社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE

A novel Fast Eigenvalue Decomposition based on Cyclic Jacobi rotation and its application in eigen-beamforming.

Dang Hai PHAM[1] Tomohisha WADA [1]

[1]Faculty of Information Engineering, University of the Ryukyus 1Senbaru Nishihara, Okinawa, 903-0129 Japan

E-mail: [1]phdang@lsi.ie.u-ryukyu.ac.jp, wada@ie.u-ryukyu.ac.jp

Abstract The eigenspace in general represents the spatial characteristics of the propagation channel. Eigen-beamforming is one approach, which use the knowledge of the eigenspace of the channel at the transmitter or receiver to achieve a MIMO-capacity gain. However, the conventional EVD (Eigenvalue Decomposition) must repeat a high computation load every time the snapshot of correlation matrix is taken. In this paper, we propose an efficient tracking method based on the exact Jacobi method. Not only eigenspaces but also noisespaces can be obtained by this proposed method. The proposed method is applicable to slowly time-variant system. In such condition, the proposed method drastically reduces computation load compared with the conventional EVD method based on the Jacobi method.

Keyword Eigenbeamforming, EVD, Jacobi.

1. Introduction

The eigenspace in general represents the spatial characteristics of the propagation channel. Eigen-beamforming is one approach, which use the knowledge of the eigenspace of the channel at the transmitter or receiver to gain a capacity in MIMO scenario [1]. The other application of subspace tracking is a class of channel parameter's estimation like MUSIC (MUltiple Signal Classification) and ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) [2]. However, these applications must repeat the high-load computation involving the eigenvector decomposition of a correlation matrix every time a snapshot is taken. Therefore, it takes a very long time to obtain the eigenvalue/vector matrix when the size of correlation matrix is large. In addition, it is quite inefficient in the case that the channel parameters vary slowly in time.

To solve this problem, BiSVD (Bi-Iteration Singular-Value Decomposition) and PAST (Projection Approximation Subspace Tracking) have been proposed and investigated, which are typical methods of successive updating (tracking) eigenvectors in the signal subspace of the correlation of a correlation matrix.

In this paper, based on the exact Jacobi rotation algorithm [3], we propose an efficient tracking method in subspaces. This method is capable of tracking the system parameters varying slowly. The computation loads are drastically reduced. Further, the calculation of square roots in the exact Jacobi method is eliminated by using a piece-wise linear approximation and a Newton method. The rest of the paper is organized as follow: Section 2 introduces the Jacobi method and the inexact Jacobi method. The fast tracking algorithm is given in Section 3. The simulation results are introduced in Section 4. Finally, the conclusions are given in Section 5.

2. The Jacobi method

2.1. The exact Jacobi method

Assuming a Hermitian matrix $\mathbf{R}_0(k)=\{a_{ij}\}\in\mathbf{C}^{L\times L}$ at time $t=kT_S$, of which the eigenvalues and eigenvectors shall be determined. One method to solve the eigenvalues and eigenvectors is symmetric matrix is the Jacobi method. The idea of the Jacobi method is that it applies series of similarity transformations to diagonalize $\mathbf{R}_0(k)$. The i-th transformation leads to

$$\mathbf{R}_{i+1}(k)\leftarrow\mathbf{J}_i^H(k)\mathbf{R}_i(k)\mathbf{J}_i(k) \quad (1)$$

The transformation matrix $\mathbf{J}_i(k)$ is an identity matrix of same size as $\mathbf{R}_0(k)$ in which entries at the matrix position (p,p), (p,q), (q,p) and (q,q) are replaced by $\cos(\phi)e^{j\varphi}$, $\sin(\phi)e^{j\varphi}$, $-\sin(\phi)$ and $\cos(\phi)$, respectively.

$$\mathbf{J}_i(k)=\mathbf{J}_i(k,p,q,\phi,\varphi)=$$

$$\begin{array}{c} \quad\quad p \quad\quad\quad q \\ \quad\quad \Downarrow \quad\quad\quad \Downarrow \end{array} \quad (2)$$

$$\begin{array}{c} \\ \\ p\Rightarrow \\ \\ q\Rightarrow \\ \\ \end{array} \begin{bmatrix} 1 & 0 & & \cdots & & 0 \\ 0 & & & & & \\ & \cos(\phi)e^{j\varphi} & & \sin(\phi)e^{j\varphi} & & \\ \vdots & & \ddots & & & \vdots \\ & -\sin(\phi) & & \cos(\phi) & & \\ & & & & & \\ 0 & & & \cdots & & 1 \end{bmatrix}$$

One rotation causes the element of position *(p,q)* as a_{pq} to be zero. Note that all matrices $\mathbf{R}_i(k), i=0,1,..$ are Hermitian so that the rotation also eliminate the element *(q,p)* as well. To determine the element of the rotation matrix $\mathbf{J}_i(k)$, the pivot matrix is used such that

$$\begin{bmatrix} a'_{pp} & a'_{pq} \\ a'_{qp} & a'_{qq} \end{bmatrix} = \begin{bmatrix} ce^{j\varphi} & se^{j\varphi} \\ -s & c \end{bmatrix}^H \begin{bmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{bmatrix} \begin{bmatrix} ce^{j\varphi} & se^{j\varphi} \\ -s & c \end{bmatrix} \quad (3)$$

Where $c = \cos(\phi), s = \sin(\phi)$

We start by looking the value of (ϕ, φ) as the rotation angles of $\mathbf{J}_i(k)$, that will make $a'_{pq}=0$.

In fact, we compute $t = \tan(\phi)$ and then calculate rotation parameters c and s from t.

Defining $\sigma=\frac{|a_{pq}|}{|a_{qq}|-|a_{pp}|}$, we solve for t as follows

$$t=\frac{2\sigma}{1+\sqrt{4\sigma^2}} \quad (4)$$

Given t, we can determine c and s using

$$c=\frac{1}{\sqrt{1+t^2}}, \quad s=ct \quad (5)$$

While $e^{j\varphi}$ is calculated as follows

$$e^{j\varphi}=\frac{a_{pq}}{|a_{pq}|} \quad (6)$$

As mentioned above that all the matrices $\mathbf{R}_i(k), i=0,1,..$ are Hermitian. Therefore, one only needs to concentrate on the upper off-diagonal elements of the matrix. After $L(L-1)/2$ rotations, all off-diagonal elements are scanned exactly once. Such a series of $L(L-1)/2$ is called a sweep.

Unfortunately, subsequent rotations might destroy already zeros in $\mathbf{R}_i(k)$. In other words, the element that has been zero in previous rotation might not be zero in next rotation. However, a measure of diagonality is the ratio between off-diagonal energy and the entire matrix energy

$$\delta=\frac{\sum_{p,q}|a_{pq}|^2-\sum_{p}|a_{pp}|^2}{\sum_{p,q}|a_{pq}|^2} \quad (7)$$

After each rotation, the value of δ becomes smaller, indicating that the matrix is more diagonal. In fact, the matrix $\mathbf{R}_i(k)$ will be diagolized after 4 to 6 sweeps.

The product of all rotation matrices yields the approximate eigenvector matrix $\mathbf{U}(k)$ of $\mathbf{R}_0(k)$

$$\mathbf{U}(k)=\prod_i \mathbf{J}_i(k) \quad (8)$$

The diagonal matrix $\mathbf{D}(k)$ that contains real value of eigenvalue of $\mathbf{R}_0(k)$ is derived as

$$\mathbf{D}(k)=\mathbf{U}^H(k)\mathbf{R}_0(k)\mathbf{U}(k) \quad (9)$$

2.2. The inexact Jacobi method

The computations of both t and c involve square roots and divisions operations to avoid in fixed-point arithmetic. We must keep in mind, however, the following requirements before selecting alternative methods to Eq. (4) and (5). First, to eliminate small off-diagonal matrix elements, t must be computed accurately for small value of ϕ. Second, to maintain the rotation matrices' orthogonality, c and s must satisfy $c^2+s^2=1$.

Instead of reducing a_{pq} to zero, we can use the inexact Jacobi method to reduce it to approximately zero. Therefore, only approximation of Eq. 4 is needed. Because a_{pq} will acquire energy as other off-diagonal elements have their energy reduced subsequently, allowing $a'_{pq}\neq 0$ does no harm. We compute a piecewise linear approximation to Eq. 4 by calculating good least square fits of a line segment to each part of σ.

$$t_{approx}=\begin{cases} sign(\sigma), & |\sigma|\geq 2 \\ \sigma/2, & 1\leq|\sigma|<2 \\ 2\sigma/3, & 1/2\leq|\sigma|<1 \\ \sigma, & |\sigma|<1/2 \end{cases} \quad (10)$$

Note that t_{approx} is accurate for small value of ϕ because the derivate of the approximation

$$\left.\frac{dt_{approx}}{d\sigma}\right|_{\sigma=0}=\left.\frac{dt}{d\sigma}\right|_{\sigma=0}=1 \tag{11}$$

The Newton method is used to iteratively compute the value of c as follow

$$\begin{aligned}&c = 0.95;\\&\mathit{for}\ k=1..5\\&\quad c=0.5(3c-t_{approx}c^3);\\&\mathit{end}\end{aligned} \tag{12}$$

After 5 to 6 iterations, the approximate value of c will converge to its exact value.

3. Fast tracking in subspaces

If the system parameters change only slightly in successive time step $k \to k+1$, we can apply the tracking of eigenvalues and eigenvectors. In case that the eigenvector matrices of $\mathbf{R}_0(k)$ and $\mathbf{R}_0(k+1)$ are similar, then the both-sided multiplication with the previous eigenvector matrix $\mathbf{U}(k)$ results in a matrix $\mathbf{R}_1(k+1)$ that is almost diagonal. In other words, the result matrix $\mathbf{R}_1(k+1)$ has the energy almost in the diagonal elements, i.e. δ is already quite small.

$$\mathbf{R}_1(k+1)=\mathbf{U}^H(k)\mathbf{R}_0(k+1)\mathbf{U}(k) \tag{13}$$

In the special case that the eigenvector stay unchanged from $k \to k+1$, e.g. $\mathbf{R}_0(k)=\mathbf{R}_0(k+1)$, the pre-multiplication yields immediately the eigenvalue matrix and no tracking is needed.

After pre-multiplying with the eigenvector matrix $\mathbf{U}(k)$ of the previous time step k, Jacobi method proceed in usual way to reduce the off-diagonal energy δ until it falls below a predetermined threshold.

The orthogonality of $\mathbf{U}(k)$ and the principle of similarity transformation ensure that $\mathbf{R}_0(k)$ and $\mathbf{R}_0(k+1)$ have identical eigenvalues. So even the system parameter change completely from $k \to k+1$, the pre-multiplication with the previous eigenvector matrix introduces no problem. However, in this case, no reduction of off-diagonal energy δ is obtained.

In correspondence with Eq. 8, the eigenvector matrix at time $k+1$ is obtained as

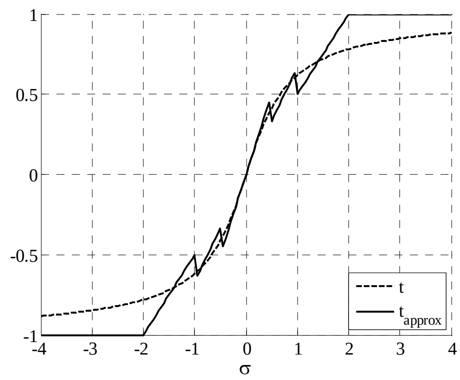


Fig.1 A piecewise linear approximation of t_{approx}.

$$\mathbf{U}(k+1)=\mathbf{U}(k)\left(\prod_i \mathbf{J}_i(k)\right) \tag{14}$$

4. Simulation results

To illustrate our proposed with its application in MIMO systems and eigen-beamforming array antenna, we conduct simulations in multipath condition. The simulation parameters are given in Table 1. The array antenna with 6 elements is employed. The distant between elements is haft of wavelength.

Figure 2 demonstrates the performance of the proposed tracking algorithm compared with no-tracking EVD using the exact Jacobi method. By applying the pre-multiplication with previous eigenvector matrix of previous time step, the off-diagonal energy is immediately reduced to about 70dB in case. After 2 sweeps the proposed tracking method reaches to the limit of computation accuracy. On the other hand, by using EVD with the exact Jacobi method, the eigenvalue/vector can be obtained after 4-5 sweeps.

To illustrate the tracking capability of the proposed algorithm in time-variant condition, AOAs of incoming signals are altered by $\Delta\theta$ such as

$$\begin{cases}\theta_1(k+1)=\theta_1(k)+\Delta\theta\\ \theta_2(k+1)=\theta_2(k)-\Delta\theta\end{cases} \tag{15}$$

Figure 3 shows the performance of the proposed tracking algorithm with inexact Jacobi method. In

Table 1. Simulation Parameters

Number of element	6
Distant	Half-wavelength
Number of samples to calculate $\mathbf{R}_0(k)$	256
Number of incoming signals	2
AOA of incoming signals	-60, 30

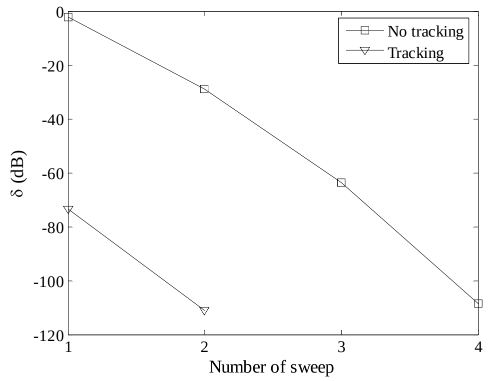


Fig.2 The performance of the proposed tracking algorithm with exact Jacobi method in static condition.

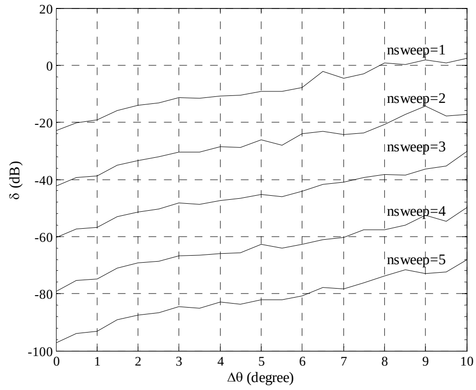
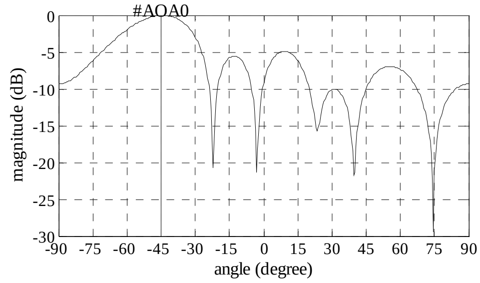


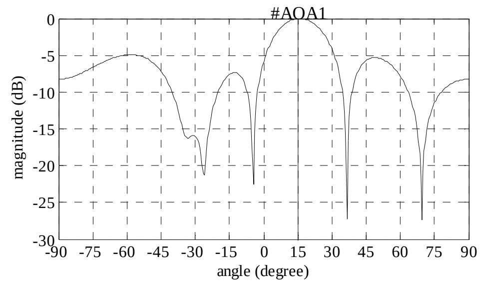
Fig.3 The performance of the proposed tracking algorithm with inexact Jacobi method.

time-variant condition, by applying two-sweep of inexact Jacobi method, the proposed algorithm is able to capture the time-variant parameters. In the region of high $\Delta\theta$, e.g. the system parameters change rapidly, the higher number of sweeps is required to tracking the time-variant parameters.

The method of eigen-beamforming is a beamforming approach in which the antenna weights are determined form eigenvalue decomposition of channel long-term correlation matrix instead of an estimation of AOA. The eigenvector corresponding to the strongest eigenvalue is used to allocate the transmit data to the array antenna. Obviously one eigenvector leads to only one distinct beam. The eigenvector corresponding to the second largest eigenvalue can also be used as the antenna weights. Figure 4 demonstrates beampatterns of the array antenna using two eigenvectors corresponding to two dominate eigenvalues as the antenna weights.



(a)



(b)

Fig.4 Beampatterns corresponding to eigenvector 1 and eigenvector 2.

5. Conclusions

The fast tracking in subspaces based on the inexact Jacobi method is applicable to eigenvalue/vector decomposition with block processing where the system parameters vary slowly from block to block. In such condition, the proposed fast tracking algorithm can reduce 3 or 4 sweeps compared with the conventional EVD method based on the exact Jacobi method. Moreover, the computation of square root, which is very expensive in hardware implementation, is provided. In this paper, eigen-beamforming, which is one of application of EVD, is also demonstrated.

The algorithm development and its application in direction estimation are subjects for future studying.

Table 2. Computation complexity

Number of element	Complex MAC for Pre-multiplication	Complex MAC for one single sweep
4	32	198
8	128	1372
16	512	9720
32	2048	71920

References

[1] Y. Karasawa, Y. Kamiyama, "Algorithm Diversity in a Software Antenna", IEIEC Trans. Comm. Vol.E83-B, No.6, June 2000.

[2] N.Kikuma, "Iterative DOA estimation using subspace tracking methods and Adaptive Beamforming," IEICE Trans. Comm. Vol.E88-B, No.5, May 2005.

[3] G. H. Golub, C.F Van Loan, "Matrix computations", The Johns Hopkins University Press, 1996.