DBFアレーアンテナによるリアルタイム到来方向推定システムとその応用に関する研究

(13650403)

平成13年度～平成15年度科学研究費補助金
（基盤研究(C)(2)）研究成果報告書

平成16年5月

研究代表者　　新井 宏之

横浜国立大学 大学院 工学研究院

はしがき

研究組織

研究代表者：　新井 宏之 （横浜国立大学 大学院 工学研究院）
研究分担者：　市毛 弘一 （横浜国立大学 大学院 工学研究院）

交付決定額（配分額，金額単位：千円）

	直接経費	間接経費	合計
平成13年度	1,400	0	1,400
平成14年度	1,100	0	1,100
平成15年度	700	0	700
総計	3,200	0	3,200

研究発表

学会誌等

1. Minseok Kim, Koichi Ichige and Hiroyuki Arai: "Design of Jacobi EVD processor based on CORDIC for DOA estimation with MUSIC algorithm", IEICE Transactions on Communications, vol. E85-B, no.12, pp. 2648-2655, December 2002.

2. Koichi Ichige, Masashi Shinagawa and Hiroyuki Arai: "A Fast Algebraic Approach to the Eigenproblems of Correlation Matrices in DOA Estimation", IEICE Transactions on Communications, vol. E86-B, no. 2, pp. 865-869, February 2003.

3. Minseok Kim, Koichi Ichige and Hiroyuki Arai: "Implementation of FPGA Based Fast Unitary MUSIC DOA Estimator", IEICE Transactions on Electronics, (accepted, in press).

口頭発表

1. Masashi Shinagawa, Koichi Ichige and Hiroyuki Arai, "Performance Evaluation of DOA Estimation Algirithms in Digital Implementation", Proceedings of URSI General Assembly, No. C2.P.3, Maastricht, The Netherlands, August 2002.

2. Koichi Ichige, Masashi Shinagawa and Hiroyuki Arai, "An Algebraic Approach to Eigenproblems toward Fast DOA Estimation", Proceedings of URSI General Assembly, No. C2.P.5, Maastricht, The Netherlands, August 2002.

3. Minseok Kim, Koichi Ichige and Hiroyuki Arai: "Design of Jacobi EVD Processor Based on CORDIC for DOA Estimation with MUSIC algorithm", Proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, No. MPO1.5.5, Lisbon, Portugal, September 2002.

4. Naoya Matsumoto, Koichi Ichige, Hiroyuki Arai, "Fixed-Point Digital Processing of Recursive Least-Square Algorithm:Toward FPGA Implementation of MMSE Adaptive Array Antenna", Proceedings of The International Conference on Signal Processing and its Applications, no. FriPmPO4-6, Paris, France, July 2003.

5. Minseok Kim, Koichi Ichige and Hiroyuki Arai: "Implementation of FPGA based Fast DOA Estimator using Unitary MUSIC Algorithm", Proceedings of IEEE Vehicular Technology Conference, Orlando, FA, U.S.A., October 2002.

6. Minseok Kim, Koichi Ichige and Hiroyuki Arai: "DOA-based Adaptive Array Antenna Testbed System", Proceedings of International Tropical Conference on Wireless Communication Technology, Honolulu, HI, U.S.A., October 2002.

研究成果による工業所有権の出願・取得状況

特になし

IEICE TRANS. COMMUN., VOL.E85–B, NO.12 DECEMBER 2002

PAPER Special Issue on Software Defined Radio Technology and Its Applications

Design of Jacobi EVD Processor Based on CORDIC for DOA Estimation with MUSIC Algorithm

Minseok KIM†, *Student Member*, Koichi ICHIGE†, *and* Hiroyuki ARAI†, *Regular Members*

SUMMARY Computing the Eigen Value Decomposition (EVD) of a symmetric matrix is a frequently encountered problem in adaptive (or smart or software) antenna signal processing, for example, super resolution DOA (Direction Of Arrival) estimation algorithms such as MUSIC (MUltiple SIgnal Classification) and ESPRIT (Estimation of Signal Parameters via Rotational Invariance Technique). In this paper the hardware architecture of the fast EVD processor of symmetric correlation matrices for the application of an adaptive antenna technology such as DOA estimation is proposed and the basic idea is also presented. Cyclic Jacobi method is well known for the simplest algorithm and easily implemented but its convergence time is slower than other factorization algorithm like QR-method. But if considering the fast parallel computation of the EVD with a hardware architecture like ASIC (Application Specific Integrated Circuit) or FPGA (Field Programmable Gate Array), the Jacobi method can be a appropriate solution, since it offers a quite higher degree of parallelism and easier implementation than other factorization algorithms. This paper computes the EVD using a Jacobi-type method, where the vector rotations and the angles of the rotations are obtained by CORDIC (COordinate Rotation DIgital Computer). The hardware architecture suitable for ASIC or FPGA with fixed-point arithmetic is presented. Because it consists of only shift and add operations, this hardware friendly feature provides easy and efficient implementation. In this paper, the computational load, the estimate of circuit scale and expected performance are discussed and the validation of fixed-point arithmetic for the practical application to MUSIC DOA estimation is examined.

key words: *adaptive antenna, FPGA implementation, DOA estimation, MUSIC, EVD*

1. Introduction

In many adaptive (or smart or software) antenna technologies, when receiving communication signals at an adaptive antenna array, usually it is desired or necessary to estimate DOAs (Directions of Arrival) of incident signals and the DOAs are used in a beamformer in order to separate and receive the desired signal spatially.

Generally, in the communication environment multi-path fading caused by a reflection by any physical structures is a serious problem. When passing through multi-path, the signals are delayed and out of phase from the signals through direct-path that causes the signal strength to be changed extremely at a receiver end, and hence receiving quality is also reduced. The requirements of wider band and higher transmission rate in the next generation communication make it more critical, and when the mobile terminal moves at high speed, the transmission rate may be confined by this problem. If the multi-path fading can be solved, the trade off between mobility and transmission rate must be dramatically improved.

An adaptive antenna technology can provide a solution of multi-path fading. The adaptive antenna can suppress the adverse effect of multi-path delayed coherent signals and interferences by steering beams toward intended directions and nulls toward the other undesired directions so that it can achieve high communication quality. This operation can make the receiving signal strength almost flat and stable over a threshold level. Therefore it is necessary that an adaptive antenna should find the DOAs of signals and form beams and steer nulls within a fading period. Considering mobility of several hundreds of km/h, the fading period becomes very short time. It is very difficult to compute them by general serial architecture DSP processors, and hence the high-speed parallel computing processor with a specified function must be needed.

In this paper, the implementation issue in MUSIC (MUltiple SIgnal Classification), a super resolution DOA estimation method, and the examination of hardware design based on FPGAs are presented. MUSIC method is one of the subspace-based methods [1]. Generally the subspace-based methods are based on the Eigen Value Decomposition (EVD) of the covariance or correlation matrix. In the EVD based system, real-time processing is very difficult to be realized because of its complex logic and heavy computational load. This paper proposes the hardware logic design of a fast EVD processor which is suitable for realtime processing and can be implemented for adaptive antenna technologies practically. It uses Cyclic Jacobi method. Cyclic Jacobi method is well known for the simplest algorithm and easily implemented but its convergence time is slower than other factorization algorithms like QR-method [2]. But if considering the fast parallel computation of the EVD with a dedicated circuit like ASIC or FPGA, the Cyclic Jacobi method can be a good choice, since it offers a very higher degree of parallelism and easier implementation than QR-method [3]. This paper uses hardware friendly CORDIC (COordinate Rotation DIgital

Manuscript received March 29, 2002.
Manuscript revised August 7, 2002.

†The authors are with the Division of Electrical & Computer Engineering Department, Yokohama National University, Yokohama-shi, 240-8501 Japan.

Computer) algorithm for vector rotators and arctangent computers, which are the basic processors of this design.

This paper is organised as follows. Section 2 presents the principle and computation flow of DOA estimation in MUSIC Method briefly. Section 3 introduces Cyclic Jacobi EVD algorithm and Sect. 4 describes the basic ideas of CORDIC algorithm and the circuit implementation. For simple architecture and practical realizability, the paper uses fixed-point or fixed bit-length arithmetic instead of floating-point operations. With fixed-point arithmetic it is desired to overcome processing speed limitation and power consumption. In Sect. 5 the number of Jacobi sweeps is determined, that is, the computational load is confined constantly, and the errors caused by the fixed-point operations are discussed. Section 6 proposes the hardware architecture and circuit design. Section 7 yields its computational load and expected performance discussion.

2. DOA Estimation by MUSIC Method

The computation flow of DOA estimation by MUSIC method is illustrated in Fig. 1. First, the correlation matrix $\boldsymbol{R}_{xx}(t)$ is computed by $E[\boldsymbol{X}(t) \cdot \boldsymbol{X}^H(t)]$ where $\boldsymbol{X}(t)$ is the data vector received at array antenna, $E[\cdot]$ is the expectation operator, and the superscript H denotes Hermitian transposition. Actually, the finite average of the correlation matrix is used to approximate a stochastic process. Then the spatial smoothing process suppresses the correlation between incident signals, which enables the estimation when the signals are correlated with one another. The correlation matrix is decomposed into signal and noise sub-space eigenvectors by EVD, and the DOAs can be found by computing the angular spectrum with inner product of noise sub-space and array mode vectors [1].

It seems to be not difficult to implement the computation of correlation matrix, spatial smoothing filter and spectrum synthesis with dedicated circuit using any fast algorithm, thanks to the simplicity of their logics.

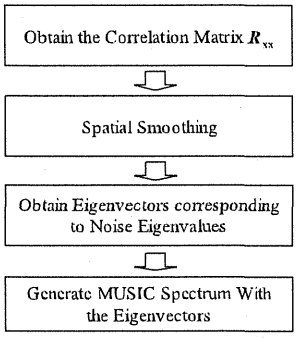


Fig. 1 Computational flow of MUSIC method.

But especially EVD computation is not so simple but rather complex. Generally it is thought that the EVD process has 30–50% of the whole computational load of DOA Estimation. In fact, there are many algorithms for EVD problems but they are just numerical solutions for serial processing on general-purpose computers. Therefore it is necessary to modify and reconstruct the serial algorithm to be suitable for parallelism of dedicated circuit in order to meet the performance requirement for the prcatical use of an adaptive antenna in the next generation communication system.

3. Cyclic Jacobi Method

This section describes the basic principle of Cyclic Jacobi, one of the EVD computation methods. It can be implemented with the simple iterative process of plane rotations. Cyclic Jacobi method computes symmetric eigenvalue problems by applying a sequence of orthonormal rotations to the left and right sides of the target $N \times N$ matrix $\boldsymbol{R}$ as

$$\boldsymbol{E}^T \cdot \boldsymbol{R} \cdot \boldsymbol{E} = \boldsymbol{D}, \qquad (1)$$
$$\left(\begin{array}{ll} \because & \boldsymbol{E} = \boldsymbol{J}_1 \cdot \boldsymbol{J}_2 \cdot \boldsymbol{J}_3 \cdots, \\ & \boldsymbol{J} = \boldsymbol{W}_{12} \cdot \boldsymbol{W}_{13} \cdot \boldsymbol{W}_{N-1,N} \end{array} \right)$$

where $\boldsymbol{W}_{pq}$ is an orthonormal plane rotation over an angle θ in the (p, q) plane whose elements are $w_{pp} = \cos\theta$, $w_{pq} = \sin\theta$, $w_{qp} = -\sin\theta$, $w_{qq} = \cos\theta$ $(p > q)$, and defined as Eq. (2). $\boldsymbol{J}$ is the multiple rotation of $\boldsymbol{W}_{pq}$'s in the cyclic-by-row manner of (p, q) which is called a Jacobi sweep, and the superscript 'T' and subscript N denote transposition and array length respectively.

$$\boldsymbol{W}_{pq} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & \cos\theta & \cdots & \sin\theta & & \\ & & \vdots & 1 & \vdots & & \\ & & -\sin\theta & \cdots & \cos\theta & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} \qquad (2)$$

Theoretically, by the infinite transform of matrix $\boldsymbol{R}$, it is certain that $\boldsymbol{E}$ and $\boldsymbol{D}$ converge into the matrix whose column vectors are composed of eigenvectors and the matrix whose diagonal elements are eigenvalues, respectively. A symmetric matrix $\boldsymbol{R}$ is transformed to $\boldsymbol{R}'$ by plane rotation as

$$\boldsymbol{R}' = \boldsymbol{W}_{pq}^{T} \cdot \boldsymbol{R} \cdot \boldsymbol{W}_{pq} = \begin{pmatrix} & \cdots & r'_{1p} & \cdots & r'_{1q} & \cdots & \\ \vdots & & \vdots & & \vdots & & \vdots \\ r'_{p1} & \cdots & r'_{pp} & \cdots & r'_{pq} & \cdots & r'_{pN} \\ \vdots & & \vdots & & \vdots & & \vdots \\ r'_{q1} & \cdots & r'_{qp} & \cdots & r'_{qq} & \cdots & r'_{qN} \\ \vdots & & \vdots & & \vdots & & \vdots \\ & \cdots & r'_{Np} & \cdots & r'_{Nq} & \cdots & \end{pmatrix}. \quad (3)$$

By the above transform, only p-th and q-th rows and columns of $\boldsymbol{R}'$ are changed as Eq. (3). The optimal rotation angle in a (p, q) plane is determined by Eq. (4). It is the basic strategy of Cyclic Jacobi method that the iterative process of the plane rotation converges (p, q) and (q, p) elements of a target matrix into zero.

$$r'_{pq} = r'_{qp} = 0 \quad (4)$$

In above manner matrix $\boldsymbol{R}$ converges into a diagonal eigenvalue matrix. This is called Cyclic Jacobi method.

In Cyclic Jacobi method, the off-diagonal quantity $S^{(h)}$ is defined by

$$S^{(h)} = \sqrt{\frac{1}{2}\left[\|\boldsymbol{R}^{(h)}\|_F^2 - \sum_{i=1}^{N}\{r_{ii}^{(h)}\}^2\right]}, \quad (5)$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Therefore, if $S^{(h)}$ converges into zero, the target matrix $\boldsymbol{R}$ becomes eigenvalue diagonal matrix as

$$\lim_{h\to\infty} S^{(h)} \to 0 \Leftrightarrow \lim_{h\to\infty} \boldsymbol{R}^{(h)} \to diag\,[\lambda_1, \cdots, \lambda_N]. \quad (6)$$

On the other hand, the execution of a similarity transform yields

$$\left[S^{(h+1)}\right]^2 = \left[S^{(h)}\right]^2 - \left[\{r_{pq}^{(h)}\}^2 - \{r_{pq}^{(h+1)}\}^2\right]. \quad (7)$$

From Eq. (7), obviously the maximal reduction of $S^{(h)}$ is obtained if $r_{pq}^{(h+1)} = 0$. The condition for an optimal angle, maximal reduction of $S^{(h)}$, is achieved as

$$\theta_{opt} = \frac{1}{2}\tan^{-1}\left[\frac{2r_{pq}}{r_{pp} - r_{qq}}\right] = \frac{1}{2}\tan^{-1}\tau, \quad (8)$$

where $\tau = \frac{2r_{pq}}{r_{pp}-r_{qq}}$ [2], [3].

4. CORDIC Algorithm for Vector Rotation and Computing Arctangent

The CORDIC algorithm is operated in one of two modes [4]. One is the rotation mode and the other is the vectoring mode. In this paper, the rotation mode is used for vector rotations and the vectoring mode for computing the optimal rotation angle by arctangent of τ, as mentioned in Sect. 3. For the rotation mode, the CORDIC equations are

$$\begin{cases} x_{i+1} = x_i - y_i \cdot d_i \cdot 2^{-i} \\ y_{i+1} = y_i + x_i \cdot d_i \cdot 2^{-i} \\ z_{i+1} = z_i - d_i \cdot \tan^{-1}\left(2^{-i}\right) \end{cases}, \quad (9)$$
$$d_i = -1 \text{ if } z_i < 0, \quad +1 \text{ otherwise}$$

where $d_i = \pm 1$ (direction of rotation) and z is called angle accumulator. It provides the following results after finite number of iterations as much as the bit-length.

$$\begin{aligned} x_n &= A_n\left(x_0 \cos z_0 - y_0 \sin z_0\right) \\ y_n &= A_n\left(y_0 \cos z_0 + x_0 \sin z_0\right) \\ z_n &= 0 \\ A_n &= \textstyle\prod_{i=0}^{n} \sqrt{1 + 2^{-2i}} \end{aligned}, \quad (10)$$

where A_n is a computational gain. On the other hand, the CORDIC equations for vectoring mode are

$$\begin{cases} x_{i+1} = x_i - y_i \cdot d_i \cdot 2^{-i} \\ y_{i+1} = y_i + x_i \cdot d_i \cdot 2^{-i} \\ z_{i+1} = z_i - d_i \cdot \tan^{-1}\left(2^{-i}\right) \end{cases}, \quad (11)$$
$$d_i = +1 \text{ if } y_i < 0, \quad -1 \text{ otherwise}$$

which also provides the following result after a finite number of iterations.

$$\begin{aligned} x_n &= A_n\sqrt{x_0^2 + x_0^2} \\ y_n &= 0 \\ z_n &= z_0 + \tan^{-1}(y_0/x_0) \\ A_n &= \textstyle\prod_{i=0}^{n} \sqrt{1 + 2^{-2i}} \end{aligned} \quad (12)$$

After the last step, the scaling operation must be performed elsewhere in the system. From Eqs. (10), (12) the scaling factor K_n is obtained by

$$K_n = \frac{1}{A_n} = \prod_{i=0}^{n} \frac{1}{\sqrt{1 + 2^{-2i}}}; \text{ Scaling Factor.} \quad (13)$$

The rotation and vectoring mode of CORDIC algorithm are restricted to rotation angles between $-\pi/2$ and $\pi/2$ due to the use of 2^0 for the tangent in the first iteration.

4.1 Architecture

The CORDIC algorithm performs only shift and add operations and hence it is easy to implement and suitable for dedicated circuit. In the implementation of CORDIC algorithm, there can be various architectures depending on which is more serious consideration, circuit resource or performance. There are a few kinds of architecture for implementing a CORDIC algorithm, for example, bit-serial or bit-parallel, and unrolled or rolled (iterative), and so on [5]. This paper uses the bit-parallel unrolled CORDIC architecture for high performance. It is a cascade structure of the consecutive CORDIC stages as shown in Fig. 2, where arctangent values are precomputed and stored at any memory

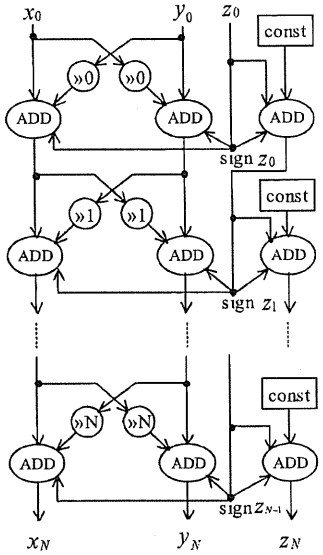


Fig. 2 Unrolled CORDIC architectures (rotation mode).

block. The unrolled design consists of only combinatorial logic components and the results can be computed fast with pipelined process.

4.2 Arctangent

If the angle accumulator z is initialized with zero ($z_0 = 0$), the arctangent, $\theta = \tan^{-1}(y/x)$, is directly computed using the vectoring mode. The result is taken from the angle accumulator as Eq. (14) [5].

$$z_n = z_0 + \tan^{-1}(y_0/x_0) \tag{14}$$

4.3 Scaling by Twice Rotation Architecture

In CORDIC process, a scaling must be required because the bit length of the processor must be finite. As Eq. (13) the scaling and normalizing using precomputed scaling value couldn't perform only with shift and add operations. It may be not easy to be implemented with a dedicated circuit.

Twice executing rotation by $z_0/2$ can solve this problem [3]. Let an elementary rotation by angle z_0 be composed of twice executing a rotation by $z_0/2$. Hence Eq. (9) is rewritten as

$$\begin{cases} x_{i+1} &= (1-2^{-2i})x_i - y_i \cdot d_i \cdot 2^{-i+1} \\ y_{i+1} &= (1-2^{-2i})y_i + x_i \cdot d_i \cdot 2^{-i+1} \\ z_{i+1} &= z_i - d_i \cdot \tan^{-1}\left(2^{-i}\right) \end{cases} . \tag{15}$$
$$d_i = -1 \text{ if } z_i < 0, +1 \text{ otherwise}$$

It requires 4-shift and 5-add operations per an iteration stage as shown in Fig. 3. The results after a finite number of iterations is as

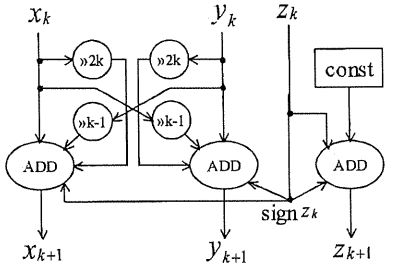


Fig. 3 k-th stage of twice executing rotation (rotation mode).

$$\begin{aligned} x_n &= A_n^{(ii)}\left(x_0 \cos z_0 - y_0 \sin z_0\right) \\ y_n &= A_n^{(ii)}\left(y_0 \cos z_0 + x_0 \sin z_0\right) \\ z_n &= 0 \\ A_n^{(ii)} &= \{A_n\}^2 = \textstyle\prod_{i=0}^{n}(1+2^{-2i}) \end{aligned} . \tag{16}$$

In Eq. (16), the computational gain $A_n^{(ii)}$ becomes square root free. In addition, twice rotation gives the scaling factor of division operation free as well as square root free as Eq. (17). It is because for a given precision b of the shift and add operations, all factors $(1-2^{-i})$ with $i > b$ do not contribute to $K_n^{(ii)}$. So the scaling factor can be approximated by simplified form as Eq. (17) [3].

$$\begin{aligned} K_n^{(ii)} = \{K_n\}^2 &= \prod_{i=0}^{n} \frac{1}{1+2^{-2i}} = \frac{1}{2}\prod_{i=1}^{n} \frac{1-2^{-2i}}{1-2^{-4i}} \\ &\approx \frac{1}{2}\prod_{i=1}^{n/4}\left(1-2^{-(4i-2)}\right) \end{aligned} \tag{17}$$

Equation (17) can be also computed only with shift and add operations. Figure 3 shows the k-th stage of the twice rotation. In this paper twice executing rotation unrolled architecture is used for efficient circuit implementation.

5. Examination of CORDIC-Jacobi EVD Processor with Fixed Point Operations

This section describes the practical implementation of Jacobi EVD processor based on CORDIC with a fixed-point arithmetic operation. The required number of Jacobi sweeps, the appropriate precision for the desired accuracy and applicability to MUSIC DOA estimator are examined. The first thing to determine is how many times of Jacobi sweeps can achieve the desired convergence.

Figure 4 illustrates the reduction of off-diagonal norm to the iteration number of Jacobi sweeps with a 6×6 random real symmetric matrix of 12-bit precision. In case of 4-element linear array antenna, the dimension of the correlation matrix is 6×6 after spatial smoothing with 3 subelements. The off-diagonal reduction is defined as Eq. (5) in Sect. 3 and the lower value means that

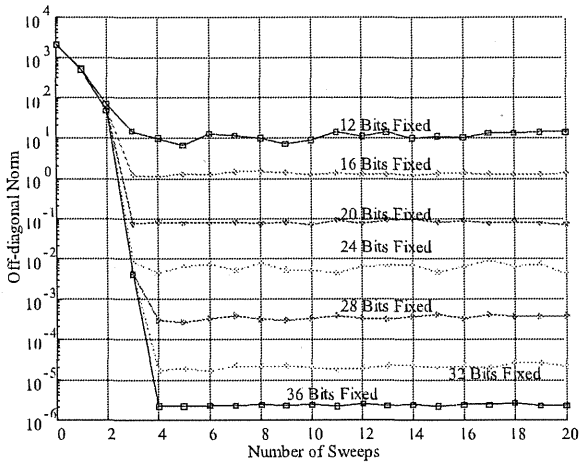


Fig. 4 Reduction of off-diagonal norm (6 × 6 random symetric matrix).

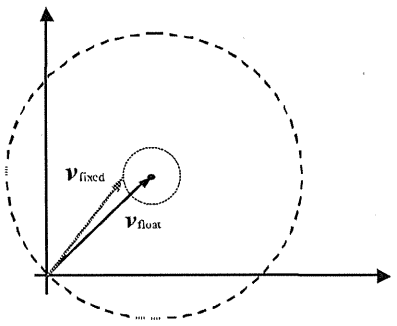


Fig. 5 Errors between vectors.

it is closer to the diagonal matrix. The 32-bit floating-point operation on general computers by C language converges to the machine zero within a given precision after several number of Jacobi sweeps, but Fig. 4 shows that fixed-point operations do not converge but only keep on vibrating after around 4 Jacobi sweeps regardless of the precision from 12 to 36 bits. This is caused by the limited-precision of the fixed-point operation. At the cost of computation accuracy, the fixed-point operation achieves simpler circuit impelementation, high performance and low power consumption. Without using additional convergence decision circuit, fixing the number of sweeps by 4 cannot only be a proper choice, but more computations must be needless for efficiency in hardware resource. Since the finite number of operation determined in advance provides the same computation time in any cases, the realtime processing can be realized.

The next thing to examine is the precision of fixed-point operation. To validate fixed-point operation, the accuracy within allowable error range must be guaranteed. Equation (18) yields the error ratio where $\boldsymbol{v}$'s are the vectors whose elements consist of eigenvalues com-

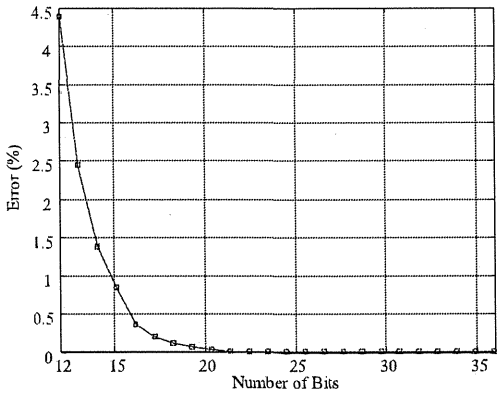


Fig. 6 Error ratio of fixed-point operations to 32-bit floating poing operation (6 × 6 random symetric matrix).

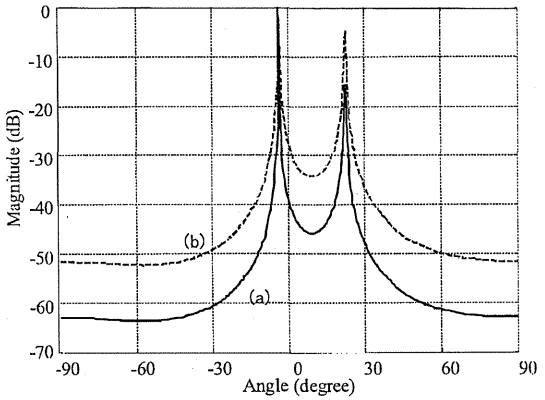


Fig. 7 MUSIC spectrum in case of 4 elements and 2 incident waves (DOAs are −5° and 20°); (a) 32-bit floating-point operation with general computer and (b) 16-bit fixed-point operation by CORDIC-Jacobi EVD processor.

puted in respective subscripted ways and $\|\cdot\|$ denotes vector norm where the error between vectors is defined as Fig. 5.

$$\text{Error} = \frac{\|\boldsymbol{v}_{float} - \boldsymbol{v}_{fixed}\|}{\|\boldsymbol{v}_{float}\|} \tag{18}$$

Figure 6 shows the error ratio of fixed-point operations with various bit-lengthes on the CORDIC-Jacobi EVD processor with respect to the 32-bit floating-point operation on general computers. Figure 6 uses an 6 × 6 random real symmetric matrix of 12-bit precision as an input matrix. Of course, the longer is the given precision in the fixed-point arithmetic, the more accuracy we can achieve. When it is implemented with 16-bit precision, it has several % of error for the floating-point operation, but in reality, around 16-bit precision is desired for practical uses. In this paper, the processor's computational load and the bit-length are 4 Jacobi sweeps and 16-bit fixed-point operation respectively. Instead of floating point arithmetic, the computation accuracy may be doubtful, but as shown in Fig. 7 the simulation

results are quite satisfactory. In Fig. 7, DOA estimation is performed by spectral MUSIC method assuming that any 2 electromagnetic waves arrive at 4-element array antenna. The EVD computations are performed with 32-bit floating-point operation on a general computer by C language and 16-bit fixed-point operation on proposed design, Jacobi EVD computer based on CORDIC. Except for EVD computation, all the other processes of MUSIC method such as correlation matrix, spartial smoothing and MUSIC specturm were computed on general computer with floating point operation. Figures 7(a) and (b) show the results respectively.

6. Hardware Design of EVD Processor for DOA Estimation

In previous sections it was mentioned that about 4 Jacobi sweeps are sufficient for convergence and the fixed-point operation of 16-bit precision gave a good enough spectral estimation result. Based on these facts, this section proposes available hardware architecture of EVD processor and designs considering the implementation on FPGAs.

Figure 8 shows the computational flow of Cyclic Jacobi EVD. As described before, Jacobi type EVD is very simple, just a sequence of vector rotations until achieving convergence. From Eq. (8) the optimal rotation angle is determined and then the processor performs the similar transform of correlation matrix $\boldsymbol{R}$ and unitary matrix $\boldsymbol{E}$(initial value is identity matrix $\boldsymbol{I}$) of eigenvectors. After 4 Jacobi sweeps the computation complete, the resulting matrix $\boldsymbol{R}$ converges the diagonal matrix of eigenvalues and $\boldsymbol{E}$ becomes the unitary matrix of eigenvectors.

The EVD processor consists of CORDIC matrix rotators and CORDIC arctangent. With the optimum angle obtained by Eq. (8), the rotation $\boldsymbol{W}_{pq}$ is determined as Eq. (2). The transform $\boldsymbol{W}_{pq}$ in Eq. (3) changes only p-th and q-th rows and columns of the matrix $\boldsymbol{R}$. Therefore the transform can be simplified as

$$\begin{pmatrix} \boldsymbol{r}'_p \\ \boldsymbol{r}'_q \end{pmatrix} = \boldsymbol{w}_{pq}^T \cdot \begin{pmatrix} \boldsymbol{r}_p \\ \boldsymbol{r}_q \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} r_{p1} & \cdots & r_{pp} & \cdots & r_{pq} & \cdots & r_{pN} \\ r_{q1} & \cdots & r_{qp} & \cdots & r_{qq} & \cdots & r_{qN} \end{pmatrix}, \quad (19)$$

where $\boldsymbol{r}_k$ and $\boldsymbol{w}_{pq}$ denote the k-th row vector of the matrix $\boldsymbol{R}$ and the (p, q) plane rotation that is a submatrix of $\boldsymbol{W}_{pq}$ in Eq. (2), respectively. Thanks to the symmetry of the matrix $\boldsymbol{R}$, the right side transform in Eq. (3) yields the same result, so it is not necessary any more if the second rotations of only two vectors $[r_{pp}\ r_{qp}]^T$ and $[r_{pq}\ r_{qq}]^T$ are performed one more time. The diagram of EVD processor is illustrated in Fig. 9, where the ESB (Embedded System Block) is a memory block of an FPGA and stores the correlation matrix and eigenvector matrix. In the CORDIC matrix rotator,

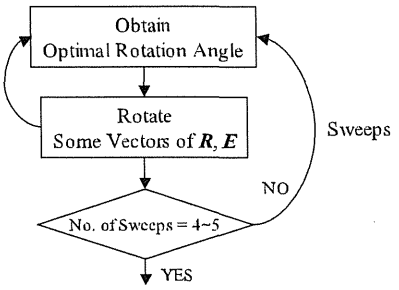


Fig. 8 Computation flow of cyclic Jacobi EVD processor.

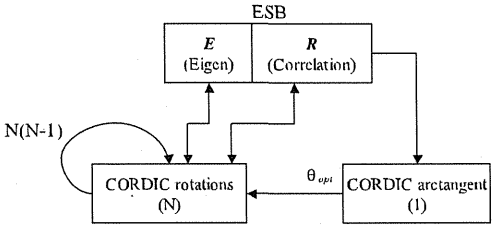


Fig. 9 Block diagram of EVD processor.

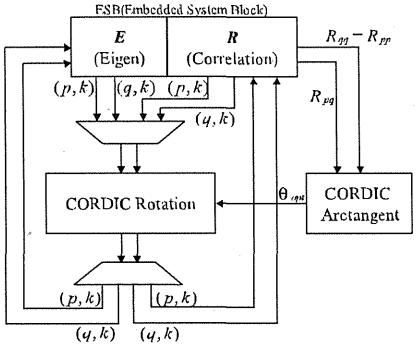


Fig. 10 Architecture of EVD processor core.

using dedicated circuit's parallelism multiple CORDIC vector rotators are performed simultaneously. It takes $N(N-1)/2$ matrix rotations per one sweep to compute $\boldsymbol{R}$ and $\boldsymbol{E}$ respectively. Figure 10 illustrates the architecture of the EVD processor core.

7. Computational Load and Expected Performance

Basic arithmetic operation of EVD processor consists of only shifts and adds. In twice executing rotation CORDIC stages, the computational load yields

$$\left(4B + \frac{1}{4}B\right) \text{ Shifts and } \left(5B + \frac{1}{4}B\right) \text{ Adds}, \quad (20)$$

where B is the bit length and $1/4B$ is approximately taken for scaling operation from Fig. 3 and Eq. (17). The number of vector rotations required computing both eigenvalues and eigenvectors is $4N(N-1)(N+2)$.

It is because $\boldsymbol{E}$ performs 4 $\boldsymbol{J}$'s, $\boldsymbol{J}$ consists of $N(N-1)/2$ $\boldsymbol{W}$'s and $\boldsymbol{W}$ requires $(N+2)$ vector rotations from Eq. (1), where N is the double number of the length of array antenna (if the length of antenna array is N, $N \times N$ correlation matrix of complex numbers is converted into the extended form of $2N \times 2N$ matrix of only real numbers [2]). By addition of arctangent, therefore total computational load of CORDIC-Jacobi EVD processor is $\{4N(N-1)(N+2)+1\} \times (\frac{17}{4}B$ Shifts $+ \frac{21}{4}B$ Adds).

On the other hand, configuring this computational load with the parallel architecture, rough estimate of circuit scale is about $(N+1) \times 15K$ equivalent gates (one 16 bit-CORDIC processor could be synthesized by about $15K$ equivalent gates). For example, if $N = 8$ (4-element array antenna, no spatial smoothing with any subelements) the total circuit scale is about $75K$ equivalent gates. We get the estimate result by synthesizing the circuit described by VHDL(Very high speed integrated circuit Hardware Description Language) with LeonardoSpectrum, Examplar Logic, where the target device is ALTERA's FPGA, APEX20KC.

If this computation load is configured by the parallel circuit architecture as Fig. 9 and it takes one clock cycle for shift or add operation (it is the worst case), the first EVD result requires $[4 \times N(N-1) \times 2+1] \times (B+1)$ clock cycles. For example, if $N = 6$, $B = 16$ (in case of 4-element array antenna and spatial smoothing with 3 subelements) and operated at the speed of 100 MHz, this system can compute EVD computations about $13,200$ times/sec ($75.8\,\mu$s/EVD). But this is just an example. Actually, the pipeline processing by placing register properly can achieve higher performance.

Considering high-speed mobility under the higher frequency area of the next generation communication, it is very difficult to realize the required performance for fading free system with general purposed processor. If DOA estimation of incident waves and beamforming toward their directions are complete within that period, fading free system can be realized. This proposed EVD processor is a combinatorial logic circuit, and hence the improvement of the performance by pipeline scheduled processing can be expected. At present, it is said that the advance of circuit technology can offer high speed operation of general combinatorial logic circuit. The EVD is the most dominant process in the whole processing load from DOA finding to beamforming. This fast parallel computation processor can provide efficient use.

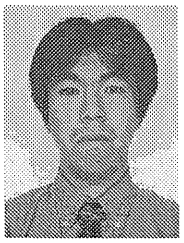
8. Conclusion

In this paper the circuit design of fast EVD computation processor for MUSIC DOA estimator was proposed. It uses CORDIC based Jacobi method and it is suitable for hardware implementation for realtime processing. Taking the practical uses in wireless communication into consideration, it is desired that arithmetic processor should perform fixed-point operation with appropriate precision below 16-bit. Adopting fixed-point arithmetic causes some error but makes the implementation easy and hence the high performance and low power consumption can be achieved. In addition, the functionality for the application of spectral MUSIC method was validated.

References

[1] R.O. Schmidt, "Multiple emitter location and signal parameter estimation," IEEE Trans. Antennas & Propag., vol.AP-34, no.2, pp.276–280, March 1986.
[2] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, "Numerical Recipes in C," in The Art of Scientific Computing, Second ed., Cambridge University Press, 1992.
[3] J. Gotze, S. Paul, and M. Sauer, "An efficient Jacobi-like algorithm for parallel eigenvalue computation," IEEE Trans. Comput., vol.42, no.9, pp.1058–1065, Sept. 1993.
[4] J.M. Muller, Elementary Functions: Algorithms and Implementation, Birkhauser, 1997.
[5] R. Andraka, "A survey of CORDIC algorithms for FPGA based computers," Proc. 1998 ACM/SIGDA Sixth International Symposium on Field Programmable Gate Arrays, pp.192–200, Monterey, CA, Feb. 1998.

Minseok Kim was born in Seoul, Korea on June 16, 1973. He received the B.S. degree in Electrical Engineering from Hanyang University, Seoul, Korea, M.E. degree in Division of Electrical and Computer Engineering, Yokohama National University, Japan in 1999 and 2002, respectively. He is currently working toward Ph.D. degree from Yokohama National University. His research interests are implementation of adaptive antennas and software defined radio.



Koichi Ichige received the B.E., M.E. and Dr.Eng. degrees in electronics and information engineering from the University of Tsukuba in 1994, 1996 and 1999, respectively. In 1999, he joined the Department of Electrical and Computer Engineering, Yokohama National University as a research associate, where he is currently an assistant professor. In 2001–2002, he has been on leave to School of Engineering, Swiss Federal Institute of Technology Lausanne (EPFL), Switzerland as a visiting researcher. His research interests include digital signal processing, image processing, approximation theory, communication and electromagnetic theory. He is a member of IEEE.

Hiroyuki Arai received the B.S. degree in Electrical and Electronic Engineering, M.E. and D.E. in Physical Electronics from Tokyo Institute of Technology in 1982, 1984 and 1987, respectively. After a research associate in Tokyo Institute of Technology, he joined Yokohama National University as a lecturer in 1989. Now he is an associate professor in Division of Electrical and Computer Engineering, Yokohama National University. He investigated microwave passive components for high power handling applications such as RF plasma heating in large Tokamaks. He developed a flat diversity base station antenna for Japanese PDC systems, and small base station antennas of in-building micro cellular system. He was awarded the "Meritorious Award on Radio" by the Association of Radio Industries and Businesses in 1997 for the development of polarization diversity antenna. He is collaborating with a large number of companies for mobile terminal antennas, cellular base station antennas, antenna measurement techniques, indoor/outdoor propagation measurement and simulation, and EMC measurements and wave absorbers. He published more than 50 reviewed journal papers and about four hundreds international and domestic conference papers. He is the co-author of four textbooks about electromagnetic waves, and he holds several US and Japanese patents of antenna.

LETTER

A Fast Algebraic Approach to the Eigenproblems of Correlation Matrices in DOA Estimation

Koichi ICHIGE†, *Member*, Masashi SHINAGAWA†, *Student Member*, *and* Hiroyuki ARAI†, *Member*

SUMMARY This paper studies on a fast approach for the eigenproblems of correlation matrices used in direction-of-arrival (DOA) estimation algorithms, especially for the case that the number of arriving waves is a few. The eigenvalues and the corresponding eigenvectors can be obtained in a very short time by the algebraic solvent of up to quartic polynomials. We also confirm that the present approach does not make the accuracy worse when it is implemented by finite word-length processors like digital signal processor (DSP) or field programmable gate array (FPGA).

key words: *adaptive array antenna, DOA estimation, eigenproblems, MUSIC method*

1. Introduction

Adaptive array antenna plays an important role in Direction-Of-Arrival (DOA) estimation and in discriminating a target wave from interferences by digital-beamforming. Many DOA estimation algorithms have been already proposed [1]–[3] and nowadays MUSIC [1] and ESPRIT [2] would be two representative superresolution algorithms (dealing with correlation matrices) that can estimate DOAs very accurately. The problem of those superresolution algorithms may be the computational complexity, especially the most time-consuming process in DOA estimation is computing eigenvalues/eigenvectors of correlation matrices. Generally the eigenproblems are solved by the combination of QR and Householder decompositions [4]. This approach has been highly-developed in the sense of both mathematics and computer-programming, however the application of such decompositions usually takes time even for small matrices.

Recalling that the derivation of eigenvalues is equivalent to solve the characteristic polynomial of the correlation matrix, the algebraic solvent can be applied if the order of the polynomial is four or less. This corresponds to the situation of adaptive array antenna with four or less antenna elements (also valid when using the space averaging technique for every four or less elements). The case of four elements is worth considering as a low-cost system and already used in practical mobile system: there have been studies on four elements array which assume to be used for mobile terminals [5] and for the base stations of Personal Handyphone System (PHS) [6],[7]. Reducing computational cost would enhance the performance of such system.

This paper first revisits the algebraic algorithm to have eigenvalues/eigenvectors of correlation matrices, supposed to be used in fast DOA estimation for a few arriving waves. And then we evaluate the algebraic approach in comparison with the general QR method. The present algorithm employs the well-known algebraic solvent of quartic characteristic polynomials to derive eigenvalues of correlation matrices instead of the numerical QR decomposition [8]. One may say that the algebraic approach makes the accuracy worse when it is implemented by a digital device due to the fixed-point quantization and the cancellation of significant digits. However, we guarantee that the quantization error in the algebraic approach does not affect to the estimated DOA when it is implemented by finite word-length processors like digital signal processor (DSP) or field programmable gate array (FPGA).

2. Preliminaries

This section summarizes the computational procedure of MUSIC method for example to see how the eigenvalues/eigenvectors are used in DOA estimation. Figure 1 illustrates the configuration of a (linear) general adaptive array antenna system with four elements. In Fig.1, θ_i and x_j denote the DOA of i-th wave and the complex amplitude of the received signal by the j-th element, respectively. The computation procedure of MUSIC method can be roughly summarized as the following three steps [1].

[STEP 1] The correlation matrix $\boldsymbol{R}_{xx}$ of the input vector $\boldsymbol{X}$ is obtained by

$$\boldsymbol{R}_{xx} = E\left[\boldsymbol{X}(t)\boldsymbol{X}^H(t)\right] = \boldsymbol{A}\boldsymbol{S}\boldsymbol{A}^H + \sigma^2\boldsymbol{I},$$

where $\boldsymbol{A}$ and $\boldsymbol{S}$ are so-called the direction and the signal correlation matrices, respectively. Note that the input vector $\boldsymbol{X}$ is formed by using the complex amplitude x_j, and the direction vector $\boldsymbol{A}$ is determined from the physical relationship between elements [1].

[STEP 2] Derive the eigenvalues λ_i and the corresponding eigenvectors $\boldsymbol{y}_i$ of the correlation matrix $\boldsymbol{R}_{xx}$.

Manuscript received August 1, 2002.
Manuscript revised September 1, 2002.

†The authors are with the Department of Electrical and Computer Engineering, Yokohama National University, Yokohama-shi, 240-8501 Japan.

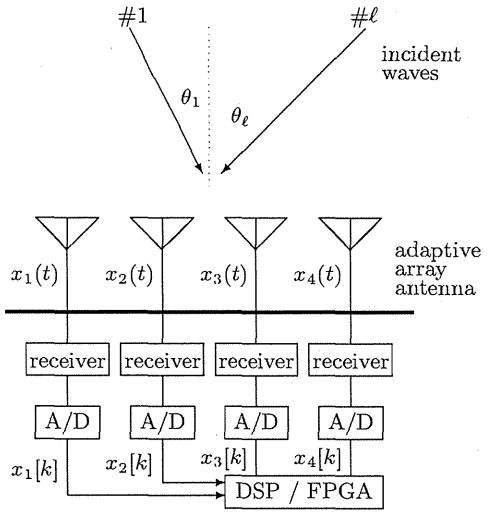


Fig. 1 Configuration of an adaptive array antenna system with four elements

[STEP 3] Using the direction matrix $\boldsymbol{A}$ and the eigenvectors $\boldsymbol{y}_i$, calculate the MUSIC spectrum $\boldsymbol{P}(\theta)$.

In MUSIC method, the time-consuming process is usually STEP 2 and STEP 3. Some fast algorithms have been already proposed for STEP 3, the representative one would be Root-MUSIC method [3] which is an algebraic algorithm without using the direction search technique. Similarly, in the next section, we aim at developing an algebraic approach for STEP 2 to make this process faster.

3. Algebraic Approach to Eigenproblems

A fast algebraic algorithm is investigated for the DOA estimation by array antenna with four elements. The eigenvalues λ_i of the matrix $\boldsymbol{R}_{xx}$ can be obtained by solving the characteristic equation

$$\det(\boldsymbol{R}_{xx} - \lambda \boldsymbol{I}) = \boldsymbol{0}, \tag{1}$$

where $\boldsymbol{I}$ and $\boldsymbol{0}$ indicate the identity and the zero matrices, respectively. If the size of the matrix $\boldsymbol{R}_{xx}$ is $n \times n$, the characteristic polynomial becomes n-th order of λ, and the equation (1) has n solutions. Here we adopt the well-known mathematical technique that the polynomials of up to quartic can be solved by an algebraic procedure [4]. For DOA estimation, we establish an algebraic approach to derive eigenvalues of correlation matrices in case of four antenna elements, also when using a space averaging technique for every four elements.

3.1 Deriving Eigenvalues

The characteristic equation (1) can be reduced into the following quartic equation of λ:

$$\lambda^4 + a_3\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0.$$

Since the matrix $\boldsymbol{R}_{xx}$ is a non-negative Hermitian matrix, it has at most four real eigenvalues [4]. Using the algebraic solvent for quartic equations, the eigenvalues of the matrix $\boldsymbol{R}_{xx}$ can be derived as

$$\lambda = \sqrt{t} \pm \sqrt{h_1 \pm 2h_2} - \frac{a_3}{4}, \tag{2}$$

where

$$\begin{aligned} h_1 &= -t - \frac{p}{2}, \quad h_2 = -\frac{q}{8\sqrt{t}}, \\ p &= -\frac{3a_3^2}{8} + a_2, \quad q = \frac{a_3^3}{8} - \frac{a_3 a_2}{2} + a_1, \\ t &= \sqrt[3]{\alpha} + \sqrt[3]{\beta} - \frac{p}{6}. \end{aligned} \tag{3}$$

In (3), the parameters α and β are also the functions of a_3, a_2, a_1 and a_0 [9]. Originally the parameters α and β are also calculated by the algebraic procedure [9], but such procedures are not required if t is numerically computed. In the programming language C (also in the languages derived from C), the cubic-root is calculated numerically by Newton-Rapson method [4]. On the other hand, the solution of t in (3) is actually the solvent of the third-order polynomial. Therefore, it is more computationally efficient to derive t numerically, not solving (3) exactly.

3.2 Deriving Eigenvectors

Now we derive the eigenvectors $\boldsymbol{y}_i = [y_{i1}, y_{i2}, y_{i3}, y_{i4}]^T$ corresponding to the eigenvalues λ_i in (2). Define the matrix $\boldsymbol{D}$ as

$$\boldsymbol{D} = [d_{ij}] := \boldsymbol{R}_{xx} - \lambda_i \boldsymbol{I} \in \boldsymbol{C}^{4\times 4},$$

then the eigenvector $\boldsymbol{y}_i$ of the matrix $\boldsymbol{D}$ corresponding to the i-th eigenvalue can be obtained by solving the following matrix equation:

$$(\boldsymbol{R}_{xx} - \lambda_i \boldsymbol{I})\boldsymbol{y}_i = \boldsymbol{D}\boldsymbol{y}_i = \boldsymbol{0}. \tag{4}$$

Equation (4) is generally solved by Gauss elimination method [4], however it is redundant since the matrix $\boldsymbol{D}$ is singular in this case. Instead, we employ a faster algorithm using an inverse matrix of the sub-matrix of $\boldsymbol{D}$.

Since $\boldsymbol{D}$ is a singular matrix, the row-vectors $\boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_3$ and $\boldsymbol{d}_4$ of the matrix $\boldsymbol{D}$ are linearly dependent. On the other hand, those four vectors include three linearly independent vectors. Suppose that $y_{i4} = 1$ and the vectors $\boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_3$ are linearly independent. Now (4) is rewritten into the following equation:

$$\boldsymbol{D}_{33}\tilde{\boldsymbol{y}}_i = -\begin{bmatrix} d_{14} \\ d_{24} \\ d_{34} \end{bmatrix}, \tag{5}$$

where

$$\boldsymbol{D}_{33} = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{12} & d_{22} & d_{23} \\ d_{13} & d_{23} & d_{33} \end{bmatrix},$$
$$\tilde{\boldsymbol{y}}_i = [y_{i1}, y_{i2}, y_{i3}]^T.$$

Since the inverse of 3x3 matrices are easily derived using cofactors, (5) can be rewritten as

$$\tilde{\boldsymbol{y}}_i = -\boldsymbol{D}_{33}^{-1} \begin{bmatrix} d_{14} \\ d_{24} \\ d_{34} \end{bmatrix}$$
$$= -\frac{1}{\det \boldsymbol{D}_{33}} \begin{bmatrix} d'_{11} & -d'_{21} & d'_{31} \\ -d'_{12} & d'_{22} & -d'_{32} \\ d'_{13} & -d'_{23} & d'_{33} \end{bmatrix} \begin{bmatrix} d_{14} \\ d_{24} \\ d_{34} \end{bmatrix}$$

where d'_{ij} denotes the cofactor of d_{ij}. Therefore, the eigenvector $\boldsymbol{y}_i$ can be obtained as

$$\boldsymbol{y_i} = [\tilde{\boldsymbol{y}}_i, 1]^T = [y_{i1}, y_{i2}, y_{i3}, 1]^T.$$

Note that the above method cannot derive eigenvectors if the vectors $\boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_3$ are linearly dependent. In this case, the vectors $\boldsymbol{d}_1, \boldsymbol{d}_2$ and $\boldsymbol{d}_4$ become linearly independent instead. Denoting y_{i3}=1, the eigenvectors can be derived similarly to the above inverse matrix approach.

4. Evaluation of the Algebraic Approach in DOA Estimation

In this section, the algebraic algorithm is evaluated in comparison with the general numerical algorithms by computer simulation.

4.1 Evaluation of the Inverse Matrix Approach

First, the inverse matrix method for deriving eigenvectors is evaluated in comparison with general Gauss elimination method. Table 1 shows the computation time of those two methods. We sat from Table 1 that the inverse matrix approach can reduce the computational cost that required in the Gauss elimination method, which means the inverse matrix approach can be more effective. Note that we tested thousands of example matrices, and the result in the Table 1 is the average of those trials.

Table 1 Computation time required in the inverse matrix approach and that in the Gauss method [4]

	Gauss method	Inverse matrix
computation time (ratio)	22.4μsec (1.000)	17.2μsec (0.768)
required memories	560KB	556KB

(CPU: Intel Celeron 433MHz, Memory:256MB)

4.2 Evaluation of the Quartic Polynomial Approach

Next, the quartic polynomial approach is evaluated in comparison with the general QR decomposition. Table 2 shows the computation time of the two methods, quartic polynomial and QR decomposition. From Table 2, the polynomial approach reduces the computational cost to only 1.74% of that required in the QR decomposition. This says the polynomial approach can greatly shorten the computation time for eigenproblems. Note that we tested thousands of example input vectors, and the result in the Table 2 is the average of those trials.

The comparison of computational time might be ambiguous but gives an evaluation from a certain sense. More objective comparison would be desired however it deeply depends on the characteristics of implemented devices. it should be studied provided that the algorithms are implemented on a real device.

Table 2 Computation time required in the algebraic approach and that in the QR decomposition [4]

	QR decomposition	Algebraic
computation time (ratio)	1.54msec (1.000)	24.2μsec (0.0174)
required memories	560KB	572KB

(CPU: Intel Celeron 433MHz, Memory:256MB)

5. Discussion

Suppose that the present algorithm is implemented on a finite word-length (digital) device. We confirm in this section that the quantization noise caused by DSP/FPGA implementation does not make the DOA estimation accuracy worse. Specifications of the simulation in Sections 5 are summarized in Table 3.

Table 3 Specifications of the simulation

antenna form	4-elements linear array
element interval	half wavelength each
incident wave	2 waves (from 0 / -30 degree)
power of waves	one for each
SNR	20dB (10dB and 5dB in Fig.7)
Number of snapshots	300 times

5.1 Effect of the Quantization Noise in A/D Converter

First, the effect of the quantization noise in A/D converter is studied. The A/D converter and its quantization noise are imperative in digital implementation. Here we assume that several waves with power one are coming to the array antenna from certain directions.

In this case, the distribution of the input voltage became as illustrated in Fig.2. From Fig.2, we see that the input voltage usually vibrates within -2 to 2 volts, and we can adjust the level of the A/D converter not to make overflow. In fact, it did not matter if we round the values more than 2 to 2.

Figure 3 illustrates the effect of quantization noise in A/D converter to the estimated DOAs. As seen in Fig.3, eight quantization bit length in A/D converter is enough for the accurate DOA estimation. Although the noise level in Fig.3 becomes slightly larger, it does not affect to the accuracy of estimated DOAs.

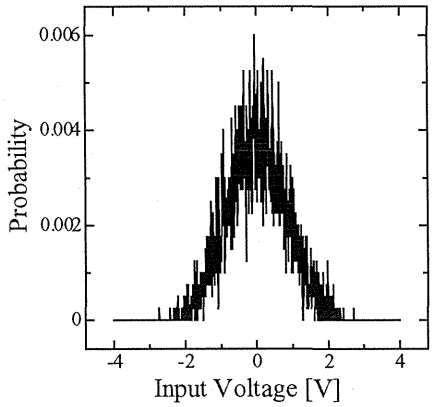


Fig. 2 Distribution of Input voltage

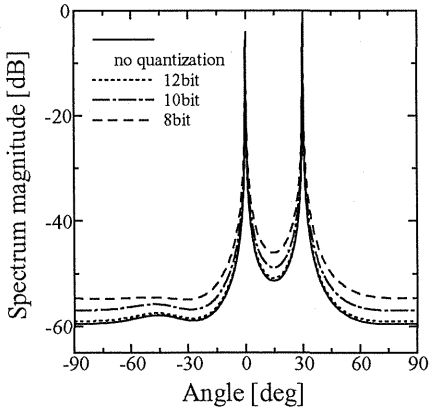


Fig. 3 Effects of quantization errors in A/D converter to the estimated DOAs

5.2 Effect of the Quantization Noise in DSP

Next, we study the effect of the quantization noise in DSP. In the algebraic approach, we should confirm the range of parameters in the middle of digital processing so that not to make overflow and the cancellation of significant digits. Figure 4 depicts the example distribution of intermediate parameters. Fig.4 says the parameters vibrates within about 10-times of the range of the input voltage. Generally the operations in the fixed-point DSP are performed with 16-bits (single precision) or 32-bits (double precision). Considering the range of parameters in Fig.4, 16-bits operation would be enough for the accurate DOA estimation.

Figure 5 shows the effect of the quantization noise in DSP for algebraic approach. The quantization bit length in A/D converter is set to be eight. Comparing with the same effect for QR method illustrated in Fig.6, we confirm that the noise-level of MUSIC spectrum is almost same, and both of the methods can preserve the accuracy of the estimated DOAs.

Moreover, in the case of low-SNRs as seen in Fig.7 for algebraic approach, the noise-level of MUSIC spectrum becomes higher but it finally doesn't affect to the accuracy of estimated DOAs. This guarantees that the algebraic approach can preserve high accuracy in the case of low-SNR like open-air.

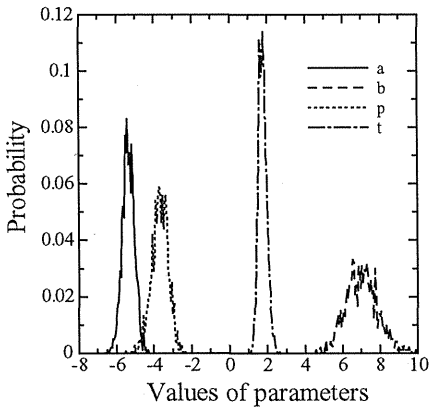


Fig. 4 Distribution of example intermediate parameters

6. Concluding Remarks

This paper studied on the fast approach to eigenproblems in DOA estimation algorithms for the case that the number of array elements is four or less. Regarding the eigenproblem as solving the quartic algebraic polynomial, the eigenvalues and eigenvectors could be obtained in a very short time. Moreover, we confirmed that the algebraic approach does not make the accuracy worse when it was implemented by finite word-length processors. This kind of approach would also be effective in ESPRIT [2] which requires two or three eigendecompositions. Besides, we should expand the algebraic procedure to correspond to matrices with higher orders, as one of future studies.

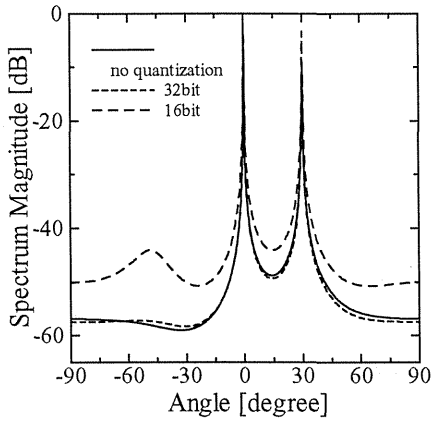


Fig. 5 Effects of quantization errors in DSP to estimated DOAs by the algebraic approach (SNR = 20dB)

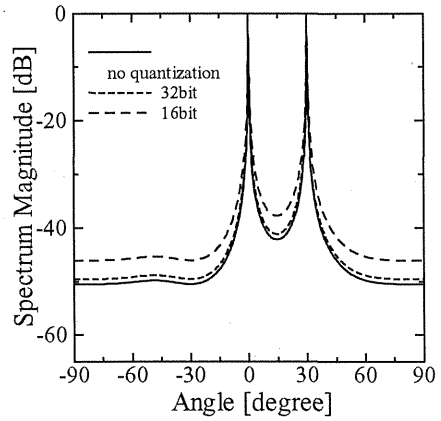
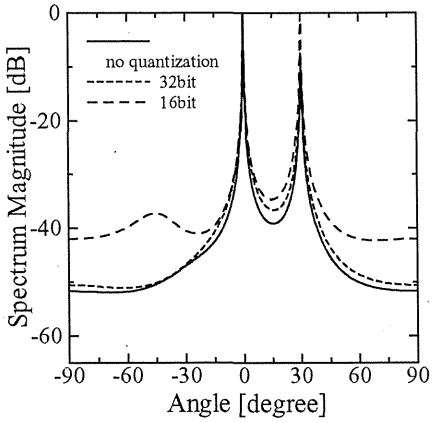
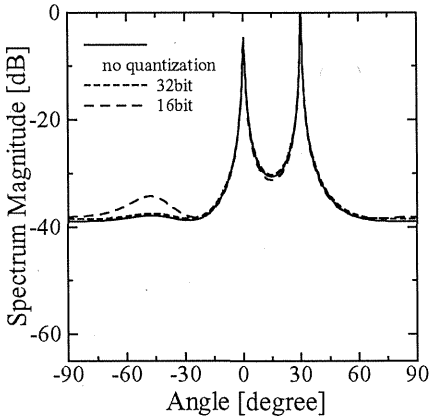


Fig. 6 Effects of quantization errors in DSP to estimated DOAs by the QR method (SNR = 20dB)



(a)



(b)

Fig. 7 Effects of the low-SNR to estimated DOAs (a) SNR = 10dB, (b) SNR = 5dB.

Acknowledgment

The authors would like to thank the anonymous reviewer's comments which considerably improve the present paper.

References

[1] R.O.Schmidt: "Multiple Emitter Location and Signal Parameter Estimation," IEEE Trans. Antenna and Propagat., vol.34, no.3, pp.276-280, Mar. 1986.

[2] R.Roy and T.Kailath: "ESPRIT—Estimation of Signal Parameters via Rotational Invariance Techniques," IEEE Trans. Accoust. Speech and Signal Proc., vol.37, no.7, pp.984-995, July 1989.

[3] B.D.Rao and K.V.S.Hari: "Performance Analysis of Root-MUSIC," IEEE Trans. Accoust. Speech and Signal Proc., vol.37, no.2, pp.1939-1949, Dec. 1989.

[4] W.H.Press, S.A.Teukolsky, W.T.Vetterling and B.P.Flannery, Numerical Recipes in C: 2nd ed., Cambridge Univ. Press, 1992.

[5] T.Ohira, "Adaptive Array Antenna Beamforming Architectures as Viewed by Microwave Circuit Designer," Proc. Asia-Pacific Microwave Conf., pp.828-833, Sydney, Australia, Dec. 2000.

[6] L.C.Godara ed., Handbook of Antennas in Wireless Communications, CRC Press, 2001.

[7] H.Arai, Measurement of Mobile Antenna Systems, Artech House, 2001.

[8] K.Ichige, M.Shinagawa, H.Arai, "An Algebraic Approach for Deriving Eigenvalues and Eigenvectors of Correlation Matrices toward the fast DOA Estimation," IEICE Tech Report, AP-2001-65, Aug. 2001.

[9] B.L. Van Der Waerden, Algebra: Volume II, Springer-Verlag, 1991.

PAPER Special Issue on Wave Technologies for Wireless and Optical Communications

Implementation of FPGA based Fast Unitary MUSIC DOA Estimator

Minseok KIM†, *Student Member*, Koichi ICHIGE†, *and* Hiroyuki ARAI†, *Members*

SUMMARY DOA (Direction Of Arrival) estimation is a useful technique in various positioning applications including the DOA-based adaptive array antenna for wireless cellular basestation. This paper presents the practical implementation of FPGA (Field Programmable Gate Array) based fast DOA estimator for wireless cellular basestation. This system incorporates spectral unitary MUSIC (MUltiple SIgnal Classification) algorithm, which is one of the representative super resolution DOA estimation techniques. This paper proposes the way of digital signal processor design suitable for an FPGA and its real hardware implementation. In this system the fast computation performance can be achieved by the inherent parallelism of FPGAs, which can process multiple tasks at the same time. The eigenvalue decomposition (EVD) and MUSIC angular spectra generation are solved by Cyclic Jacobi processor based on CORDIC (COordinate Rotation DIgital Computer) and the spatial DFT (Discrete Fourier Transform), respectively. The perfomance will be analyzed by the hardware level simulations and experiments in a radio anechoic chamber. All digital signal processing procedures are computed by the only fixed-point operation with finite wordlength for fast processing and low power consumption.

key words: *Adaptive Antenna, Smart Antenna, FPGA Implementation, DOA Estimation, MUSIC, EVD*

1. Introduction

Adaptive array antennas are expected to be a promising technology in the wireless communication systems at present and in the future. The radio system controlling the radiation pattern of antenna by software adaptively will allow us to overcome a lack of reconfigurability and flexibility in a fixed beam antenna system [1].

DOA (Direction Of Arrival) information of incoming signals is frequently useful in various positioning applications such as an emergency system, a radio surveillance system, a radar system, a millitary system and so forth. Moreover, from the theoretical point of view, many excellent adaptive array antenna techniques for wireless cellular communications also need DOAs of desired signal and interferers in advance [2]. As a matter of fact, we have another choice of the MMSE (Minimizing Mean Square Error) based combining techniques like LMS (Least Mean Square) and RLS (Recursive Least Square). They employ a temporal reference signal instead of explicit DOA informations, while the DOA-based systems exploit the estimated DOAs in beamforming to separate the desired signal from interferers spatially. Such DOA-based systems, however, have many advantages over the conventional temporal reference based solutions. First of them is that they are more applicable to the downlink (forward link) beamforming thanks to the knowledge of the explicit directional information, thus the basestations can steer maximum transmitting power toward the desired user direction. And it is also aready known that DOA-based beamforming has superior SINR (Signal to Interference and Noise Ratio) performance to that of other combining techniques for small angular spread [3]. That may be reasonable in macro cellular suburban environment having far field scatterers. However, the drawback is that it requires time-consuming task of DOA estimation. But the lack of the cost effective digital processing devices to resolve the high computational burden has been a distress in the applications to the practical systems commercially. With the general Von Neumann architecture processors, it may be difficult to meet the requirements of the high speed computation and the compact architecture with low power consumption at the same time in the future communication systems [4].

Although the fast DOA estimator is indispensable for the ideal beamforming with the explicit directional information, we have many difficulties of the actual implementation. In fact, there have been some practical works with dedicated circuits [5]. However the quantitative evaluation of the fixed-point operation of FPGAs including the hardware implementation details have rarely been described. The recent study implements the DOA-based smart antenna for European GSM system [6]. It made use of the unitary ESPRIT (Estimation of Signal Parameters Rotational Invariance Technique) and MVM (Minimum Variance Method) as DOA estimators. However, it used general purposed processor (DEC Alpha 500MHz), which might not be optimized for the dedicated tasks and consume large amount of electrical power.

In this paper, the FPGA based digital signal processor design of the DOA estimator and its hardware implementation will be presented. The inherent parallelism, reconfigurability and optimisability of FPGAs give us more benefits than general purpose processors. The specific circuit implementation of a DOA estimator has not been published yet, thus the design concept and

Manuscript received January 16, 2004.
Manuscript revised April 12, 2004.

†The author is with the Division of Electrical & Computer Engineering Dept., Yokohama National University, Japan

practical implementation in this work will be a valuable example. This system will be useful for high speed DOA-based beamforming in wireless cellular systems. It incorporates unitary MUSIC (MUltiple SIgnal Classification) algorithm, which is a super resolution DOA estimation technique. MUSIC-like algorithm has many advantages in the real hardware implementation due to its simplicity compared with other well-known subspace based techniques like ESPRIT. However, there still remains the computational complication of the complex number arithmetic, which is a great distress to the fast computation and compact system scale. With a unitary transform, the eigendecomposition of the correlation matrix can be solved with real number only [7][8]. The unitary MUSIC processor (call UMP, hereafter) performs all digital signal processing procedures by the only fixed-point operation with finite word-length. The eigenvalue decomposition (EVD) and MUSIC angular spectra generation are solved by Cyclic Jacobi processor based on CORDIC (COordinate Rotation DIgital Computer) [9] and the spatial DFT (Discrete Fourier Transform), respectively.

This paper is organized as follows. In Sect.2, the data model and basic principle of unitary MUSIC DOA estimator will be described. Section 3 will present the digital signal processing concepts and key features. In this section, the fast EVD processor incorporated in UMP will be introduced briefly. In UMP, the DOAs are treated as some number of discrete wavefronts. The computation concept of the discrete angular spectra via spatial DFT will be also described. Sections 4 and 5 will present the real hardware implementation details and the performance analysis, respectively. In Sect.6, whole system operation will be demonstrated by experiments in a radio anechoic chamber. Finally, Sect.7 will conclude this paper.

2. Preliminaries

2.1 Data Model

We assume the basic model of the narrowband signal $s_i(t)$ for i-th source, where $i = 1, 2 \cdots, L$. The signals received at K antenna array spaced by half wavelength can be modeled by

$$\boldsymbol{x}(t) = \boldsymbol{V}\boldsymbol{s}(t) + \boldsymbol{n}(t), \tag{1}$$

where the array output $\boldsymbol{x}(t)$ is a snapshot vector, and $\boldsymbol{s}(t)$ and $\boldsymbol{n}(t)$ are the signal and complex Additive White Gaussian Noise (AWGN) vectors, at time t, respectively. The columns of the channel matrix $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_L]$ consist of the spatial channel vectors for L sources. The spatial channel vector $\boldsymbol{v}_i$ for i-th source can be given by the array response vector $\boldsymbol{a}_i$ under the assumption that the plane waves arrive at an ideal omni-directional antenna array from the point sources as

$$\begin{aligned}\boldsymbol{v}_i &= \boldsymbol{a}(\theta_i) \\ &= \left[1, e^{-j\pi \sin\theta_i}, \cdots, e^{-j\pi(K-1)\sin\theta_i}\right]^T,\end{aligned} \tag{2}$$

where θ_i is the DOA for the i-th source and superscript ^{T} denotes transpose operator.

2.2 Unitary MUSIC DOA Estimation

MUSIC algorithm is a kind of DOA estimation technique based on eigenvalue decomposition, which is also called subspace-based method [10]. It has many advantages of the implementation simplicity as well as the capability of estimating DOA in much higher resolution over any other conventional methods. The correlation matrix of $\boldsymbol{x}(t)$ is given by

$$\boldsymbol{R}_{xx} = E[\boldsymbol{x}(t)\boldsymbol{x}^H(t)] = \boldsymbol{V}\boldsymbol{R}_{ss}\boldsymbol{V}^H + \sigma^2 I, \tag{3}$$

where $E[\cdot]$ and superscript ^{H} denote expectation and Hermitian transpose operators, respectively. And $\boldsymbol{R}_{ss} = E[\boldsymbol{s}(t)\boldsymbol{s}^H(t)]$ is the signal covariance matrix and σ^2 is the noise variance. Since the correlation matrix $\boldsymbol{R}_{xx}$ is a positive definite Hermitian matrix, it can be decomposed to signal and noise subspaces by eigenvalue decomposition. The noise subspace eigenvectors of corresponding eigenvalues of σ^2 are orthogonal to the signal subspace, and eventually orthogonal to the array resoponse vectors. From this fact, the MUSIC spectrum is typically given by

$$P_{MU}(\theta) = \frac{\boldsymbol{a}^H(\theta)\boldsymbol{a}(\theta)}{\boldsymbol{a}^H(\theta)\boldsymbol{E}_n\boldsymbol{E}_n^H\boldsymbol{a}(\theta)}, \tag{4}$$

where $\boldsymbol{E}_n$ is the matrix whose columns consist of noise subspace eigenvectors. In the result spectrum of Eq.(4), the peaks appear at the DOAs of incident signals.

Generally the correlation matrix in Eq.(3) is complex-valued. It is clear that the EVD with complex-valued correlation matrix should be high computational burden. Reducing the computational complexity via unitary transform allows real-valued eigenvalue decomposition with the transformed real number correlation matrix [7]. Since the EVD procedure has a large portion of whole computational load of MUSIC-like subspace based algorithms, real-valued eigenvalue decomposition is attractive. If the steering vector of Eq.(2) is rearranged in the conjugate centro-symmetric manner, the correlation matrix $\boldsymbol{R}_{xx}$ becomes centro-Hermitian. The real-valued correlation matrix $\hat{\boldsymbol{R}}_{xx}$ can be obtained with an appropriate unitary transform $\boldsymbol{Q}$ as

$$\hat{\boldsymbol{R}}_{xx} = \mathrm{Re}\{\boldsymbol{Q}^H\boldsymbol{R}_{xx}\boldsymbol{Q}\}. \tag{5}$$

3. Digital Signal Processor Design Concepts

The computation flow of the unitary MUSIC DOA estimation is involved in 4 steps largely [7][9]; (1) Estimation of correlation matrix including unitary transform

and spatial smoothing, (2) EVD (Eigen Value Decomposition) of the correlation matrix, (3) Computation of MUSIC spectrum and (4) 1-dimensional peak search. In this section, the digital signal processor design concepts for the dominant procedures in the unitary MUSIC DOA estimation algorithm will be described.

3.1 Correlation Matirix with Unitary Transform

In this time, UMP was designed to discriminate somewhat correlated signals from the mixed signals by using the spatial smoothing technique. By the unitary transform, the EVD which takes the largest computational cost of the whole procedures can be solved in real number arithmetic. In addition, with the selection of real part only in Eq.(5) provides FB (Forward-Backward) averaging, thus backward spatial smoothing can be achieved simultaneously [7]. The first step in the unitary MUSIC algorithm is to transform the input data vector $\boldsymbol{x}$ to $\boldsymbol{y}$ with a unitary matrix $\boldsymbol{Q}$ as

$$\boldsymbol{y}_i = \boldsymbol{Q}^H \boldsymbol{x}_i, \tag{6}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ $(i = 1, \cdots, M)$ are the divided M subvectors of the snapshot vector and the corresponding transformed sub-vectors, for spatial smoothing respectively, as

$$\begin{aligned} \boldsymbol{x}_i &= [x_i, \cdots, x_{i+K-M}]^T, \\ \boldsymbol{y}_i &= [y_i, \cdots, y_{i+K-M}]^T. \end{aligned}$$

By the unitary transform, the real-valued correlation matrix $\hat{\boldsymbol{R}}_{yy}$ is computed as Eqs.(3) and (5). In reality, however, it is approximated by the uniform averaging with some number of snapshots sampled at the time nT, where T is a sampling period, as

$$\hat{\boldsymbol{R}}_{yy}(n) = \frac{1}{N} \sum_{\substack{m= \\ n-N+1}}^{n} \sum_{i=1}^{M} \mathrm{Re}\{\bar{\boldsymbol{R}}_{yy,i}(m)\}, \tag{7}$$

where

$$\bar{\boldsymbol{R}}_{yy,i}(m) = \boldsymbol{y}_i(m)\boldsymbol{y}_i^H(m)$$

and N is the number of snapshots. This type of averaging can be implemented by a sliding boxcar FIR (Finite Impulse Response) filter with unit gain. From the view points of hardware implementation, however, this filter needs such a long shift register that memory resources turn out to be exhaustive. Thus the first order IIR (Infinite Impulse Response) exponential averaging filter as shown in Fig.1 is a reasonable choice. It requires single register only. In this figure, the correlation matix can be written by

$$\begin{aligned} \hat{\boldsymbol{R}}_{yy}(n) &= \beta \hat{\boldsymbol{R}}_{yy}(n-1) + \\ &\quad (1-\beta) \sum_{i=1}^{M} \mathrm{Re}\{\boldsymbol{y}_i(n)\boldsymbol{y}_i^H(n)\}, \end{aligned} \tag{8}$$

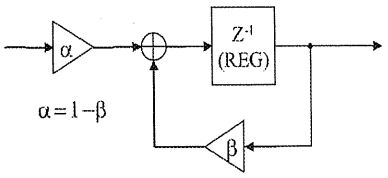


Fig. 1 Diagram of Exponential Averaging Filter

where β (< 1) are a real number forgetting factor. This type of filter can respond to the non-stationary environment quickly if the smoothing factor β is determined appropriately.

3.2 Eigenvalue Decomposition via CORDIC based Jacobi Processor

In the next step, the correlation matrix is eigendecomposed by the EVD processor. In our former work, the EVD processor design for MUSIC-like subspace based techniques has been studied [9]. It used CORDIC based Cyclic Jacobi method suitable for logic circuit implementation with FPGA. Cyclic Jacobi method is well known for the simplest algorithm, but usually its convergence time is slower than other factorization algorithms like QR-method. But if making use of the high parallelism with a dedicated circuit like FPGA or ASIC (Application Specific Integrated Circuit), it would rather be a good choice. In this system, the EVD processor computes real symmetric eigenvalue problems by applying orthonormal rotations sequentially to the target matrix. This system employed the hardware friendly CORDIC algorithm for vector rotators and arctangent computers, which were the basic processing units in this design. As far as the fixed-point operation was applied, there exist the truncation errors caused by the fixed word-length. However, we could confirm that if the computation word-length was longer than 16-bit, it had reasonable performance through the fixed-point computer simulation. In UMP, the number of iterations and the computation word-length of the EVD processor are 4 Jacobi sweeps and 16-bit long, respectively.

3.3 MUSIC Spectrum Computation via Spatial DFT

After the EVD step, the MUSIC spectrum of Eq.(4) is computed. In order to reduce the system complexity, only denominator in Eq.(4) is taken into account. This reciprocal spectrum can be generated from the sum of spatial DFT spectra of $(K - L)$ noise eigenvectors returned to complex values by the inverse unitary transform as

$$P_{MU,reciprocal} = \sum_{i=L+1}^{K} \left|\mathrm{DFT}\{\boldsymbol{Q} \cdot \boldsymbol{E}_i\}\right|^2, \tag{9}$$

where $\boldsymbol{E}_i$ is the i-th eigenvector belonging to the noise subspace.

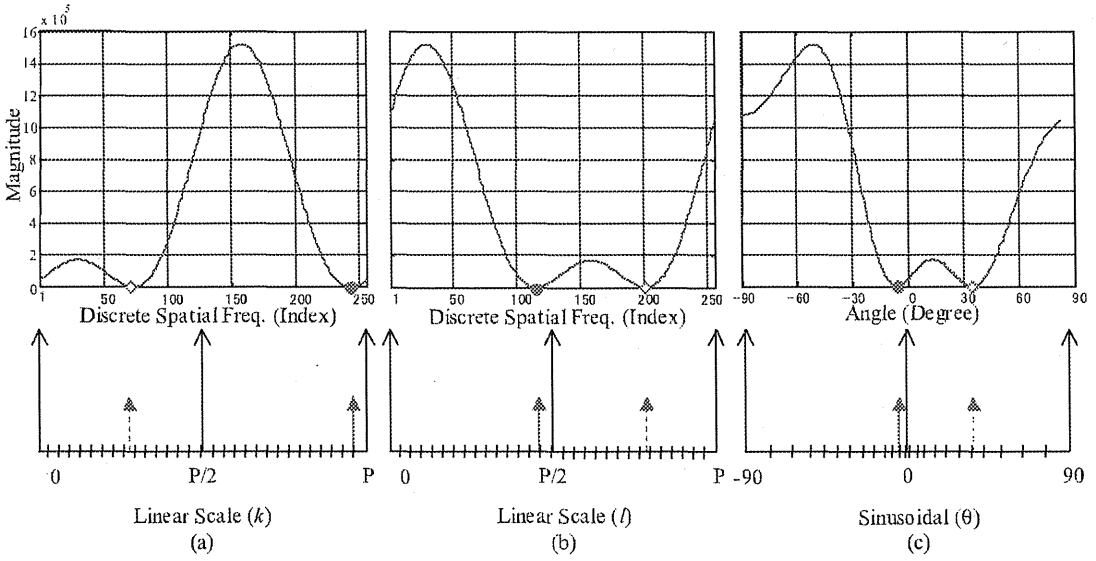


Fig. 2 Relationship between DFT indexes (k and l) and non-uniform discrete wavefront(θ) in reciprocal MUSIC spectrum ($P = 256$)

In specral MUSIC algorithm, in order to find out the DOAs of the incoming signals, the angular spectrum should be computed. In fact, there exist another alternative solutions in MUSIC algorithms. It is based on solving roots of the MUSIC polynomial, called root-MUSIC [8]. However the root finding problem of the complex number coefficient polynomial is less suitable for the dedicated circuit computer with the fixed-point operation like FPGAs. For fast digital signal processor implementation on FPGAs, a simple iterative algorithm must be the best solution. In order to compute the angular MUSIC spectrum, the spatial DFT technique may be an attractive solution due to the well-known performance guarantee as well as the simplicity with FFT algorithm. This subsection will describe how to apply DFT to the computation of the MUSIC angular spectrum.

The simple continuous spatial signal model for 1-dimensional distance is typically given by

$$x_d = u(t) \cdot \exp(-j2\pi \cdot f_{spa} \cdot d), \tag{10}$$

where $u(t)$ includes all complex-valued time-varying components, and d and f_{spa} are the distance from the first reference antenna element and spatial frequency, respectively. In case of array antenna signal processing, we can consider the snapshot vector elements as the sampled data at the distance of $m \cdot D_{spacing}$ in each element of antenna sensors. Actually, however, the distance between each antenna is a discrete value as

$$d \rightarrow m \cdot D_{spacing}, \tag{11}$$

where m and $D_{spacing}$ are the index of discrete distance and the antenna spacing (or spatial period), respectively. But the spatial frequency depending on the

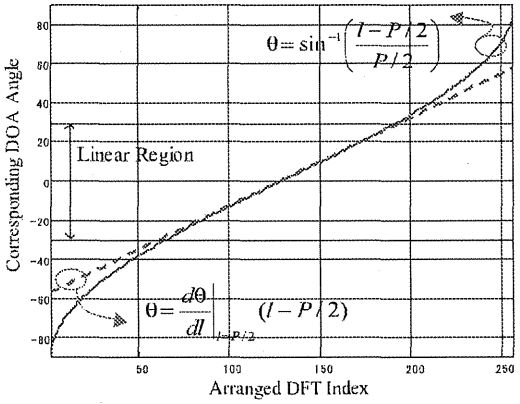


Fig. 3 Non-uniform Discrete Wavefront (DOA) corresponding to DFT index ($P = 256$)

incident DOA still remains continuous value. In order to apply digital signal processing, the continuous spatial frequency should be digitized by P discrete spatial frequencies as

$$2\pi \cdot f_{spa} \rightarrow \frac{2\pi}{P \cdot D_{spacing}} k, \tag{12}$$

where k is the index of the discrete spatial frequency. From this fact, by applying spatial P-point DFT, the discrete spatial frequency domain components can be obtained by

$$X_d[k] = \frac{1}{P} \sum_{m=0}^{P-1} x_d[m] \cdot e^{-j\frac{2\pi}{P} m \cdot k}, \tag{13}$$

When $D_{spacing} = \lambda/2$, eventually the discrete wavefront θ can be computed from the following relationships of Eqs.(14)-(15).

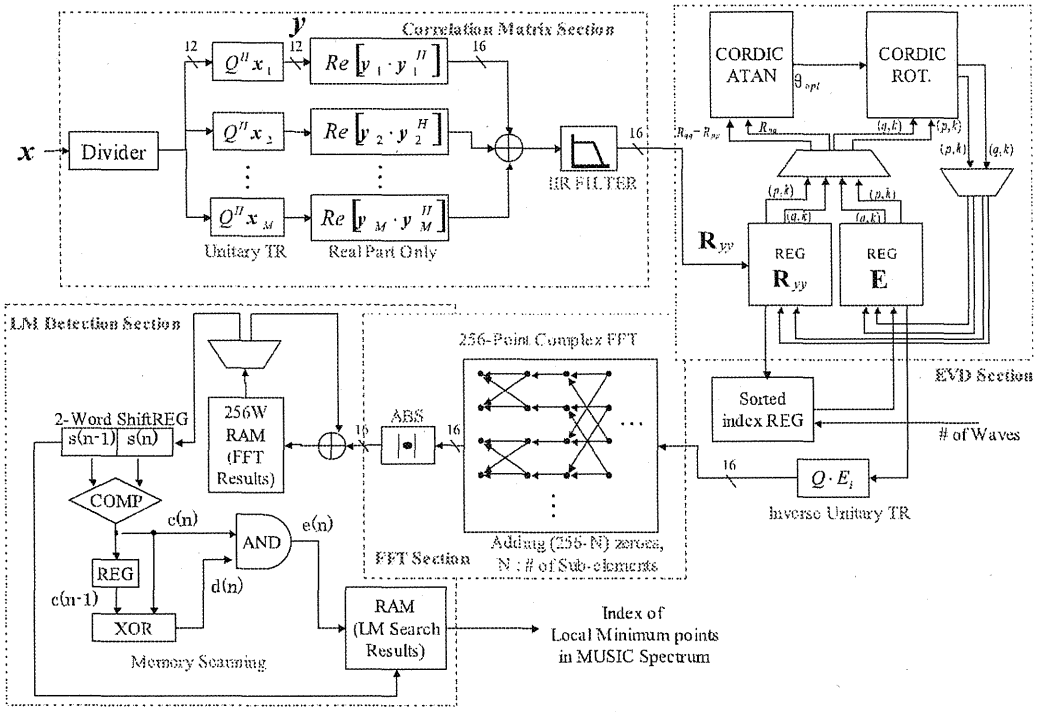


Fig. 4 Diagram of UMP Digital Signal Processor

$$f_{spa,discrete} = \frac{\sin\theta}{\lambda} = \frac{k}{P \cdot D_{spacing}} \tag{14}$$

$$\theta = \sin^{-1}\left(\frac{k}{P/2}\right) \tag{15}$$

From above facts, in the MUSIC algorithm, it can be seen that the spatial DFT of the noise subspace eigenvectors in Eq.(9) gives the distribution of the spatial frequency. However the spatial spectrum obtained by DFT of only a few spatial data as many as antenna array length has coarse resolution not to estimate ccurately. In reality, the antenna array length is limitted for several reasons on hardware implementation. Thus the high resolution estimation will not be available from the coarse spectrum. In that regards, the interpolation of the spectrum should be taken into consideration. According to digital signal processing theory, the DFT spectrum can be generated fine and smoothly by adding a few number of zeroes to the spatial data of the noise eigenvector elements. The spectrum generated by the spatial DFT is completely equivalent to the denominator of Eq.(4) which means scanning main-beam toward whole directions.

Figure 2 shows an example of the relationship between the DFT index k, rearranged index l and corresponding discrete wavefront θ in the reciprocal MUSIC spectrum (denominator of Eq.(5)) generated by spatial DFT (DFT bin $P = 256$). That is, Fig.2(b) is rearranged version of Fig.2(a) (actual order stored in a memery) by the relation as

$$l = \begin{cases} k + P/2 & (k < P/2) \\ k - P/2 & (k \geq P/2) \end{cases} \tag{16}$$

From Eqs.(15) and (16), the concrete discrete wavefronts (DOAs) in Fig.2(c) are obtained as

$$\theta = \sin^{-1}\left(\frac{l - P/2}{P/2}\right). \tag{17}$$

As shown in Fig.2(c), the discrete wavefronts generated by the spatial DFT spectrum is not uniformly spaced. Thus the estimation resolution becomes lower when the signals arrive from close to endfire direction, since the angular spacing gets wider. That, however, may be of no concern in the practical sectorized basestation configuration. From Eq.(17), θ is an inverse sinusoidal function of l. Figure 3 shows the effect of non-uniform discrete wavefront. In the region between -30 to 30 degrees, the estimation resolution can be approximated by linear function whose gradient is given by the derivative of Eq.(17) at $l = P/2$ (broadside) as

$$\left.\frac{d\theta}{dl}\right|_{l=P/2} = \frac{1}{P/2}[\text{rad}]. \tag{18}$$

In this linear region, the estimation resolution (angular spacing) can be regarded as almost uniform. Futhermore, the estimation resolution becomes higher with larger P of DFT bin length, because that is inversely proportional to P. When P is 256, the estimation resolution is about 0.4476 degrees from Eq.(18).

Table 1 Core Performances of Dominant Procedures

	Required Clocks(M_{clk})	LEs(Logic Elements)	f_{max} (MHz)	t_{min} (μs)
$\hat{R}_{yy}$	N	8301	27.4	$0.04 \times N$
EVD	1836	4045	110	16.69
FFT & LM Det.	1356	2303	114	11.89

3.4 Local Minima Detection

Instead of finding peaks in the MUSIC spectrum written in Eq.(4), local minima (LM) detection of the reciprocal MUSIC spectrum generated by DFT in Eq.(9) can be applied for the implementation simplicity. The LM detection can be implemented by memory scanning circuit consisting of 2-word shift register, comparator and logic gates of XOR and AND as shown in LM detection section of Fig.4. The comparator compares 2 words loaded in the shift register to output the 1-bit decision result $c(n)$, which means the sign of derivative between at the indices $(n-1)$ and n. And then, the AND($\oplus$) output of $c(n)$ and XOR($\otimes$) output $d(n)$ notifies the the memory writting controller of the transition of $c(n)$ $(0 \rightarrow 1)$ as

$$\begin{aligned} c(n) &= \begin{cases} 0 & \text{if } s(n-1) > s(n) \\ 1 & \text{otherwise} \end{cases} \\ d(n) &= c(n-1) \otimes c(n) \\ e(n) &= c(n) \oplus d(n). \end{aligned} \tag{19}$$

This procedure can be implemented by simple logic circuit with high speed operation.

4. Practical Hardware Implementation

Not only the theoretical study, we also tried to implement it on single FPGA (STRATIX EP1S25, Altera) which has about 0.6 million equivalent gates, 200 Kbytes internal memory blocks and optimised digital signal processing (DSP) blocks [11]. The whole block diagram of the digital signal processing procedures described in previous section is shown in Fig.4. It is involved in 4 major procedure sections including Correlation Matrix Section, EVD Section, FFT Section and LM Detection Section. The word-length of every section is also shown in this figure. In this time, it is assumed that the exact number of waves were predetermined and known already from any other preprocessing. We described the procedures with VHDL (Very high speed integrated circuits Hardware Description Language). The VHDL sources were synthesized and place-and-routed by Leonardospectrum (Mentor Graphics) and Quartus II (Altera), respectively.

Table 1 illustrates the roughly estimated performance of the dominant processing core, where LEs (Logic Elements) means the number of occupied logic

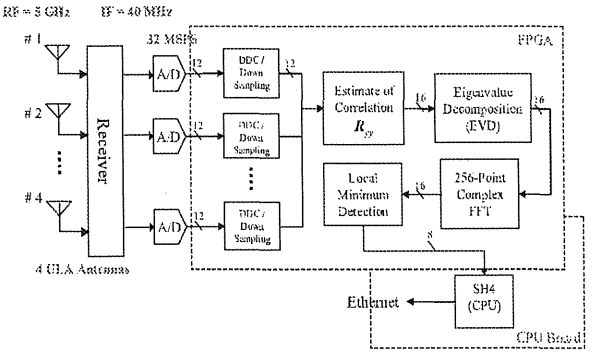


Fig. 5 Whole Evaluation System Configuration

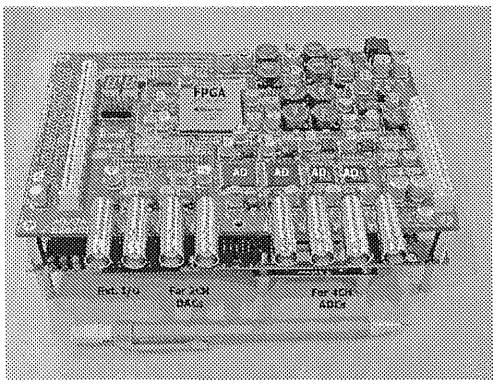


Fig. 6 Appearance of UMP Digital Processing Unit

blocks in FPGAs and f_{max} is the maximum clock frequency at which normal operation can be guaranteed, respectively. The minimum computation time t_{min} is calculated by M_{clk}/f_{max} and N is the number of snapshots. We assumed that less than 2 coherent/incoherent waves arrive at only 4-element uniform linear array (ULA) antenna in order to reduce system complexity. For spectrum generation, 256-point radix-4 complex FFT was employed [12], and the FFT with 256 spatial data composed of 3 elements of the noise eigenvector (1 dimension of array is used for spatial smoothing) and $(256-3)$ zeroes interpolates the spectrum fine and smoothly. All computations were performed by fixed-point arithmetic with 12-bit input data from ADCs. As shown in Table 1, EVD took the longest computational time if the FFT was performed only one time (in case of two incident waves). The pipeline scheduling can allow us to divide the whole processing into some sub-blocks. That is, if considering 3 sub-blocks as illustrated in Table 1, this system can perform single realization of DOA estimation within about 17 μs. In this result, the inherent parallelism and optimisability of FPGAs provided the fast computation performance. Additionally, the recent FPGA manufacturing technology allowed the electric power consumption to be dramatically low by about 2 watts below. Figures 5

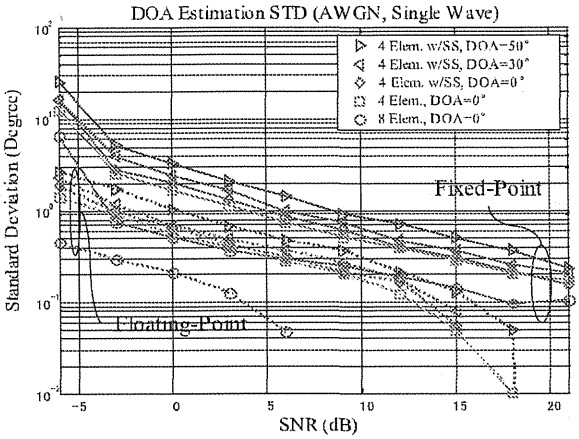


Fig. 7 Standard Deviation of Estimated DOA when single wave impinging at 0, 30 and 50 degrees from broadside

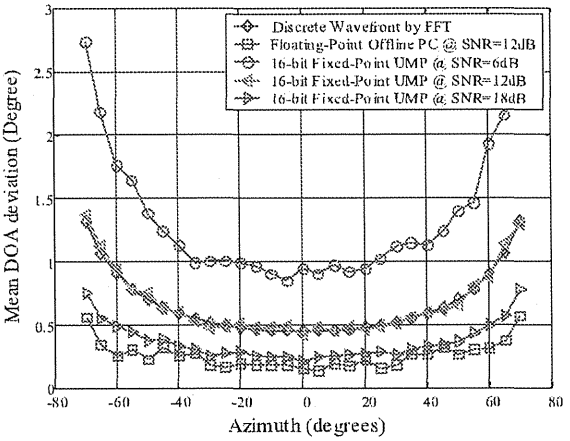


Fig. 8 DOA Dependency of Estimation Performance

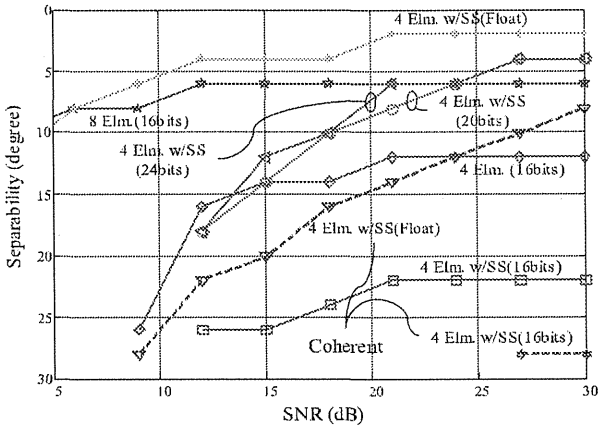
and 6 show the whole evaluation system configuration and its appearance, respectively. The system architecture is a super-heterodyne IF sampling receiver with quasi-coherent detection. RF (Radio Frequency) signals received at antenna array are down-converted to IF (Intermediate Frequency) band in analog DC (Down-conversion) receiver, where the RF and IF frequencies are 5 GHz and 40 MHz, respectively. And then the IF signals are digitized by ADCs (Analog to Digital Converters) at the rates of 32 MSPS. The undersampled IF signals are digitally down-converted (DDC) again to complex baseband and then downsampled by L-times, where L is an appropriate integer number. IF passband processing including DDC in FPGAs has been studied in [13]. As shown in Fig.5, single FPGA on AD board performs the digital signal processing of UMP. The user terminal PC communicates with the CPU SH4 via Ethernet, and CPU controls the UMP via direct 32-bit data bus connection.

5. Performance Assessment

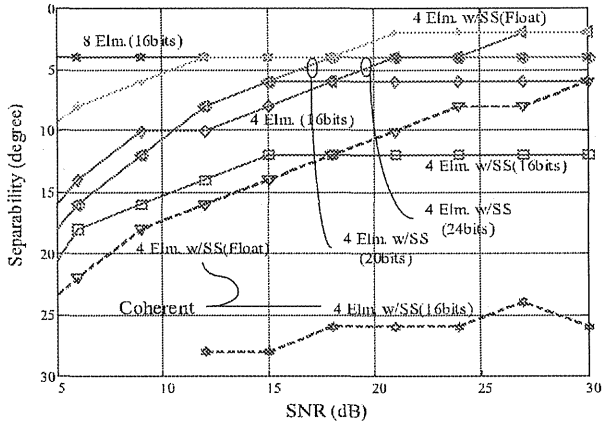
In this section, the estimation performance of UMP will be discussed by the hardware level simulations with offline PC with Matlab (Mathworks). The hardware level simulation means that the fixed-point operation behavior of UMP in VHDL was described exactly in Matlab m-file. It was assumed that the antennas and analog components prior to UMP had ideal characteristics or they were well calibrated. That is to say, this analysis considered only the digital signal processing part and neglected system-specific effects of analog parts. It may be efficient to assess the system without working the whole system components in various scenarios. The input level adjustment circuit like AGC (Automatic Gain Control) was considered in order to make use of full scale range of ADCs (12 bits), otherwise the performance of UMP will be degraded because of the low dynamic range of the fixed-point operation with bit truncation. In this section, all simulations included 200 burst frame data and single burst consisted of 136 snapshots (symbols). The source waves were a $\pi/4$ QPSK modulated signals.

5.1 AWGN Channel

In AWGN channel, the standard deviation of the estimated DOA when the single wave impinging from broadside is a good overall performance assessment as an estimation variation to SNR (Signal to Noise Ratio). The same approach has been used in [6]. Figure 7 shows the simulation results computed by the 4-element UMP with 16-bit fixed-point operation with spatial smoothing (SS) and the offline PC with 64-bit floating-point (double precision) operation, where the diamonded, right-triangled and left-triangled line denote the incident DOAs of 0, 30 and 50 degrees, respectively. In addition, the results of 4- and 8-element UMP without SS at 0 degree are also illustrated by the sqaured and circular lines, respecively. In this results, it is clear that the estimation accuracy is below 2 degree if the input SNR is greater than 5 dB in the linear region between -30 and +30 degrees. And it can be also seen that the UMP has a pretty good performance for the offline PC, although the estimation variation is larger to some degree because of its finite word-length (16-bit) fixed-point operation. As a matter of fact, the floating-point operation has the SNR gain of about 10dB more than the fixed-point operation, but it is an unavoidable trade-off between them. In Fig.7, it can be seen that the estimation variation becomes larger as the DOA angle gets far off from the broadside. That is caused by only non-uniform discrete wavefronts generated by the spatial DFT if considering ideal omni-directional antenna pattern. Non-uniform discrete wavefront effect is illustrated well in Fig.8, which shows the DOA dependency of the mean DOA deviation ϑ, where



(a) ρ=1 deg.



(a) ρ=3 deg.

Fig. 9 Separability Performance of two incident waves

$$\vartheta = E\{|\theta_{Estimated} - \theta_{DOA}|\}. \tag{20}$$

When there existed two incident waves, the separability performance which represented how spatially close signals can be discriminated from each other was assessed. The criteria of the successful estimation was

$$r^{\varphi} = \arg\max_{\theta}\{\mathcal{R}^{\varphi}_{std<\rho} \cap \mathcal{R}^{\varphi}_{mean<\rho}\}, \tag{21}$$

where φ, θ and ρ are a given SNR, incident DOA and condition value, respectively. r^{φ} and $\mathcal{R}^{\varphi}$ denote the separable angle and the range of θ satisfying subscripted condition to each estimation at a given SNR φ, respectively. Herein, the condition of $(mean < \rho)$ eliminates far-off estimation with small estimation variation. Figure 9 shows the separability performance of UMP where solid line and dotted line denote incoherent and coherent, respectively. Two waves had the same powers and temporally correlated with each other. The spatial smoothing (SS) capability of UMP reduces the correlation to some degree. In Fig.9, the separable performance of UMP increases as the EVD word-length gets longer. And it can be also seen that the separability performance with highly correlated (coherent) signals

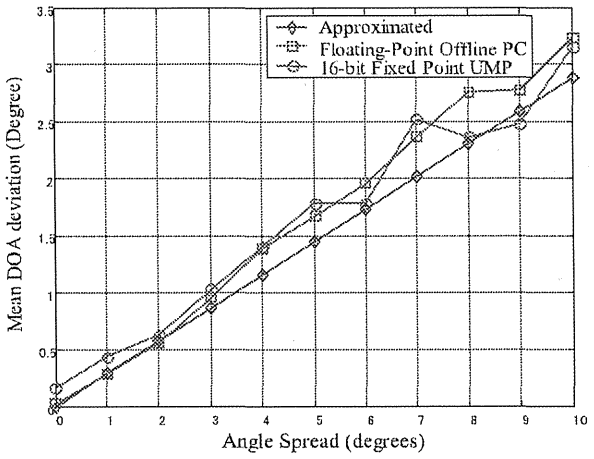


Fig. 10 Mean DOA Deviation to Angle Spread in UMP (average SNR=10dB)

was degraded. That is because the spatial smoothing with two sub-matrices in fixed-point operation could not provide sufficient reduction of the correlation between each signal. But it will be imploved if with larger number of sub-matrices in longer array antenna and longer word-length. On the other hand, the 16-bit fixed-point UMP can separate two incoherent waves close by about 10 degrees when the SNR more than 10 dB under the condition of $\rho = 3°$, but the separability performance can be imploved with more antenna elements as shown in Fig.9.

5.2 Rayleigh Fading Channel

Above analyses may be quite meaningful to evaluate the behavior of the fixed-point UMP at a given SNR. Actually, however, the realistic channel was not modeled appropriately in above analyses. In wireless cellular communication, a fading phenomenon occurs by multipath propagation. However, in such a fading environment, the SNR becomes a random variable with any distribution. In addition, the great number of coherent signals caused by far field local scatters arrived from spread DOA angle to reduce the fading correlation between antenna elements. From this fact, the plane wave model from a point source may not be valid any more, thus the estimation performace will be degraded in multi-path fading environment [14]. In that regard, it is important that we investigate the behaviour of UMP assumming multi-path fading environment. In multipath propagation, by neglecting the time delays between sub-pathes, the channel vector for i-th source in Eq.(2) can be rewritten approximately by

$$\boldsymbol{v}_i = \sum_{j=1}^{N_j} \beta_{ij} \boldsymbol{a}(\theta_i + \tilde{\theta}_j), \tag{22}$$

where β_{ij}, $\tilde{\theta}_j$ and N_j are the complex amplitude with

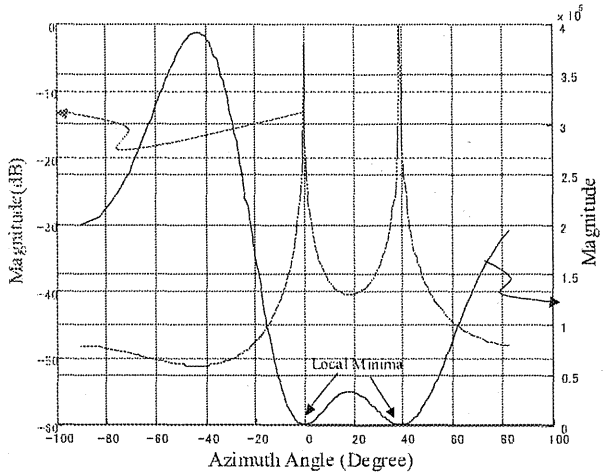


Fig. 11 Experimental Result of MUSIC Spectrum and its reciprocal (4 antennas, two waves impinging at 0 and 40 degrees, same powers, average SNR=19dB)

Rayleigh distributed magnitude of j-th scattered sub-path, DOA variation and number of the sub-pathes, respectively. In Fig.10, the relationship between the angle spread and the estimated mean DOA deviation as Eq.(20) is shown, where $N_j = 30$ sub-pathes are uniformly distributed at the center of single wave source $\theta_1 = 30°$ and average SNR was 10 dB, respectively. In this figure, the diamond line is theoretically approximated result [14]. The UMP result had a good agreement to the theoretical approximation. Thus it can be seen that the functionality of UMP in a realistic multi-path channel model is still valid.

6. Experimental Example

In this section, the whole system operation of UMP will be demonstrated experimentally in radio anechoic chamber. We used 4 omni-directional sleeve antennas which were half wavelength spaced in ULA. In this experiment, the fine calibration procedure was not considered, but the coarse adjustment of amplitude and phase was just conducted manually before measurement. In this example, it was assumed that there existed 2 incident waves. The data snapshot vector of two incident waves was generated by linear combination of each data snapshot vector of sigle incident wave. Table 2 illustrates the experimental parameters. The RF carrier frequency was 5 GHz. The transmitted waves were CW (Continuous Wave) at different frequencies (the difference was about 1.3 MHz). Figure 11 shows an experimental result. The DOAs of each wave were set by 0 and 40 degrees, respectively. Two waves were transmitted at the same power and the average SNR at each antenna element was around 19 dB. In Fig.11, the indices of the DFT spectrum corresponding to the LM points below any appropriate threshold level (5, 000

Table 2 Experimental Parameters

Antennas	4-element sleeve antenna in ULA
Antenna spacing	$\lambda/2$
RF Frequency	5 GHz
IF Frequency	40 MHz
Modulation	CW
Sampling Frequency	32 MHz
ADC resolution	12 bits

herein) were 127 and 208. With these indices, the concrete discrete wavefronts could be converted to -0.4476 and 38.6822 degrees from the relationship of Eq.(17), respectively. It was reasonable that the estimation error was caused by lack of the fine calibraition for the antennas and analog components, but the system operation could be confirmed to some extent. The calibration of whole system should be cleared for practical use, and the digital calibration processor will be our future works.

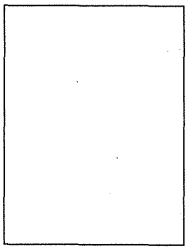
7. Conclusion

In this paper, the FPGA design of the fast DOA estimator using the unitary MUSIC algorithm was proposed and its real hardware implementation was also introduced. The unique features of this system are the fast computation and compact architecture of the EVD and MUSIC angular spectrum generation with Cyclic Jacobi processor based on CORDIC and spatial DFT, respectively. All procedures of digital signal processing were computed by only fixed-point operation with finite word-length for high speed and low power consumption. As far as the optimization depends on the design technique, the hardware friendly parallel algorithm and processing concept will be still valid and outperform the serial computer like general purpose CPU. Finally, we are expecting that this system provides a useful example of the real hardware implementation of a high speed DOA estimator for the wireless communication applications with dedicated circuits.

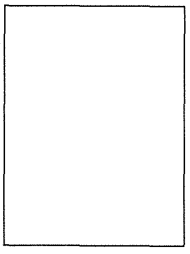
References

[1] R. Kohno, "Structures and Theories of Software Antennas for Software Defined Radio," IEICE Trans. Commun., vol. E83-B, No. 6, pp. 1189-1199, Jun 2000.
[2] N. Kikuma and M. Fujimoto, "Adaptive Antennas," IEICE Trans. Commun., vol. E86-B, No. 3, pp. 968-979, March 2003.
[3] S. Anderson, et al., "An Adaptive Array for Mobile Communication Systems," IEEE Trans. Vehicular Tech., vol. 40, No. 1, Feb 1991
[4] M. Cummings and S. Haruyama, "FPGA in the Software Radio," IEEE Commun. Mag., vol. 37, No. 2, pp.108-112, Feb 1999.
[5] T. Asai, S. Tomisato, T. Matsumoto, "Field Test Results for a Beam and Null Simultaneous Steering S/T-Equalizer in Broadband Mobile Communication Environments," IEICE Trans. Commun., vol. E84-B, No. 7, pp. 1760-1767, July 2001.

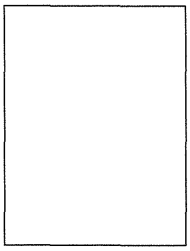
[6] A. Kuchar, M. Tangemann and E. Bonek, "A Real-Time DOA-Based Smart Antenna Processor," IEEE Trans. Vehicular Technology, vol. 51, No. 6, pp. 1279-1293, Nov 2002.
[7] M. Pesavento, A. B. Gershman, and M. Haardt, On Unitary Root-MUSIC with a real-valued eigendecomposition: A theoretical and experimental performance study," IEEE Trans. Signal Processing, vol. 48, pp. 1306-1314, May 2000.
[8] M. D. Zoltowski, G. M. Kautz, and S. D. Silverstein, "Beamspace root- MUSIC," IEEE Trans. Signal Processing, vol. 41, pp. 344-364, Jan 1993.
[9] M. Kim, K. Ichige, H. Arai, "Design of Jacobi EVD processor based on CORDIC for DOA estimation with MUSIC algorithm", IEICE Trans. Commun., vol. E85-B, No.12, pp. 2648-2655, Dec 2002.
[10] R.O. Schmidt, "Multiple emitter location and signal parameter estimation," IEEE Trans. on Antenna and Propag., vol. 34, no. 3, pp. 276-280, March 1986.
[11] "Stratix Device Handbook," Altera Corp., Web document is available at http://www.altera.com/literature/hb/stx/stratix_handbook.pdf.
[12] "FFT MegaCore Function User's Guide," Altera Corp., Web document is available at http://www.altera.com/literature/ug/fftug.pdf.
[13] M. Kim, K. Ichige, H. Arai, "Adaptive Array Antenna Experimental System based on FPGAs and its evaluation through implementation of MRC beamformer," IEICE Antenna and Propagation Tech. Report, AP2002-38, June 2002.
[14] D. Astely and B. Ottersten, "The effects of Local Scattering on Direction of Arrival Estimation with MUSIC," IEEE Trans. Signal Processing, vol. 47, No. 12, pp. 3220-3234, Dec 1999.



Minseok Kim was born in Seoul, Korea on June 16, 1973. He received the B.S. degrees in Electrical Engineering from Hanyang University, Seoul, Korea, M.E. degree in Division of Electrical and Computer Engineering, Yokohama National University, Japan in 1999 and 2002, respectively. He is currently working toward Ph. D. degree from Yokohama National University. His research interests are hardware implementation of adaptive antenna system and software defined radio.



Koichi Ichige received the B.E., M.E. and Dr. Eng. degrees in electronics and information engineering from the University of Tsukuba in 1994, 1996 and 1999, respectively. In 1999, he joined the Department of Electrical and Computer Engineering, Yokohama National University as a research associate, where he is currently an assistant professor. In 2001-2002, he has been on leave to School of Engineering, Swiss Federal Institute of Technology Lausanne (EPFL), Switzerland as a visiting researcher. His research interests include digital signal processing, image processing, approximation theory, communication and electromagnetic theory. He is a member of IEEE.



Hiroyuki Arai received the B.S. degree in Electrical and Electronic Engineering, M.E. and D.E. in Physical Electronics from Tokyo Institute of Technology in 1982, 1984 and 1987, respectively. After a research associate in Tokyo Institute of Technology, he joined Yokohama National University as a lecturer in 1989. Now he is a professor in Division of Electrical and Computer Engineering, Yokohama National University. He investigated microwave passive components for high power handling applications such as RF plasma heating in large Tokamaks. He developed a flat diversity base station antenna for Japanese PDC systems, and small base station antennas of In-building micro cellular system. He was awarded the "Meritorious Award on Radio" by the Association of Radio Industries and Businesses in 1997 for the development of polarization diversity antenna. He is collaborating with a large number of companies for mobile terminal antennas, cellular base station antennas, antenna measurement techniques, indoor / outdoor propagation measurement and simulation, and EMC measurements and wave absorbers. He published more than 50 reviewed journal papers and about four hundreds international and domestic conference papers. He is the co-author of four textbooks about electromagnetic waves, and he holds several US and Japanese patents of antenna.

PERFORMANCE EVALUATION OF DOA ESTIMATION ALGORITHMS IN DIGITAL IMPLEMENTATION

Masashi SHINAGAWA[(1)], Koichi ICHIGE[(2)], Hiroyuki ARAI[(3)]

[(1)] Department of Electrical and Computer Engineering, Yokohama National University Tokiwadai 79-5, Hodogaya-ku, Yokohama, 240-8501 Japan, E-mail: masa@arailab.dnj.ynu.ac.jp
[(2)] As (1) above, but E-mail: koichi@dnj.ynu.ac.jp
[(3)] As (1) above, but E-mail: arai@arailab.dnj.ynu.ac.jp

ABSTRUCT

In this paper, we study the properties and drawbacks of representative DOA (direction of arrival) estimation algorithms such as MUSIC, Root-MUSIC [1][2], and Unitary-ESPRIT by investigating the influence of quantization errors in digital operations for various cases. We also verify which algorithm is the most suitable for the implementation by finite word-length digital processors. First we examine the characteristics of DOA algorithms through some computer simulation [4]. Then we study the internal quantization error of DOA algorithms, and derive the bit length required for the accurate DOA estimation.

INTRODUCTION

Recently eigendecomposition based algorithms such as MUSIC, Root-MUSIC, and Unitary-ESPRIT have been used in DOA estimation. Many authors examine the characteristics on each algorithm. But few quantitative investigations for digital implementation are done. At the same time, FPGA has attracted the attention as low power and high speed device which is based on parallel processing and fixed-point operation.
In this paper, we study the properties and drawbacks of representative DOA estimation algorithms by investigating the influence of quantization errors in digital operations for various cases. We also verify which algorithm is the most suitable for the implementation by finite word-length digital processors. First we examine the characteristics of DOA algorithms through some computer simulation. Here we change DOA angle, the number of snapshots, SNR, array elements, and the correlation of incident waves. Then we study the internal quantization error of DOA algorithms, and derive the bit length required for the accurate DOA estimation. We change a bit length of the internal operation of each algorithm in fixed-point operation, and examine the relation between estimation accuracy and bit length.

SPECIFICATIONS OF SIMULATION

Simulation on Characteristics of DOA Algorithms

In this simulation, as it is given in Fig.1, we used uniform linear array antenna. The algorithm used in the simulation is as follows: Spectral (original) MUSIC, Root-MUSIC, Unitary Spectral MUSIC, Unitary Root-MUSIC, ESPRIT, and Unitary ESPRIT. Here we change arrival angles, the number of snapshots, SNR, array elements, and the correlation of incident waves. We used the FB (forward-backward) space averaging technique [3] as a correlation suppressing method. We defined DOA angle as θ_{DOA} and estimation angle as θ_{EST}. The estimation error used for this evaluation is:

$$\left|\theta_{DOA} - \theta_{EST}\right| [\text{deg}] \tag{1}$$

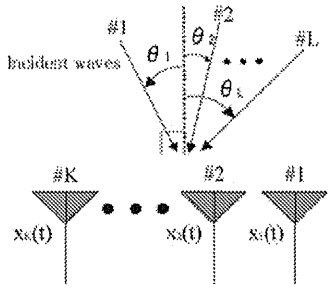


Fig.1. K-elements uniform linear array antenna

Simulation on The Internal Quantization Error of DOA Algorithms

We studied on the required bit length for the internal operations of each DOA algorithm. We examined the relation between estimation accuracy and bit length, when we changed a bit length of the decimal part and the integer part respectively in a fixed-point operation from 4bits to 32bits. But we calculated MUSIC spectrum in a floating-point operation, because the value of MUSIC spectrum was very large. Moreover, we used the values of $\sqrt{\ }$ and $\sin\theta$, $\cos\theta$, $\tan\theta$ which were transformed from the calculation results in PC into the fixed-point values.

RESULTS OF SIMULATION

Simulation on Characteristics of DOA Algorithms

First, we investigated the accuracy of DOA estimation when we changed arrival angles, the number of snapshots, and SNR to see how the above algorithms work. Fig.2 shows the characteristic when changing arrival angle. We verified that the estimation accuracy deteriorates in nearly ± 90 degrees. Fig.3 shows the characteristic when changing the number of snapshots. Fig.4 shows the characteristic when changing SNR. We verified that the estimation accuracy could be improved by increasing the number of snapshots or SNR. These simulation conditions are given in Table 2. The obtained characteristics by the algorithms using MUSIC spectrum are different from those by the algorithms estimating DOA numerically. In other words, the estimation accuracy of the algorithms using MUSIC spectrum depends on how precise we measure the steering vectors. We used the steering vectors whose minimum interval of value was 0.5 degrees that can be considered sufficiently precise.
Fig.5 shows the characteristic when changing SNR in the case where uncorrelated 3 waves come. Compared with Fig.4, we verified that the estimation accuracy deteriorates by increasing the number of incident waves. Fig.6 shows the characteristic when changing SNR in the case where full-correlated 3 waves come. Compared with Fig.5, we verified that the estimation accuracy deteriorate when incident waves were correlated.

Table 1. Meanings of Abbreviation

Abbr.	Meanings
S	Spectral MUSIC
US	Unitary Spectral MUSIC
R	Root-MUSIC
UR	Unitary Root-MUSIC
E	ESPRIT
UE	Unitary ESPRIT

Table 2. Settings of Parameters in Simulation

	Fig.2.	Fig.3.	Fig.4.	Fig.5.	Fig.6.	Fig.7.	Fig.8.	Fig.9.
Array Elements	4	4	4	4	6	Change	Change	10
Sub-Array Elements					4		4	Change
Incident waves	1	1	1	3	3	2	2	2
Arrival Angle [deg] (Power)	Change (1.0)	-35.2 (1.0)	-35.2 (1.0)	-35.2(1.0) 10.2(1.0) 25.2(1.0)	-35.2(1.0) 10.2(1.0) 25.2(1.0)	15.2 (1.0) Change	15.2 (1.0) Change	15.2 (1.0) Change
Snapshots	300	Change	300	300	300	300	300	300
SNR [dB]	7	7	Change	Change	Change	7	7	7

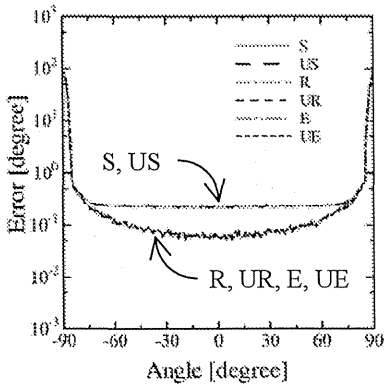


Fig.2. Estimation Error when changing arrival angle

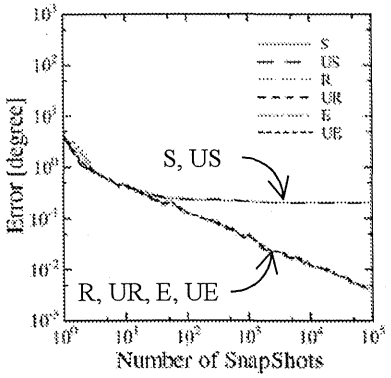


Fig.3. Estimation Error when changing the number of snapshots

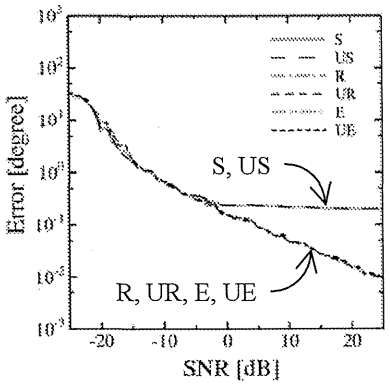


Fig.4. Estimation Error when changing SNR

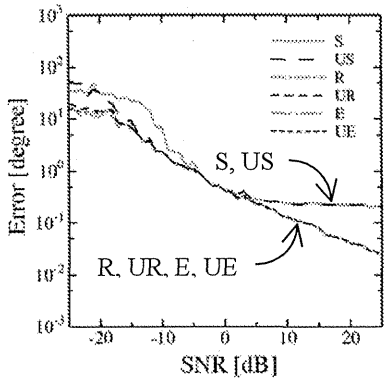


Fig.5. Estimation Error when changing SNR

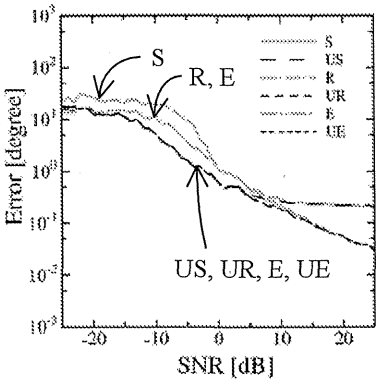


Fig.6. Estimation Error when changing SNR

Next, for the cases that two incident waves were uncorrelated and full-correlated waves, we investigated the relation between the number of array elements and the range of angles which could be achieved enough accuracy (the range of angles which can be estimated with the error less than 1 degree). Fig.7 shows the characteristics in the case where uncorrelated 2 waves come. As shown in the previous simulation result, we confirmed the difference of the characteristics between what use MUSIC spectrum for DOA estimation, and what estimate DOA numerically. Note that the estimation accuracy can be improved in any algorithm by increasing the number of elements.

Especially in the case where full-correlated 2 waves come, we simulated two cases. One is that changing the number of array elements (the number of sub-array elements are fixed to 4 elements), and the other is that changing the number of sub-array elements (the number of array elements are fixed to 10 elements). Simulation results are given in Fig.8 and Fig.9 respectively. Consequently, we observed the followings: When increasing the number of array elements, the algorithms not employing unitary transform (non-unitary algorithms) cannot improve the DOA estimation accuracy at all, they just keep the accuracy at the same level, but the algorithms employing unitary transform (unitary algorithms) can improve the estimation accuracy. On the other hand, when increasing the number of sub-array elements, the non-unitary algorithms can improve the DOA estimation accuracy, while the unitary algorithms cannot improve the estimation accuracy. From this fact, we can remark that the estimation accuracy for the non-unitary algorithms and that for the unitary ones are dependent on the number of sub-array elements and the number of sub-arrays, respectively, in the case that full-correlated waves come.

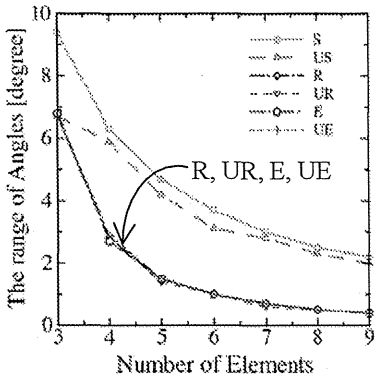


Fig.7. The range of angles when changing the number of array elements

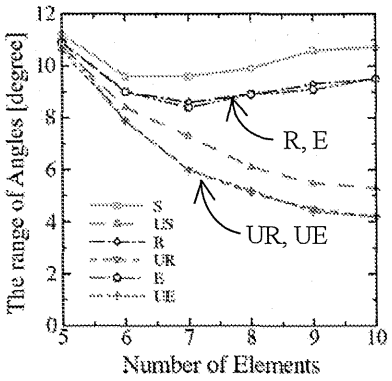


Fig.8. The range of angles when changing the number of array elements

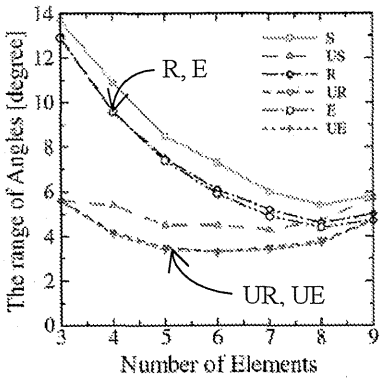


Fig.8. The range of angles when changing the number of sub-array

Simulation on The Internal Quantization Error of DOA Algorithms

We derive the bit length required for the accurate DOA estimation for implementation in fixed-point operation. Simulation condition is as follows: full-correlated 3 waves come because the estimation accuracy when 3 waves come is lower than that when 1 wave comes, and the estimation accuracy when correlated waves come is lower than that when uncorrelated waves come. Arrival angles are −5.2 degrees, 15.2 degrees and 35.2 degrees (intervals of 20 degrees). SNR is 7 dB because we consider actual environment. The number of snapshot is 300 because we consider that 300 snapshots are proper for the estimation accuracy and the amount of calculation.

Fig.10 shows the characteristic when changing a bit length of the decimal part in a fixed-point operation at 6 elements array and 4 elements sub-array (a bit length of integer part is fixed to 64 bits). We confirmed, as compared to non-unitary algorithms, unitary algorithms could estimate with the error less than 1 degree with 16 bits. So only about unitary algorithms, we simulated the characteristics when changing a bit length of the integer part in a fixed-point operation in the case where a bit length of decimal part was fixed to 16 bits. The result is given in Fig.11. We confirmed that we needed 12 bits to estimate with the error less than 1 degree.

Fig.12 and Fig.13 show the characteristics when changing a bit length of the decimal part and the integer part respectively, at 10 elements array and 6 elements sub-array (in Fig.13, a bit length of decimal part is fixed to 12 bits). In Fig.13, we confirmed, by increasing the number of array elements, the required bit length of Root MUSIC and Unitary Root-MUSIC were very large because the calculation amount of Root-MUSIC polynomial depended on the number of array element: $2(K-1)$.

As far as the result of our simulation, in the case where full-correlated 3 waves come at 6 elements array and 4 elements sub-array, at least 28 bits (16bits + 12bits) is necessary for the DOA estimation with sufficient accuracy.

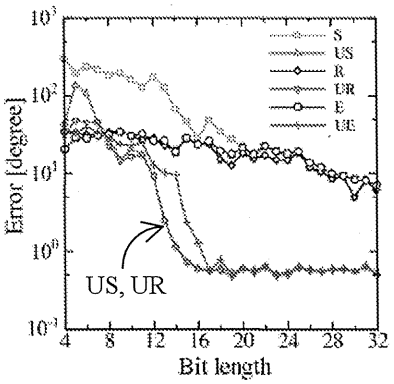


Fig.10. Estimation Error when changing a bit length of the decimal part at 6 elements array and 4 elements sub-array

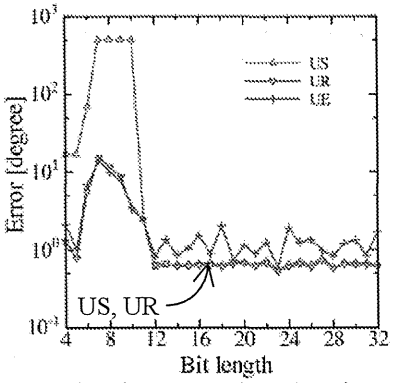


Fig.11. Estimation Error when changing a bit length of the integer part at 6 elements array and 4 elements sub-array

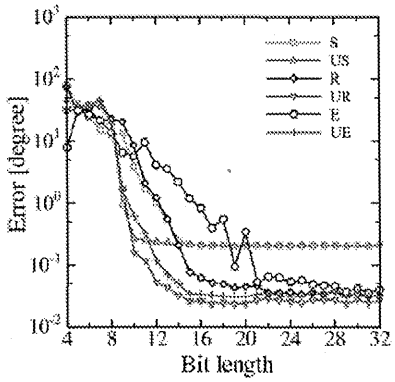


Fig.12. Estimation Error when changing a bit length of the decimal part at 10 elements array and 6 elements sub-array

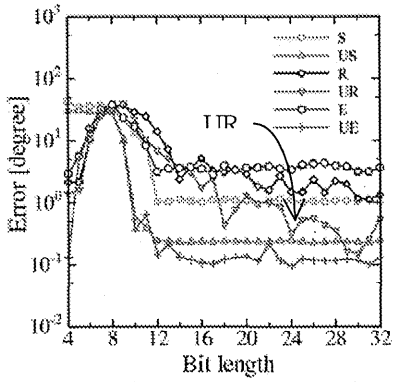


Fig.13. Estimation Error when changing a bit length of the integer part at 10 elements array and 6 elements sub-array

CONCLUSION

In this work, we investigated the characteristics of various DOA estimation algorithms. Throughout the simulations, we confirmed and summarized that unitary algorithms were superior to the others, and unitary algorithms were more effective for finite bit-length operation but as the number of array elements increased, the required bit length of Unitary Root-MUSIC increased. Consequently, we conclude that Unitary ESPRIT is the best algorithm in 6 algorithms we examine for digital implementation. In the case where full-correlated 3 waves come at 6 elements array and 4 elements sub-array, at least 28 bits (16bits + 12bits) is necessary for the DOA estimation with sufficient accuracy.

REFERENCES

[1] Marius Pesavento, Alex B. Gershman, and Martin Haardt, "Unitary Root-MUSIC with a Real-Valued Eigen decomposition," IEEE Trans. Signal Processing, vol.48, no.5, pp.1306-1314, May 2000.

[2] Bhaskar D. Rao, and K. V. S. Hari, "Performance Analysis of Root-MUSIC," IEEE Trans. Acoust. , Speech, Signal Processing, vol.37, no.12, pp.1939-1949, December 1989.

[3] Ronald T. Williams, Surendra Prasad, and A. K. Mahalanabis, "An Improved Spatial Smoothing Technique for Bearing Estimation in a Multipath Environment," IEEE Trans. Acoust. , Speech, Signal Processing, vol.36, no.4, pp.425-431, April 1988.

[4] William H. Press et al, Numerical Recipes in C: The Art of Scientific Computing, Cambridge University Press, Cambridge, 1992.

AN ALGEBRAIC APPROACH TO EIGENPROBLEMS AND ITS APPLICATION TO DOA ESTIMATION

Koichi ICHIGE[(1)(*)], Masashi SHINAGAWA[(2)], Hiroyuki ARAI[(3)]

[(1)] *Department of Electrical and Computer Engineering, Yokohama National University 79-5 Tokiwadai, Hodogaya-ku, Yokohama 240-8501, Japan, E-mail: koichi@dnj.ynu.ac.jp*
[(2)] *As (1) above, but E-mail: masa@arailab.dnj.ynu.ac.jp*
[(3)] *As (1) above, but E-mail: arai@arailab.dnj.ynu.ac.jp*
[(*)] *On leave to IOA/FSTI, Swiss Federal Institute of Technology Lausanne (EPFL), CH-1015 Lausanne, Switzerland, E-mail: koichi.ichige@epfl.ch*

ABSTRACT

This paper discusses the eigenproblems of correlation matrices in direction-of-arrival (DOA) estimation algorithms, especially for the case that the number of arriving waves is a few. The eigenvalues and eigenvectors can be obtained in a very short time (only 1.74% of the time by the conventional method) by regarding the eigenproblem as solving a fourth-order algebraic polynomial. It is also confirmed that the proposed algebraic approach does not make the accuracy worse when it is implemented by finite word-length processors like digital signal processor (DSP) or field programmable gate array (FPGA).

INTRODUCTION

Mobile communication systems are intensively developing toward the next generation technology. To distinguish the target wave with the interference waves, DOA estimation is very significant for digital beam forming (DBF) array antenna system. Many DOA estimation algorithms have been already proposed [1]–[3], nowadays MUSIC [1] and ESPRIT [2] are two representative algorithms that can estimate DOAs accurately. The problem of the DOA estimation algorithms with correlation matrices (like MUSIC and ESPRIT) may be the computational complexity. One of the time-consuming processes in DOA estimation is computing eigenvalues/eigenvectors of correlation matrices.

Recalling that the derivation of eigenvalues is equivalent to solve the characteristic polynomial of the correlation matrix, the algebraic solvent can be applied if the order of the polynomial is four or less. This corresponds to the situation of DBF array antenna with four or less antenna elements (also valid when using a space averaging technique for every four or less elements). This paper presents a fast algebraic algorithm to derive eigenvalues/eigenvectors of correlation matrices, supposed to be used in the real-time DOA estimation for a small number of arriving waves [4]. The proposed algorithm employs the algebraic solvent of fourth-order characteristic polynomial to derive eigenvalues of correlation matrices instead of the numerical QR decomposition algorithm. The algebraic approach tends to make the accuracy worse when it is implemented by a digital device due to the quantization, however, the proposed approach guarantees that the quantization error does not affect to the estimated DOA when it is implemented by finite word-length processors like DSP or FPGA.

PRELIMINARIES

Figure 1 illustrates the configuration of an (linear) array antenna system. The computation procedure of MUSIC method can be roughly summarized as the following three steps, for instance to see how the eigenvalues and eigenvectors are used in DOA estimation.

[STEP 1] The correlation matrix $\boldsymbol{R}_{xx}$ of the input vector $\boldsymbol{X}$ is obtained by

$$\boldsymbol{R}_{xx} = E\left[\boldsymbol{X}(t)\boldsymbol{X}^H(t)\right] = \boldsymbol{A}\boldsymbol{S}\boldsymbol{A}^H + \sigma^2\boldsymbol{I},$$

where $\boldsymbol{X}$, $\boldsymbol{A}$ and $\boldsymbol{S}$ are so-called the signal, the direction and the signal correlation matrices, respectively.

[STEP 2] Derive the eigenvalues λ_i and the corresponding eigenvectors $\boldsymbol{y}_i$ of the correlation matrix $\boldsymbol{R}_{xx}$.

[STEP 3] Using the direction matrix $\boldsymbol{A}$ and the eigenvectors $\boldsymbol{y}_i$, calculate the MUSIC spectrum $\boldsymbol{P}(\theta)$.

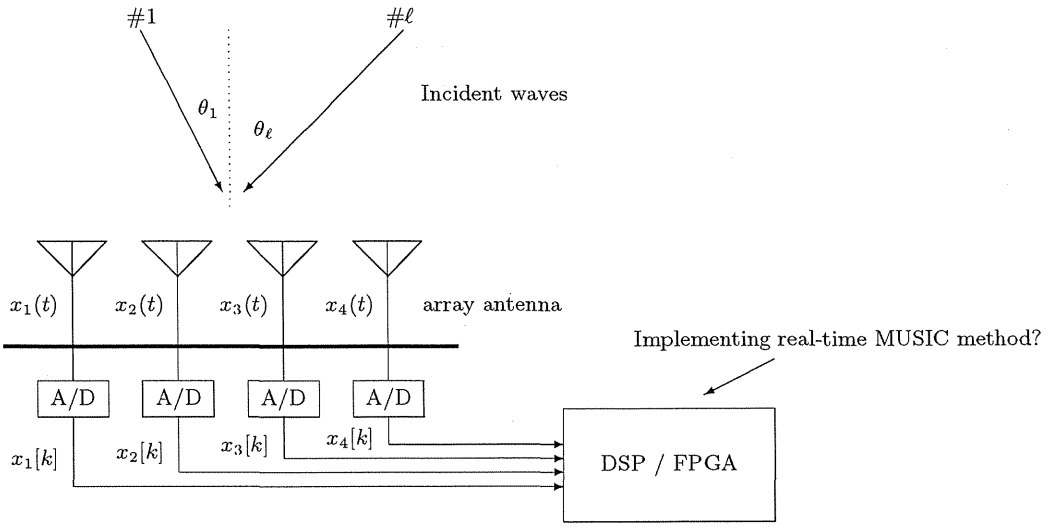


Figure 1: Configuration of array antenna system (with four elements)

In MUSIC method, the time-consuming process is usually either STEP 2 or STEP 3, and some fast algorithms have been already proposed for STEP 3. Representative one would be Root-MUSIC method [3] which is an algebraic algorithm without the direction search. Similarly, in the next section, we aim at developing an algebraic approach for STEP 2 to make this process faster.

PROPOSED APPROACH

In this section, a fast algebraic algorithm is investigated for the DOA estimation by array antenna with four elements. The eigenvalues λ_i of the matrix $\boldsymbol{R}_{xx}$ can be derived by solving the characteristic polynomial

$$\det(\boldsymbol{R}_{xx} - \lambda \boldsymbol{I}) = 0. \tag{1}$$

If the size of the matrix $\boldsymbol{R}_{xx}$ is $n \times n$, the characteristic polynomial becomes n-th order polynomial of λ, and it has n solutions. Here, we adopt the well-known mathematical technique that the polynomials of up to fourth-order can be solved by an algebraic procedure. For the DOA estimation problems, we can establish an algebraic approach to derive eigenvalues of correlation matrices in case of four antenna elements, also when using a space averaging technique for every four elements. Now we investigate the algebraic approach.

Deriving Eigenvalues

In case of four antenna elements, (1) can be reduced into the following fourth-order polynomial equation of λ:

$$\lambda^4 + a\lambda^3 + b\lambda^2 + c\lambda + d = 0. \tag{2}$$

Since the matrix $\boldsymbol{R}_{xx}$ is a non-negative Hermitian matrix with a full rank 4 [5], it has four real eigenvalues different to each other. By simply solving th equation (2), it's simple and well-known, the eigenvalues of the matrix $\boldsymbol{R}_{xx}$ can be derived as

$$\lambda_i = \sqrt{t} \pm \sqrt{h_1 \pm 2h_2} - \frac{a}{4}$$

where h_1, h_2 and t are the functions of a, b, c and d.

Deriving Eigenvectors

Define the matrix $\boldsymbol{D}$ by

$$\boldsymbol{D} = [d_{ij}] := \boldsymbol{R}_{xx} - \lambda_i \boldsymbol{I} \in \boldsymbol{C}^{4\times 4},$$

then the eigenvector $\boldsymbol{y}_i = [y_{i1}, y_{i2}, y_{i3}, y_{i4}]^T$ of the matrix $\boldsymbol{D}$ corresponding to the i-th eigenvalue can be obtained by solving the following matrix equation:

$$(\boldsymbol{R}_{xx} - \lambda_i \boldsymbol{I})\boldsymbol{y}_i = \boldsymbol{D}\boldsymbol{y}_i = 0. \tag{3}$$

Equation (3) is generally solved by Gauss elimination method, however it is redundant since the matrix $\boldsymbol{D}$ is singular in this case. Hence, we employ a faster algorithm using an inverse matrix of the sub-matrix of $\boldsymbol{D}$. Since $\boldsymbol{D}$ is a singular matrix, the row-vectors $\boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_3$ and $\boldsymbol{d}_4$ of the matrix $\boldsymbol{D}$ are linearly dependent. On the other hand, those four vectors include three linearly independent vectors. Suppose that $y_{i4} = 1$ and the vectors $\boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_3$ are linearly independent. Based on this relation, the eigenvectors $\boldsymbol{y}_i$ can be calculated. (Indeed it's faster than Gaussian method, however space does not prove to show the results.)

SIMULATION

The proposed fourth-order polynomial approach is evaluated in comparison with the general QR decomposition. Table 1 shows the computation times of the two methods, polynomial and QR decomposition. From Table 1, the polynomial approach requires only 1.74% of the computation time that required in the QR decomposition. From this fact, the proposed polynomial approach is very effective to shorten the computation time to solve eigenproblems. Indeed the computation time of this process can almost be ignored. Also note that we tested thousands of example input vectors, and the result in the Table 1 is the average of those trials.

Table 1: Comparison of the required computation time

	QR decomp.	Proposed
computation time	1.54msec	24.2μsec
(ratio)	(1.000)	(0.0174)
required memories	560KB	572KB

(CPU: Intel Celeron 433MHz, Memory:256MB)

DISCUSSION

Assuming to implement the proposed algorithm by finite word-length (digital) devices, we confirm that the quantization noise caused by DSP/FPGA implementation does not make the DOA estimation accuracy worse.

Effect of the Quantization Noise in A/D Converter

First, the effect of the quantization noise in A/D converter is studied. The A/D converter and its quantization noise are imperative in digital implementation. Here we assume that some signals with power one are coming to the array antenna. In this case, the distribution of the input voltage becomes as illustrated in Fig. 2(a). From Fig. 2(a), we see that the input voltage usually vibrates within -2 to 2 volts, and we can adjust the level of the A/D converter not to make overflow. Actually, it does not matter if the numbers more than 2 is rounded to 2. Figure 3(a) shows the effect of quantization noise in A/D converter to the estimated DOAs. As seen in Fig.3(a), eight quantization bit length in A/D converter is enough for the accurate DOA estimation. Although the noise level in Fig.3(a) becomes slightly larger, it does not affect to the accuracy of DOA estimation.

Effect of the Quantization Noise in DSP

Next, we study the effect of the quantization noise in DSP. In the proposed algebraic approach, we should confirm the range of parameters in the middle of digital processing, not to make overflow. Figure 2(b) shows the example distribution of intermediate parameters. From Fig.2(b), the parameters vibrates within about 10-times of the range of the input voltage. Generally the operations in the fixed-point DSP are performed with 16-bits (single precision) or 32-bits (double precision). Considering the range of parameters, 16-bits operation would be enough for the accurate DOA estimation. Figure 3(b) shows the effect of the quantization noise in DSP. The quantization bit length in A/D converter is set to be eight. From Fig.3(b), the fixed-point DSP operation does not make the accuracy of the DOA estimation worse.

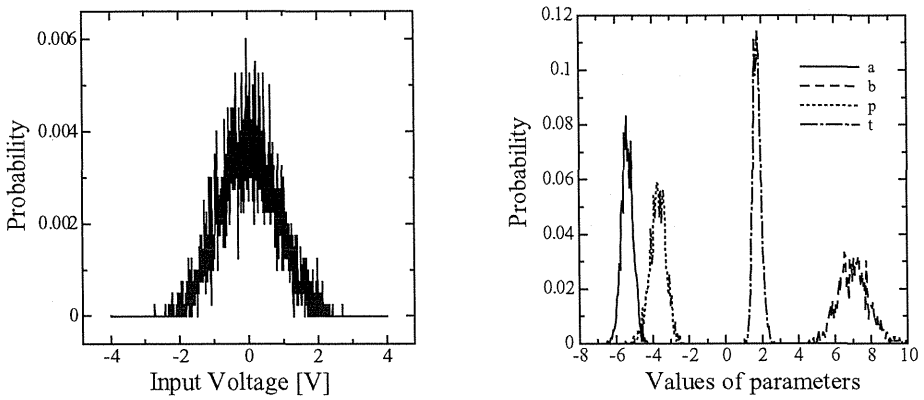


Figure 2: Distribution of Parameters: (a) that of Input voltage, (b) that of example intermediate parameters

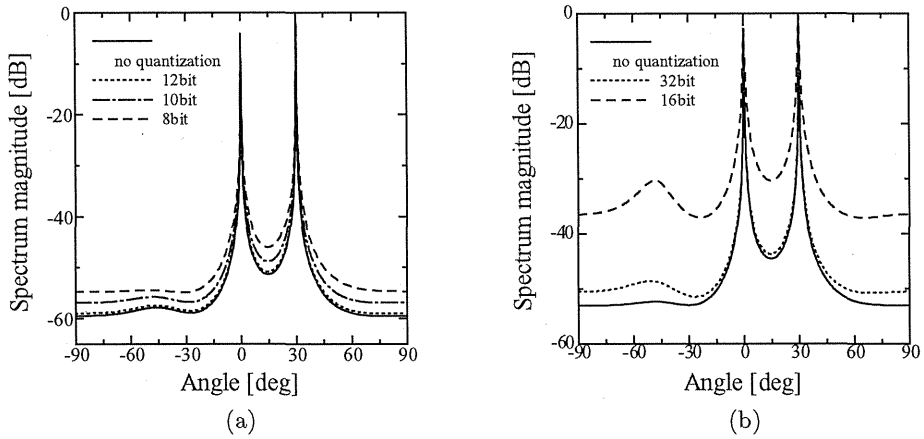


Figure 3: Effects of quantization errors in DOA estimation: (a) that in A/D converter, (b) that in DSP

CONCLUDING REMARKS

This paper focused on the eigenproblems for that the number of the arriving wave was a few, say, the order of the correlation matrix was four or less. Regarding the eigenproblem as solving the fourth-order algebraic polynomial, the eigenvalues and eigenvectors could be obtained in a very short time (only 1.74% of that by the conventional method). Moreover, we confirmed that the proposed algebraic approach does not make the accuracy worse when it was implemented by finite word-length processors.

REFERENCE

[1] R.O.Schmidt: "Multiple Emitter Location and Signal Parameter Estimation," *IEEE Trans. Antenna and Propagat.*, vol. 34, no. 3, pp.276-280, Mar. 1986.

[2] R.Roy and T.Kailath: "ESPRIT—Estimation of Signal Parameters via Rotational Invariance Techniques," *IEEE Trans. Accoust. Speech and Signal Proc.*, vol. 37, no. 7, pp.984–995, July 1989.

[3] B.D.Rao and K.V.S.Hari: "Performance Analysis of Root-MUSIC," *IEEE Trans. Accoust. Speech and Signal Proc.*, vol. 37, no. 2, pp. 1939–1949, Dec. 1989.

[4] K.Ichige, M.Shinagawa, H.Arai, "A Fast Algebraic Approach to Eigenproblems of Correlation Matrices in DOA Estimation," (submitted).

[5] W.H.Press et al., *Numerical Recipes in C: 2nd ed.*, Cambridge Univ. Press, 1992.

DESIGN OF JACOBI EVD PROCESSOR BASED ON CORDIC FOR DOA ESTIMATION WITH MUSIC ALGORITHM

Minseok Kim, Koichi Ichige, Hiroyuki Arai

Division of Electrical & Computer Engineering Dept.,
Yokohama National University, Japan,
{mskim,koichi.arai}@arailab.dnj.ynu.ac.jp

Abstract - Computing the Eigen Value Decomposition (EVD) of a symmetric matrix is a frequently encountered problem in adaptive (or smart or software) antenna signal processing, for example, super resolution DOA (Direction Of Arrival) estimation algorithms such as MUSIC (MUltiple SIgnal Classification) and ESPRIT(Estimation of Signal Parameters via Rotational Invariance Technique). In this paper the hardware architecture of the fast EVD processor of a symmetric correlation matrix for the application of an adaptive antenna technology such as DOA estimation is proposed and the basic idea is also presented.

Keywords - Adaptive Antenna, Smart Antenna, FPGA Implementation, DOA Estimation, MUSIC, EVD

I. INTRODUCTION

An adaptive antenna technology can provide a solution of multi-path fading. The adaptive antenna can suppress the adverse effect of multi-path delayed coherent signals and interferences by steering beams toward intended directions and nulls toward the other undesired directions so that it can achieve high communication quality. This operation can make the receiving signal strength almost flat and stable over a threshold level. Therefore it is necessary that an adaptive antenna should find the DOAs of signals and form beams and steer nulls within a fading period. Considering mobility of several hundreds of *km/h*, the fading period becomes very short time. It is very difficult to compute them by general serial architecture DSP processors, and hence the high-speed parallel computing processor with a specified function must be needed.

In this paper, the implementation issue in MUSIC (MUltiple SIgnal Classification), a super resolution DOA estimation method, and the examination of hardware design based on FPGAs are presented. MUSIC method is one of the subspace-based methods [1]. Generally the subspace-based methods are based on the Eigen Value Decomposition (EVD) of the covariance or correlation matrix. In the EVD based system, real-time processing is very difficult to be realized because of its complex logic and heavy computational load. This paper proposes the hardware logic design of a fast EVD processor which is suitable for realtime processing and can be implemented for adaptive antenna technologies practically. It uses Cyclic Jacobi method. Cyclic Jacobi method is well known for the simplest algorithm and easily implemented but its convergence time is slower than other factorization algorithms like QR-method [2]. But if considering the fast parallel computation of the EVD with a dedicated circuit like ASIC or FPGA, the Cyclic Jacobi method can be a good choice, since it offers a very higher degree of parallelism and easier implementation than QR-method [3]. This paper uses hardware friendly CORDIC (COordinate Rotation DIgital Computer) algorithm for vector rotators and arctangent computers, which are the basic processors of this design.

This paper is organised as follows. Sect.II presents the principle and computation flow of DOA estimation in MUSIC Method briefly. Sect.III introduces Cyclic Jacobi EVD algorithm and Sect.IV describes the basic ideas of CORDIC algorithm and the circuit implementation. For simple architecture and practical realizability, the paper uses fixed-point or fixed bit-length arithmetic instead of floating-point operations. With fixed-point arithmetic it is desired to overcome processing speed limitation and power consumption. In Sect.V the number of Jacobi sweeps is determined, that is, the computational load is confined constantly, and the errors caused by the fixed-point operations are discussed. Sect.VI proposes the hardware architecture and circuit design. Sect.VII yields its computational load discussion.

II. DOA ESTIMATION BY MUSIC METHOD

The computation flow of DOA estimation by MUSIC method is as follows. First, the correlation matrix $\boldsymbol{R}_{xx}(t)$ is computed by $E[\boldsymbol{X}(t) \cdot \boldsymbol{X}^H(t)]$ where $\boldsymbol{X}(t)$ is the data vector received at array antenna, $E[\cdot]$ is the expectation operator, and the superscript H denotes Hermitian transposition. Actually, the finite average of the correlation matrix is used to approximate a stochastic process. Then the spatial smoothing process suppresses the correlation between incident signals, which enables the estimation when the signals are correlated with one another. The correlation matrix is decomposed into signal and noise sub-space eigenvectors by EVD, and the DOAs can be found by computing the angular spectrum with inner product of noise sub-space and array mode vectors [1].

It seems to be not difficult to implement the computation of correlation matrix, spatial smoothing filter and spectrum synthesis with dedicated circuit using any fast algorithm, thanks

0-7803-7589-0/02/$17.00 ©2002 IEEE

to the simplicity of their logics. But especially EVD computation is not so simple but rather complex. Generally it is thought that the EVD process has 30 $\sim$ 50% of the whole computational load of DOA Estimation. In fact, there are many algorithms for EVD problems but they are just numerical solutions for serial processing on general-purpose computers. Therefore it is necessary to modify and reconstruct the serial algorithm to be suitable for parallelism of dedicated circuit in order to meet the performance requirement for the prcatical use of an adaptive antenna in the next generation communication system.

III. Cyclic Jacobi Method

This section describes the basic principle of Cyclic Jacobi, one of the EVD computation methods. It can be implemented with the simple iterative process of plane rotations. Cyclic Jacobi method computes symmetric eigenvalue problems by applying a sequence of orthonormal rotations to the left and right sides of the target $N \times N$ matrix $\boldsymbol{R}$ as

$$\begin{array}{c} \boldsymbol{E}^T \cdot \boldsymbol{R} \cdot \boldsymbol{E} = \boldsymbol{D}, \\ \left(\begin{array}{ll} \because & \boldsymbol{E} = \boldsymbol{J}_1 \cdot \boldsymbol{J}_2 \cdot \boldsymbol{J}_3 \cdots, \\ & \boldsymbol{J} = \boldsymbol{W}_{12} \cdot \boldsymbol{W}_{13} \cdot \boldsymbol{W}_{N-1,N} \end{array} \right) \end{array} \tag{1}$$

where $\boldsymbol{W}_{pq}$ is an orthonormal plane rotation over an angle θ in the (p, q) plane whose elements are $w_{pp} = \cos\theta$, $w_{pq} = \sin\theta$, $w_{qp} = -\sin\theta$, $w_{qq} = \cos\theta$ $(p > q)$, and defined as Eq.(2). $\boldsymbol{J}$ is the multiple rotation of $\boldsymbol{W}_{pq}$'s in the cyclic-by-row manner of (p, q) which is called a Jacobi sweep, and the superscript T and subscript N denote transposition and array length respectively.

$$\boldsymbol{W}_{pq} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & \cos\theta & \cdots & \sin\theta & & \\ & & \vdots & 1 & \vdots & & \\ & & -\sin\theta & \cdots & \cos\theta & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} \tag{2}$$

Theoretically, by the infinite transform of matrix $\boldsymbol{R}$, it is certain that $\boldsymbol{E}$ and $\boldsymbol{D}$ converge into the matrix whose column vectors are composed of eigenvectors and the matrix whose diagonal elements are eigenvalues, respectively. A symmetric matrix $\boldsymbol{R}$ is transformed to $\boldsymbol{R}'$ by plane rotation as

$$\boldsymbol{R}' = \boldsymbol{W}_{pq}^T \cdot \boldsymbol{R} \cdot \boldsymbol{W}_{pq} \tag{3}$$

By the above transform, only p-th and q-th rows and columns of $\boldsymbol{R}'$ are changed as Eq.(3). The optimal rotation angle in a (p, q) plane is determined by Eq.(4). It is the basic strategy of Cyclic Jacobi method that the iterative process of the plane rotation converges (p, q) and (q, p) elements of a target matrix into zero.

$$r'_{pq} = r'_{qp} = 0 \tag{4}$$

In above manner matrix $\boldsymbol{R}$ converges into a diagonal eigenvalue matrix. This is called Cyclic Jacobi method.

In Cyclic Jacobi method, the off-diagonal quantity $S^{(h)}$ is defined by

$$S^{(h)} = \sqrt{\frac{1}{2}\left[\|\boldsymbol{R}^{(h)}\|_F^2 - \sum_{i=1}^{N}\{r_{ii}^{(h)}\}^2\right]}, \tag{5}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Therefore, if $S^{(h)}$ converges into zero, the target matrix $\boldsymbol{R}$ becomes eigenvalue diagonal matrix as

$$\lim_{h\to\infty} S^{(h)} \to 0 \Leftrightarrow \lim_{h\to\infty} \boldsymbol{R}^{(h)} \to diag\left[\lambda_1, \cdots, \lambda_N\right]. \tag{6}$$

On the other hand, the execution of a similarity transform yields

$$\left[S^{(h+1)}\right]^2 = \left[S^{(h)}\right]^2 - \left[\{r_{pq}^{(h)}\}^2 - \{r_{pq}^{(h+1)}\}^2\right]. \tag{7}$$

From Eq.(7), obviously the maximal reduction of $S^{(h)}$ is obtained if $r_{pq}^{(h+1)} = 0$. The condition for an optimal angle, maximal reduction of $S^{(h)}$, is achieved as

$$\theta_{opt} = \frac{1}{2}\tan^{-1}\left[\frac{2r_{pq}}{r_{pp} - r_{qq}}\right] = \frac{1}{2}\tan^{-1}\tau, \tag{8}$$

where $\tau = \frac{2r_{pq}}{r_{pp}-r_{qq}}$ [2][3].

IV. CORDIC Algorithm for Vector Rotation and Computing Arctangent

The CORDIC algorithm is operated in one of two modes [4]. One is the rotation mode and the other is the vectoring mode. In this paper, the rotation mode is used for vector rotations and the vectoring mode for computing the optimal rotation angle by arctangent of τ, as mentioned in Sect.III. For the rotation mode, the CORDIC equations are

$$\begin{array}{l} \left\{ \begin{array}{lcl} x_{i+1} & = & x_i - y_i \cdot d_i \cdot 2^{-i} \\ y_{i+1} & = & y_i + x_i \cdot d_i \cdot 2^{-i} \\ z_{i+1} & = & z_i - d_i \cdot \tan^{-1}\left(2^{-i}\right) \end{array} \right., \\ d_i = -1 \ if \ z_i < 0, \ +1 \ otherwise \end{array} \tag{9}$$

where $d_i = \pm 1$ (direction of rotation) and z is called angle accumulator. It provides the following results after finite number of iterations as much as the bit-length.

$$\begin{array}{lcl} x_n & = & A_n\left(x_0\cos z_0 - y_0 \sin z_0\right) \\ y_n & = & A_n\left(y_0\cos z_0 + x_0 \sin z_0\right) \\ z_n & = & 0 \\ A_n & = & \prod_{i=0}^{n}\sqrt{1+2^{-2i}} \end{array}, \tag{10}$$

where A_n is a computational gain.

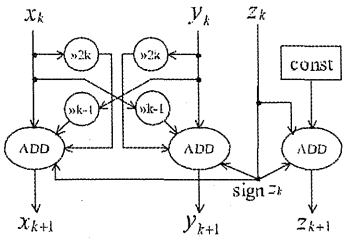


Fig. 1. k-th stage of twice executing rotation (Rotation Mode)

After the last step, the scaling operation must be performed elsewhere in the system. From Eq.(10) the scaling factor K_n is obtained by

$$K_n = \frac{1}{A_n} = \prod_{i=0}^{n} \frac{1}{\sqrt{1+2^{-2i}}};\ Scaling\ Factor. \qquad (11)$$

The CORDIC algorithm performs only shift and add operations and hence it is easy to implement and suitable for dedicated circuit. In the implementation of CORDIC algorithm, there can be various architectures depending on which is more serious consideration, circuit resource or performance. There are a few kinds of architecture for implementing a CORDIC algorithm, for example, bit-serial or bit-parallel, and unrolled or rolled(iterative), and so on [5]. This paper uses the bit-parallel unrolled CORDIC architecture for high performance. It is a cascade structure of the consecutive CORDIC stages, where arctangent values are precomputed and stored at any memory block. The unrolled design consists of only combinatorial logic components and the results can be computed fast with pipelined process.

If the angle accumulator z is initialized with zero ($z_0 = 0$), the arctangent, $\theta = tan^{-1}(y/x)$, is directly computed using the vectoring mode. The result is taken from the angle accumulator as Eq.(12) [5].

$$z_n = z_0 + \tan^{-1}(y_0/x_0) \qquad (12)$$

In CORDIC process, a scaling must be required because the bit length of the processor must be finite. As Eq.(11) the scaling and normalizing using pre-computed scaling value couldn't perform only with shift and add operations. It may be not easy to be implemented with a dedicated circuit.

Twice executing rotation by $z_0/2$ can solve this problem [3]. Let an elementary rotation by angle z_0 be composed of twice executing a rotation by $z_0/2$. Hence Eq.(9) is rewritten as

$$\begin{cases} x_{i+1} &= (1-2^{-2i})x_i - y_i \cdot d_i \cdot 2^{-i+1} \\ y_{i+1} &= (1-2^{-2i})y_i + x_i \cdot d_i \cdot 2^{-i+1} \\ z_{i+1} &= z_i - d_i \cdot \tan^{-1}\left(2^{-i}\right) \end{cases} . \qquad (13)$$
$$d_i = -1\ if\ z_i < 0, +1\ otherwise$$

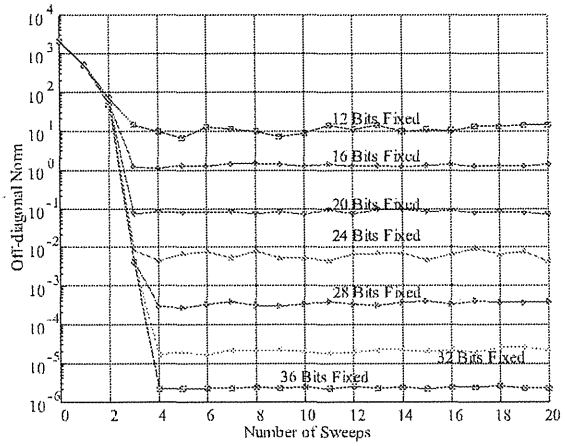


Fig. 2. Reduction of Off-diagonal Norm(6×6 Random Symetric Matrix)

It requires 4-shift and 5-add operations per an iteration stage as shown in Fig.1. The results after a finite number of iterations is as

$$\begin{aligned} x_n &= A_n^{(ii)}\left(x_0 \cos z_0 - y_0 \sin z_0\right) \\ y_n &= A_n^{(ii)}\left(y_0 \cos z_0 + x_0 \sin z_0\right) \\ z_n &= 0 \\ A_n^{(ii)} &= \{A_n\}^2 = \textstyle\prod_{i=0}^{n}(1+2^{-2i}) \end{aligned} . \qquad (14)$$

In Eq.(14), the computational gain $A_n^{(ii)}$ becomes square root free. In addition, twice rotation gives the scaling factor of division operation free as well as square root free as Eq.(15). It is because for a given precision b of the shift and add operations, all factors $(1-2^{-n})$ with $n > b$ do not contribute to $K_n^{(ii)}$. So the scaling factor can be approximated by simplified form as Eq.(15) [3].

$$\begin{aligned} K_n^{(ii)} &= \{K_n\}^2 = \prod_{i=0}^{n} \frac{1}{1+2^{-2i}} = \frac{1}{2}\prod_{i=1}^{n} \frac{1-2^{-2i}}{1-2^{-4i}} \\ &\approx \frac{1}{2}\prod_{i=1}^{n/4}\left(1-2^{-(4i-2)}\right) \end{aligned} \qquad (15)$$

Eq.(15) can be also computed only with shift and add operations. Fig.1 shows the k-th stage of the twice rotation. In this paper twice executing rotation unrolled architecture is used for efficient circuit implementation.

V. Examination of CORDIC-Jacobi EVD Processor with Fixed Point Operations

This section describes the practical implementation of Jacobi EVD processor based on CORDIC with a fixed-point arithmetic operation. The required number of Jacobi sweeps, the appropriate precision for the desired accuracy and applicability to MUSIC DOA estimator are examined. The first

thing to determine is how many times of Jacobi sweeps can achieve the desired convergence.

Fig.2 illustrates the reduction of off-diagonal norm to the iteration number of Jacobi sweeps with a 6×6 random real symmetric matrix of 12-bit precision. In case of 4-element linear array antenna, the dimension of the correlation matrix is 6×6 after spatial smoothing with 3 subelements. The off-diagonal reduction is defined as Eq.(5) in Sect.III and the lower value means that it is closer to the diagonal matrix. The 32-bit floating-point operation on general computers by C language converges to the machine zero within a given precision after several number of Jacobi sweeps, but Fig.2 shows that fixed-point operations do not converge but only keep on vibrating after around 4 Jacobi sweeps regardless of the precision from 12 to 36 bits. This is caused by the limited-precision of the fixed-point operation. At the cost of computation accuracy, the fixed-point operation achieves simpler circuit impelementation, high performance and low power consumption. Without using additional convergence decision circuit, fixing the number of sweeps by 4 cannot only be a proper choice, but more computations must be needless for efficiency in hardware resource. Since the finite number of operation determined in advance provides the same computation time in any cases, the realtime processing can be realized.

The next thing to examine is the precision of fixed-point operation. To validate fixed-point operation, the accuracy within allowable error range must be guaranteed. Eq.(16) yields the error ratio where $\boldsymbol{v}$'s are the vectors whose elements consist of eigenvalues computed in respective subscripted ways and $\|\cdot\|$ denotes vector norm where the error between vectors is defined as

$$\text{Error} = \frac{\|\boldsymbol{v}_{float} - \boldsymbol{v}_{fixed}\|}{\|\boldsymbol{v}_{float}\|}. \tag{16}$$

Fig.3 shows the error ratio of fixed-point operations with various bit-lengthes on the CORDIC-Jacobi EVD processor with respect to the 32-bit floating-point operation on general computers. Fig.3 uses an 6×6 random real symmetric matrix of 12-bit precision as an input matrix. Of course, the longer is the given precision in the fixed-point arithmetic, the more accuracy we can achieve. When it is implemented with 16-bit precision, it has several % of error for the floating-point operation, but in reality, around 16-bit precision is desired for practical uses. In this paper, the processor's computational load and the bit-length are 4 Jacobi sweeps and 16-bit fixed-point operation respectively. Instead of floating point arithmetic, the computation accuracy may be doubtful, but as shown in Fig.4 the simulation results are quite satisfactory. In Fig.4, DOA estimation is performed by spectral MUSIC method assuming that any 2 electromagnetic waves arrive at 4-element array antenna. The EVD computations are performed with 32-bit floating-point operation on a general computer by C language and 16-bit fixed-point operation on proposed design, Jacobi EVD computer based on CORDIC. Except for EVD computation, all the other processes of MUSIC method such as correlation matrix, spartial smoothing and MUSIC specturm were computed on general computer with floating point operation. Fig.4 (a) and (b) show the results respectively.

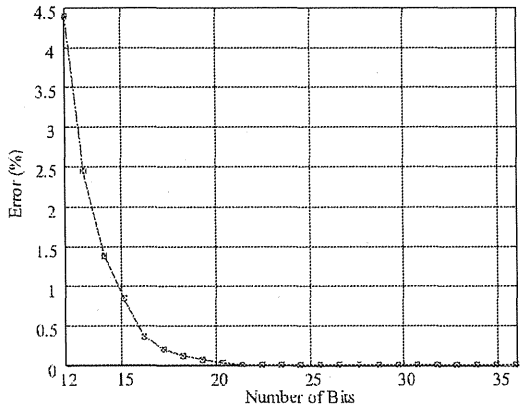


Fig. 3. Error Ratio of fixed-point operations to 32-bit floating poing operation(6×6 Random Symetric Matrix)

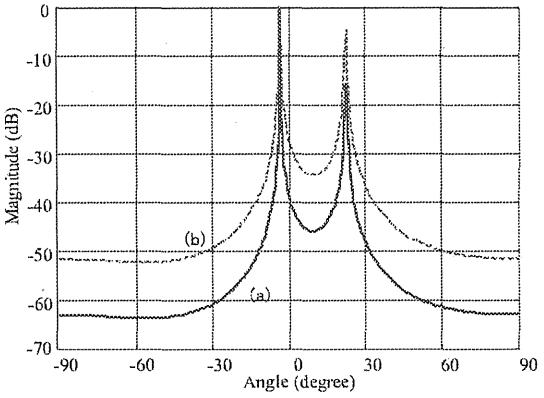


Fig. 4. MUSIC Spectrum in case of 4 elements and 2 incident waves (DOAs are $-5°$ and $20°$); (a) 32-bit floating-point operation with general computer and (b) 16-bit fixed-point operation by CORDIC-Jacobi EVD processor

VI. Hardware Design of EVD Processor for DOA Estimation

Jacobi type EVD is very simple, just a sequence of vector rotations until achieving convergence. From Eq.(8) the optimal rotation angle is determined and then the processor performs the similar transform of correlation matrix $\boldsymbol{R}$ and unitary matrix $\boldsymbol{E}$(initial value is identity matrix $\boldsymbol{I}$) of eigenvectors. After 4 Jacobi sweeps the computation complete, the resulting matrix $\boldsymbol{R}$ converges the diagonal matrix of eigenvalues and $\boldsymbol{E}$ becomes the unitary matrix of eigenvectors.

The EVD processor consists of CORDIC matrix rotators and CORDIC arctangent. With the optimum angle obtained

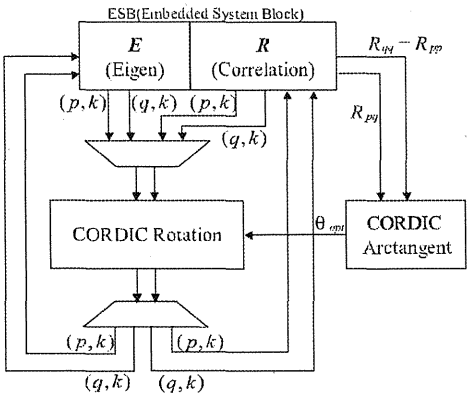


Fig. 5. Architecture of EVD Processor Core

by Eq.(8), the rotation $\boldsymbol{W}_{pq}$ is determined as Eq.(2). The transform $\boldsymbol{W}_{pq}$ in Eq.(3) changes only p-th and q-th rows and columns of the matrix $\boldsymbol{R}$. Therefore the transform can be simplified as

$$\begin{pmatrix} r'_p \\ r'_q \end{pmatrix} = \boldsymbol{w}_{pq}^T \cdot \begin{pmatrix} r_p \\ r_q \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} r_{p1} & \cdots & r_{pp} & \cdots & r_{pq} & \cdots & r_{pN} \\ r_{q1} & \cdots & r_{qp} & \cdots & r_{qq} & \cdots & r_{qN} \end{pmatrix}, \quad (17)$$

where $\boldsymbol{r}_k$ and $\boldsymbol{w}_{pq}$ denote the k-th row vector of the matrix $\boldsymbol{R}$ and the (p, q) plane rotation that is a submatrix of $\boldsymbol{W}_{pq}$ in Eq.(2), respectively. Thanks to the symmetry of the matrix $\boldsymbol{R}$, the right side transform in Eq.(3) yields the same result, so it is not necessary any more if the second rotations of only two vectors $[r_{pp}\ r_{qp}]^T$ and $[r_{pq}\ r_{qq}]^T$ are performed one more time. In the CORDIC matrix rotator, using dedicated circuit's parallelism multiple CORDIC vector rotators are performed simultaneously. It takes $N(N-1)/2$ matrix rotations per one sweep to compute $\boldsymbol{R}$ and $\boldsymbol{E}$ respectively. Fig.5 illustrates the architecture of the EVD processor core.

VII. Computational Load

Basic arithmetic operation of EVD processor consists of only shifts and adds. In twice executing rotation CORDIC stages, the computational load yields

$$\left(4B + \frac{1}{4}B\right)\ Shifts\ and\ \left(5B + \frac{1}{4}B\right)\ Adds, \quad (18)$$

where B is the bit length and $1/4B$ is approximately taken for scaling operation from Fig.1 and Eq.(15). The number of vector rotations required computing both eigenvalues and eigenvectors is $4N(N-1)(N+2)$. It is because $\boldsymbol{E}$ performs 4 $\boldsymbol{J}$'s, $\boldsymbol{J}$ consists of $N(N-1)/2$ $\boldsymbol{W}$'s and $\boldsymbol{W}$ requires $(N+2)$ vector rotations from Eq.(1), where N is the double number of the length of array antenna (if the length of antenna array is N, $N \times N$ correlation matrix of complex numbers is converted into the extended form of $2N \times 2N$ matrix of only real numbers [2]). By addition of arctangent, therefore total computational load of CORDIC-Jacobi EVD processor is $\{4N(N-1)(N+2)+1\} \times (\frac{17}{4}B\ Shifts + \frac{21}{4}B\ Adds)$.

Considering high-speed mobility under the higher frequency area of the next generation communication, it is very difficult to realize the required performance for fading free system with general purposed processor. If DOA estimation of incident waves and beamforming toward their directions are complete within that period, fading free system can be realized. This proposed EVD processor is a combinatorial logic circuit, and hence the improvement of the performance by pipeline scheduled processing can be expected. At present, it is said that the advance of circuit technology can offer high speed operation of general combinatorial logic circuit. The EVD is the most dominant process in the whole processing load from DOA finding to beamforming. This fast parallel computation processor can provide efficient use.

VIII. Conclusion

In this paper the circuit design of fast EVD computation processor for MUSIC DOA estimator was proposed. It uses CORDIC based Jacobi method and it is suitable for hardware implementation for realtime processing. Taking the practical uses in wireless communication into consideration, it is desired that arithmetic processor should perform fixed-point operation with appropriate precision below 16-bit. Adopting fixed-point arithmetic causes some error but makes the implementation easy and hence the high performance and low power consumption can be achieved. In addition, the functionality for the application of spectral MUSIC method was validated.

References

[1] R. O. Schmidt, "Multiple Emitter Location and Signal Parameter Estimation," IEEE Trans. on Antennas & Propagation, vol. AP-34, no. 2, pp. 276~280, Mar 1986

[2] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, Numerical Recipes in C, The Art of Scientific Computing, Second Edition, Cambridge University Press, 1992

[3] J. Gotze, S. Paul, and M. Sauer, "An Efficient Jacobi-like Algorithm for Parallel Eigenvalue Computation," IEEE Trans. on Computer, vol. 42, no. 9, pp. 1058~1065, Sep. 1993

[4] J. M. Muller, Elementary Functions: Algorithms and Implementation, Birkhauser, 1997

[5] Andraka R. "A survey of CORDIC algorithms for FPGA based computers," Proceeding of the 1998 ACM/SIGDA sixth international symposium on Field Programmable Gate Arrays, pp 192~200, Monterey, CA, Feb. 1998

FIXED-POINT DIGITAL PROCESSING OF RECURSIVE LEAST-SQUARE ALGORITHM: TOWARD FPGA IMPLEMENTATION OF MMSE ADAPTIVE ARRAY ANTENNA

Naoya Matsumoto, Koichi Ichige, Hiroyuki Arai

Department of Electrical and Computer Engineering, Yokohama National University
79-5 Tokiwadai, Hodogaya-ku, Yokohama, 240-8501 Japan
E-mail: naoya@ichilab.dnj.ynu.ac.jp, koichi@ynu.ac.jp, arai@ynu.ac.jp

ABSTRACT

In this paper, we try to implement RLS (Recursive Least Square) algorithm on FPGA with fixed-point operation to be used in 4-elements MMSE (Minimum Mean Square Error) adaptive array antenna. RLS algorithm is known to the fast convergence property and broadly used in the optimization process of MMSE adaptive array. An inherent problem of RLS algorithm is the large computational cost. Hence it was difficult to implement on fixed-point DSP (Digital Signal Processor) or FPGA (Field Programmable Gate Array). However, the computation must be simplified for the case with small number of array elements. Through some simulations with 4-elements array antenna, we confirm that RLS algorithm can be accurately implemented on fixed-point digital processors.

1. INTRODUCTION

MMSE adaptive array antenna optimizes the array weight parameters in order to discriminate a desired wave from interferences by digital beamforming, and is now broadly used in mobile communication. In MMSE adaptive array, RLS algorithm is often used for optimization procedure due to its fast convergence property [1],[2]. But its computational cost is still large for digital implementation [3]. Since RLS algorithm contains many vector and matrix operations in optimization procedure, FPGA would be suitable to implement RLS algorithm.

We have already implemented some adaptive algorithms on FPGA [4]–[6]. In this paper, we aim at implementing RLS algorithm used in MMSE adaptive array on FPGA, and study the case of 4-elements array antenna . Throughout some simulations, we confirm that RLS algorithm for 4-elements adaptive array can be accurately implemented on fixed-point digital processors.

2. SYSTEM CONFIGURATION

Consider 4-elements MMSE adaptive array as shown in Fig. 1. Incident waves are received at array elements, then downconverted to baseband signals, digitized by A/D converters, and finally sent to FPGA. RLS algorithm implemented on FPGA determines the optimum weight. The baseband signals are synchronized with the reference signal which is known beforehand and memorized [7].

3. FIXED-POINT OPERATION OF RLS ALGORITHM

In this section, we see that RLS algorithm for 4-elements adaptive array can be implemented on fixed-point digital processors. Specifications of fixed-point simulation is shown in Table 1. We assume

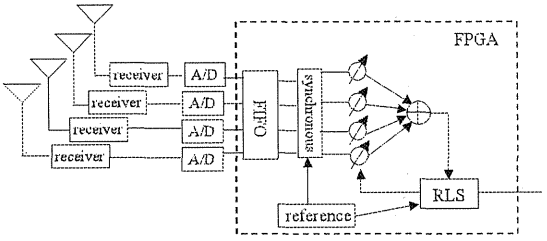


Figure 1: Circuit diagram

Table 1: Specifications of simulation

Array form	$\lambda/2$ uniform linear array
Number of elements	4
Incident waves	2 correlated waves
DOAs	30[deg] (desired) -10[deg] (interference)
SNR	5[dB]
forgetting factor	$q = 0.8$

the situation that 2 correlated incident waves arrive at 4-elements array antenna.

3.1. Case of sinusoidal incident waves

Here we study the case of sinusoidal incident waves. RLS algorithm is generally implemented by the double precision floating-point operation [1]. Figure 2 depicts the convergence property of mainbeam (the beam for the direction of desired wave) and null-beam (for that of interference wave) calculated by RLS algorithm with the double precision floating-point operation. We confirm that both of the main- and null-beams converge very quickly.

In order to use fixed-point operation on FPGA, all the input signals are normalized to integer values. In case RLS algorithm is executed with fixed-point operation, the convergence property of main- and null-beams deeply relates on the bit length.

Figure 3 illustrates the convergence property of the mainbeam for the cases with 4, 8, 12 and 16bits fixed-point representation including one sign bit. Similarly, Figure 4 shows the convergence property of the nullbeam for the same cases. From Figs.3 and 4, we see that the case of 4 or 8 bit cannot achieve enough accuracy and make the convergence property worse. In cases of 12 and 16 bits, The convergence property is fast enough and close to the floating-point case in Fig.2. From the above discussion, we hereafter use 12-bit A/D converters to input 12-bit integer signals to FPGA.

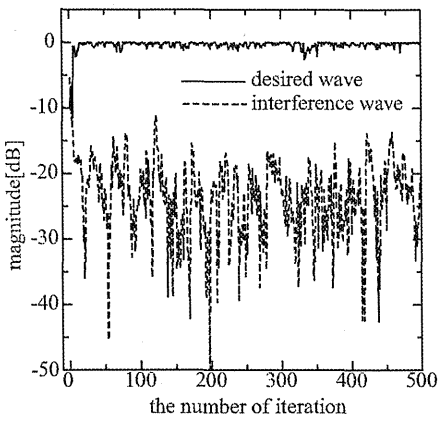


Figure 2: Behaivor of desired & interference waves , floating-point operation

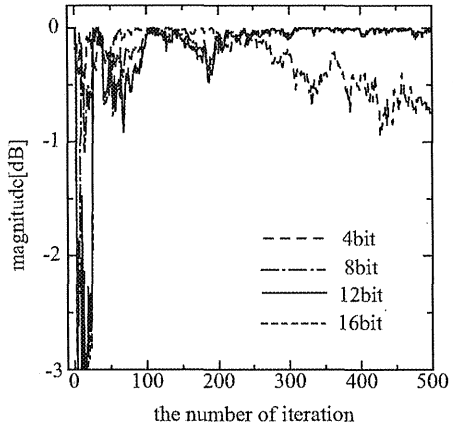


Figure 3: Behaivor of desired wave, fixed-point operation

3.2. Case of modulated incident waves

Next, we test the case of modulated incident waves because $\pi/4$-shift QPSK modulated waves are often used in real communication system. Preserving the condition in Table 1 but with $\pi/4$-shift QPSK modulated incident waves, we again evaluate RLS algorithm with fixed-point operation. The convergence property is drawn in Fig.5. Figure 5 shows that the interference wave can be well suppressed similarly to the case of sinusoidal incident waves.

3.3. Required bit length for multipliers and dividers

If we implement long-bit multiplication or division on FPGA, they require huge number of gates. Therefore, the bit-length of signals must be as short as possible.

The left column of Table 2 represents the originally required bit length for multipliers and dividers. The input signal from A/D converter is with 12-bit only, but some signals become huge and require very long bit in the middle of RLS algorithm. The following operations are adopted in order to reduce the required gates in multipliers and dividers.

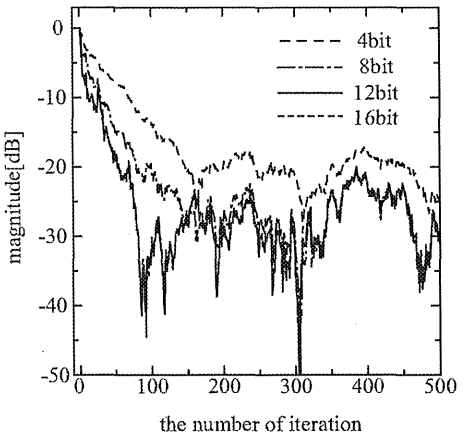


Figure 4: Behaivor of interference wave, fixed-point operation

Table 2: Minimum bit length for multipliers and dividers

original bit length	reduced (1)	reduced (2)
18×12	18×12	16×12
16×12	16×12	16×12
12×31	12×24	12×16
50×50	38×38	16×16
50×31	38×24	16×16
18×31	18×25	16×16
18×34	18×29	16×16
102/101	86/77	38/25

1. For multipliers and dividers, the higher bits (superior digits) become always redundant, in other words, never used. After removing those redundant bits, the required bits for multipliers and dividers are reduced to the center column of Table 2.
2. When we do the multiplication of 32×32bits, for example, the lower bits (inferior digits) don't affect to the higher bits of the multiplication result. Therefore, we can remove the lower bits in advance and execute 16×16bits multiplication instead. After that the removed bit length is added to the result of multiplication. With these simplification process, the required bits can be further reduced to the right column of Table 2.

Figure 6 shows the convergence property of main- and null- beams in the case the above procedure is adopted in multiplications and divisions. We see from Fig.6 that the above reduced-bit operation does not affect to the convergence property.

4. CONCLUSION

We discussed the problems of the fixed-point digital signal processing of RLS algorithm. We confirmed that RLS algorithm for 4-elements adaptive array could be accurately implemented on FPGA with fixed-point operation. Moreover, we consider implementing RLS and synchronous scheme on FPGA of 600,000 gates developed in [6] (as seen in Fig.7).

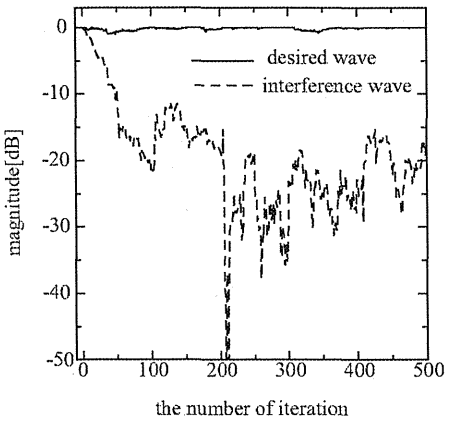


Figure 5: Behavior of desired & interference waves, modulated wave, fixed-point operation

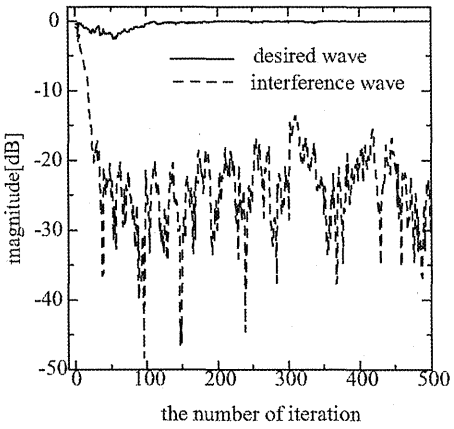


Figure 6: Behavior of desired & interference waves, reduced-fixed-point operation

5. REFERENCES

[1] J.G.Proakis, *Digital Communications: 3rd Ed.*, McGraw-Hill, New York, 1995.

[2] L.C.Godara, "Applications of Anntena Arrays to Mobile Communications, Part 2: Beam-Forming and Direction-of-Arrival Considerations," Proc. IEEE, vol.85, no.8, pp.1195-1245, Aug. 1997.

[3] Application Note: A Smart Antenna System for 3G Wireless Using the MSC8102 DSP Device, Motorora Inc., Dec 2002. available at http://e-www.motorola.com/brdata/PDFDB/docs/AN2383.pdf

[4] A.Suzuki, S.Muramatsu, K.Ichige and A.Arai, "A hardware implementation of LS-CMA adaptive array for high-speed mobile communication", Proc. Int'l Sympo. Personal, Indoor & Mobile Radio Comm., No. MPO2.6, Lisbon, Portugal, Sep. 2002.

[5] M.S.Kim, K.Ichige and H.Arai, "Design of Jacobi EVD Processor Based on CORDIC for DOA Estimation with MUSIC Algorithm", IEICE Trans. Commun., vol.E85-B, no.12, pp. 2648-2655, Dec, 2002.

[6] M.S.Kim, K.Ichige and H.Arai, "FPGA based Fast DOA Estimator using Unitary MUSIC Algorithm", IEICE Trans. Commun., (submitted for Publication).

[7] A.V.Keerthi and J.J.Shynk, "Separation of Cochannel Signals in TDMA Mobile Radio", IEEE Trans. Signal Processing, vol.46, no.10, pp2684-2697, Oct.1987.

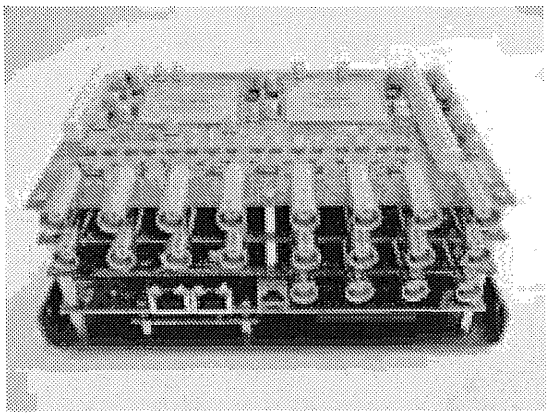


Figure 7: target FPGA

Implementation of FPGA based Fast DOA Estimator using Unitary MUSIC Algorithm

Minseok Kim, Koichi Ichige and Hiroyuki Arai
Graduate School of Engineering, Yokohama National University
Tokiwadai 79–5, Hodogaya-ku, Yokohama, 240–8501, Japan
Email: mskim@arailab.dnj.ynu.ac.jp

Abstract— **This paper proposes the practical implementation of DOA estimation system using FPGA (Field Programmable Gate Array) that is a key technique in the realization of the DOA-based adaptive array antenna for cellular wireless basestation. It incorporates spectral unitary MUSIC (MUltiple SIgnal Classification) algorithm, which is one of the representative super resolution DOA estimation techniques [1]. This paper describes the way of DSP design and real hardware implementation of the unitary MUSIC algorithm. This system achieves the high performance in the eigenvalue decomposition (EVD) and MUSIC angular spectra computation with Cyclic Jacobi processor based on CORDIC (COordinate Rotation DIgital Computer) [2] and spatial DFT (Discrete Fourier Transform), respectively. All DSP functions are computed by only fixed-point operation with finite bit-length to meet the requirements of fast processing and low power consumption due to the simplified and optimized architecture.**

I. INTRODUCTION

Exploiting adaptive array antenna technologies, the wireless system capacity will be dramatically increased and the harmful effects by multipath fading can be combated as well. From the theoretical point of view, many useful algorithms of the adaptive array antenna techniques need DOAs (Directions Of Arrival) of desired and interferer signals in advance. Of course, the practical researches often have used Wiener-solution based algorithms like LMS and RLS employing a temporal reference signal instead of the DOA informations, while the DOA-based systems exploit the exact DOAs in a beamformer to separate the desired signal from interferers spatially. However the DOA-based systems have many advantages over the conventional temporal reference based solutions. For exaple, they are more applicable to the downlink solution thanks to the exact directional information. And the performance of DOA-based beamforming is superior to that of other types of algorithms for small angular spread, while it has time-consuming task of DOA estimation [3]. In order to implement such a DOA-based system, the most time-consuming DOA estimation step shoud be processed as fast as possible. But such processing has been very difficult to realize in the practical systems from the lack of cost effective digital processing devices to solve the hard computational burden. We believe that general Von Neumann architecture processors can never usually meet the requirements of the fast and compact architecture and low power consumption at the same time.

Thus, in this paper, the FPGA based DSP (Digital Signal Processing) design and hardware implementation of the fast DOA estimator will be presented. It can be applied to cellular wireless basestation for DOA-based beamforming and a realtime DOA monitoring system usefully, if it is tuned up appropriately correspoding to the appied environment. It incorporates unitary MUSIC (MUltiple SIgnal Classification) algorithm, which is one of the representative super resolution DOA estimation techniques. MUSIC based algorithm has many advantages in the real hardware implementation due to its simplicity compared with other well-known subspace based techniques like ESPRIT. However, there still remains the computational complication of the complex number arithmetic, which is a great distress to the fast and compact architecture. With a unitary transform, the eigendecomposition of the correlation (or covariance) matrix in the MUSIC algorithm can be solved with real number only [1] [6]. The unitary MUSIC processor (UMP) performs all DSP functions with only fixed-point operation with finite bit-length in order to meet the requirement of fast processing and low power consumption by simplified and optimized architecture. This system performs the fast computation of EVD and MUSIC angular spectra with Cyclic Jacobi processor based on CORDIC (COordinate Rotation DIgital Computer) [2] and spatial DFT (Discrete Fourier Transform), respectively.

II. UNITARY MUSIC DOA ESTIMATOR

MUSIC algorithm is a kind of DOA (Direction Of Arrival) estimation technique based on eigenvalue decomposition, which is also called subspace-based method [5]. It is well known for the implementation simplicity as well as the capability of estimating DOA in much higher resolution than any other conventional algorithms.

We assume the basic model of a narrowband signal $s(n)$. The signals received at K antenna array spaced by half wavelength can be written by linear combination of L incident signals from far-field and white Gaussian noise as

$$\boldsymbol{X}(n) = \boldsymbol{A} \cdot \boldsymbol{S}(n) + \boldsymbol{N}(n), \tag{1}$$

where $\boldsymbol{S}(n)$ is a signal matrix and $\boldsymbol{X}(n)$ is a $K \times 1$ vector of array output at any sampling time n, which is called a snapshot. The columns of $\boldsymbol{A} = [\boldsymbol{a}(\theta_1), \boldsymbol{a}(\theta_2), \cdots, \boldsymbol{a}(\theta_L)]$ are the steering vectors. The correlation matrix of $\boldsymbol{X}(n)$ is given by

$$\boldsymbol{R}_{xx} = E[\boldsymbol{X}(n)\boldsymbol{X}^H(n)] = \boldsymbol{A}\boldsymbol{R}_{ss}\boldsymbol{A}^H + \sigma^2 I, \tag{2}$$

0-7803-7955-1/03/$17.00 (C) 2003 IEEE

but, in reality, the correlation matrix is approximated by uniform averaging by some number of snapshots as

$$\boldsymbol{R}_{xx}(n) \approx \frac{1}{snapshots} \sum_{n=1}^{snapshots} \boldsymbol{X}(n)\boldsymbol{X}^H(n), \quad (3)$$

where $E[\cdot]$ and superscript H denote expectation and hermitian operator, respectively. And $\boldsymbol{R}_{ss} = E[\boldsymbol{S}(n)\boldsymbol{S}^H(n)]$ is signal covariance matrix and σ^2 is noise variance.

Since $\boldsymbol{R}_{xx}$ is a positive definite hermitian matrix, the correlation matrix of $\boldsymbol{X}(n)$ can be decomposed to signal and noise subspaces. The noise subspace eigenvectors of corresponding eigenvalue of σ^2 lies in the nullspace of $\boldsymbol{A}^H$, that is to say, they are orthogonal to the signal subspace, and eventually orthogonal to the steering vectors of incident signals. Using this principle, the MUSIC spectrum is computed using noise subspace eigenvectors as

$$P_{MU} = \frac{\boldsymbol{a}^H(\theta)\boldsymbol{a}(\theta)}{\boldsymbol{a}^H(\theta)\boldsymbol{E}_N\boldsymbol{E}_N^H\boldsymbol{a}(\theta)}, \quad (4)$$

where $\boldsymbol{E}_N$ is the matrix whose column vectors are noise subspace eigenvectors. In the result spectrum of Eq.(4), the peaks appear at the corresponding angles to the DOAs of incident signals, since they are reciprocal of the nulls. Exploiting null steering toward DOAs makes the super-resolution estimation available.

Generally the correlation matrix in Eqs.(2)-(3) is complex-valued. It is certain that the EVD with complex-valued correlation matrix should be high computational burden. Reducing the computational complexity via unitary transform allows real-valued eigenvalue decomposition of the transformed real number correlation matrix [1]. Since the EVD process has a large portion of the whole computational load of MUSIC based algorithms, the real-valued eigenvalue decomposition can provide the fast and compact computation. If the steering vectors are arranged conjugate centro-symmetric as

$$\boldsymbol{a}(\theta_l) = \left[e^{j\pi\frac{(K-1)}{2}\sin\theta_l}, \cdots, e^{-j\pi\frac{(K-1)}{2}\sin\theta_l}\right]^T, \quad (5)$$

the correlation matrix $\boldsymbol{R}_{xx}$ becomes centro-Hermitian. The real-valued correlation matrix $\hat{\boldsymbol{R}}_{xx}$ can be obtained via any unitary transform $\boldsymbol{Q}$ as

$$\hat{\boldsymbol{R}}_{xx} = Re\{\boldsymbol{Q}^H\boldsymbol{R}_{xx}\boldsymbol{Q}\}, \quad (6)$$

The unitary transform $\boldsymbol{Q}$ can be chosen as

$$\begin{aligned} \boldsymbol{Q} &= \frac{1}{\sqrt{2}}\begin{pmatrix} \boldsymbol{I} & j\boldsymbol{I} \\ \boldsymbol{I}\boldsymbol{I} & -j\boldsymbol{I}\boldsymbol{I} \end{pmatrix}, \\ \boldsymbol{Q} &= \frac{1}{\sqrt{2}}\begin{pmatrix} \boldsymbol{I} & 0 & j\boldsymbol{I} \\ \boldsymbol{0} & \sqrt{2} & -j\boldsymbol{0} \\ \boldsymbol{I}\boldsymbol{I} & 0 & -j\boldsymbol{I}\boldsymbol{I} \end{pmatrix}, \end{aligned} \quad (7)$$

according to the even and odd number of arrays respectively, where the vector $\boldsymbol{0} = [0, 0, \cdots, 0]^T$, and $\boldsymbol{I}$ and $\boldsymbol{I}\boldsymbol{I}$ are the identity matrix and column flipped identity matrix in the left-right direction, respectively. In Eq.(6), the selection of the real part only provides FB (Forward-Backward) averaging [1].

III. DSP Design Concepts for Dominant Procedures

The unitary MUSIC computational flow is involved in 4 steps largely; Estimation of correlation matrix including unitary transform and spatial smoothing if needed, EVD (Eigen Value Decomposition) of the correlation matrix, Computation of MUSIC spectrum and 1-dimensional peak search (local maximum detection). In this section, the DSP concepts for the dominant procedures in the unitary MUSIC algorithm will be described.

A. Eigenvalue Decomposition via CORDIC based Jacobi Processor

In our former work, the circuit design of EVD computation processor for MUSIC DOA estimator was studied [2]. It used CORDIC based Jacobi method, and it was suitable for hardware implementation for fast parallel processing. Cyclic Jacobi processor computes real symmetric eigenvalue problems by applying a sequence of orthonormal rotations to the left and right sides of the target matrix (unitary tranformed $K \times K$ real symmetric correlation matrix $\boldsymbol{R}_{yy}$) as

$$\begin{aligned} &\boldsymbol{E}^T \cdot \boldsymbol{R}_{yy} \cdot \boldsymbol{E} = \boldsymbol{D}, \\ &\left(\because \quad \boldsymbol{E} = \boldsymbol{J}_1 \cdot \boldsymbol{J}_2 \cdot \boldsymbol{J}_3 \cdots, \right. \\ &\left. \quad \boldsymbol{J} = \boldsymbol{W}_{12} \cdot \boldsymbol{W}_{13} \cdot \boldsymbol{W}_{K-1,K} \right) \end{aligned} \quad (8)$$

where $\boldsymbol{W}_{pq}$ is an orthonormal plane rotation over an angle θ in the (p, q) plane whose elements are $w_{pp} = \cos\theta$, $w_{pq} = \sin\theta$, $w_{qp} = -\sin\theta$, $w_{qq} = \cos\theta$ $(p > q)$. $\boldsymbol{J}$ is the multiple rotation of $\boldsymbol{W}_{pq}$'s in the cyclic-by-row manner of (p, q) which is called a Jacobi sweep, and the superscript T and subscript K denote transposition and array length, respectively. This processor employed the hardware friendly CORDIC (COordinate Rotation DIgital Computer) algorithm for vector rotators and arctangent computers to solve Eq.(8), which were the basic processing unit. Because the fixed-point operation is applied, of course there exist the approximation errors. But when it was implemented with above 16-bit precision, we could get the reasonable performance. In this EVD processor, the number of iterations and the computation bit-length are 4 Jacobi sweeps and 16-bit long, respectively.

B. MUSIC Spectrum Computation via Spatial DFT

In specral MUSIC algorithm, in order to find out DOA angles of the incident signals, the angular spectrum should be computed after the EVD step. Of course, there exist another alternative solutions for direction finding problem in MUSIC based algorithms. It is the technique based on root finding of the MUSIC polynomial, which called root-MUSIC [6]. However complex number coefficient polynomial is very complicated and not suitable to solve with the dedicated circuit computer with the fixed-point operation like FPGAs.

For fast DSP implementation on FPGAs, a simple iterative algorithm should be the best solution. To compute the angular MUSIC spectrum, spatial DFT (Discrete Fourier Transform) technique is very attractive due to the well-known performance guarantee as well as the simplicity. This section will describe

how to apply DFT to the computation of the MUSIC angular spectrum.

The simple continuous spatial signal model of ULA (Uniform Linear Array) is given typically by

$$x_d = U(t) \cdot \exp(-j2\pi \cdot f_{spa} \cdot d), \tag{9}$$

where $U(t)$ includes all time varying components and complex amplitude, and d and f_{spa} are the distance from the first reference antenna element and spatial frequency of $sin\theta/\lambda$, respectively. By applying spatial P-point DFT, the discrete spatial frequency distribution function can be obtained by

$$X_d[k] = \frac{1}{P} \sum_{m=0}^{P-1} x_d[m] \cdot e^{-j\frac{2\pi}{P}m \cdot k}, \tag{10}$$

where k and m is the indices of the discrete spatial frequency and discrete distance, respectively. When the antenna spacing $D_{spacing} = \lambda/2$, eventually the discrete wavefront θ can be computed from the relation of Eqs.(11)-(12).

$$f_{spa,discrete} = \frac{\sin\theta}{\lambda} = \frac{k}{P \cdot D_{spacing}} \tag{11}$$

$$\theta = \sin^{-1}\left(\frac{k}{P/2}\right) \tag{12}$$

From above, in the MUSIC algorithm, it is certain that the spatial DFT of the noise subspace eigenvectors as Eq. (10) provide the distribution of the spatial frequency. But if DFTed with only a few spatial samples of antenna array, the resolution of the spatial spectrum becomes very coarse. Thus any estimation will not be available from the coarse spectrum. In that regards, the interpolation of the spectrum should be taken into consideration. According to digital signal processing theory, the DFT spectrum can be generated fine and smoothly by adding a few number of zeroes to the spatial data of the noise eigenvector elements. The spectrum generated by the spatial DFT is completely equivalent to that by steering main-beam toward whole directions as Eq. (4) mentioned in Sect.II.

Instead of finding peaks in the MUSIC spectrum written in Eq. (4), local minima (LM) detection of the DFT spectrum as Eq. (10), which is equivalent to the denominator of Eq. (4), can be applied for implementation simplicity. Rearranging the spectrum of Eq. (10) the concrete discrete wavefronts (DOAs) are obtained from Eq. (12) by

$$\theta = \sin^{-1}\left(\frac{l - P/2}{P/2}\right), \tag{13}$$

As shown in Eq. (13), the discrete wavefronts from spatial DFT spectrum are not uniformly spaced. That, however, may be of no concern in the practical sectorized basestation configuration. From Eq. (13), θ is an inverse sinusoidal function of l. In the region between -30 to 30 degrees, the function of Eq. (13) can be approximated by linear function whose gradient is given by derivative of Eq. (14) at $l = P/2$ (0 degree) as

$$\left.\frac{d\theta}{dl}\right|_{l=P/2} = \frac{1}{P/2} rad = 0.4476\,\mathrm{deg}. \tag{14}$$

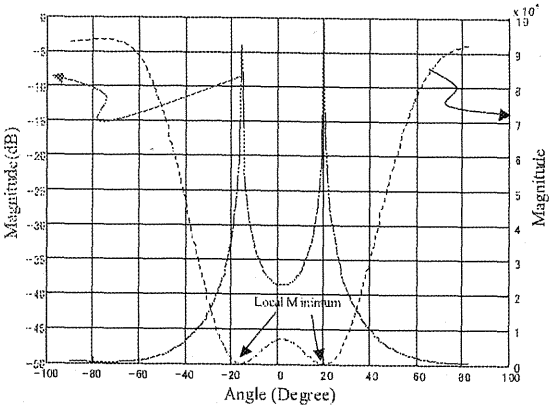


Fig. 1. Hardware Level Simulation Result of MUSIC Spectrum and the reciprocal (4 antennas, coherent 2 waves at -15 and 20 degrees, same powers, SNR=10dB)

In this linear region, the angular spacing can be regarded as almost uniform and the estimation resolution is inversely proportional to DFT length P. When P is 256, the estimation resolution is about 0.4476 degree.

IV. Practical DSP Design and Hardware Level Simulation

In this time, DOA estimation under stationary case without any fading was assumed for the simple implementation, and the system was designed to classify highly correlated (coherent) signals via spatial smoothing technique for easy experiment with single wave source only. In unitary MUSIC, with the forward only spatial smoothing of the correlation matrix, backward spatial smoothing can be achieved simultaneously as mentioned in Sect.II. The first step of the unitary MUSIC procedures is to transform the input data vector $\boldsymbol{X}$ to $\boldsymbol{Y}$ with a unitary transform $\boldsymbol{Q}$ as written by

$$\boldsymbol{Y}_i = \boldsymbol{Q}^H \boldsymbol{X}_i, \tag{15}$$

where $\boldsymbol{X}_i$ and $\boldsymbol{Y}_i$ are the sub-vectors and the corresponding transformed sub-vectors divided by M for spatial smoothing, respectively. From above unitary transform, the correlation matrix $\boldsymbol{R}_{yy}$ can be obtained by

$$\begin{aligned} \boldsymbol{R}_{yy}(n) &= \beta \boldsymbol{R}_{yy}(n-1) \\ &+ (1-\beta)\sum_{i=1}^{M} Re\{\boldsymbol{Y}_i(n)\boldsymbol{Y}_i^H(n)\}, \end{aligned} \tag{16}$$

where β is an appropriate real smoothing factor and M is the number of sub-matrices. It was implemented by first-order IIR (Infinite Impulse Response) exponential averaging filter.

Next step, the correlation matrix is eigen-decomposed by the EVD processor. As described in Sect.III-A, CORDIC based Jacobi EVD processor was incorporated in this system. It had only simple iterative process of vector rotation after obtaining optimal rotation angle. After the EVD step, the reciprocal MUSIC spectrum written in Eq.(4) of Sect.II is computed via

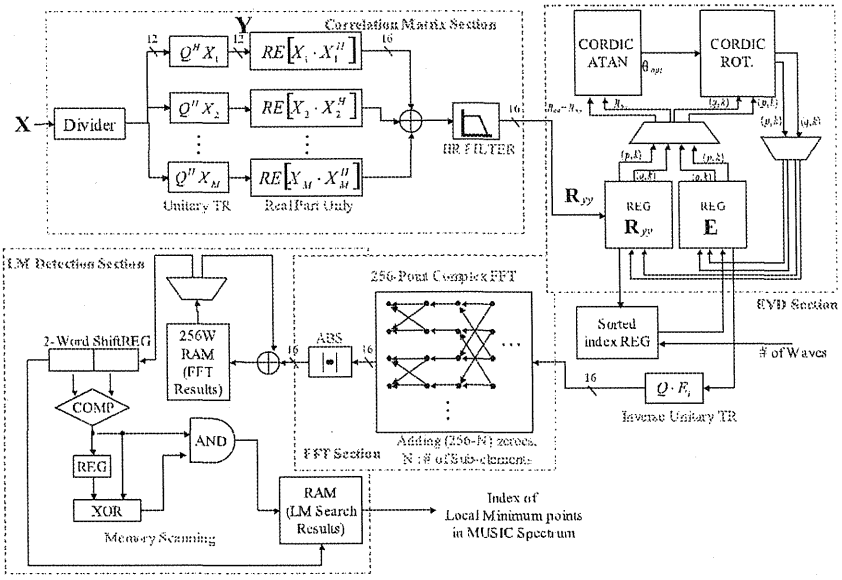


Fig. 2. DSP block Diagram

spatial DFT of the noise eigenvectors returned to complex values by the unitary inverse transform as

$$P_{MU,reciprocal} = \sum_{i=L+1}^{K} |DFT\{\boldsymbol{Q} \cdot \boldsymbol{E}_i\}|^2, \qquad (17)$$

where $\boldsymbol{E}_i$ is the i-th eigenvector belonging to the noise subspace, K and L are the number of antenna elements and the number of waves, respectively.

Figure 1 shows a hardware level simulation result. Hardware level simulations were performed by the direct measuments with only DSP part of a real hardware to evaluate the validity of the system efficiently avoiding working the whole system components. We used the input data made by an offline PC in advance and obtained the results with real hardware operation. With these hardware level simulations, we could verify the function of the digital signal processor. In this simulation, it was assumed that 2 coherent (or fully correlated) waves were impinging at 4 ULA antennas from the DOAs of -15 and 20 degrees, respectively. And two waves were same powers and the input SNR was 10 dB. For the spectrum computation, the FFT (Fast Fourier Transform) of 256 points including 3-spatial data of the noise eigenvector's elements (1 dimension was used for spatial smoothing) and 253 zeroes, was applied.

In final step, we found out the DOAs by the detection of the LM (Local Minimum) points in the reciprocal MUSIC spectrum as shown in Fig.1. It could be easily implemented with whole memory scanning circuit. In this case, as shown in Fig.1, the index numbers of the DFT spectrum corresponding to the LM points below any appropriate threshold level are 95 and 170. With these index values, the concrete discrete wavefronts could be obtained as -14.94 and 19.16 degrees respectively from the relation of the spatial DFT index (or discrete frequency number) and discrete wavefront (DOA angle) as Eq. (13)

V. FPGA Implementation and Performance

Not only the theoretical design, we also tried to implement it on 2 FPGAs (EP20K600, Altera) which had about 1.2 million equivalent gates and 80 Kbytes internal memory block totally. The whole block diagram of the DSP procedures described in previous sections is shown in Fig.2. It is involved in 4 major procedure sections including Correlation Matrix Section, EVD Section, FFT Section and LM Detection Section. The bit precision of every section is also shown in this figure. For the present, it was assumed that the exact number of waves were predetermined and known already from any other process.

In our evaluation testbed as shown in Figs.3 and 4, the RF (Radio Frequency) signals are down-converted to IF (Intermediate Frequency) signals centered at 10 MHz in analog DC (Downconversion) receiver, and then digitized by ADCs (Analog to Digital Converters) at the rates of 40 MSPS. The 4 times oversampled IF signals are digitally down-converted once again to complex baseband and then downsampled by L-times, where L is an appropriate integer number. The FPGAs perform the digital signal processing of the unitary MUSIC algorithm.

Table I illustrates the roughly estimated performance of the dominant core functions, where LEs (Logic Elements) means the number of occupied logic blocks in FPGAs and f_{max} is the maximum clock frequency at which normal operation can be guaranteed. And the minimum computation time t_{min} is calculated by required clks $\times f_{max}$. In this time, we assumed that less than 2 coherent/incoherent waves arrive at only 4-element uniform linear array antenna. For spectrum generation, 256-point radix-4 complex FFT was employed [7],

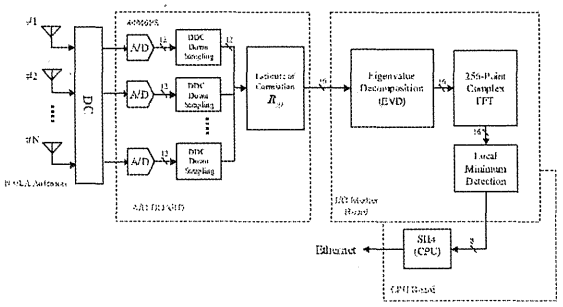


Fig. 3. Whole Evaluation System Configuration

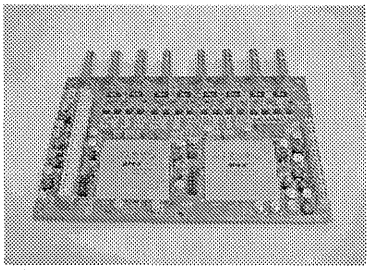


Fig. 4. Appearance of UMP Digital Processing Unit (AD board)

and the FFT with 256 spatial data composed of N elements of the noise eigenvector and $(256-N)$ zeroes interpolates the spectrum fine and smoothly. All computations were performed by fixed-point arithmetic with 12-bit input data from ADCs.

On the other hand, the estimation accuracy of this system depends on so many factors that the proper assessment has some difficulties in detail analysis. For example, the effect of finite bit-length and bit-truncation by scaling in the fixed-point operation, the estimation errors caused by non-uniform discrete wavefront, and so forth. Thus, in order to assess them totally, we would better evaluate the overall accuracy. As [4], the standard deviation of the estimated DOA when the single wave impinging at 0 degree from broadside was one of good overall performance assessment methods of the estimation accuracy. Fig.5 shows the hardware level simulation results, where the squared, diamonded and triangled line at 0, 30 and 60 degrees respectively were processed by UMP, and the circled line was obtained by an offline PC with 64-bit floating-point operation. This measurement included 1000 trials(bursts) data of 32 snapshots. The source wave was a CW signal. In this result, it is clear that the estimation accuracy is below 2 degree if the input SNR is greater than 5 dB in the linear region between -30 and +30 degrees. And it can be also seen that the UMP has a good performance for the offline PC in spite of the compact fixed-point operation.

VI. Conclusion

In this paper, the FPGA design of the fast DOA estimator using the unitary MUSIC algorithm was proposed and its real hardware implementation was also introduced. The unique

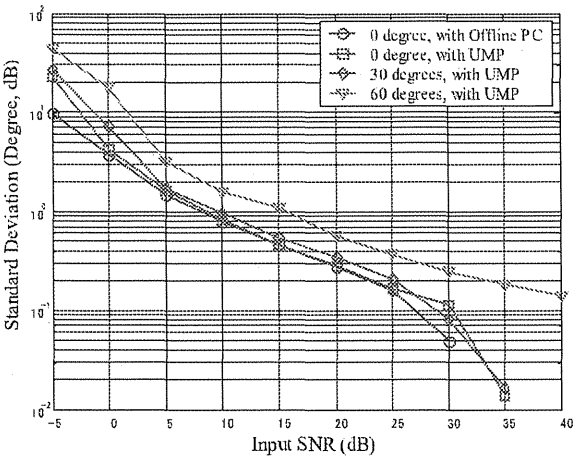


Fig. 5. Standard Deviation of Estimated DOA when single wave impinging at 0, 30 and 60 degrees each from broadside

TABLE I
Core Performances of Dominant Procedures

	Required CLKs	LEs(Logic Elements)	f_{max}(MHz)	$t_{min}(\mu s)$
R_{yy}	32	8301	27.4	1.28
EVD	1836	4045	110	16.69
FFT	1102	2303	114	9.67

features of this system are the fast and compact computation of the EVD and MUSIC angular spectrum generation with Cyclic Jacobi processor based on CORDIC and spatial DFT, respectively. All DSP functions are computed by only fixed-point operation with finite bit-length in order to meet the requirement of fast processing and low power consumption due to the simplified and optimized architecture. we are expecting that this system design provides a useful application of a high speed DOA estimator for the wireless communication.

References

[1] M. Pesavento, A. B. Gershman, and M. Haardt, "On Unitary Root-MUSIC with a real-valued eigendecomposition: A theoretical and experimental performance study," IEEE Trans. Signal Processing, vol. 48, pp. 1306-1314, May 2000.

[2] M. Kim, K. Ichige, H. Arai, "Design of JACOBI EVD Processor Based on CORDIC for DOA Estimation with MUSIC Algorithm", The 13th IEEE Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC2002), Lisbon, Portugal, Sept. 2002

[3] S. Anderson, et al., "An Adaptive Array for Mobile Communication Systems," IEEE Trans. on Vehicular Tech., Vol. 40, No. 1, Feb 1991

[4] A. Kuchar, M. Tangemann and E. Bonek, "A Real-Time DOA-Based Smart Antenna Processor," IEEE Trans. Vehicular Technology, vol. 51, No. 6, pp. 1279-1293, Nov 2002.

[5] R.O. Schmidt, "Multiple emitter location and signal parameter estimation," IEEE Trans. on Antenna and Propag., vol. 34, no. 3, pp. 276-280, March 1986.

[6] M. D. Zoltowski, G. M. Kautz, and S. D. Silverstein, "Beamspace root-MUSIC," IEEE Trans. Signal Processing, vol. 41, pp. 344-364, Jan 1993.

[7] "FFT MegaCore Function User's Guide", Altera Corp., http://www.altera.com/literature/ug/fftug.pdf.

DOA-based Adaptive Array Antenna Testbed System

Minseok Kim[†*] Koichi Ichige[‡] Hiroyuki Arai[‡]
Division of Electrical & Computer Engineering, Yokohama National University
Tel: +81-45-339-4270, Fax: +81-45-338-1157
E-mail: [†*] mskim@arailab.dnj.ynu.ac.jp, [‡] {koichi, arai}@ynu.ac.jp

In recent and future wireless cellular communications, Co-Channel Interference (CCI) is very critical problem. At basestation, adaptive array antenna technology exploiting spatial domain filtering with antenna array and digital signal processing can combat the CCI, thus the system capacity can be increased. Most algorithms for adaptive array antenna system have been studied. Most algorithms usually need directions of arrival (DOAs) of desired and interferer signals in advance. The DOA-based adaptive array antenna systems make use of the estimated DOAs in a beamformer in order to separate the desired signal from interferers spatially. But, of course, there are other types of algorithms that exploit temporal reference and not need the DOA information. The performance of DOA-based beamforming is superior to that of other types of algorithms for small angular spread, while it has time-consuming tasks due to its large amount of computational burden [1].

In this paper, the real hardware basestation testbed system of DOA-based adaptive array antenna is presented. This system includes transmitting part for downlink as well as receiving one for uplink. The efficient beamforming based on the estimated incident signals' DOAs separates only the signal of interest from CCIs. To estimate DOAs, we chose unitary MUSIC (MUltiple SIgnal Classification) algorithm because it can achieve super-resolution estimation and be implemented relatively simple with fast dedicated processor such as FPGA (Field Programmable Gate Array). The schematic diagram of the system architecture and photograph of digital signal processor are illustrated in Figs. 1 and 2, respectively. The geometry of antennas is a uniform linear array consisting of 8 and 4 antennas equally spaced by half wavelength for receiving and transmitting, respectively. In this testbed system, the carrier frequency is 8.45GHz. In the receiving part, the filtered and amplified RF signals are downconverted to low-IF of 10MHz. The resolution and conversion rate of ADCs (Analog to Digital Converters) and DACs (Digital to Analog Converters) are 12 and 14 bits, respectively, and both 40MHz. In the receiving part, the low-IF signals captured by 4-times oversampling scheme are downconverted again to baseband and detected by digital downconverter (DDC) on FPGAs. Reciprocally, in the transmitting part the digital baseband signals are modulated and upconverted by digital upconverter (DUC) on FPGAs, and converted analog low-IF signals by DACs. After a few stages of upconversion to RF, the transmitting signals are emitted from array antenna. In this system, QPSK and $\pi/4$-QPSK modulation is applied [2][3].

In receiving part, uplink processing is performed with downconverted and calibrated complex digital low-IF or decimated baseband signal. Some memories store finite length of symbols, e.g. some timeslots of a frame data. Thus burst-by-burst real-time processing is available. In burst-wise communication system, if assuming that the channels are stationary within burst duration, the burst-by-burst processing will be sufficient to the beamforming. However, considering severe multi-path propagation environment and the channels become non-stationary, it is desired that adaptive array antenna systems should complete beamforming including the DOA estimation within a very short time to track fading. In this system, the dedicated signal processor with FPGAs can be applied in order

0-7803-8197-1/03/$17.00 (c)2003 IEEE

to perform faster time-consuming task such as DOA estimation procedure. General purposed processor (RISC CPU SH4, Hitachi) is also available for any tasks demanding high precision floating point calculation. In order to achieve high-speed computation, this system estimates DOAs with unitary MUSIC processor (UMP) that is performed with only fixed-point operation with finite bit-length on FPGAs [4]. We are expecting that various adaptive antenna technologies can be applied and evaluated usefully on this system.

REFERENCES

[1] S. Anderson, et al., "An Adaptive Array for Mobile Communication Systems," IEEE Trans. on Vehicular Tech., Vol. 40, No. 1, Feb 1991

[2] M. Kim, K. Ichige and H. Arai, "Adaptive Array Antenna Experimental System," IEICE AP Technical Report, AP2002-38

[3] S. Muramatsu, M. Kim, K. Ichige and H. Arai, "A Prototype Transmitter Using FPGA ay," IEICE AP Technical Report, Wireless Communication Workshop, AP2003-14

[4] M. Kim, K. Ichige and H. Arai, "Design of Fast DOA Estimator using Unitary MUSIC Algorithm based on FPGA," 2003 IEICE General Conference, B-1-118, Mar 2003.

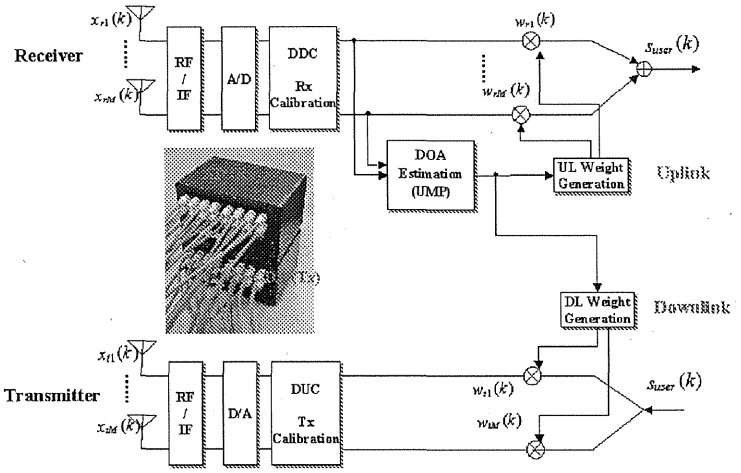


Fig. 1 System Configuration

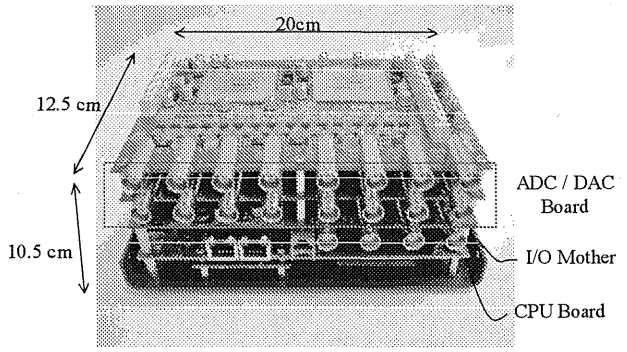


Fig. 2 Digital Signal Processor