논문 2011-48IE-1-2

신호부공간 추정 성능 향상을 위한 전후방 상관과 제곱근행렬 갱신을 이용한 COPAST(correlation-based projection approximation for subspace-tracking) 알고리즘 연구

(A Square-Root Forward Backward Correlation-based Projection Approximation for Subspace Tracking)

임 준 석*, 편 용 국**

(June-Seok Lim and Yong-Kug Pyeon)

요 약

본 논문에서는 상관계수를 바탕으로 신호부공간을 추정하는 COPAST(correlation-based projection approximation subspace tracking)의 성능을 향상시키기 위하여 상관계수를 구하는 부분을 순방향 신호 벡터로부터 상관계수를 구하고 동시에 역방향 신호 벡터에서도 상관계수를 구하여 신호 부공간을 추정하는 방법을 제안한다. 또 매번 갱신되는 상관행렬의 안정성을 도모하고자 제곱근 행렬 갱신을 하도록 하였다. 컴퓨터 모의 실험을 통해서 제안된 방법이 기존의 COPAST에 비해서 약 5dB의 신호 부공간 추정 정확도에 향상이 있었음을 확인하였다.

Abstract

In this paper, we propose a correlation-based subspace estimation technique, which is called square-root forward/backward correlation-based projection approximation subspace tracking(SRFB-COPAST). The SRFB-COPAST utilizes the forward and backward correlation matrix as well as square-root recursive matrix update in projection approximation approach to develop the subspace tracking algorithm. With the projection approximation, the square-root recursive FB-COPAST is presented. The proposed algorithm has the better performance than the recently developed COPAST method.

Keywords : Subspace Estimation, PAST, COPAST, Square-Root Update

Ⅰ. Introduction

In recent years, subspace-tracking algorithms have been intensively studied and widely applied to reduce the computational complexity of subspace estimation. Instead of updating the whole eigen-structure, subspace-tracking algorithm only works with the signal or noise subspace. This makes subspace-tracking algorithm more efficient than conventional methods using eigenvalue decomposition (ED) or singular value decomposition (SVD). One of the attractive subspace-tracking algorithms is the projection approximation subspace-tracking (PAST) algorithm[1]. The idea of the PAST is to make the expectation of the squared difference between the input vector and the projected vector minimum. With proper projection approximation, the PAST derives a recursive least squares (R LS) algorithm for tracking the signal subspace. However, the PAST still has

* 정회원, 세종대학교 전자공학과
(Dept. of Electronics Eng., Sejong University)
** 정회원, 강원도립대학 정보통신과
(Dept. of Information & Communication, GangWon Provincial College)
접수일자: 2010년10월1일, 수정완료일: 2011년2월21일

room for the improvement in the subspace estimation accuracy. To improve the PAST algorithm, many different algorithms have been proposed. For example, Jung-Lang Yu develop ed a correlation-based projection approximation subspace tracking (CO PAST), to improve the convergence property of the subspace tracking[2] and Gustafsson proposed an instrumental variable based PAST to cope with the colored noise case[3]. Lim proposed an algorithm to control the estimation window size automatically to handle time- varying signals[4].

In this paper, we propose a new algorithm to improve the subspace estimation accuracy and the stability in matrix inversion in the COPAST algorithm. The proposed algorithm utilizes the normal forward ordered input vector and the reversal ordered input vector simultaneously and also applies the square-root algorithm to the update of inversion matrix. The forward and backward input vectors can build up the better sample covariance and the square-root update can also improve the stability in the inverse matrix, which plays a key role in COPAST algorithm which is a kind of the overdetermined recursive algorithm[5].

Therefore, we expect the improved COPAST algorithm with the better subspace estimation accuracy. The outline of the paper is as follows. In Section Ⅱ, the problem formulation is introduced and the PAST algorithm. In Section Ⅲ, the COPAST is summarized and a new algorithm is derived based on the square-root matrix update. Secion Ⅳ discusses the property of covariance matrix and the estimated covariance matrix. Then the square-root forward and backward COPAST (SRFB- COPAST) algorithm is derived. Section Ⅴ shows simulation results to demonstrate that the FB-COPAST performs very well. Finally, Section Ⅵ contains the conclusions.

Ⅱ. PAST algorithm

Consider an M-element narrowband uniformly linear array (ULA) illuminated by J signals. The input vector to the array sensors can be written as

$$x(t)=\sum_{j=1}^{J}s_j(t)\mathbf{a}(\theta_j)+\mathbf{n}(t) \tag{1}$$

where, $s_j(t)$, j==1,..., J represent the wave- forms of the signals which are not fully correlated, $\mathbf{a}(\theta_j)$ is the phase vector of the jth signal with θ_j indicating the arriving direction, and $\mathbf{n}(t)$ is the spatially and temporally uncorrelated background noise. The correlation matrix of the input vector is

$$\mathbf{C}=E\left[x(t)x^H(t)\right]=\sum_{j=1}^{J}\sigma_j^2\mathbf{a}(\theta_j)\mathbf{a}^H(\theta_j)+\sigma_n^2\mathbf{I} \tag{2}$$

where E[•] denotes expectation, H denotes complex conjugate transpose, σ_j^2 and σ_n^2 represent the powers of the jth signal and noise, respectively. With the assumption of $J<M$, the correlation matrix C can be eigen decomposed as

$$\mathbf{C}=\sum_{i=1}^{M}\lambda_i\mathbf{e}_i\mathbf{e}_i^H \tag{3}$$

where $\lambda_1\geq\lambda_2\geq\Lambda\geq\lambda_{J+1}=\Lambda=\lambda_M=\sigma_n^2$ are eig- envalues in the descending order, and $\mathbf{e}_i$, i=1,...,M, are the corresponding orthonormal eigenvectors. The eigenvectors $\mathbf{e}_1,...,\mathbf{e}_J$ span the subspace same as that spanned by $\mathbf{a}_1,...,\mathbf{a}_J$,which is called the signal subspace. The remaining eigenvectors, $\mathbf{e}_{J+1},...,\mathbf{e}_M$ are orthogonal to the signal subspace and span the noise subspace[6~7].

$$J(\mathbf{W}(t))=\sum_{i=1}^{t}\beta^{t-i}\left\|\mathbf{x}(i)-\mathbf{W}(t)\mathbf{W}(t)\mathbf{x}(i)\right\| \tag{4}$$

PAST (Projection Approximation Subspace Tracking) algorithm is based on the minimum property of the unconstrained cost function.

where $\mathbf{x}(i)=\left[x_1(i)\quad\Lambda\quad x_N(i)\right]^T$ is the input vector and $\mathbf{W}(t)$ is a $N\times r$ matrix represent- ing the signal subspace. To derive recursive update of $\mathbf{W}(t)$ from $\mathbf{W}(t-1)$, Yang in [1] approximates $\mathbf{W}^H(t)\mathbf{x}(i)$ by the expression $\mathbf{y}(i)=\mathbf{W}^H(i-1)\mathbf{x}(i)$ which can be calculated for $1 \le i \le t$ at the time instant t. Then (1) results in a modified cost function,

$$J'(\mathbf{W}(t))=\sum_{i=1}^{t}\beta^{t-i}\left\|\mathbf{x}(i)-\mathbf{W}(t)\mathbf{y}(i)\right\| \quad (5)$$

This becomes the exponentially weighted least squares cost function which is well studied in adaptive filtering. $J'(\mathbf{W}(t))$ is minimized if

$$\begin{aligned}
&\mathbf{W}(t)=\mathbf{C}_{\mathbf{xy}}(t)\mathbf{C}_{\mathbf{yy}}^{-1}(t),\\
&\mathbf{C}_{\mathbf{xy}}(t)=\sum_{i=1}^{t}\beta^{t-i}\mathbf{x}(i)\mathbf{y}^H(i)=\beta\mathbf{C}_{\mathbf{xy}}(t-1)+\mathbf{x}(t)\mathbf{y}^H(t),\\
&\mathbf{C}_{\mathbf{yy}}(t)=\sum_{i=1}^{t}\beta^{t-i}\mathbf{y}(i)\mathbf{y}^H(i)=\beta\mathbf{C}_{\mathbf{yy}}(t-1)+\mathbf{y}(t)\mathbf{y}^H(t).
\end{aligned} \quad (6)$$

Ⅲ. Correlation-based PAST and its square-root recursive Algorithm

Yu developed the COPAST algorithm in [2]. The COPAST defines the cost function in terms of the correlation matrix as follows

$$\begin{aligned}
J(\mathbf{W})&=\left\|\mathbf{C}_{\mathbf{xx}}-\mathbf{W}\mathbf{W}^H\mathbf{C}_{\mathbf{xx}}\right\|_F^2\\
&=\mathrm{Tr}\left\{\begin{matrix}\mathbf{C}_{\mathbf{xx}}\mathbf{C}_{\mathbf{xx}}^H-2\mathbf{W}^H\mathbf{C}_{\mathbf{xx}}\mathbf{C}_{\mathbf{xx}}^H\mathbf{W}\\+\mathbf{W}^H\mathbf{C}_{\mathbf{xx}}\mathbf{C}_{\mathbf{xx}}^H\mathbf{W}\mathbf{W}^H\mathbf{W}\end{matrix}\right\}
\end{aligned} \quad (7)$$

where $Tr\{\bullet\}$ denotes the trace of a matrix and F indicates the Frobenius norm. Since the correlation matrix is the second-order statistic of the input vector, the input SNR seems to be square of that in the PAST. Consequently, the COPAST using the correlation matrix improves the performance of the PAST. For convenience of finding the solution of (7), the ensemble correlation matrix is replaced with the exponentially weighted sum

$$\mathbf{C}_{\mathbf{xx}}(t)=\sum_{i=1}^{t}\beta^{t-i}\mathbf{x}(i)\mathbf{x}^H(i) \quad (8)$$

Thus, the cost function becomes

$$J(\mathbf{W}(t))=\left\|\mathbf{C}_{\mathbf{xx}}(t)-\mathbf{W}(t)\mathbf{W}^H(t)\mathbf{C}_{\mathbf{xx}}(t)\right\|_F^2 \quad (9)$$

Using the projection approximation concept and (8), $\mathbf{W}^H(t)\mathbf{C}_{\mathbf{xx}}(t)$ in(9),which is termed $\mathbf{C}_{\mathbf{yx}}(t)$, is approximated to

$$\mathbf{C}_{\mathbf{yx}}(t)=\mathbf{W}^H(t)\mathbf{C}_{\mathbf{xx}}(t)\cong\sum_{i=1}^{t}\beta^{t-i}\mathbf{W}^H(i-1)\mathbf{x}(i)\mathbf{x}^H(i) \quad (10)$$

Using (10), the criterion of (9) degenerates to a quadratic optimization problem, and is minimized by

$$\mathbf{W}(t)=\mathbf{C}_{\mathbf{xx}}(t)\mathbf{C}_{\mathbf{yx}}^H(t)\left(\mathbf{C}_{\mathbf{yx}}(t)\mathbf{C}_{\mathbf{yx}}^H(t)\right)^{-1}=\mathbf{C}_{\mathbf{xx}}(t)\mathbf{C}_{\mathbf{yx}}^{+}(t) \quad (11a)$$

where $\mathbf{C}_{\mathbf{xx}}(t)=\beta\mathbf{C}_{\mathbf{xx}}(t-1)+\mathbf{x}(t)\mathbf{x}^H(t)$ and $^+$ denotes the Moore-Penrose pseudo-inverse[2]. The COPAST algorithm is summarized in Table 1.

표 1. COPAST 알고리즘 요약

Table 1. Summary of COPAST Algorithm.

$$\mathbf{W}(0)=\begin{bmatrix}1&0&\cdots&0\\0&1&&\vdots\\\vdots&\vdots&\ddots&\vdots\\0&0&\cdots&\vdots\end{bmatrix},\quad [L\times r]$$

$$\mathbf{P}(0)=\begin{bmatrix}1&0&\cdots&0\\0&1&&0\\\vdots&\vdots&\ddots&\vdots\\0&0&\cdots&1\end{bmatrix},\quad [r\times r]$$

Do t=1,⋯

$$\mathbf{y}(t)=\mathbf{W}^H(t-1)\mathbf{x}(t)$$

$$\mathbf{V}(t)=\left[\mathbf{C}_{\mathbf{xx}}(t-1)\mathbf{x}(t)\quad \mathbf{x}(t)\right]$$

$$\boldsymbol{\omega}(t)=\mathbf{C}_{\mathbf{yx}}(t-1)\mathbf{x}(t)$$

$$\beta^2\boldsymbol{\Lambda}=\begin{bmatrix}-\mathbf{x}^H(t)\mathbf{x}(t)&\beta\\\beta&0\end{bmatrix}$$

$$\boldsymbol{\Phi}(t)=\left[\boldsymbol{\omega}(t)\quad \mathbf{y}(t)\right]$$

$$\mathbf{K}(t)=\left[\beta^2\boldsymbol{\Lambda}+\boldsymbol{\Phi}^H(t)\mathbf{P}(t-1)\boldsymbol{\Phi}(t)\right]^{-1}\boldsymbol{\Phi}^H(t)\mathbf{P}(t-1)$$

$$\mathbf{P}(t)=\left[\mathbf{P}(t-1)-\mathbf{P}(t-1)\boldsymbol{\Phi}(t)\mathbf{K}(t)\right]/\beta^2$$

$$\mathbf{C}_{\mathbf{yx}}(t)=\beta\mathbf{C}_{\mathbf{yx}}(t-1)+\mathbf{y}(t)\mathbf{x}^H(t)$$

$$\mathbf{C}_{\mathbf{xx}}(t)=\beta\mathbf{C}_{\mathbf{xx}}(t-1)+\mathbf{x}(t)\mathbf{x}^H(t)$$

$$\mathbf{W}(t)=\mathbf{W}(t-1)+\left[\mathbf{V}(t)-\mathbf{W}(t-1)\boldsymbol{\Phi}(t)\right]\mathbf{K}(t)$$

END

A major drawback of the overdetermined recursive algorithm such as COPAST, is likely to be the poor numerical behavior. The poor numerical behavior comes from the instability in the matrix inversion part of (11a). One of the methods to improve the stability is the square-root update of the matrix square-root of the inverse matrix P(t) in Table1. In this paper, in order to improve the stability in the inverse matrix in the overdetermined recursive algorithm, we follows the Porat's approach in [5].

First of all, the hyperbolic transformation needs to update the matrix square-root in the COPAST and it is as follows.

$$\beta^2\mathbf{\Lambda} = \tilde{\mathbf{\Lambda}} = \begin{bmatrix} -\mathbf{x}^H(t)\mathbf{x}(t) & \beta \\ \beta & 0 \end{bmatrix} = \begin{bmatrix} \alpha & 0 \\ -\beta/\alpha & \beta/\alpha \end{bmatrix}\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} \alpha & -\beta/\alpha \\ 0 & \beta/\alpha \end{bmatrix} = \mathbf{AJA}^T \tag{11b}$$

where $\alpha = \left(\mathbf{x}^H(t)\mathbf{x}(t)\right)^{1/2}$.

Assume that we have found a matrix Q of dimension $(N+2)\times(N+2)$ such that

$$\mathbf{Q}\begin{bmatrix} \mathbf{J} & 0 \\ 0 & \mathbf{I}_N \end{bmatrix}\mathbf{Q}^T = \begin{bmatrix} \mathbf{J} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix} \tag{12}$$

Such a matrix is called J-orthogonal. Furthermore, assume that Q was chosen so that

$$\begin{bmatrix} \mathbf{A} & \mathbf{\Phi}^H(t)\mathbf{P}^{1/2}(t) \\ 0 & \mathbf{P}^{1/2}(t) \end{bmatrix}\mathbf{Q} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{0} \\ \mathbf{M} & \mathbf{L}_2 \end{bmatrix} \tag{13}$$

where $\mathbf{L}_1$ and $\mathbf{L}_2$ are lower trianglar of 2×2 and $N\times N$, respectively. By(12), we will have

$$\begin{bmatrix} \mathbf{A} & \mathbf{\Phi}^H(t)\mathbf{P}^{1/2}(t) \\ \mathbf{0} & \mathbf{P}^{1/2}(t) \end{bmatrix}\begin{bmatrix} \mathbf{J} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix}\begin{bmatrix} \mathbf{A}^T & \mathbf{0} \\ \mathbf{\Phi}^H(t)\mathbf{P}^{1/2}(t) & (\mathbf{P}^{1/2}(t))^T \end{bmatrix} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{0} \\ \mathbf{M} & \mathbf{L}_2 \end{bmatrix}\begin{bmatrix} \mathbf{J} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix}\begin{bmatrix} \mathbf{L}_1^T & \mathbf{M}^T \\ \mathbf{0} & \mathbf{L}_2^T \end{bmatrix} \tag{14}$$

$$\mathbf{AJA}^H + \mathbf{\Phi}^H(t)\mathbf{P}(t-1)\mathbf{\Phi}(t) = \mathbf{\Lambda}(t) + \mathbf{\Phi}^H(t)\mathbf{P}(t-1)\mathbf{\Phi}(t) = \mathbf{L}_1\mathbf{JL}_1^H \tag{15a}$$

$$\mathbf{P}(t-1)\mathbf{\Phi}(t) = \mathbf{MJL}_1^H \tag{15b}$$

$$\mathbf{P}(t-1) = \mathbf{MJM}^H + \mathbf{L}_2\mathbf{L}_2^H \tag{15c}$$

From (15b) we get

$$\mathbf{M} = \mathbf{P}(t-1)\mathbf{\Phi}(t)(\mathbf{L}_1^H)^{-1}\mathbf{J} \tag{16}$$

Hence, substituting in (15c),

$$\begin{aligned} \mathbf{L}_2\mathbf{L}_2^H &= \mathbf{P}(t-1) - \mathbf{MJM}^H \\ &= \mathbf{P}(t-1) - \mathbf{P}(t-1)\mathbf{\Phi}(t)(\mathbf{L}_1^H)^{-1}\mathbf{JL}_1^{-1}\mathbf{\Phi}^H(t)\mathbf{P}(t-1) \\ &= \mathbf{P}(t-1) - \mathbf{P}(t-1)\mathbf{\Phi}(t)\left[\mathbf{\Lambda}(t) + \mathbf{\Phi}^H(t)\mathbf{P}(t-1)\mathbf{\Phi}(t)\right]^{-1} \\ &\quad \mathbf{\Phi}^H(t)\mathbf{P}(t-1) \\ &= \beta^2\mathbf{P}(t) \end{aligned} \tag{17}$$

$$\therefore \quad \mathbf{P}^{1/2}(t) = \beta^{-1}\mathbf{L}_2 \tag{18}$$

Note also that

표 2. 제곱근행렬 갱신 COPAST 알고리즘 요약
Table 2. Summary of Square Root Based COPAST Algorithm.

$$\mathbf{W}(0) = \begin{bmatrix} 1 & 0 & \Lambda & 0 \\ 0 & 1 & & M \\ M & M & O & M \\ 0 & 0 & \Lambda & M \end{bmatrix}, \quad [L\times r]$$

$$\mathbf{P}(0) = \begin{bmatrix} 1 & 0 & \Lambda & 0 \\ 0 & 1 & & 0 \\ M & M & O & M \\ 0 & 0 & \Lambda & 1 \end{bmatrix}, \quad [r\times r]$$

Do t=1,⋯

$$\mathbf{y}(t) = \mathbf{W}^H(t-1)\mathbf{x}(t)$$

$$\mathbf{V}(t) = \left[\mathbf{C}_{\mathbf{xx}}(t-1)\mathbf{x}(t) \quad \mathbf{x}(t)\right]$$

$$\boldsymbol{\omega}(t) = \mathbf{C}_{\mathbf{yx}}(t-1)\mathbf{x}(t)$$

$$\mathbf{\Phi}(t) = \left[\boldsymbol{\omega}(t) \quad \mathbf{y}(t)\right]$$

$$\alpha = \left(\mathbf{x}^H(t)\mathbf{x}(t)\right)^{1/2}$$

$$\mathbf{A} = \begin{bmatrix} \alpha & 0 \\ -\beta/\alpha & \beta/\alpha \end{bmatrix}$$

Find Q

$$\begin{bmatrix} \mathbf{A} & \mathbf{\Phi}^H(t)\mathbf{P}^{1/2}(t) \\ 0 & \mathbf{P}^{1/2}(t) \end{bmatrix}\mathbf{Q} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{0} \\ \mathbf{M} & \mathbf{L}_2 \end{bmatrix}$$

$$\mathbf{P}^{1/2}(t) = \beta^{-1}\mathbf{L}_2$$

$$\mathbf{K} = \left(\mathbf{ML}_1^{-1}\right)^H$$

$$\mathbf{C}_{\mathbf{yx}}(t) = \beta\mathbf{C}_{\mathbf{yx}}(t-1) + \mathbf{y}(t)\mathbf{x}^H(t)$$

$$\mathbf{C}_{\mathbf{xx}}(t) = \beta\mathbf{C}_{\mathbf{xx}}(t-1) + \mathbf{x}(t)\mathbf{x}^H(t)$$

$$\mathbf{W}(t) = \mathbf{W}(t-1) + \left[\mathbf{V}(t) - \mathbf{W}(t-1)\mathbf{\Phi}(t)\right]\mathbf{K}(t)$$

END

$$\begin{aligned}(\mathbf{ML}_1^{-1})^H &= \{\mathbf{P}(t-1)\mathbf{\Phi}(t)(\mathbf{L}_1^H)^{-1}\mathbf{JL}_1^{-1}\}^H \\ &= \left\{\mathbf{P}(t-1)\mathbf{\Phi}(t)\left[\mathbf{\Lambda}(t)+\mathbf{\Phi}^H(t)\mathbf{P}(t-1)\mathbf{\Phi}(t)\right]^{-1}\right\}^H \\ &= \mathbf{K}\end{aligned} \tag{19}$$

A new square-root recursive COPAST (SR-COPAST) is summarized in Table 2.

Ⅳ. Square-Root Based Forward-Backward COPAST Algorithm

PAST style algorithms such as PAST and COPAST, use the sample covariance matrix $\mathbf{C}_{\mathbf{xx}}(t)$ which is,

$$\mathbf{C}_{\mathbf{xx}}(t) = \beta\mathbf{C}_{\mathbf{xx}}(t-1) + \begin{bmatrix} x_1(t) \\ \mathrm{M} \\ x_L(t)\end{bmatrix}\begin{bmatrix} x_1^*(t) & \Lambda & x_L^*(t)\end{bmatrix} \tag{20}$$

Generally, the theoretical covariance matrix $\mathbf{C}_{\mathbf{xx}}$ is Toeplitz and persymmetric[8].

However, the estimated covariance matrix $\mathbf{C}_{\mathbf{xx}}(t)$ in (20) does not guarantee all the properties, because it does not satisfy the persymmetric property such as $\mathbf{J}(\mathbf{C}_{\mathbf{xx}}(t))^T\mathbf{J} \neq \mathbf{C}(t)$, where $\mathbf{J} = \begin{bmatrix} 0 & & 1 \\ & \mathrm{N} & \\ 1 & & 0\end{bmatrix}$ is the so-called reversal matrix.

We can make the sample covariance matrix Toeplitz and persymmetric by some modification of (20). The modified sample covariance matrix $\widetilde{\mathbf{C}}_{\mathbf{xx}}$ is,

$$\widetilde{\mathbf{R}}_{\mathbf{xx}}(t) = \frac{1}{2}\left(\mathbf{C}_{\mathbf{xx}}(t) + \mathbf{JC}_{\mathbf{xx}}^T(t)\mathbf{J}\right) \tag{21}$$

The second term of (21) has the following recursive form,

$$\mathbf{JC}_{\mathbf{xx}}^T(t)\mathbf{J} = \beta\mathbf{JC}_{\mathbf{xx}}^T(t-1)\mathbf{J} + \begin{bmatrix} x_L^*(t) \\ \mathrm{M} \\ x_1^*(t)\end{bmatrix}\begin{bmatrix} x_L(t) & \Lambda & x_1(t)\end{bmatrix} \tag{22}$$

The modified sample covariance matrix $\widetilde{\mathbf{C}}_{\mathbf{xx}}(t)$ utilizes the sample covariance matrix of the reverse ordered vector as well as that of the normal forward ordered vector. Therefore, we can call the $\widetilde{\mathbf{C}}_{\mathbf{xx}}(t)$ as the forward- backward covariance matrix. The forward- backward covariance matrix, $\widetilde{\mathbf{C}}_{\mathbf{xx}}(t)$, is invariant to the transform, $\mathbf{J}(\bullet)^T\mathbf{J}$,

$$\mathbf{J}(\widetilde{\mathbf{C}}_{\mathbf{xx}}(t))^T\mathbf{J} = \widetilde{\mathbf{C}}_{\mathbf{xx}}(t) \tag{23}$$

(23) shows that $\widetilde{\mathbf{C}}_{\mathbf{xx}}(t)$ is persymmetric[8].

Therefore, We may expect $\widetilde{\mathbf{C}}_{\mathbf{xx}}(t)$ to be a better estimate of $\mathbf{C}_{\mathbf{xx}}$ than $\mathbf{C}_{\mathbf{xx}}(t)$. In turn, this means that the estimated principal components derived from $\widetilde{\mathbf{C}}_{\mathbf{xx}}(t)$ are likely to be more accurate than those obtained from $\mathbf{C}_{\mathbf{xx}}(t)$. To apply the forward-backward covariance matrix to the PAST style algorithm, we should modify (20) considering a forward- backward covariance matrix. The recursive forward-backward covariance matrix is as follows.

$$\widetilde{\mathbf{C}}_{\mathbf{xx}}(t) = \beta\widetilde{\mathbf{C}}_{\mathbf{xx}}(t-1) + \begin{bmatrix} x_1(t) & x_L^*(t) \\ \mathrm{M} & \mathrm{M} \\ x_L(t) & x_1^*(t)\end{bmatrix}\begin{bmatrix} x_1^*(t) & \Lambda & x_L^*(t) \\ x_L(t) & \Lambda & x_1(t)\end{bmatrix} \tag{24}$$

Comparing (24) with (21), (24) needs a scaling factor, 1/2. However, the scaling factor does not affect the subspace so that we dismiss the factor. Applying (24) to COPAST algorithm, we can derive a new COPAST algorithm with the forward-backward covariance matrix.

Considering the forward and backward data vectors in this square root algorithm, it is enough to apply the forward and backward data vectors only to the recursive estimations of $\mathbf{C}_{\mathbf{yx}}(t)$ and $\mathbf{C}_{\mathbf{xx}}(t)$, respectively. It is beca- use the square root algorithm keeps the good matrix property of $(\mathbf{P}^{-1})^H = \mathbf{P}^{-1}$ in the matrix $\mathbf{P}^{-1}$. When applying the forward and backward data vectors, we should replace $\mathbf{C}_{\mathbf{yx}}(t)$ and $\mathbf{C}_{\mathbf{xx}}(t)$ as follows, respectively.

$$\mathbf{C}_{\widetilde{\mathbf{Y}}\widetilde{\mathbf{X}}}(t) = \beta\mathbf{C}_{\widetilde{\mathbf{Y}}\widetilde{\mathbf{X}}}(t-1) + \widetilde{\mathbf{Y}}(t)\widetilde{\mathbf{X}}^H(t) \tag{25a}$$

$$\mathbf{C}_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}(t) = \beta\mathbf{C}_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}(t-1) + \tilde{\mathbf{X}}(t)\tilde{\mathbf{X}}^H(t) \tag{25b}$$

We should also replace $\boldsymbol{\omega}(t)$, $\boldsymbol{\Phi}(t)$ and $\mathbf{V}(t)$ with $\boldsymbol{\Omega}(t)$, $\tilde{\boldsymbol{\Phi}}(t)$ and $\tilde{\mathbf{V}}(t)$, respectively, asfollows. $\boldsymbol{\Omega}(t) = \mathbf{C}_{\tilde{\mathbf{Y}}\tilde{\mathbf{X}}}(t-1)\tilde{\mathbf{X}}(t)$, $\tilde{\boldsymbol{\Phi}}(t) = [\boldsymbol{\Omega}(t) \quad \tilde{\mathbf{Y}}(t)]$ an $\tilde{\mathbf{V}}(t) = [\mathbf{C}_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}(t-1)\tilde{\mathbf{X}}(t) \quad \tilde{\mathbf{X}}(t)]$.

The proposed square-root forward backward COPAST (SRFB-COPAST) is summarized in Table 3.

표 3. 전후방 상관과 제곱근행렬 갱신 COPAST 알고리즘 요약

Table 3. Summary of Square Root Based FB-COPAST Algorithm.

$$\mathbf{W}(0) = \begin{bmatrix} 1 & 0 & \Lambda & 0 \\ 0 & 1 & & M \\ M & M & O & M \\ 0 & 0 & \Lambda & M \end{bmatrix}, \quad [L \times r]$$

$$\mathbf{C}_{\tilde{\mathbf{P}}}(0) = \begin{bmatrix} 1 & 0 & \Lambda & 0 \\ 0 & 1 & & 0 \\ M & M & O & M \\ 0 & 0 & \Lambda & 1 \end{bmatrix}, \quad [r \times r]$$

Do t=1,⋯

$$\tilde{\mathbf{X}}(t) = \begin{bmatrix} x_1(t)\, x_L^*(t) \\ M \quad M \\ x_L(t)\, x_1^*(t) \end{bmatrix}$$

$$\tilde{\mathbf{Y}}(t) = \mathbf{W}^H(t-1)\tilde{\mathbf{X}}(t)$$

$$\tilde{\mathbf{V}}(t) = [\mathbf{C}_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}(t-1)\tilde{\mathbf{X}}(t) \quad \tilde{\mathbf{X}}(t)]$$

$$\boldsymbol{\Omega}(t) = \mathbf{C}_{\tilde{\mathbf{Y}}\tilde{\mathbf{X}}}(t)\tilde{\mathbf{X}}(t)$$

$$\tilde{\boldsymbol{\Phi}}(t) = [\boldsymbol{\Omega}(t) \quad \tilde{\mathbf{Y}}(t)]$$

$$\alpha = \left(\mathbf{x}^H(t)\mathbf{x}(t)\right)^{1/2}$$

$$\mathbf{A} = \begin{bmatrix} \alpha & 0 \\ -\beta/\alpha & \beta/\alpha \end{bmatrix}$$

Find Q

$$\begin{bmatrix} \mathbf{A} & \boldsymbol{\Phi}^H(t)\mathbf{P}^{1/2}(t) \\ 0 & \mathbf{P}^{1/2}(t) \end{bmatrix}\mathbf{Q} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{0} \\ \mathbf{M} & \mathbf{L}_2 \end{bmatrix}$$

$$\mathbf{P}^{1/2}(t) = \beta^{-1}\mathbf{L}_2$$

$$\mathbf{K} = \left(\mathbf{M}\mathbf{L}_1^{-1}\right)^H$$

$$\mathbf{C}_{\tilde{\mathbf{Y}}\tilde{\mathbf{X}}}(t) = \beta\mathbf{C}_{\tilde{\mathbf{Y}}\tilde{\mathbf{X}}}(t-1) + \tilde{\mathbf{Y}}(t)\tilde{\mathbf{X}}^H(t)$$

$$\mathbf{C}_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}(t) = \beta\mathbf{C}_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}(t-1) + \tilde{\mathbf{X}}(t)\tilde{\mathbf{X}}^H(t)$$

$$\mathbf{W}(t) = \mathbf{W}(t-1) + [\tilde{\mathbf{V}}(t) - \mathbf{W}(t-1)\tilde{\boldsymbol{\Phi}}(t)]\mathbf{K}(t)$$

END

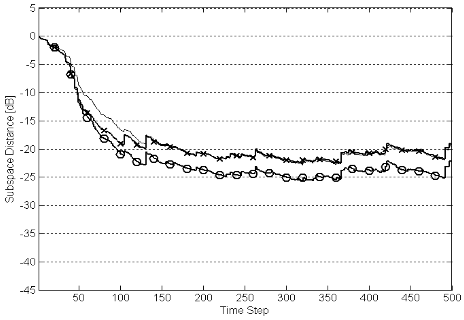
Ⅴ. Simulation Results

In this section, we demonstrate the applicability of the proposed algorithm to the subspace estimation. We assume the signal subspace comes from a narrow band far-field source by using a linear uniform array with 8 sensors. For the experimental purpose, we set the scenario that the angle of arrival of the signal comes from -30^o.

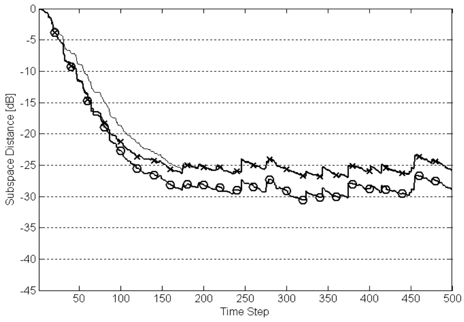
In Fig.1, we compare the estimation accuracy of the proposed algorithm (SRFB-COPAST) with the conventional PAST algorithm and COPAST algorithm in the fixed forgetting factors of 0.98 under the four different SNR cases of 5dB, 10dB, 15dB and 20dB, respectively. To compare the quality of the estimated subspace, we show the distance between the estimated subspace and the true subspace, which is defined in [9].

$$\sin\theta(\mathbf{S},\tilde{\mathbf{S}}) \equiv \left\|(\mathbf{I}-\mathbf{P})\tilde{\mathbf{P}}\right\| \tag{26}$$

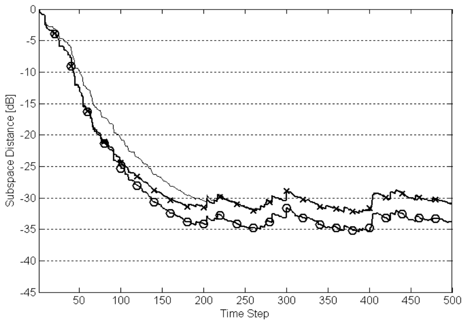
where $\mathbf{S}$ is the true subspace, $\tilde{\mathbf{S}}$ is the estimated sub-space, $\mathbf{P}$ is the projector onto $\mathbf{S}$ and $\tilde{\mathbf{P}}$ is the projector onto $\tilde{\mathbf{S}}$. The results in Fig.1 show that the proposed algorithm estimated the subspace more accurately than the conventional PAST algorithm and the COPAST algorithm in all SNR cases. In Fig.2, we also compare the estimation accuracy of the proposed algorithm (SRFB-COPAST) with the conventional PAST algorithm and COPAST algorithm in the fixed forgetting factors of 0.9 under the four different SNR cases of 5dB, 10dB, 15dB and 20dB, respectively. The results in Fig.2 also show that the proposed algorithm estimated the subspace more accurately than the conventional PAST algorithm and the COPAST algorithm in all the same SNR cases. In the same SNR, the result from the forgetting factor of 0.98 outperforms the result from the forgetting factor of 0.9. It is because the larger forgetting factor guarantees the longer effective data window for estimation than the smaller forgetting factor does. In table 4, we summarize the subspace



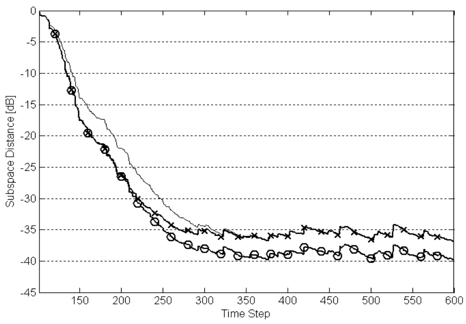
(a) SNR 5dB



(b) SNR 10dB



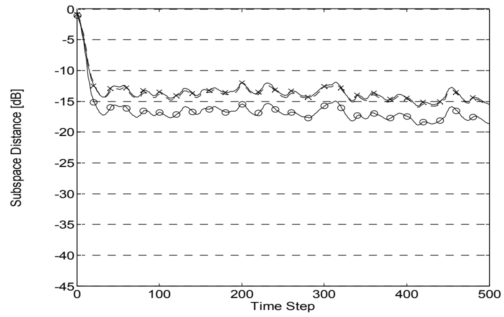
(c) SNR 15dB



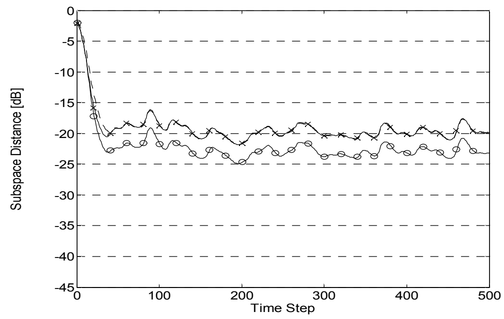
(d) SNR 20dB

그림 1. 망각인자 0.98 정재환경에서의 부공간 추정 정확도 비교(–o–: 제안된알고리즘, –x–: COPAST, – – –: 기존 PAST).

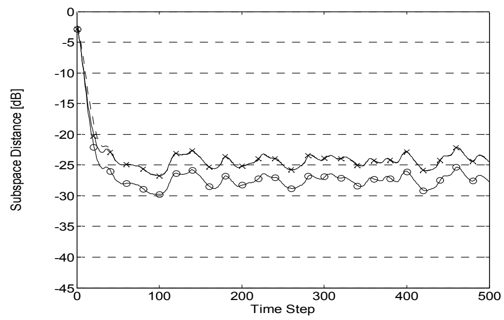
Fig. 1. The subspace estimation accuracy comparisons in stationary environment with forgetting factor of 0.98 (–o–: proposed algorithm, –x–: COPAST, – – –: conventional PAST).



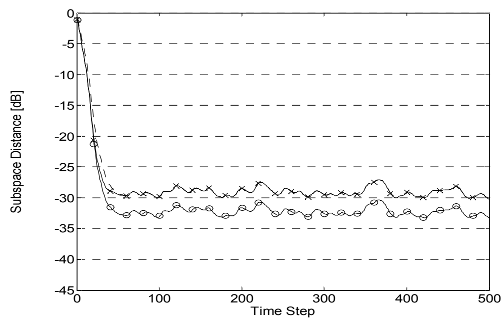
(a) SNR 5dB



(b) SNR 10dB



(c) SNR 15dB



(d) SNR 20dB

그림 2. 망각인자 0.9 정재환경에서의 부공간 추정 정확도 비교 (–o–: 제안된알고리즘, –x–: COPAST, – – –: 기존 PAST).

Fig. 2. The subspace estimation accuracy comparisons in stationary environment with forgetting factor of 0.9 (–o–: proposed algorithm, –x–: COPAST, – – –: conventional PAST).

표 4. 부공간거리에서의 신호부공간 추정성능비교
Table 4. Subspace Estimation Performance Comparison in Subspace Distance.

SNR	Subspace Distance [dB] Forgetting factor (0.98)		Subspace Distance [dB] Forgetting factor (0.9)	
5dB	PAST	-21.2	PAST	-14.2
	COPAST	-21.3	COPAST	-14.5
	Proposed Method	-24.0	Proposed Method	-17.5
10dB	PAST	-26.0	PAST	-19.2
	COPAST	-26.2	COPAST	-19.3
	Proposed Method	-29.1	Proposed Method	-22.2
15dB	PAST	-31.1	PAST	-24.2
	COPAST	-31.2	COPAST	-24.2
	Proposed Method	-34.1	Proposed Method	-27.5
20dB	PAST	-35.0	PAST	-29.1
	COPAST	-35.2	COPAST	-29.2
	Proposed Method	-39.0	Proposed Method	-33.2

estimation accuracy by comparing the proposed method, PAST and COPAST in different SNR cases.

Ⅵ. Conclusion

In this paper, we have proposed the square-root forward-backward COPAST (SRFB-COPAST) algorithm to estimate the signal subspace. The SRFB-COPAST applies the forward and backward ordered data vectors as well as square root update algorithm to the COPAST. It improves the property of the estimated covariance matrix to get closer to the ideal covariance matrix. From the simulation results, we can confirm the proposed SRFB-COPAST outperforms the COPAST as well as the conventional PAST in the estimation accuracy.

References

[1] Bin Yang, "Projection Approximation Subsp- ace Tracking", IEEE Trans. Signal Proc., 1995, 43, (1), pp. 95-107
[2] Jung-Lang Yu, "A Novel Subspace Tracking Using Correlation-Based Projection Approximation", Signal Procerssing, 2000, 80, pp. 2517-2525.
[3] Tony Gustafsson, "Instrumental Variable Subspace Tracking Using Projection Approximation", IEEE Trans. Signal Proc., 1998, 46, (3), pp. 669-681.
[4] Jun-Seok Lim, Seongwook Song and Koeng -Mo Sung, "Variable forgetting factor PASTd algorithm for time-varying subspace estimation", Electronics Letters, 2000, 36, (16), pp.1434-1435.
[5] Boaz Porat and Benjamin Friedlander, "The Square-Root Overdetermined Recursive Instrumental Variable Algorithm", IEEE Trans. Automatic Control, 1989, 34, (6), pp.656~658.
[6] R.O. Schmidt, "Multiple Emitter Location and Signal Parameter Estimation', IEEE Trans. Antennas Prop., 1986, 34, (3), pp. 276-280.
[7] R. Kumaresan, D.W. Tufts, "Estimation the Angles of Arrival of Mutiple Plane Waves", IEEE Trans. Aerosp. Electron. System, 1983, 19, (1), pp 134-139.
[8] P. Stoica and R. L. Moses, Spectral Analysis of Signals, Prentice Hall, Upper Saddle River, N.J., 2005.
[9] Srinath Hosur, Ahmed H. Tewfik and Daniel Boley, "ULV and Generalized ULV Subspace Tracking Adaptive Algorithms", IEEE Trans. Signal Proc., 1996, 46, (5), pp. 1282-1297.

저 자 소 개

임 준 석(정회원)
1986년 서울대학교 전자공학과 학사 졸업.
1988년 서울대학교 전자공학과 학사 졸업.
1996년 서울대학교 전자공학과 학사 졸업.
1996년 7월～1997년10월 LG종합기술원.
현재 세종대학교 전자공학과 교수
<주관심분야 : 통신, 배열신호처리>

편 용 국(정회원)
1993년 강원대학교 전자공학과 학사 졸업.
1996년 관동대학교 전자공학과 석사 졸업.
2004년 세종대학교 전자공학과 박사 졸업.
현재 SE로보틱스 대표
현재 강원도립대학 정보통신과 조교수
<주관심분야 : 통신, 배열신호처리, 지능형로봇 >