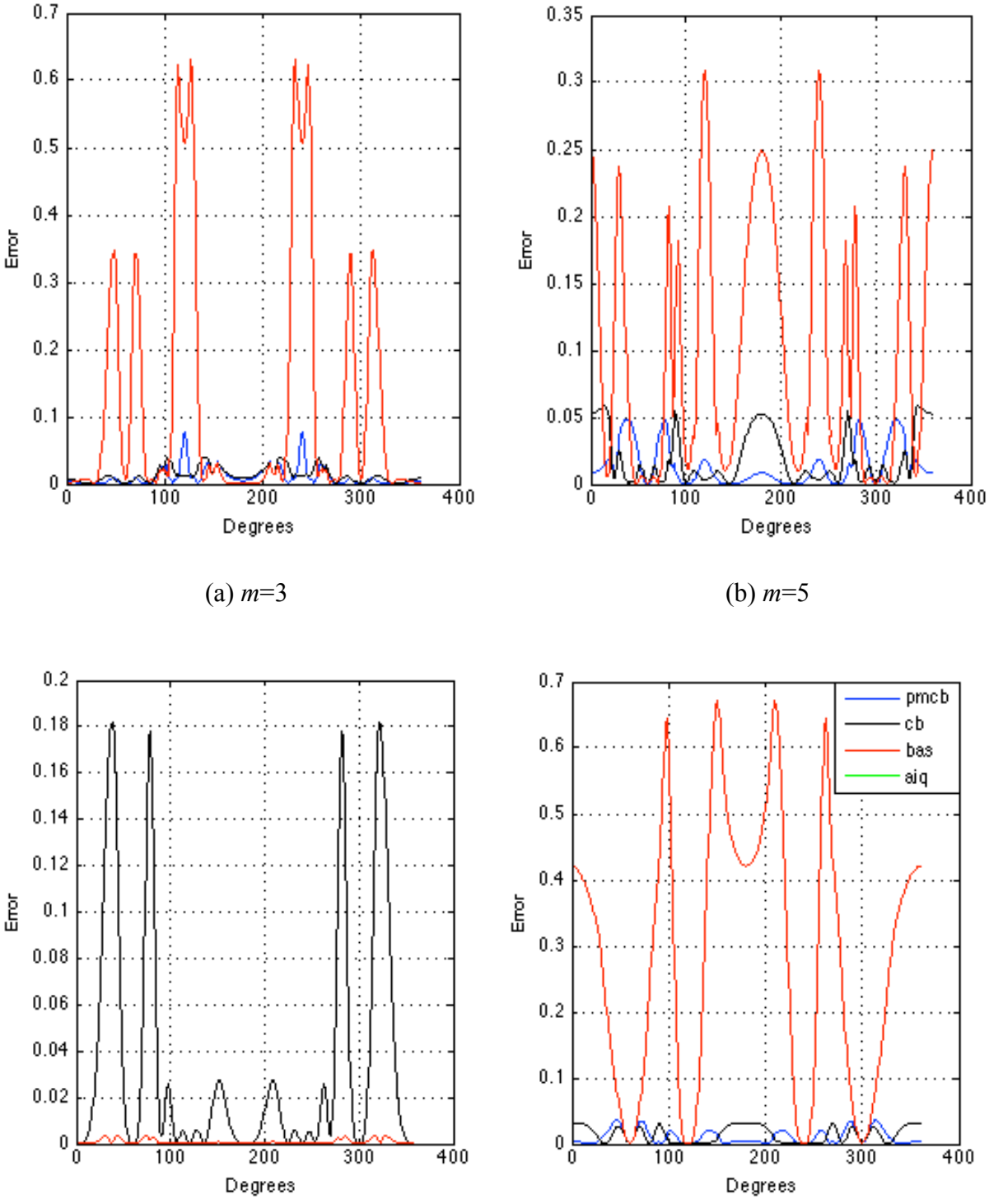
**Table 4.2 Locations (Degrees) and levels (not normalized) of main and side lobes in all transforms**

| FP DCT and AIQ DCT | | PMCB [7] | | BAS [18] | | CB [19] | |
|---|---|---|---|---|---|---|---|
| **Location** | **Level** | **Location** | **Level** | **Location** | **Level** | **Location** | **Level** |
| 9 | 0.7590 | 14 | 0.8769 | 10 | 1.4327 | 91 | 0.5795 |
| 89 | 0.7572 | 88 | 0.8757 | 89 | 1.4328 | 104 | 0.1359 |
| 104 | 0.5560 | 103 | 0.6692 | 107 | 0.5936 | 120 | 0.8165 |
| 120 | 0.5098 | 120 | 1.1547 | 136 | 0.5940 | 140 | 0.1381 |
| 139 | 0.5565 | 141 | 0.6697 | 180 | 1.4142 | 172 | 0.5819 |
| 180 | 0.7507 | 180 | 0.8165 | 224 | 0.5940 | 188 | 0.5819 |
| 221 | 0.5565 | 219 | 0.6697 | 253 | 0.5936 | 220 | 0.1381 |
| 240 | 0.5098 | 240 | 1.1547 | 271 | 1.4328 | 240 | 0.8165 |
| 256 | 0.5560 | 257 | 0.6692 | 350 | 1.4327 | 256 | 0.1359 |
| 271 | 0.7572 | 272 | 0.8757 | | | 269 | 0.5795 |
| 351 | 0.7590 | 346 | 0.8769 | | | | |




(a) $m$=1 (b) $m$=2

Fig. 4.3. Normalized error Plot of PMCB, CB, BAS and AIQ compared to FP DCT for $m$=1 and 2




(a) *m*=3

(b) *m*=5

(c) *m*=6

(d)*m*=7

Fig. 4.4. Normalized error Plot of PMCB, CB, BAS and AIQ compared to FP DCT for *m*=3,5,6 and 7

BAS, presented with red line on the plots, shows the maximum error for the second, fourth, sixth and eighth rows. Maximum deviation for third and fifth rows is found with CB. PMCB shows noticeable deviation for $m$=2,4,6 and 8 with the maximum being at $m$=2.

As it can be seen from the plots of Fig. 4.3 and Fig. 4.4, the deviation for different approximations at different indices vary from zero to 1.4, which makes it difficult to prefer one approximation to another. Furthermore, this deviation would have unwanted side lobe widths in the case of multi-beamforming applications where precision of the direction of the received beam and interference are of severe importance.

## 4.4 Hardware Flow Graph and Comparison

The hardware flow graph of the DCT kernel is shown in Fig. 4.5. The blocks with $c1$, $c2$,…., $c7$ which are the algebraic integer coefficients for 8-point DCT present only additions and shift operations required for the implementation. Therefore an error-free DCT mapping without any multiplication can be obtained. *a[0], a[1], a[2], a[3],.....,a[8]* are the inputs and *y[0], y[1], y[2], y[3],....., y[8]* are the outputs of the system.

A cost comparison is provided in Table 4.3. The proposed method is computationally efficient compared with others. In BAS approximations, ±1, ±½ and 0 are the only coefficients used in the matrix in place of the floating-point coefficients in classical DCT, which offers a great advantage in implementation. A diagonal matrix is used in the quantization stage to obtain the DCT coefficients. While this process reduces hardware in one stage, it sacrifices hardware efficiency in the other stage.

The same technique is adopted in CB and PMCB approximations. In CB, only ±1 and 0 are used in place of the DCT coefficients with a diagonal matrix added to the quantization stage.

Fewer coefficient options in CB, however, make the algorithm more error prone. ±1, ±2 and 0 are the chosen DCT coefficients in PMCB algorithm. The approximation methods used in [7], [18] and [19] may cost fewer operations in the transformation stage, but require additional computation steps in the next level (i.e., quantization followed by transform). Moreover, these algorithms are observed as being erroneous in multibeamforming application.

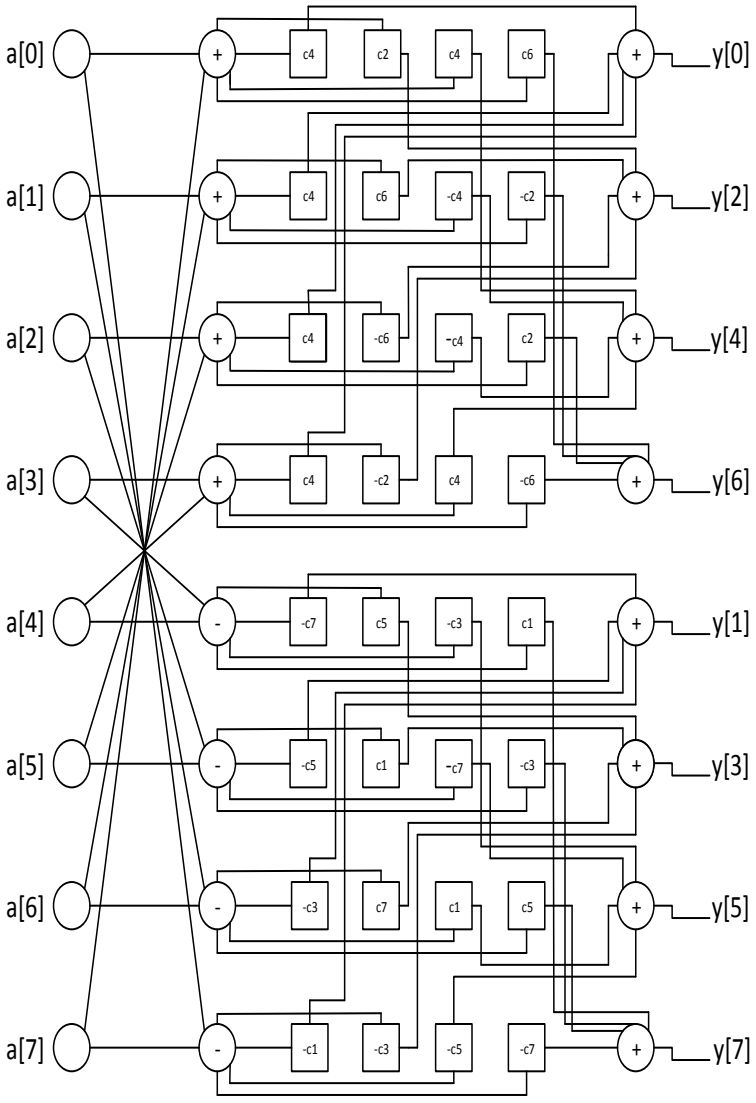



Figure. 4.5. Hardware flow graph for 8-point DCT

Multidimensional AIQ has been presented in [13] along with 1D AIQ for 12-bit and 10-bit consecutively. However, the number of operations required for the proposed algorithm in this work is less as the decomposition of the DCT matrix into different matrices reduces the number of operations.

**Table 4.3 Comparison of hardware cost**

| Methods | Adder | Multiplier | Shift | Error Free? | Operations left for next step? |
|---|---|---|---|---|---|
| DCT [20] | 56 | 64 | 0 | Yes | No |
| DFT [21] | 58 | 6 | 0 | Yes | No |
| Arai [22] | 29 | 5 | 0 | No | No |
| BAS [18] | 18 | 0 | 2 | No | Yes |
| CB [19] | 22 | 0 | 0 | No | Yes |
| PMCB [7] | 24 | 0 | 6 | No | Yes |
| AIQ [13] | 132 | 0 | - | Yes | No |
| Ours | 47 | 0 | 31 | Yes | No |

## 4.5 Summary

As expected, the 8-point AIQ provides us with an algorithm that can be implemented without any multipliers, resulting in less hardware complexity compared to other FP DCT algorithms. It also gives an error free output in multibeamforming which makes it reliable compared to other approximations proposed.

# CHAPTER 5

## PROPOSED 16-POINT ALGORITHM

As mentioned earlier, an 8-point DCT would provide us with beams in eight directions. Increasing the size of the DCT matrix can increase the number of beams. A 16-point DCT would provide 16 beams in 16 different directions.

Let us consider a 16-element uniform linear aperture array arranged along the z-axis. An array can be planar or circular as well, being uniform or non-uniform in terms of distance in between them. In an ULA, each element is separated by half wavelength distance from each other. Let us assume a uniform plane wave arriving at the array at an angle $\theta$ with respect to the $z$-axis. The array pattern would be:

$$re = \sum_{n=1}^{16} e^{-nj\pi\cos\theta} \tag{5.1}$$

If we take all the other coordinates to be zero except that of $z$-axis and consider the weighting such as to direct the beam to $\theta_e$ direction with shifting phases of each element, the array factor, *AF,* would be:

$$AF = \sum_{N=1}^{16} e^{-Nj\pi(\cos\theta_e - \cos\theta)} \tag{5.2}$$

Here, $\theta$ is from 0 to 2π, $\theta_e$ is the expected beam direction and $N$ is the number of sensors in the array.

This array factor defines the response of an array assuming all the elements have identical radiation patterns. Taking this as the transfer function and transforming with Discrete Cosine

Transform, this would give us a multiple beamformer output at specific directions corresponding to the expected direction.

The equation would be:

$$\hat{Y} = x.\hat{V} \tag{5.3}$$

where $\hat{Y}$ are $N$ beams for input snapshot for $N$ element ULA and $x$ is a transformation matrix. This can be elaborated as:

$$\hat{Y} = \sum_{N=1}^{16} x(m,n).e^{-Nj\pi(\cos\theta_e - \cos\theta)} \tag{5.4}$$

Here *x(m,n)* would be the preferred transformation matrix with *m* and *n* being the row and column of the matrix.

## 5.1 One Dimensional AIQ Mapping

The 16-point DCT matrix is shown below where,

$$c_k = \cos\frac{k\pi}{32} \tag{5.5}$$

1D AIQ for 16-point DCT can be implemented by considering,

$$z = 2\cos\frac{\pi}{32}$$

As shown earlier for 8-point DCT, all the other coefficients for 16-point DCT can be obtained and implemented using *z*. The general equation is as follow,

$$f(z) = \sum a_i z^i \tag{5.6}$$

The integer coefficients for all the cosine angles are presented in table 5.1. The maximum degree for $z$ is 15 as can be seen from the table.

$$C(16,16)=\begin{bmatrix}
c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 \\
c_1 & c_3 & c_5 & c_7 & c_9 & c_{11} & c_{13} & c_{15} & -c_{15} & -c_{13} & -c_{11} & -c_9 & -c_7 & -c_5 & -c_3 & -c_1 \\
c_2 & c_6 & c_{10} & c_{14} & -c_{14} & -c_{10} & -c_6 & -c_2 & -c_2 & -c_6 & -c_{10} & -c_{14} & c_{14} & c_{10} & c_6 & c_2 \\
c_3 & c_9 & c_{15} & -c_{11} & -c_5 & -c_1 & -c_7 & -c_{13} & c_{13} & c_7 & c_1 & c_5 & c_{11} & -c_{15} & -c_9 & -c_3 \\
c_4 & c_{12} & -c_{12} & -c_4 & -c_4 & -c_{12} & c_{12} & c_4 & c_4 & c_{12} & -c_{12} & -c_4 & -c_4 & -c_{12} & c_{12} & c_4 \\
c_5 & c_{15} & -c_7 & -c_3 & -c_{13} & c_9 & c_1 & c_{11} & -c_{11} & -c_1 & -c_9 & c_{13} & c_3 & c_7 & -c_{15} & -c_5 \\
c_6 & -c_{14} & -c_2 & -c_{10} & c_{10} & c_2 & c_{14} & -c_6 & -c_6 & c_{14} & c_2 & c_{10} & -c_{10} & -c_2 & -c_{14} & c_6 \\
c_7 & -c_{11} & -c_3 & c_{15} & c_1 & c_{13} & -c_5 & -c_9 & c_9 & c_5 & -c_{13} & -c_1 & -c_{15} & c_3 & c_{11} & -c_7 \\
c_8 & -c_8 & -c_8 & c_8 & c_8 & -c_8 & -c_8 & c_8 & c_8 & -c_8 & -c_8 & c_8 & c_8 & -c_8 & -c_8 & c_8 \\
c_9 & -c_5 & -c_{13} & c_1 & -c_{15} & -c_3 & c_{11} & c_7 & -c_7 & -c_{11} & c_3 & c_{15} & -c_1 & c_{13} & c_5 & -c_9 \\
c_{10} & -c_2 & c_{14} & c_6 & -c_6 & -c_{14} & c_2 & -c_{10} & -c_{10} & c_2 & -c_{14} & -c_6 & c_6 & c_{14} & -c_2 & c_{10} \\
c_{11} & -c_1 & c_9 & c_{13} & -c_3 & c_7 & c_{15} & -c_5 & c_5 & -c_{15} & -c_7 & c_3 & -c_{13} & -c_9 & c_1 & -c_{11} \\
c_{12} & -c_4 & c_4 & -c_{12} & -c_{12} & c_4 & -c_4 & c_{12} & c_{12} & -c_{14} & c_4 & -c_{12} & -c_{12} & c_4 & -c_4 & c_{12} \\
c_{13} & -c_7 & c_1 & -c_5 & c_{11} & c_{15} & -c_9 & c_3 & -c_3 & c_9 & -c_{15} & -c_{11} & c_5 & -c_1 & c_7 & -c_{13} \\
c_{14} & -c_{10} & c_6 & -c_2 & c_2 & -c_6 & c_{10} & -c_{14} & -c_{14} & c_{10} & -c_6 & c_2 & -c_2 & c_6 & -c_{10} & c_{14} \\
c_{15} & -c_{13} & c_{11} & -c_9 & c_7 & -c_5 & c_3 & -c_1 & c_1 & -c_3 & c_5 & -c_7 & c_9 & -c_{11} & c_{13} & -c_{15}
\end{bmatrix}$$

From Table 5.1, the equation for $C_{10}$ would be,

$$c_{10} = -2 \times z^0 + 25 \times z^2 - 50 \times z^4 + 35 \times z^6 - 10 \times z^8 + 1 \times z^{10}$$

Realization of all the cosine angles of 16-point DCT is presented in Table 5.2. As can be seen and also mentioned earlier, the maximum degree for $z$ is 15. This can be further simplified using multidimensional AIQ instead of 1D. Degree of considered algebraic integer tends to reduce as the number of integers considered is increased. On the other hand, the hardware efficiency achieved using algebraic integers reduces with the increased number of integers considered. Therefore, considering an optimum number of algebraic integers for maximum hardware efficiency is critical.

In the following chapter, a multidimensional algebraic integer implementation of classical DCT is proposed and analyzed with three algebraic integers.

**Table 5.1 1D AIQ Representations of 16-point DCT**

| $a_i$ / $c_i$ | $a_0$ | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ | $a_6$ | $a_7$ | $a_8$ | $a_9$ | $a_{10}$ | $a_{11}$ | $a_{12}$ | $a_{13}$ | $a_{14}$ | $a_{15}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $c_0$ | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_1$ | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_2$ | -2 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_3$ | 0 | -3 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_4$ | 2 | 0 | -4 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_5$ | 0 | 5 | 0 | -5 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_6$ | -2 | 0 | 9 | 0 | -6 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_7$ | -7 | 0 | 0 | 14 | 0 | -7 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_8$ | 2 | 0 | -16 | 0 | 20 | 0 | -8 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_9$ | 0 | 9 | 0 | -30 | 0 | 27 | 0 | -9 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| $c_{10}$ | -2 | 0 | 25 | 0 | -50 | 0 | 35 | 0 | -10 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| $c_{11}$ | 0 | -11 | 0 | 55 | 0 | -77 | 0 | 44 | 0 | -11 | 0 | 1 | 0 | 0 | 0 | 0 |
| $c_{12}$ | 2 | 0 | -36 | 0 | 105 | 0 | -112 | 0 | 54 | 0 | -12 | 0 | 1 | 0 | 0 | 0 |
| $c_{13}$ | 0 | 13 | 0 | -91 | 0 | 182 | 0 | -156 | 0 | 65 | 0 | -13 | 0 | 1 | 0 | 0 |
| $c_{14}$ | -2 | 0 | 49 | 0 | -196 | 0 | 294 | 0 | -210 | 0 | 77 | 0 | -14 | 0 | 1 | 0 |
| $c_{15}$ | 0 | -15 | 0 | 140 | 0 | -378 | 0 | 450 | 0 | -275 | 0 | 90 | 0 | -15 | 0 | 1 |

**Table 5.2 1D AIQ Representations of 16-point DCT (in terms of *Z*)**

| Coefficients ($c_i$) | Expressions |
|---|---|
| $c_0$ | 2 |
| $c_1$ | z |
| $c_2$ | $z^2-2$ |
| $c_3$ | $z^3-3z$ |
| $c_4$ | $z^4-4z^2+2$ |
| $c_5$ | $z^5-5z^3+5z$ |
| $c_6$ | $z^6-6z^4+9z^2-2$ |
| $c_7$ | $z^7-7z^5+14z^3-7z$ |
| $c_8$ | $z^8-8z^6+20z^4-16z^2+2$ |
| $c_9$ | $z^9-9z^7+27z^5-30z^3+9z$ |
| $c_{10}$ | $z^{10}-10z^8+35z^6-50z^4+25z^2-2$ |
| $c_{11}$ | $z^{11}-11z^9+44z^7-77z^5+55z^3-11z$ |
| $c_{12}$ | $z^{12}-12z^{10}+54z^8-112z^6+105z^4-36z^2+2$ |
| $c_{13}$ | $z^{13}-13z^{11}+65z^9-156z^7+182z^5-91z^3+13z$ |
| $c_{14}$ | $z^{14}-14z^{12}+77z^{10}-210z^8+294z^6-196z^4+49z^2-2$ |
| $c_{15}$ | $z^{15}-15z^{13}+90z^{11}-275z^9+450z^7-378z^5+140z^3-15z$ |

## 5.2 Three Dimensional AIQ Mapping

An efficient algorithm for sixteen point DCT implementation with multidimensional algebraic integer is proposed here. We consider,

$$z_1 = 2\cos\frac{\pi}{32} = \sqrt{2+\sqrt{2+\sqrt{2+\sqrt{2}}}} \quad 5.7$$

$$z_2 = 2\cos\frac{2\pi}{32} = \sqrt{2+\sqrt{2+\sqrt{2}}} \quad 5.8$$

$$z_3 = 2\cos\frac{8\pi}{32} = \sqrt{2} \quad 5.9$$

This leads to expressions for all the other constants encountered in 8-point DCT in terms of these three coefficients only. All the other cosine angles in sixteen-point classical DCT can be presented as

$$f(z_1, z_2, z_3) = \sum a_{i,j,k} {z_1}^i {z_2}^j {z_3}^k \quad 5.10$$

Here,

$i$=0,1

$j$=0,1,2,3 and

$k$=0,1.

As an example, $2\cos(\frac{9\pi}{32})$ can be represented as

$$2\cos(\frac{9\pi}{32}) = -z_1 + 2z_1z_2 + z_1z_3 + z_1{z_2}^2 - z_1{z_2}^3$$

The coefficients are presented in Table 5.3. The maximum degree of the algebraic integers observed in this case is 3 which is a great improvement over 1D AIQ mapping where the degree reaches up to 15. The coefficients in 3D AIQ mapping are also much smaller compared to those of 1D AIQ. This promises an efficient hardware implementation of 3D AIQ.

**Table 5.3 3D AIQ Representations of 16-Point DCT Coefficients**

| Cosine Angles | $a_{ijk}$ | $f(z_1,z_2,z_3)$ |
|---|---|---|
| $2\cos(\frac{0\pi}{32})$ | 2 0 0 0 0 0 0 0<br>0 0 0 0 0 0 0 0 | $2$ |
| $2\cos(\frac{1\pi}{32})$ | 0 1 0 0 0 0 0 0<br>0 0 0 0 0 0 0 0 | $z_1$ |
| $2\cos(\frac{2\pi}{32})$ | 0 0 1 0 0 0 0 0<br>0 0 0 0 0 0 0 0 | $z_2$ |
| $2\cos(\frac{3\pi}{32})$ | 0 −1 0 0 1 0 0 0<br>0 0 0 0 0 0 0 0 | $-z_1 + z_1z_2$ |
| $2\cos(\frac{4\pi}{32})$ | −2 0 0 0 0 0 0 0<br>1 0 0 0 0 0 0 0 | $-2 + z_2^2$ |
| $2\cos(\frac{5\pi}{32})$ | 0 −1 0 0 −1 0 0 0<br>0 1 0 0 0 0 0 0 | $-z_1 + z_1z_2^2 - z_1z_2$ |
| $2\cos(\frac{6\pi}{32})$ | 0 0 −3 0 0 0 0 0<br>0 0 0 0 1 0 0 0 | $-3z_2 + z_2^3$ |
| $2\cos(\frac{7\pi}{32})$ | 0 1 0 0 −2 0 0 0<br>0 −1 0 0 0 1 0 0 | $-2z_1z_2 + z_1z_2^3 + z_1 - z_1z_2^2$ |
| $2\cos(\frac{8\pi}{32})$ | 0 0 0 1 0 0 0 0<br>0 0 0 0 0 0 0 0 | $z_3$ |
| $2\cos(\frac{9\pi}{32})$ | 0 −1 0 0 2 1 0 0<br>0 1 0 0 0 −1 0 0 | $z_1z_3 + 2z_1z_2 - z_1z_2^3 - z_1 + z_1z_2^2$ |
| $2\cos(\frac{10\pi}{32})$ | 0 0 3 0 0 0 1 0<br>0 0 0 0 −1 0 0 0 | $3z_2 - z_2^3 + z_2z_3$ |
| $2\cos(\frac{11\pi}{32})$ | 0 1 0 0 1 −1 0 1<br>0 −1 0 0 0 0 0 0 | $z_1z_2 + z_1z_2z_3 - z_1z_3 + z_1 - z_1z_2^2$ |
| $2\cos(\frac{12\pi}{32})$ | 2 0 0 −2 0 0 0 0<br>−1 0 1 0 0 0 0 0 | $2 - z_2^2 - 2z_3 + z_2^2z_3$ |
| $2\cos(\frac{13\pi}{32})$ | 0 1 0 0 −1 −1 0 −1<br>0 0 0 1 0 0 0 0 | $z_1 - z_1z_3 + z_1z_2^2z_3 - z_1z_2 - z_1z_2z_3$ |
| $2\cos(\frac{14\pi}{32})$ | 0 0 0 0 0 0 1 0<br>0 0 −1 0 0 0 −3 0 | $-z_2 - 3z_2z_3 + z_2^3z_3$ |
| $2\cos(\frac{15\pi}{32})$ | 0 0 0 −1 0 0 0 1<br>0 −1 0 0 0 1 0 −3 | $-2z_1z_2z_3 + z_1z_2^3z_3 - z_1 + z_1z_3 - z_1z_2^2z_3$ |

These coefficients, while transformed into binary numbers in the final reconstruction state can be represented applying booth's algorithm [35] as:

$$z_1 = 1.11111101 = 1.00000\bar{1}1\bar{1} = 2 - 2^{-6} + 2^{-7} - 2^{-8}$$

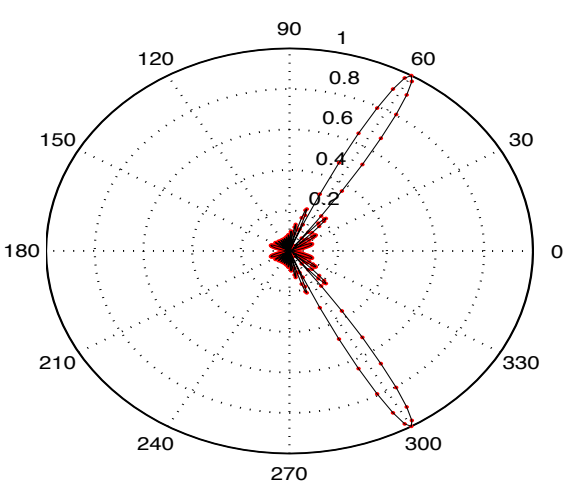
$$z_2 = 1.11110110 = 1.000\bar{1}10\bar{1}0 = 2 - 2^{-4} + 2^{-5} - 2^{-7}$$

$$z_3 = 1.01101010 = 1 + 2^{-2} + 2^{-3} + 2^{-5} + 2^{-7}$$

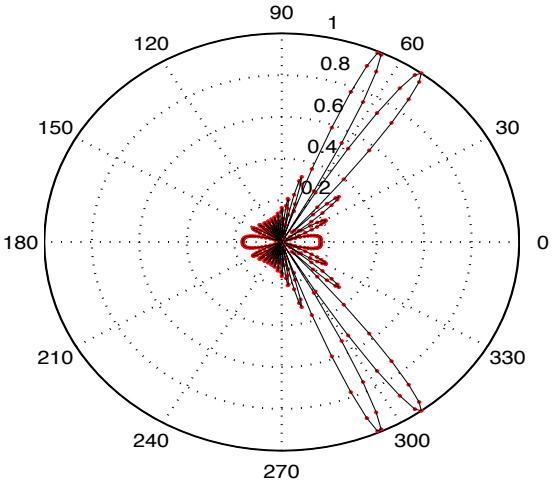
## 5.3 Performance Analysis in Multibeamforming

The proposed multiplier-less 16 point AIQ DCT offers us an accurate implementation of 16 point FP DCT. Furthermore, due to additions and shifts only throughout the operation, it has the least error after the final reconstruction stage. Consequently, AIQ DCT tops FP DCT in terms of error measurement. Evidently, 16 point DCT provides significant signal input/output in sixteen different directions as characterized by its property. 8 point DCT, on the other hand, would provide 8 receivable directions. Since the same area is distributed in 16 and 8 directions with 16 point and 8 point DCTs respectively, the 16 point DCT results in narrower beams compared to those of 8 point DCT. This would offer better Signal to Interference and Noise Ratio (SINR) and directionality.

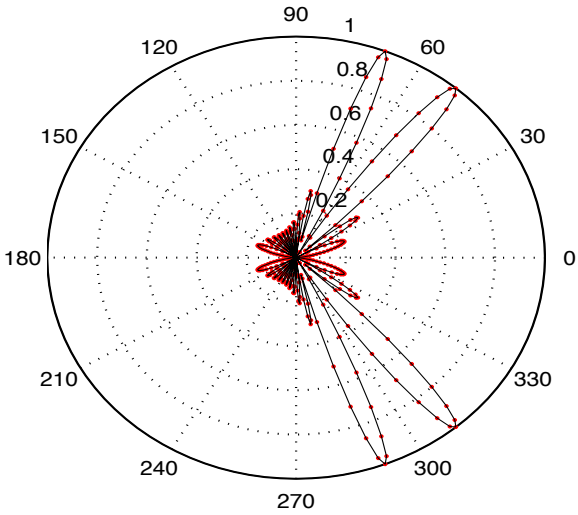
Fig. 5.3.1, Fig. 5.3.2, Fig. 5.3.3 and Fig. 5.3.4 present the polar plot of 16-point FP and AIQ DCT in multibeamforming. Main lobes in 16 different directions can be achieved as can be seen from the plots. The beam width is much narrower in case of 16-point DCT as the same view area is being covered with more number of beams compared to 8-point DCT. This would enable less noise pickups in this case. With AIQ, the beam directions and peaks for both the main lobes and side lobes are the same.



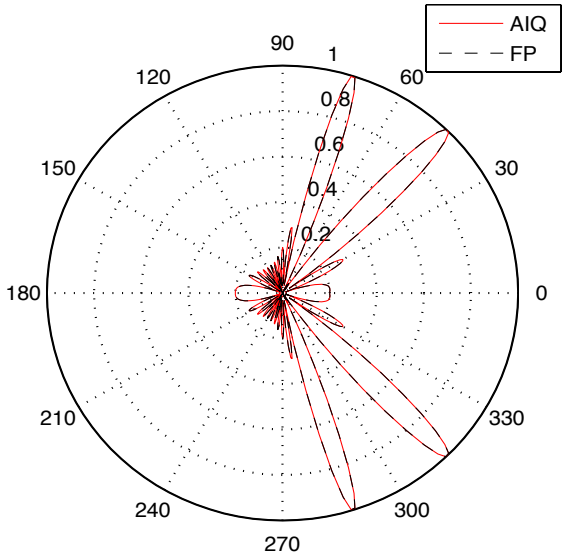

(a) $m$=0




(b) $m$=1

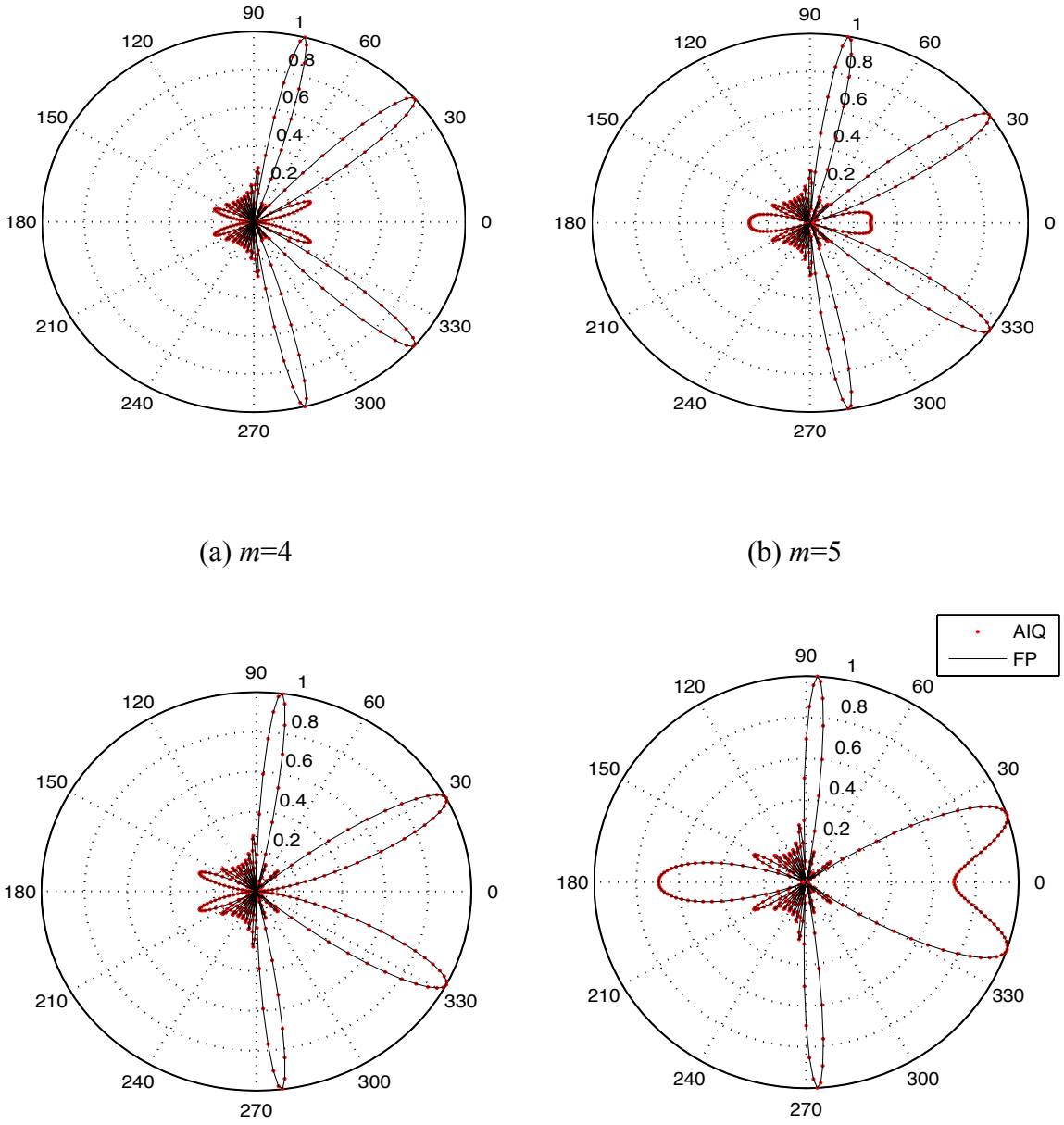



(c) $m$=2




(d) $m$=3

Fig. 5.1. Polar Plot of 16-point FP and AIQ DCT for $m$=0,1,2 and 3



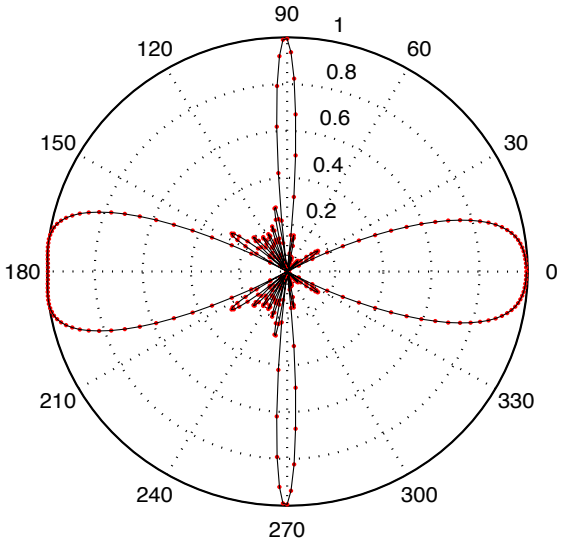

(a) *m*=4

(b) *m*=5

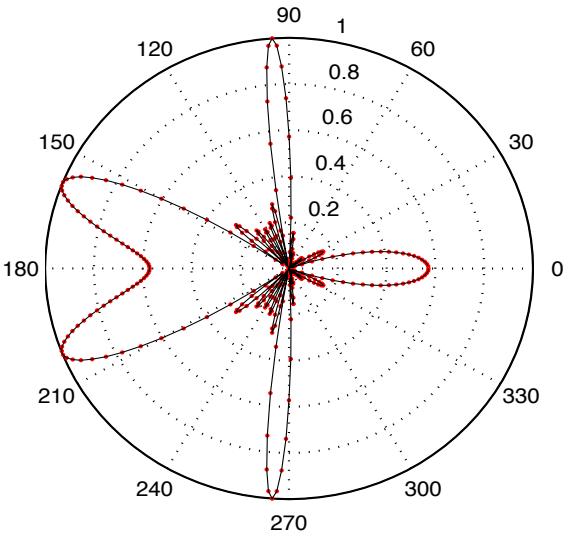
(c) *m*=6

(d) *m*=7

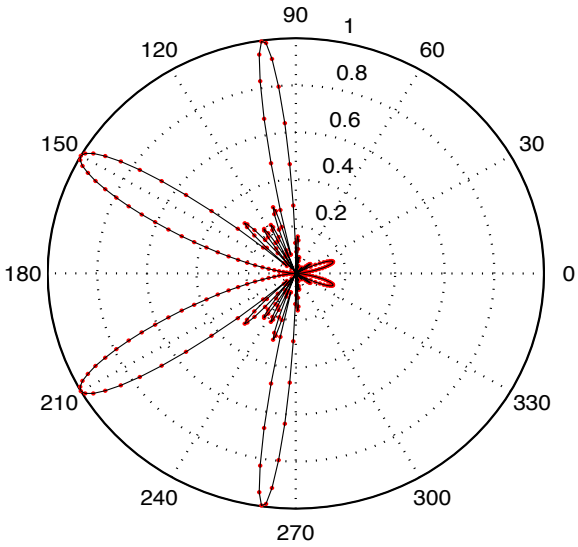
Fig. 5.2. Polar Plot of 16-point FP and AIQ DCT for *m*=4,5,6 and 7



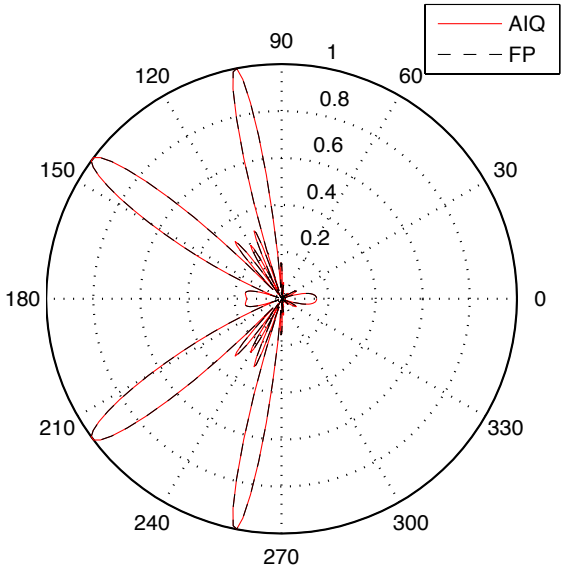

(a) *m*=8




(b) *m*=9

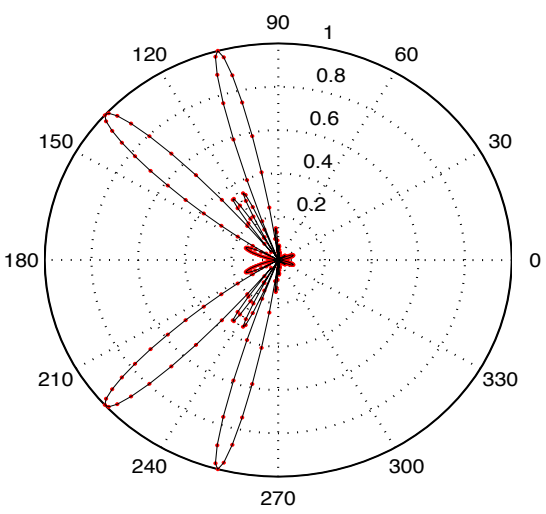



(c) *m*=10

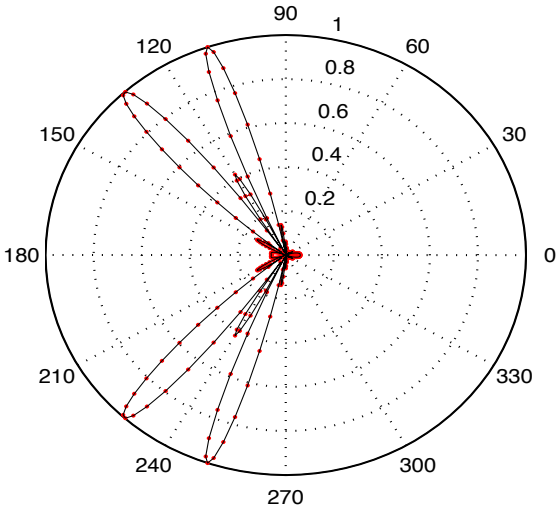



(d) *m*=11

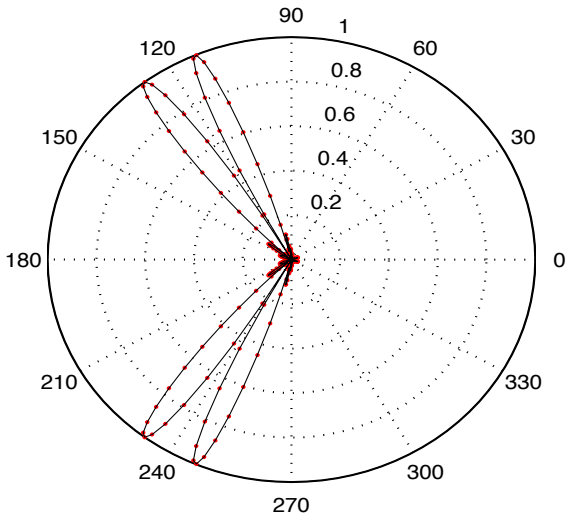
Fig. 5.3. Polar Plot of 16-point FP and AIQ DCT for *m*=8,9,10 and 11



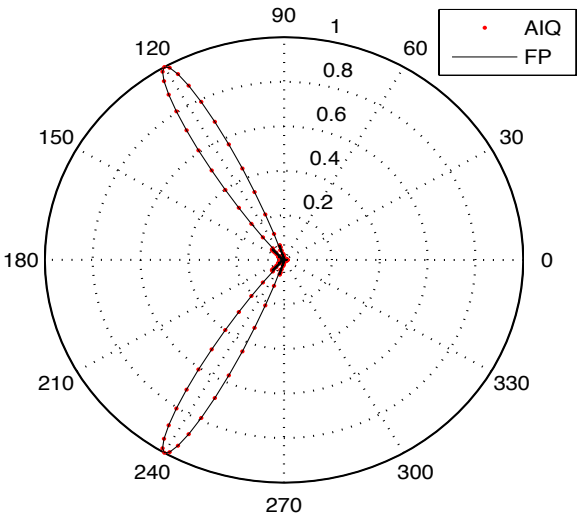

(a) *m*=12




(b) *m*=13




(c) *m*=14




(d) *m*=15

Fig. 5.4. Polar Plot of 16-point FP and AIQ DCT for *m*=12,13,14 and 15

Table 5.4 provides a clear picture of the locations and peaks of main lobes and side lobes for *m*=1 for 8 point and 16 point AIQ DCT and 16 point classical DCT. Being more segmented, 16 point picks up more signals from the same area causing more number of side lobes. But if we observe the peak value, it is much lower than that of the 8 point one. This promises a better SINR in case of the 16 point AIQ DCT. Locations in degrees for minimum and maximum side lobes and that of main lobe with respective normalized peak values are presented in Table 5.4 for 16-point DFT, DCT, AIQ and 8-point AIQ.

**Table 5.4 Locations (Degrees) and Peaks (Normalized) of lobes in different algorithms**

| **Algorithm** | **Location (Min-SL)** | **Peak (Min-SL)** | **Location (Max-SL)** | **Peak (Max-SL)** | **Location (ML)** | **Peak (ML)** |
|---|---|---|---|---|---|---|
| 16p Classical DCT [6] | 124 | 0.0626 | 47 | 0.2195 | 60 | 1.0000 |
| 8p Classical DCT [6] | 129 | 0.1274 | 31 | 0.2291 | 60 | 1.0000 |
| 16p DFT [7] | 124 | 0.0626 | 47 | 0.2195 | 60 | 1.0000 |
| Proposed 16p AIQ | 124 | 0.0626 | 47 | 0.2195 | 60 | 1.0000 |

## 5.4 Matrix Decomposition

The symmetric structure of the 16 point classical DCT, *C(16,16)*, can be exploited to reduce the hardware by decomposing it into sub-matrices. *C(16,16)* can be represented as,

$$C(16,16) = P_1 \times C_p \times P_2 \qquad 5.11$$

Where, $P_1$ is a permutation matrix with only one non-zero element which is 1 in each row and column adding no additional computational cost at all. $P_2$ is the matrix having $\pm 1$'s on the main diagonal and antidiagonal. Thus, only one addition for each row resulting in a total of 16 additions is required for the implementation of $P_2$.

$C_p$ can be considered as the output of direct sum of two sub-matrices:

$$C_p = C_{p1} \oplus C_{p2} \tag{5.12}$$

Applying similar scheme on $C_{p1}$,

$$C_{p1} = P_{11} \times C_{p11} \times P_{12} \tag{5.13}$$

$P_{11}$ is the permutation matrix here, which needs rewiring only for implementation with no additional arithmetic computation. $P_{12}$ is the $8 \times 8$ version of $P_2$, which needs eight simple additions for the implementation. $C_{p11}$ can be obtained with the direct sum as:

$$C_{p11} = C_{p111} \oplus C_{p112} \tag{5.14}$$

Every decomposition reduces the number of non zero values in the matrices which can be seen from the matrices. This consequently reduces the number of arithmetic operations and thereby hardware cost.

$C_{p111}$ can further be implemented as:

$$C_{p111} = C_{p1111} \times P_{1111} \tag{5.15}$$

The final equation would be:

$$C(16,16) = P_1 \times [\left\{ P_{11} \times \{ \left( C_{p1111} \times P_{1111} \right) \oplus C_{p112} \} \times P_{12} \right\} \oplus C_{p2}] \times P_2 \tag{5.16}$$

$$C_p = \begin{bmatrix}
c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
c_2 & c_6 & c_{10} & c_{14} & -c_{14} & -c_{10} & -c_6 & -c_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
c_4 & c_{12} & -c_{12} & -c_4 & -c_4 & -c_{12} & c_{12} & c_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
c_6 & -c_{14} & -c_2 & -c_{10} & c_{10} & c_2 & c_{14} & -c_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
c_8 & -c_8 & -c_8 & c_8 & c_8 & -c_8 & -c_8 & c_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
c_{10} & -c_2 & c_{14} & c_6 & -c_6 & -c_{14} & c_2 & -c_{10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
c_{12} & -c_4 & c_4 & -c_{12} & -c_{12} & c_4 & -c_4 & c_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
c_{14} & -c_{10} & c_6 & -c_2 & c_2 & -c_6 & c_{10} & -c_{14} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -c_{15} & -c_{13} & -c_{11} & -c_9 & -c_7 & -c_5 & -c_3 & -c_1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{13} & c_7 & c_1 & c_5 & c_{11} & -c_{15} & -c_9 & -c_3 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -c_{11} & -c_1 & -c_9 & c_{13} & c_3 & c_7 & -c_{15} & -c_5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_9 & c_5 & -c_{13} & -c_1 & -c_{15} & c_3 & c_{11} & -c_7 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -c_7 & -c_{11} & c_3 & c_{15} & -c_1 & c_{13} & c_5 & -c_9 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_5 & -c_{15} & -c_7 & c_3 & -c_{13} & -c_9 & c_1 & -c_{11} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -c_3 & c_9 & -c_{15} & -c_{11} & c_5 & -c_1 & c_7 & -c_{13} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_1 & -c_3 & c_5 & -c_7 & c_9 & -c_{11} & c_{13} & -c_{15}
\end{bmatrix}$$

$$P_1 = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}$$

$$P_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C_{p1} = \begin{bmatrix} c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 & c_8 \\ c_2 & c_6 & c_{10} & c_{14} & -c_{14} & c_{10} & -c_6 & -c_2 \\ c_4 & c_{12} & -c_{12} & -c_4 & -c_4 & -c_{12} & c_{12} & c_4 \\ c_6 & -c_{14} & -c_2 & -c_{10} & c_{10} & c_2 & c_{14} & -c_6 \\ c_8 & -c_8 & -c_8 & c_8 & c_8 & -c_8 & -c_8 & c_8 \\ c_{10} & -c_2 & c_{14} & c_6 & -c_6 & -c_{14} & c_2 & -c_{10} \\ c_{12} & -c_4 & c_4 & -c_{12} & -c_{12} & c_4 & -c_4 & c_{12} \\ c_{14} & -c_{10} & c_6 & -c_2 & c_2 & -c_6 & c_{10} & -c_{14} \end{bmatrix}$$

$$C_{p11} = \begin{bmatrix} c_8 & c_8 & c_8 & c_8 & 0 & 0 & 0 & 0 \\ c_4 & c_{12} & -c_{12} & -c_4 & 0 & 0 & 0 & 0 \\ c_8 & -c_8 & -c_8 & c_8 & 0 & 0 & 0 & 0 \\ c_{12} & -c_4 & c_4 & -c_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -c_{14} & -c_{10} & -c_6 & -c_2 \\ 0 & 0 & 0 & 0 & c_{10} & c_2 & c_{14} & -c_6 \\ 0 & 0 & 0 & 0 & -c_6 & -c_{14} & c_2 & -c_{10} \\ 0 & 0 & 0 & 0 & c_2 & -c_6 & c_{10} & -c_{14} \end{bmatrix}$$

$$P_{11} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P_{12} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C_{p1111} = \begin{bmatrix} c_8 & c_8 & 0 & 0 \\ 0 & 0 & -c_{12} & -c_4 \\ c_8 & -c_8 & 0 & 0 \\ 0 & 0 & c_4 & -c_{12} \end{bmatrix}, \; P_{1111} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}$$

## 5.5 Hardware Cost and Comparison

Segmented hardware cost for each matrix of our proposed 16-poind AIQ DCT and other specifications are presented in Table 5.5. Apparently, permutation matrices $P_1$ and $P_{11}$ do not require any computations but instead wiring for their implementation. $P_2$ and $P_{12}$ need 16 and 8 additions respectively. The three coefficients $Z_1$, $Z_2$ and $Z_3$ can be implemented with a total of nine additions when represented with 8 bits. The number of bits can be selected according to the precision of the requirement, which offers a flexible implementation.

Considering all the decomposed matrices, a total of 148 adders would implement our proposed 16-point AIQ with a total LUT of 8944.

**Table 5.5 Segmented hardware cost for each decomposed matrix**

| Component | Additions | LUT |
|---|---|---|
| $P_1$ | 0 | 0 |
| $P_2$ | 16 | 136 |
| $P_{11}$ | 0 | 0 |
| $P_{12}$ | 8 | 62 |
| $C_{p11}$ | 25 | 1677 |
| $C_{p2}$ | 56 | 7066 |
| Coefficients | 34 | |
| $z_1+z_2+z_3$ | 9 | |
| Total ($C_p$) | 148 | 8944 |

Table 5.6 provides a comparative figure for the hardware cost of AIQ and classical implementation of DCT. As it shows, proposed AIQ architecture is multiplier free and offers a significant improvement on the classical approach of FP DCT implementation. In place of 31 and 24 multipliers in case of the DCT [6] and DFT [7] consequently, the proposed AIQ DCT offers a completely multiplier free implementation. In case of additions, the proposed algorithm needs 148 additions where as the other two algorithms compared here need 81 and 116 additions as can be seen from the table.

**Table 5.6 Hardware comparison**

| Algorithm | Multiplication | Addition |
|---|---|---|
| 16p Classical DCT [6] | 31 | 81 |
| 16p DFT [7] | 24 | 116 |
| Proposed 16p AIQ | 0 | 148 |

## 5.6 Summary

The Proposed 16-point AIQ implementation of classical DCT offers an error free realization in multibeamforming with significant savings in terms of hardware cost and complexity. Both 3D and 1D AIQ have been considered to find the optimal mapping in terms of hardware complexity. The proposed 3D AIQ offers significant improvement in terms of hardware cost over classical DCT and DFT.

# CHAPTER 6

## CONCLUSION

In real time applications, it is required to steer the radiation direction in real time. Mechanical movement of gigantic antenna systems to direct radiation to a particular direction in real time is not feasible. It has to be done electronically using beamforming techniques to cope with the advancement in technology and demand on exploration. If multibeamforming is of interest, DFT has been considered to be the preferred algorithm. However, DFT has been replaced with DCT in many applications due to the hardware efficiency and competitive accuracy in many applications. Several DCT algorithms have been considered and analyzed in this work as alternatives of DFT in multibeamforming applications. The next generation radio observatories are one of the many fields where our proposed algorithm for multibeamforming could be utilized. Proposed AIQ DCT in multibeamforming can be applied along with ADC and other required digital circuitry in the tile level to produce multiple narrow beams in multiple directions with high accuracy and low hardware complexity. These beams then can be collected and correlated in the station level main computer system for further processing to retrieve the desired information.

For high frequency signals, the physical dimension of the antennas is practical as the wavelength is low. But for low frequency signals, it is necessary to increase the electrical size of the antenna by forming aperture array. A Multibeamforming algorithm such as 16-point AIQ DCT with low complexity and high accuracy and capable of producing multiple directional beams is a required solution in aperture arrays

In wireless communication systems, the limited frequency spectrum is a constraint in regards to meeting the increasing demand. Multibeamforming can be applied to form multiple localized beams in multiple directions using the same frequency. This would greatly enhance the service capacity. Smart antennas with beamformers, these days, are used in wireless communication systems for this purpose. Multiple spot beams with 8-point or 16-point AIQ DCT would be a good option for the smart antenna systems in wireless communications.

## 6.1 Thesis Accomplishments

The accomplishments of this thesis can be concluded as below:

- Discrete transforms such as, DST, WHT and DCT along with DFT have been applied to the array factor to produce multiple beams in multiple directions. As it has been observed, all of these transforms are capable of producing multiple beams. However, the directions for different indices appear to be different. Also, the shapes of the main beams, though comparable, are not same. The number of side lobes and their normalized peaks also make a difference. Depending on the specification and requirement of the application where the multibeamformer is to be deployed, the algorithm can be chosen. Considering performance and hardware efficiency, DCT, however, appears to be a competitive option.
- The standardized Integer DCT's such as JPEG, MPEG, AVS, AVC, and HEVC have been considered in multibeamforming and a comparative study has been performed. When compared among themselves, similar performance has been observed for all the approximations in multibeamforming. All of these hardware-efficient algorithms

showed promising performance. However, these approximations, when compared with the performance of classical DCT, are not completely error free.

- The 8-point AIQ DCT has been considered here in multibeamforming. The performance has been analyzed in comparison with BAS [18], CB [19] and PMCB [7] approximations. As expected, the performance of AIQ DCT has been observed to be completely error free in multibeamforming. An efficient hardware implementation of 2D AIQ DCT with matrix decomposition has also been considered.
- 16-point DCT has been considered to produce main beams in 16 different directions. It showed better performance compared to 8-point DCTs in terms of beam width and peaks of side lobes. To make it more efficient in terms of hardware complexity, 16-point AIQ DCT has been considered. Both 1D and 3D mapping of 16-point AIQ DCT have been developed and analyzed to find the optimum one. Naturally, the 3D mapping out performs the 1D one in terms of hardware complexity. In multibeamforming, this offers an error free implementation of classical DCT.

## 6.2 Future Works

The following extensions of this work can be considered to be studied in future:

- In this work, the far field has been considered to predict the beam patterns; so the inclusion of near field considerations are likely to alter the shape in terms of beam width which is required to be considered in practical applications.
- ULA with uniform plane wave has been considered to evaluate the performance of AIQ DCT in multibeamforming. Consideration of other array architectures and

inclusion of TE and TM waves along with accompanying circuitry are to be done to ensure its feasibility in practical applications.

- Furthermore, increasing the DCT matrix size can increase the number of beams and performance can be analyzed.

# LIST OF REFERENCES


[1] Weib, M. (2009) Digital Antennas. In Multistatic Surveillance and Reconnaissance: Sensor, Signals and Data Fusion (pp. 5-1 – 5-29). Educational Notes RTO-EN-SET-133, Paper 5. Neuilly-sur-Seine, France: RTO. Available from: http://www.rto.nato.int.abstract.aps.

[2] Van Veen, B.D.; Buckley, K.M., "Beamforming: a versatile approach to spatial filtering," ASSP Magazine, IEEE , vol.5, no.2, pp.4,24, April 1988 doi: 10.1109/53.665.

[3] Constantine A. Balanis, "Arrays: Linear, Planar, and Circular" in Antenna Theory Analysis and Design, Third Edition, A John Wiley & Sons, Inc., Publication, 2005, ch. 6, pp. 283-365.

[4] Godara, L.C., "Application of antenna arrays to mobile communications. II. Beam-forming and direction-of-arrival considerations," Proceedings of the IEEE , vol.85, no.8, pp.1195,1245, Aug 1997.

[5] Zarb-Adami, K.; Faulkner, A.; bij de Vaate, J.G.; Kant, G.W.; Picard, P., "Beamforming techniques for large-N aperture arrays," Phased Array Systems and Technology (ARRAY), 2010 IEEE International Symposium on , vol., no., pp.883,890, 12-15 Oct. 2010.

[6] Wei Liu and Stephan Weiss, "Introduction" in Wideband Beamforming *Concepts and Techniques*, First Edition, A John Wiley and Sons, Ltd., Publication, 2010, ch. 1, pp. 1-18.

[7] S. Potluri, A. Madanayake, R. Cintra, et al., "Multiplier-free DCT approximations for RF multi-beam digital aperture-array space imaging and directional sensing", Measurement Science and Technology, IOP Publishing Ltd., 2012.

[8] Ahmed, N.; Natarajan, T.; Rao, K.R., "Discrete Cosine Transform," Computers, IEEE Transactions on, vol.C-23, no.1, pp.90,93, Jan. 1974.

[9] K. Wahid, M. Martuza, M. Das, C. McCrosky. "Efficient hardware implementation of 8×8 integer cosine transforms for multiple video codecs." J Real-Time Image Proc, Vol. 8, pp. 403-410, Dec. 2013.

[10] Muhammad Martuza, Khan A. Wahid. "Implementation of a cost-shared transform architecture for multiple video codecs", J Real-Time Image Proc, pp. 1-12, Aug. 2012.

[11] Chih-Peng Fan; Guo-An Su, "Fast Algorithm and Low-Cost Hardware-Sharing Design of Multiple Integer Transforms for VC-1," IEEE TCAS-II, vol.56, no.10, pp.788-792, Oct. 2009.

[12] Meher, P.K.; Sang Yoon Park; Mohanty, B.K.; Khoon Seong Lim; Chuohao Yeo, "Efficient Integer DCT Architectures for HEVC," IEEE TCASVT, vol.24, no.1, pp.168-178, Jan. 2014.

[13] Dimitrov, V.; Jullien, G.A., "Multidimensional algebraic-integer encoding for high performance implementation of DCT and IDCT," Electronics Letters, vol.39, no.7, pp.602-603, 3 April 2003.

[14] K. Wahid, V. Dimitrov, G. Jullien, "New Encoding of 8x8 DCT to make H.264 lossless", Proceedings of APCCAS, pp. 781-784, 2006.

[15] Wahid, K.; Dimitrov, V.; Jullien, G., "Error-free computation of 8×8 2D DCT and IDCT using two-dimensional algebraic integer quantization," 17th IEEE Sym. on Computer Arithmetic, pp.214,221, 27-29 2005.

[16] Cozzens, J.H.; Finkelstein, L., "Computing the discrete Fourier transform using residue number systems in a ring of algebraic integers," Information Theory, IEEE Transactions on , vol.31, no.5, pp.580,588, Sep 1985.

[17] Games, R.; Moulin, D.; O'Neil, S.D.; Rushanan, J.J., "Algebraic-integer quantization an residue number system processing," Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on , vol., no., pp.948,951 vol.2, 23-26 May 1989.

[18] Bouguezel S, Ahmad M O and Swamy M N S, "Low-complexity 8×8 transform for image compression", Electron. Lett. (44) 1249-50, 2008.

[19] Cintra R J and Bayer F M, "A DCT approximation for image compression", IEEE Signal Process. Lett. 18, 579-82, 2011.

[20] Britanak V,Yip P and Rao K R, Discrete Cosine and Sine Transforms, (New York: Academic), 2007.

[21] Blahut R E, Fast Algorithms for Digital Signal Processing (Reading, MA: Addison-Wesley), 1985.

[22] Arai Y,Agui T and Nakajima M, "A fast DCT-SQ scheme for images", Trans. IEICE E-71, 1095–7, 1988.

[23] Van Ardenne, A., "Concepts of the Square Kilometre Array; toward the new generation radio telescopes," Antennas and Propagation Society International Symposium, 2000. IEEE , vol.1, no., pp.158,161 vol.1, 16-21 July 2000.

[24] van Ardenne, A.; Bentum, M.J.; Boonstra, A.J., "SKA antenna systems; Outlook for non-astronomy applications," Antennas and Propagation (EUCAP), 2012 6th European Conference on , vol., no., pp.1199,1203, 26-30 March 2012.

[25] A. van Ardenne, A. Smolders, and G. Hampson, “Active adaptive an- tennas for radio astronomy: Results of the R&D program toward the square kilometer array,” in Proc. Astronomical Telescopes and Instru- mentation 2000—Radio Telescopes, Munich, Germany, Mar. 2000, SPIE Conf. 4015.

[26] N. E. Kassim, T. Joseph, W. Lazio, and W. C. Erickson. (1999, Sept.) Opening a New Window on the Universe: High Reso- lution, Long Wavelength Radio Astronomy [Online]. Available: http://rsd-www.nrl.navy.mil/7213/lazio/decade_web/decade_web.html, http://www.lofar.org.

[27] J. D. Bregman, “Concept design for a low-frequency array,” in Proc. SPIE Astronomical Telescopes and Instrumentation, vol. 4015, Munich, Germany, Mar. 2000.

[28] J. W. Dreher, “The one hectare telescope (1hT) project,” in Proc. Per- spectives in Radio Astronomy: Technologies for Large Antenna Arrays , Dwingeloo, The Netherlands, Apr. 1999. (ISBN 90-805 434-2-X; [On- line]. Available: http://www.astron.nl).

[29] R. S. Dixon, “Argus: A next-generation omnidirectional radio tele- scope,” in Proc. High-Sensitivity Radio Astronomy , U.K., Jan. 1997. N. Jackson and R. J. Davies (eds.), Cambridge Univ. Press, pp. 287–291.

[30] Ellingson, S.W.; Cazemier, W., "Efficient multibeam synthesis with interference ing for large arrays," Antennas and Propagation, IEEE Transactions on , vol.51, no.3, pp.503,511, March 2003.

[31] Mucci, Ronald A., "A comparison of efficient beamforming algorithms," Acoustics, Speech and Signal Processing, IEEE Transactions on , vol.32, no.3, pp.548,558, Jun 1984.

[32] Li, W.; Xinping Huang; Leung, H., "Performance evaluation of digital beamforming strategies for satellite communications," Aerospace and Electronic Systems, IEEE Transactions on , vol.40, no.1, pp.12,26, Jan 2004.

[33] Ellingson, S.W.; Cazemier, W., "Efficient multibeam synthesis with interference ing for large arrays," Antennas and Propagation, IEEE Transactions on , vol.51, no.3, pp.503,511, March 2003.

[34] https://www.skatelescope.org/ last accessed: 12-04-2015

[35] http://courses.cs.washington.edu/courses/cse378/00sp/Sec5-1.htm/ last accessed: 07-04-2015

[36] http://www.news.com.au/technology/science/largest-radio-telescope-funded/story-fn5fsgyc-1226544081909 last accessed: 07-04-2015

# APPENDIX

Accepted Conference Papers:

- Ziad Gias, Mehedi Hasan, Khan A. Wahid, "Multi-Beamforming with Uniform Linear Array and Algebraic Integer Quantization Based DCT", IEEE Int'l Symposium on Circuits & Systems (ISCAS), 2015, Paper-2253.
- Ziad Gias, Mehedi Hasan, Khan A. Wahid, “Performance Analysis of Integer DCT Algorithms in Multi-beamforming with Uniform Linear Array”, IEEE CCECE, 2015, Submission-1345.