A Matrix Pseudo-Inversion Lemma for Positive Semidefinite Hermitian Matrices and Its Application to Adaptive Blind Deconvolution of MIMO Systems

Kiyotaka Kohno, *Member, IEEE*, Yujiro Inouye, and Mitsuru Kawamoto

Abstract—In the simplest case, the matrix inversion lemma gives an explicit formula of the inverse of a positive-definite matrix $\boldsymbol{A}$ added to a rank-one matrix $\boldsymbol{bb}^H$ as follows: $(\boldsymbol{A} + \boldsymbol{bb}^H)^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{b}(1 + \boldsymbol{b}^H\boldsymbol{A}^{-1}\boldsymbol{b})^{-1}\boldsymbol{b}^H\boldsymbol{A}^{-1}$. It is well known in the literature that this formula is very useful to develop a recursive least-squares algorithm for the recursive identification of linear systems or the design of adaptive filters. We extend this result to the case when the matrix $\boldsymbol{A}$ is singular and present a matrix pseudo-inversion lemma along with some illustrative examples. Such a singular case may occur in a situation where a given problem is overdetermined in the sense that it has more equations than unknowns. This lemma is important in its own right, but in order to show the usefulness of the lemma, we apply it to develop an adaptive super-exponential algorithm for the blind deconvolution of multi-input multi-output systems.

Index Terms—Adaptive super-exponential algorithm (SEA), matrix pseudo-inversion lemma, recursive algorithms.

I. INTRODUCTION

THE familiar matrix inversion lemma states in the simplest case that the inverse of a positive-definite matrix $\boldsymbol{A}$ added a rank-one matrix $\boldsymbol{bb}^H$ can be represented as

$$(\boldsymbol{A} + \boldsymbol{bb}^H)^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{b}(1 + \boldsymbol{b}^H\boldsymbol{A}^{-1}\boldsymbol{b})^{-1}\boldsymbol{b}^H\boldsymbol{A}^{-1} \quad (1)$$

where the superscript H denotes the complex conjugate transpose (or Hermitian) operation. It is well known in the literature that this formula is very useful to develop a recursive least-squares algorithm for the recursive identification [1], [2] or the design of adaptive filters [3].

In this paper, we extend this matrix inversion lemma to the case when the matrix $\boldsymbol{A}$ is singular and show a matrix pseudo-inversion lemma together with some illustrative examples. Such a singular case may occur in a situation where a problem dealt with is overdetermined in the sense that it has more equations than unknowns. In particular, we encountered this singular situation when we developed an adaptive version of the super-exponential method (SEM) for the blind deconvolution of a multi-input multi-output (MIMO) system, where the number of its outputs is greater than the number of its inputs [9]–[11].

We show briefly how the matrix pseudo-inversion lemma can be applied to obtain an adaptive super-exponential algorithm (SEA) for the blind deconvolution of a MIMO system. We include simulation results for the performance of the algorithm in order to show the usefulness of the lemma, where we compare the performance of the algorithm using the lemma with that of the algorithm using the built-in function in MATLAB Version 7.1.0 for calculating pseudo-inverse instead of using the lemma.

This paper uses the following notation. Let Z denote the set of all integers. Let C denote the set of all complex numbers. Let $\boldsymbol{C}^{m \times n}$ denote the set of all $m \times n$ matrices with complex components. The superscripts T, $*$, and $\dagger$ denote, respectively, the transpose, the complex conjugate and the (Moore–Penrose) pseudo-inverse operations of a matrix. The symbol $\oplus$ denotes the direct sum of subspaces or the direct sum of matrices and the superscript $\perp$ denotes the orthogonal complement of a subspace [12]. The symbol $:=$ means equality by definition. Let $i = \overline{1, n}$ stand for $i = 1, 2, \cdots, n$.

II. MATRIX PSEUDO-INVERSION LEMMA

The following lemma gives an explicit formula of the pseudo-inverse for a positive semidefinite Hermitian matrix $\boldsymbol{A}$ added to a general rank-one matrix $\boldsymbol{bb}^H$ called a dyad.

Lemma 1: Let $\boldsymbol{A} \in \boldsymbol{C}^{m \times n}$ be a positive semidefinite Hermitian matrix, and let $\boldsymbol{b} \in \boldsymbol{C}^n$ be a nonzero vector. Let the linear vector space $\boldsymbol{C}^n$ be uniquely decomposed as $\boldsymbol{C}^n = \operatorname{Im}\boldsymbol{A} \oplus (\operatorname{Im}\boldsymbol{A})^\perp$, where $\operatorname{Im}\boldsymbol{A}$ denotes the image (or range) space of $\boldsymbol{A}$ and $(\operatorname{Im}\boldsymbol{A})^\perp$ denotes the orthogonal complement of $\operatorname{Im}\boldsymbol{A}$. Let $\boldsymbol{b} \in \boldsymbol{C}^n$ be decomposed uniquely as

$$\boldsymbol{b} = \boldsymbol{b}_1 \oplus \boldsymbol{b}_2 \text{ with } \boldsymbol{b}_1 \in \operatorname{Im}\boldsymbol{A} \text{ and } \boldsymbol{b}_2 \in (\operatorname{Im}\boldsymbol{A})^\perp. \quad (2)$$

Let $\boldsymbol{Q}$ be defined as

$$\begin{aligned} \boldsymbol{Q} &:= \boldsymbol{A} + \boldsymbol{bb}^H \\ &= \boldsymbol{A} + (\boldsymbol{b}_1 + \boldsymbol{b}_2)(\boldsymbol{b}_1 + \boldsymbol{b}_2)^H \\ &= \boldsymbol{A} + \boldsymbol{b}_1\boldsymbol{b}_1^H + \boldsymbol{b}_2\boldsymbol{b}_2^H + \boldsymbol{b}_1\boldsymbol{b}_2^H + \boldsymbol{b}_2\boldsymbol{b}_1^H \in \boldsymbol{C}^{n \times n}. \quad (3) \end{aligned}$$

Then, the pseudo-inverse $\boldsymbol{Q}^\dagger$ of the matrix $\boldsymbol{Q}$ is explicitly expressed, depending on the values of vectors $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$, as follows.

K. Kohno and Y. Inouye are with the Department of Electronic and Control Systems Engineering, Shimane University, Shimane 690-8504, Japan (e-mail: kohno@yonago-k.ac.jp; inouye@riko.shimane-u.ac.jp).

M. Kawamoto is with the Information Technology Research Institute, National Institute of Advanced Science and Technology (AIST), Ibaraki 305-8568, Japan (e-mail: m.kawamoto@aist.go.jp).

1) If $\boldsymbol{b}_2 = 0$, then

$$Q^\dagger = A^\dagger - \frac{A^\dagger \boldsymbol{b}_1 \boldsymbol{b}_1^H A^\dagger}{1 + \boldsymbol{b}_1^H A^\dagger \boldsymbol{b}_1}. \qquad (4)$$

2) If $\boldsymbol{b}_2 \neq 0$ and $\boldsymbol{b}_1 = 0$, then

$$Q^\dagger = A^\dagger + \left(\boldsymbol{b}_2^H\right)^\dagger \boldsymbol{b}_2^\dagger. \qquad (5)$$

3) If $\boldsymbol{b}_2 \neq 0$ and $\boldsymbol{b}_1 \neq 0$, then

$$Q^\dagger = Q_b^\dagger - Q_b^\dagger [\boldsymbol{b}_1, \boldsymbol{b}_2] Q_d^{-1} [\boldsymbol{b}_1, \boldsymbol{b}_2]^H Q_b^\dagger \qquad (6)$$

where $Q_b^\dagger$ and Q_d^{-1} are respectively defined by

$$Q_b^\dagger := A^\dagger - \frac{A^\dagger \boldsymbol{b}_1 \boldsymbol{b}_1^H A^\dagger}{1 + \boldsymbol{b}_1^H A^\dagger \boldsymbol{b}_1} + \left(\boldsymbol{b}_2^H\right)^\dagger \boldsymbol{b}_2^\dagger \qquad (7)$$

$$\begin{aligned} Q_d^{-1} &:= \left(P + [\boldsymbol{b}_1, \boldsymbol{b}_2]^H Q_b^\dagger [\boldsymbol{b}_1, \boldsymbol{b}_2] \right)^{-1} \\ &= \frac{1}{1 - \boldsymbol{b}_2^H Q_b^\dagger \boldsymbol{b}_2 \boldsymbol{b}_1^H Q_b^\dagger \boldsymbol{b}_1} \left[\begin{array}{c|c} -\boldsymbol{b}_2^H Q_b^\dagger \boldsymbol{b}_2 & 1 \\ \hline 1 & -\boldsymbol{b}_1^H Q_b^\dagger \boldsymbol{b}_1 \end{array} \right] \\ &\in C^{2 \times 2} \end{aligned} \qquad (8)$$

with

$$P := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \in R^{2 \times 2} \qquad (9)$$

where $R^{2 \times 2}$ is the set of all 2×2 matrices with real components.

The proof of Lemma 1 is very lengthy and is thus relegated to Appendix A.

It can be seen that the first and second expressions of the pseudo-inverse given in (4) and (5) can be included as special cases in the third expression of the pseudo-inverse given in (6). Namely, we have the following theorem.

Theorem 1: Under the same conditions in Lemma 1, it follows that

$$\begin{aligned} Q^\dagger &= (A + \boldsymbol{b}\boldsymbol{b}^H)^\dagger \\ &= Q_b^\dagger - Q_b^\dagger [\boldsymbol{b}_1, \boldsymbol{b}_2] Q_d^{-1} [\boldsymbol{b}_1, \boldsymbol{b}_2]^H Q_b^\dagger \end{aligned} \qquad (10)$$

where $Q_b^\dagger$ and Q_d^{-1} are defined by (7) and (8), respectively.

The proof of Theorem 1 is shown in Appendix B.

Remark 1: In the late 1980s, Ogawa extended the matrix inversion lemma to the singular case and presented an operator pseudo-inversion lemma [15]. Instead of the adding term $\boldsymbol{b}\boldsymbol{b}^H$ in (3), he treated a more general adding term $BC^{-1}B^H$, where B is an operator and C is an positive definite operator, but he gave the operator pseudo-inversion lemma under the condition

$$\mathrm{Im}\, B \subset \mathrm{Im}\, A. \qquad (11)$$

Therefore, case 1) of Lemma 1 is included in the case he just considered, but cases 2) and 3) are not treated by him. We should note that the above condition does not hold true in a nonstationary environment for blind deconvolution of MIMO systems (see Section IV for details).

In order to present illustrations of the matrix pseudo-inversion lemma, we demonstrate illustrative examples. We consider a single matrix A with three different values of vector $\boldsymbol{b}$, as follows:

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} [1 \quad 1 \quad 1] = \boldsymbol{x}\boldsymbol{x}^H. \qquad (12)$$

Case 1)

$$\boldsymbol{b} = [1 \quad 1 \quad 1]^H. \qquad (13)$$

Case 2)

$$\boldsymbol{b} = [1 \quad 1 \quad -2]^H. \qquad (14)$$

Case 3)

$$\boldsymbol{b} = [1 \quad 2 \quad 3]^H \qquad (15)$$

where $\boldsymbol{x}$ is a vector that is defined by $\boldsymbol{x} := [1\,1\,1]^H$. Then putting $Q := A + \boldsymbol{b}\boldsymbol{b}^H$, the matrix Q is calculated for each vector $\boldsymbol{b}$, as follows.

Case 1)

$$Q = 2 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \qquad (16)$$

Case 2)

$$Q = \begin{bmatrix} 2 & 2 & -1 \\ 2 & 2 & -1 \\ -1 & -1 & 5 \end{bmatrix}. \qquad (17)$$

Case 3)

$$Q = \begin{bmatrix} 2 & 3 & 4 \\ 3 & 5 & 7 \\ 4 & 7 & 10 \end{bmatrix}. \qquad (18)$$

Since $AA^\dagger$ is the orthogonal projection of C^3 onto $\mathrm{Im}\, A$ and $(I - AA^\dagger)$ is the orthogonal projection of C^3 onto $(\mathrm{Im}\, A)^\perp$ [7], it follows from (2) that the vector $\boldsymbol{b}$ is decomposed into two vectors $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ by using the following relations:

$$\boldsymbol{b}_1 = AA^\dagger \boldsymbol{b} \in \mathrm{Im}\, A, \qquad (19)$$
$$\boldsymbol{b}_2 = (I - AA^\dagger)\boldsymbol{b} \in (\mathrm{Im}\, A)^\perp \qquad (20)$$

where

$$A^\dagger = (\boldsymbol{x}\boldsymbol{x}^H)^\dagger = \frac{\boldsymbol{x}\boldsymbol{x}^H}{\|\boldsymbol{x}\|^4} = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad (21)$$

(cf. (96) and (97) in Appendix A), I is the 3×3 identity matrix, and $\|\boldsymbol{x}\|$ denotes the Euclidean norm of the vector $\boldsymbol{x}$.

First, we consider the case when $\boldsymbol{b} = [1\,1\,1]^H$ [Case 1)]. From (19)–(21), we have

$$\boldsymbol{b}_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}; \quad \boldsymbol{b}_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \qquad (22)$$

where we note $\boldsymbol{b} \in \operatorname{Im} \boldsymbol{A}$. Therefore, the pseudo-inverse $\boldsymbol{Q}^{\dagger}$ of the matrix $\boldsymbol{Q} = \boldsymbol{A} + \boldsymbol{b}\boldsymbol{b}^{H}$ is obtained by applying (4) which is the case when $\boldsymbol{b}_2 = 0$ in Lemma 1, as follows:

$$\boldsymbol{Q}^{\dagger} = \boldsymbol{A}^{\dagger} - \frac{\boldsymbol{A}^{\dagger}\boldsymbol{b}_1\boldsymbol{b}_1^{H}\boldsymbol{A}^{\dagger}}{1 + \boldsymbol{b}_1^{H}\boldsymbol{A}^{\dagger}\boldsymbol{b}_1} = \frac{1}{18}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \quad (23)$$

As an evaluation of the value of the pseudo-inverse $\boldsymbol{Q}^{\dagger}$, we consider the following error matrices defined by using the Moore–Penrose conditions [5], [6]:

$$E_1 := QQ^{\dagger}Q - Q \quad (24)$$
$$E_2 := Q^{\dagger}QQ^{\dagger} - Q^{\dagger} \quad (25)$$
$$E_3 := (QQ^{\dagger})^{H} - QQ^{\dagger} \quad (26)$$
$$E_4 := (Q^{\dagger}Q)^{H} - Q^{\dagger}Q. \quad (27)$$

We note that all of the error matrices vanish when the value of $\boldsymbol{Q}^{\dagger}$ is correct. Substituting (16) and (23) into (24)–(27), we obtain $\boldsymbol{E}_1 = 0$, $\boldsymbol{E}_2 = 0$, $\boldsymbol{E}_3 = 0$, and $\boldsymbol{E}_4 = 0$. Therefore, these results prove that the value of the pseudo-inverse $\boldsymbol{Q}^{\dagger}$ is correct in Case 1).

Similarly, we obtain the pseudo-inverse $\boldsymbol{Q}^{\dagger}$ and evaluate its value in the case when $\boldsymbol{b} = [1\,1\,-2]^{H}$ [Case 2)] and $\boldsymbol{b} = [1\,2\,3]^{H}$ [Case 3)]. From (19)–(21), two vectors $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ are obtained in each case, as follows.

Case 2)

$$\boldsymbol{b}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}; \qquad \boldsymbol{b}_2 = \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix}. \quad (28)$$

Case 3)

$$\boldsymbol{b}_1 = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}; \qquad \boldsymbol{b}_2 = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}. \quad (29)$$

The pseudo-inverse $\boldsymbol{Q}^{\dagger}$ of the matrix $\boldsymbol{Q} = \boldsymbol{A} + \boldsymbol{b}\boldsymbol{b}^{H}$ is calculated from (5) in Case 2) and from (6)–(8) in Case 3) as follows.

Case 2)

$$\boldsymbol{Q}^{\dagger} = \frac{1}{36}\begin{bmatrix} 5 & 5 & 2 \\ 5 & 5 & 2 \\ 2 & 2 & 8 \end{bmatrix}. \quad (30)$$

Case 3)

$$\boldsymbol{Q}^{\dagger} = \frac{1}{36}\begin{bmatrix} 73 & 16 & -41 \\ 16 & 4 & -8 \\ -41 & -8 & 25 \end{bmatrix}. \quad (31)$$

Case 2) corresponds to the case when $\boldsymbol{b}_2 \neq 0$ and $\boldsymbol{b}_1 = 0$ in Lemma 1, and Case 3) corresponds to the case when $\boldsymbol{b}_2 \neq 0$ and $\boldsymbol{b}_1 \neq 0$ in Lemma 1.

Then, the values of the pseudo-inverse $\boldsymbol{Q}^{\dagger}$ are evaluated by using the error matrices in (24)–(27). Because all of the error matrices become zero, it can be seen that the values of the pseudo-inverse $\boldsymbol{Q}^{\dagger}$ are correct in Cases 2) and 3).

Therefore, these examples may convince ourselves that Lemma 1 is true.

III. APPLICATION TO ADAPTIVE BLIND DECONVOLUTION

This section provides a brief explanation of the problem of blind deconvolution of MIMO systems and introduce the multi-channel SEA (MSEA) proposed in [8] for reader convenience, but the reader who is familiar with this problem along with the MSEA may proceed to the line below (37).

Let us consider a MIMO system with n inputs and m outputs as described by

$$\boldsymbol{y}(t) = \sum_{k=-\infty}^{\infty} \boldsymbol{H}^{(k)}\boldsymbol{s}(t - k), \quad t \in Z \quad (32)$$

where $\boldsymbol{y}(t)$ is an m-column vector of output signals, $\boldsymbol{s}(t)$ is an n-column vector of input (or source) signals, and $\{\boldsymbol{H}^{(k)}\}$ is an $m \times n$ matrix sequence called the impulse response. The transfer function of the channel is defined by

$$\boldsymbol{H}(z) = \sum_{k=-\infty}^{\infty} \boldsymbol{H}^{(k)}z^{-k}, \quad z \in C. \quad (33)$$

It is assumed for theoretical analysis that the noise is absent in (32).

To recover the source signals, we process the output signals by an $n \times m$ equalizer (or deconvolver) $\boldsymbol{W}(z)$ described by

$$\boldsymbol{z}(t) = \sum_{k=-\infty}^{\infty} \boldsymbol{W}^{(k)}\boldsymbol{y}(t - k), \quad t \in Z. \quad (34)$$

The objective of multichannel blind deconvolution is to construct an equalizer that recovers the original source signals only from the measurements of the corresponding outputs.

We put the following assumptions on the channel, the source signals, and the deconvolver [8].

A1) The transfer function $\boldsymbol{H}(z)$ is stable and has full column rank on the unit circle $|z| = 1$ (this implies that the unknown system has less inputs than outputs, i.e., $n \leq m$, and there exists a left stable inverse of the unknown system).

A2) The input sequence $\{\boldsymbol{s}(t)\}$ is a complex, zero-mean, non-Gaussian random vector process with element processes $\{s_i(t)\}$, $i = \overline{1, n}$. The element processes are mutually independent. Moreover, each element process $\{s_i(t)\}$ is an i.i.d. process with a variance $\sigma_i^2 \neq 0$ and a fourth-order cumulant $\gamma_i \neq 0$. The variances σ_i^2's and the fourth-order cumulants γ_i's are unknown.

A3) The equalizer $\boldsymbol{W}(z)$ is a finite-impulse-response (FIR) channel of sufficient length L so that the truncation effect can be ignored.

Under A3), let us consider an FIR equalizer with the transfer function $\boldsymbol{W}(z)$ given by

$$\boldsymbol{W}(z) = \sum_{k=L_1}^{L_2} \boldsymbol{W}^{(k)}z^{-k} \quad (35)$$

where L_1 and L_2 are respectively the first and the last super-scripted numbers of the tap coefficients $\boldsymbol{W}^{(k)}$'s of the equalizer $\boldsymbol{W}(z)$, and the length $L := L_2 - L_1 + 1$ is taken to be sufficiently large. Let $\tilde{\boldsymbol{w}}_i$ be the mL-column vector consisting of the tap coefficients (corresponding to the ith output) of the equalizer defined by

$$\tilde{\boldsymbol{w}}_i := \left[\boldsymbol{w}_{i,1}^T, \boldsymbol{w}_{i,2}^T, \cdots, \boldsymbol{w}_{i,m}^T\right]^T \in \boldsymbol{C}^{mL} \qquad (36)$$

$$\boldsymbol{w}_{i,j} := \left[w_{i,j}^{(L_1)}, w_{i,j}^{(L_1+1)}, \cdots, w_{i,j}^{(L_2)}\right]^T \in \boldsymbol{C}^L \qquad (37)$$

where $w_{i,j}^{(k)}$ is the (i,j)th element of matrix $\boldsymbol{W}^{(k)}$.

Inouye and Tanebe [8] proposed the MSEA for finding the tap coefficient vectors $\tilde{\boldsymbol{w}}_i$'s of the equalizer $\boldsymbol{W}(z)$, each iteration of which consists of the following two steps:

$$\tilde{\boldsymbol{w}}_i^{[1]} = \tilde{\boldsymbol{R}}^\dagger \tilde{\boldsymbol{d}}_i, \qquad \text{for} \quad i = \overline{1, n} \qquad (38)$$

$$\tilde{\boldsymbol{w}}_i^{[2]} = \frac{\tilde{\boldsymbol{w}}_i^{[1]}}{\sqrt{\tilde{\boldsymbol{w}}_i^{[1]H} \tilde{\boldsymbol{R}} \tilde{\boldsymbol{w}}_i^{[1]}}}, \qquad \text{for} \quad i = \overline{1, n} \qquad (39)$$

where $(\cdot)^{[1]}$ and $(\cdot)^{[2]}$ stand, respectively, for the result of the first step and the result of the second step. Let $\tilde{\boldsymbol{y}}(t)$ be the mL-column vector consisting of the L consecutive inputs of the equalizer defined by

$$\tilde{\boldsymbol{y}}(t) := \left[\bar{\boldsymbol{y}}_1(t)^T, \bar{\boldsymbol{y}}_2(t)^T, \cdots, \bar{\boldsymbol{y}}_m(t)^T\right]^T \in \boldsymbol{C}^{mL} \qquad (40)$$

$$\bar{\boldsymbol{y}}_i(t) := [y_i(t - L_1), y_i(t - L_1 - 1),$$
$$\cdots, y_i(t - L_2)]^T \in \boldsymbol{C}^L \qquad (41)$$

where $y_i(t)$ is the ith element of the output vector $\boldsymbol{y}(t)$ of the channel in (32). Then, the correlation matrix $\tilde{\boldsymbol{R}}$ is represented as

$$\tilde{\boldsymbol{R}} = E\left[\tilde{\boldsymbol{y}}^*(t)\tilde{\boldsymbol{y}}^T(t)\right] \in \boldsymbol{C}^{mL \times mL} \qquad (42)$$

and the fourth-order cumulant vector $\tilde{\boldsymbol{d}}_i$ is represented as

$$\tilde{\boldsymbol{d}}_i = \text{cum}\left(z_i(t), z_i(t), z_i^*(t), \tilde{\boldsymbol{y}}^*(t)\right)$$
$$= E\left[|z_i(t)|^2 z_i(t)\tilde{\boldsymbol{y}}^*(t)\right] - 2E\left[|z_i(t)|^2\right] E\left[z_i(t)\tilde{\boldsymbol{y}}^*(t)\right]$$
$$- E\left[z_i^2(t)\right] E\left[z_i^*(t)\tilde{\boldsymbol{y}}^*(t)\right] \in \boldsymbol{C}^{mL} \qquad (43)$$

where $E[x]$ denotes the expectation of a random variable x. We note that the last term can be ignored in case of $E[s_i(t)^2] = 0$, in which case $E[z_i(t)^2] = 0$ for all $i = \overline{1, n}$.

In addition, we obtain (from [8, eqs. (32) and (39)])

$$\tilde{\boldsymbol{d}}_i \in \text{Im} \, \tilde{\boldsymbol{R}}, \qquad \text{for} \quad i = \overline{1, n}. \qquad (44)$$

We should note that the stationarity of the input process $\{\boldsymbol{s}(t)\}$ [or the assumption A2)] ensures the relation (44) and that the relation (44) means that the vector $\tilde{\boldsymbol{w}}_i^{[1]}$ obtained by (38) satisfies

$$\tilde{\boldsymbol{R}}\tilde{\boldsymbol{w}}_i^{[1]} = \tilde{\boldsymbol{d}}_i. \qquad (45)$$

Consider the batch algorithm in (38) and (39). Equation (39) constrains a weighted norm of vector $\tilde{\boldsymbol{w}}_i$ to equal one, and thus

we assume this constraint is always satisfied using a normalization or an automatic gain control (AGC) of $\tilde{\boldsymbol{w}}_i$ at each time t. To develop an adaptive version of (38), we must specify the dependency of each time t and rewrite (38) as

$$\tilde{\boldsymbol{w}}_i(t) = \tilde{\boldsymbol{R}}^\dagger(t)\tilde{\boldsymbol{d}}_i(t), \qquad i = \overline{1, n}. \qquad (46)$$

In order to develop an adaptive version of the MSEA, we should obtain recursion formulas for time-updating of matrix $\tilde{\boldsymbol{R}}(t)$, vector $\tilde{\boldsymbol{d}}_i(t)$, and pseudo-inverse $\tilde{\boldsymbol{R}}^\dagger(t)$ in (46), respectively, as

$$\tilde{\boldsymbol{R}}(t) = (1 - \alpha_t)\tilde{\boldsymbol{R}}(t-1) + \alpha_t \tilde{\boldsymbol{y}}^*(t)\tilde{\boldsymbol{y}}^T(t) \qquad (47)$$

$$\tilde{\boldsymbol{d}}_i(t) = (1 - \alpha_t)\tilde{\boldsymbol{d}}_i(t-1) + \alpha_t \tilde{\boldsymbol{y}}^*(t)\tilde{z}_i(t) \qquad (48)$$

where

$$\tilde{z}_i(t) := \left(|z_i(t)|^2 - 2\left\langle|z_i(t)|^2\right\rangle\right) z_i(t) - \left\langle z_i^2(t)\right\rangle z_i^*(t). \quad (49)$$

Here, $\langle|z_i(t)|^2\rangle$ and $\langle z_i^2(t)\rangle$ denote the estimates of $E[|z_i(t)|^2]$ and $E[z_i(t)^2]$ at time t, respectively, and α_t is a positive number close to, but greater than zero, which accounts for some exponential weighting factor or forgetting factor [3]. For example, we may take $\alpha_t = 1/t$.

Because we consider the case when the number of inputs n is less than the number of output m, i.e., $n < m$, the correlation matrix $\tilde{\boldsymbol{R}}(t)$ is not of full rank and a singular matrix [9]. Therefore, we may apply the matrix pseudo-inversion lemma to the recursive equation (47).

By applying Lemma 1 to (47) for obtaining a recursive formula for time-updating of pseudo-inverse $\boldsymbol{P}(t) = \tilde{\boldsymbol{R}}^\dagger(t)$, we have the following lemma.

Lemma 2: Let $\boldsymbol{Q}, \boldsymbol{Q}^\dagger, \boldsymbol{A}, \boldsymbol{A}^\dagger, \boldsymbol{b}, \boldsymbol{b}_1,$ and $\boldsymbol{b}_2$ in Lemma 1 be respectively defined as

$$\boldsymbol{Q} := \tilde{\boldsymbol{R}}(t), \qquad (50)$$

$$\boldsymbol{Q}^\dagger := \boldsymbol{P}(t) = \tilde{\boldsymbol{R}}^\dagger(t) \qquad (51)$$

$$\boldsymbol{A} := (1 - \alpha_t)\tilde{\boldsymbol{R}}(t-1) \qquad (52)$$

$$\boldsymbol{A}^\dagger := \frac{1}{1 - \alpha_t}\tilde{\boldsymbol{R}}^\dagger(t-1) = \frac{1}{1 - \alpha_t}\boldsymbol{P}(t-1) \qquad (53)$$

$$\boldsymbol{b} := \boldsymbol{b}(t) = \sqrt{\alpha_t}\tilde{\boldsymbol{y}}^*(t) \qquad (54)$$

$$\boldsymbol{b}_1 := \boldsymbol{b}_1(t) = \tilde{\boldsymbol{R}}(t-1)\boldsymbol{P}(t-1)\boldsymbol{b}(t) \qquad (55)$$

$$\boldsymbol{b}_2 := \boldsymbol{b}_2(t) = \left\{\boldsymbol{I} - \tilde{\boldsymbol{R}}(t-1)\boldsymbol{P}(t-1)\right\}\boldsymbol{b}(t). \qquad (56)$$

Then, substituting these definitions into Lemma 1, the recursion for the pseudo-inverse $\boldsymbol{P}(t) = \tilde{\boldsymbol{R}}^\dagger$ of the correlation matrix $\tilde{\boldsymbol{R}}(t)$ from $\boldsymbol{P}(t-1)$ is explicitly expressed, depending on the values of vectors $\boldsymbol{b}_1(t)$ and $\boldsymbol{b}_2(t)$, as follows.

1) If $\boldsymbol{b}_2(t) = 0$, then

$$\boldsymbol{P}(t) = \frac{1}{1 - \alpha_t}\left[\boldsymbol{P}(t-1) - \frac{\boldsymbol{P}(t-1)\boldsymbol{b}_1(t)\boldsymbol{b}_1^H(t)\boldsymbol{P}(t-1)}{1 - \alpha_t + \boldsymbol{b}_1^H(t)\boldsymbol{P}(t-1)\boldsymbol{b}_1(t)}\right].$$
$$(57)$$

2) If $\boldsymbol{b}_2(t) \neq 0$ and $\boldsymbol{b}_1(t) = 0$, then

$$\boldsymbol{P}(t) = \frac{1}{1 - \alpha_t}\boldsymbol{P}(t-1) + \left\{\boldsymbol{b}_2^H(t)\right\}^\dagger \boldsymbol{b}_2^\dagger(t). \qquad (58)$$

3) If $\boldsymbol{b}_2 \neq 0$ and $\boldsymbol{b}_1 \neq 0$, then

$$\boldsymbol{P}(t) = \boldsymbol{P}_b^\dagger(t) - \boldsymbol{P}_b^\dagger(t) \left[\boldsymbol{b}_1(t), \boldsymbol{b}_2(t)\right] \boldsymbol{P}_d^{-1}(t) \left[\boldsymbol{b}_1(t), \boldsymbol{b}_2(t)\right]^H \boldsymbol{P}_b^\dagger(t) \tag{59}$$

where $\boldsymbol{P}_b^\dagger(t)$ and $\boldsymbol{P}_d^{-1}(t)$ are, respectively, defined by

$$\boldsymbol{P}_b^\dagger(t) := \frac{1}{1-\alpha_t} \left[\boldsymbol{P}(t-1) - \frac{\boldsymbol{P}(t-1)\boldsymbol{b}_1(t)\boldsymbol{b}_1^H(t)\boldsymbol{P}(t-1)}{1-\alpha_t + \boldsymbol{b}_1^H(t)\boldsymbol{P}(t-1)\boldsymbol{b}_1(t)}\right]$$
$$+ \left\{\boldsymbol{b}_2^H(t)\right\}^\dagger \boldsymbol{b}_2^\dagger(t) \tag{60}$$

$$\boldsymbol{P}_d^{-1}(t) := \frac{1}{1 - \boldsymbol{b}_2^H(t)\boldsymbol{P}_b^\dagger(t)\boldsymbol{b}_2(t)\boldsymbol{b}_1^H(t)\boldsymbol{P}_b^\dagger(t)\boldsymbol{b}_1(t)}$$
$$\times \left[\begin{array}{c|c} -\boldsymbol{b}_2^H(t)\boldsymbol{P}_b^\dagger(t)\boldsymbol{b}_2(t) & 1 \\ \hline 1 & -\boldsymbol{b}_1^H(t)\boldsymbol{P}_b^\dagger(t)\boldsymbol{b}_1(t) \end{array}\right] \tag{61}$$

These equations are initialized by their values appropriately selected or calculated by the batch algorithm in (38) and (39) at initial time t_0 and used for $t = t_0 + 1, t_0 + 2, \cdots$.

The proof of Lemma 2 is easy and follows from Lemma 1 along with simple calculations.

Based on Lemma 2 along with (46)–(48), we have the following theorem, which gives a recursion formula for time-updating of the tap vector $\tilde{\boldsymbol{w}}_i(t)$ for $i = \overline{1, n}$.

Theorem 2: The recursion for $\tilde{\boldsymbol{w}}_i(t)$ is

$$\tilde{\boldsymbol{w}}_i(t) = \boldsymbol{P}(t)\tilde{\boldsymbol{R}}(t)\tilde{\boldsymbol{w}}_i(t-1) + \boldsymbol{k}(t)\left[\tilde{z}_i(t) - \tilde{\boldsymbol{y}}^T(t)\tilde{\boldsymbol{w}}_i(t-1)\right] \tag{62}$$

where

$$\boldsymbol{k}(t) := \alpha_t \boldsymbol{P}(t)\tilde{\boldsymbol{y}}^*(t) \tag{63}$$

$$\tilde{z}_i(t) := \left(|z_i(t)|^2 - 2\left\langle|z_i(t)|^2\right\rangle\right) z_i(t)$$
$$- \left\langle z_i^2(t)\right\rangle z_i^*(t) \tag{64}$$

$$\left\langle|z_i(t)|^2\right\rangle := (1-\beta_t)\left\langle|z_i(t-1)|^2\right\rangle + \beta_t |z_i(t)|^2 \tag{65}$$

$$\left\langle z_i^2(t)\right\rangle := (1-\beta_t)\left\langle z_i^2(t-1)\right\rangle + \beta_t z_i^2(t). \tag{66}$$

Here, β_t is a positive constant greater than α_t but less than one, and $\boldsymbol{P}(t) = \tilde{\boldsymbol{R}}^\dagger(t)$ is the pseudo-inverse of $\tilde{\boldsymbol{R}}(t)$ and is calculated using the formula (59).

The proof of Theorem 2 is shown in Appendix C.

Remark 2: The recursive algorithm proposed by Shalvi and Weinstein [13] can be shown to correspond to the particular case of Theorem 2, where $n = m = 1$ and the correlation matrices $\tilde{\boldsymbol{R}}(t)$'s are nonsingular.

Remark 3: Based on Theorem 2, we have developed a recursive algorithm implementing the recursion formula (62) [14], but we have not yet theoretically analyzed the stability and rate of convergence of the algorithm. See [16] for details of the stability analysis of the standard recursive least-squares (RLS) algorithm and its variants.

Remark 4: It is well known that the standard RLS algorithm uses the matrix inversion lemma and the order of N^2 (i.e., $O(N^2)$) computations per recursive iteration, where N is the

filter order [16]. The same computational complexity of $O(N^2)$ holds true for Theorem 2. Namely, the recursive algorithm implementing the recursion (62) requires $O(N^2)$ computations per recursive iteration, where $N = mL$. On the other hand, the inversion or the pseudo-inversion of an $N \times N$ matrix requires $O(N^3)$ computations [17, p. 239]. We shall see this computational difference between the two methods through computer simulations in Section IV.

IV. SIMULATION RESULTS

To demonstrate the usefulness of the matrix pseudo-inversion lemma, some computer simulations for obtaining the pseudo-inverse $\boldsymbol{P}(t) = \tilde{\boldsymbol{R}}^\dagger(t)$ of the correlation matrix $\tilde{\boldsymbol{R}}(t)$ in (47) by using Lemma 2 were conducted. We note here that we do not use Theorem 2, because we are not interested in finding $\tilde{\boldsymbol{w}}_i(t)$'s but interested in calculating $\tilde{\boldsymbol{R}}^\dagger(t)$. The results of calculating $\tilde{\boldsymbol{w}}_i(t)$'s and recovering original sources $\boldsymbol{s}(t)$'s are found in [11] and [14].

We considered a MIMO system $\boldsymbol{H}(z)$ with two inputs ($n = 2$) and three outputs ($m = 3$) and assumed that the system $\boldsymbol{H}(z)$ is FIR and the length of channel is three, that is, $\boldsymbol{H}^{(k)}$'s in (33) were set to be

$$\boldsymbol{H}(z)$$
$$= \sum_{k=0}^{2} \boldsymbol{H}^{(k)} z^{-k}$$
$$= \begin{bmatrix} 1.00+0.15z^{-1}+0.10z^{-2} & 0.65+0.25z^{-1}+0.15z^{-2} \\ 0.50-0.10z^{-1}+0.20z^{-2} & 1.00+0.25z^{-1}+0.10z^{-2} \\ 0.60+0.10z^{-1}+0.40z^{-2} & 0.10+0.20z^{-1}+0.10z^{-2} \end{bmatrix}. \tag{67}$$

Two source signals were 4-PSK and 8-PSK signals, respectively. The length of the deconvolver was chosen to be five ($L = 5$). For obtaining the pseudo-inverse of the correlation matrix, the initial values of $\tilde{\boldsymbol{R}}, \tilde{\boldsymbol{d}}_i$, and $\boldsymbol{P}$ were estimated by approximating ensemble averages with empirical averages using 100 data samples. The value of α_t was chosen as $\alpha_t = 1/t$ for each t. As a measure of performance, we use the following sum of the Euclidean or Frobenius norms of the four error matrices in (24) through (27) for each t:

$$M(t) := \frac{\|\boldsymbol{E}_1(t)\| + \|\boldsymbol{E}_2(t)\| + \|\boldsymbol{E}_3(t)\| + \|\boldsymbol{E}_4(t)\|}{\left\|\tilde{\boldsymbol{R}}(t)\right\|}. \tag{68}$$

First, we considered the time-invariant MIMO system in (67). In this case, from A2), the input process $\{\boldsymbol{s}(t)\}$ is stationary, and this means that the random process $\{\tilde{\boldsymbol{y}}^*(t)\}$ is also stationary. Thus, it follows from (42) that $\tilde{\boldsymbol{y}}^*(t)$ belongs almost surely (a.s.) (or with probability 1) to $\text{Im } \tilde{\boldsymbol{R}}$, that is

$$\tilde{\boldsymbol{y}}^*(t) \in \text{Im } \tilde{\boldsymbol{R}}, \quad \text{a.s.} \tag{69}$$

The proof of this relation is shown in Appendix D. On the other hand, it follows from (47) with $\tilde{\boldsymbol{R}}(0) = \boldsymbol{0}$ and $\alpha_t = 1/t$ that

$$\tilde{\boldsymbol{R}}(t) = \frac{1}{t}\sum_{k=1}^{t} \tilde{\boldsymbol{y}}^*(k)\tilde{\boldsymbol{y}}^T(k). \tag{70}$$
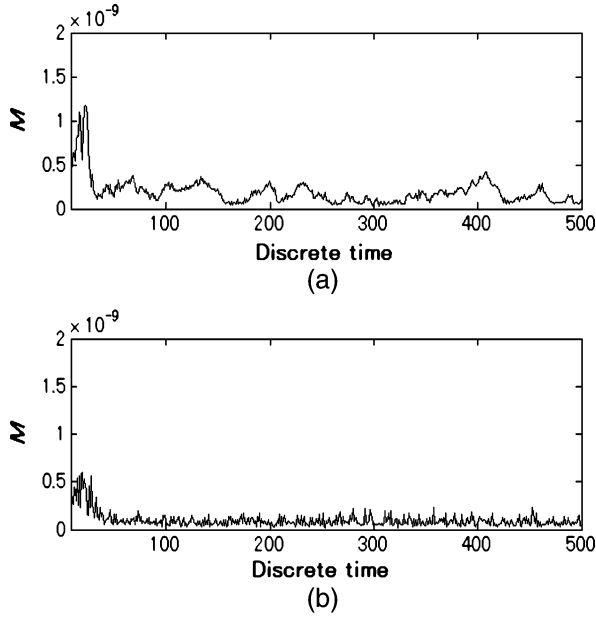
Fig. 1. Values of the M versus iterations (or discrete times) (a) using the matrix pseudo-inversion lemma and (b) using the built-in function "pinv" for the time-invariant MIMO system.

TABLE I
COMPARISON OF THE NUMBERS OF FLOATING POINT OPERATIONS $[\times 10^6]$

The length of the deconvolver	The matrix pseudo-inversion lemma	The built-in function "pinv"
$L = 5$	9.09	96.81
$L = 10$	35.60	659.58

TABLE II
COMPARISON OF THE EXECUTION TIMES [s]

The length of the deconvolver	The matrix pseudo-inversion lemma	The built-in function "pinv"
$L = 5$	0.1942	0.3485
$L = 10$	0.3925	1.1517

If the input process $\{\boldsymbol{s}(t)\}$ is ergodic, then

$$\lim_{t \to \infty} \tilde{\boldsymbol{R}}(t) = \tilde{\boldsymbol{R}}, \quad \text{a.s.} \tag{71}$$

Since the sequence $\{\text{rank } \tilde{\boldsymbol{R}}(t)\}$ is monotonically increasing, (71) implies that there exists a time T such that

$$\text{rank } \tilde{\boldsymbol{R}}(t) = \text{rank } \tilde{\boldsymbol{R}}, \quad \text{for any } t \geq T. \tag{72}$$

Therefore, it follows from (69) and (72) that

$$\tilde{\boldsymbol{y}}^*(t) \in \text{Im } \tilde{\boldsymbol{R}}(t-1), \quad \text{for} \quad t \geq T+1 \tag{73}$$

which means from (52) and (54) that

$$\boldsymbol{b} \in \text{Im } \boldsymbol{A} \tag{74}$$

where $\boldsymbol{b}$ and Im $\boldsymbol{A}$ are, respectively, defined by (54) and (52). Therefore, we can assume in this application that the component vector $\boldsymbol{b}_2 = \boldsymbol{b}_2(t)$ always vanishes, that is, $\boldsymbol{b}_2(t) = 0$, and we can use the recursion formula (57) for calculating the pseudo-inverse $\boldsymbol{P}(t) = \tilde{\boldsymbol{R}}^\dagger(t)$ at each iteration (or time) t.

We compared the performance of the proposed method (i.e., the method using the matrix pseudo-inversion lemma) with the performance of the method using the built-in function "pinv" in MATLAB Version 7.1.0 for calculating the pseudo-inverse $\boldsymbol{P}(t) = \tilde{\boldsymbol{R}}^\dagger(t)$ of correlation matrix $\tilde{\boldsymbol{R}}(t)$. The pseudo-inverses are calculated iteratively (or recursively) for each iteration (or recursion) number t for the two methods. Fig. 1(a) shows the performance results of the performance measure M for the proposed method by using 500 data samples. Fig. 1(b) shows the performance results of the M for the latter method by using the same data samples.

We also compared performances of the two methods in computational complexity by using the built-in function "flops" in

MATLAB Version 5.2. Table I shows the average of the numbers of floating point operations (flops) over 100 independent Monte Carlo runs using 500 data samples of the outputs for each Monte Carlo run by changing the length of the deconvolver L from 5 to 10.

Moreover, we compared performances of the two methods in execution time by using a PC with a 3.0-GHz processor and 1-GB main memories used in simulation experiments. Table II shows the average of the execution times over 100 independent Monte Carlo runs using 500 data samples of the outputs for each Monte Carlo run by changing the length of the deconvolver L from 5 to 10.

It can be seen from Fig. 1 that the accuracy of the matrix pseudo-inversion lemma is equivalent to the built-in function "pinv." However, Table I shows that the average of the numbers of floating-point operations for the proposed method is much smaller than that for the method using the built-in function "pinv." The ratios of the former average to the latter average are about 0.094 and 0.054 for $L = 5$ and $L = 10$, respectively. Table II shows that the average of the execution times for the proposed method is smaller than that for the method using the built-in function "pinv." The ratios of the former average to the latter average are about 0.56 and 0.34 for $L = 5$ and $L = 10$, respectively.

The computational complexity of the method using the built-in function "pinv" increases more than the computational complexity of the proposed method when the length of the deconvolver L increases. We consider that one of reasons why the matrix pseudo-inversion lemma is superior to the built-in function "pinv" in the number of floating-point operations and the execution times is that it is not necessary to calculate the pseudo-inverse $\boldsymbol{A}^\dagger$ of (4) in Lemma 1, because the results of the previous iteration $\boldsymbol{P}(t-1) = \tilde{\boldsymbol{R}}^\dagger(t-1)$ can be used instead of $\boldsymbol{A}^\dagger$ in (57) of Lemma 2 (see Remark 4 for details).

Next, we considered a time-variant MIMO system, where the system is the same in (67) except that the last matrix $\boldsymbol{H}^{(2)}$ of the impulse response of the channel was varied by adding 0.3 to all of its elements at discrete time $t = 200$. In this case, the random process $\{\tilde{\boldsymbol{y}}^*(t)\}$ is not stationary, that is, $\boldsymbol{b}_2(t) \neq 0$. Fig. 2(a) and (b) shows the performance results of the performance measure M for the proposed method and the method using the built-in function "pinv" by using 500 data samples of the outputs of the MIMO time-variant system. Fig. 2(c) shows
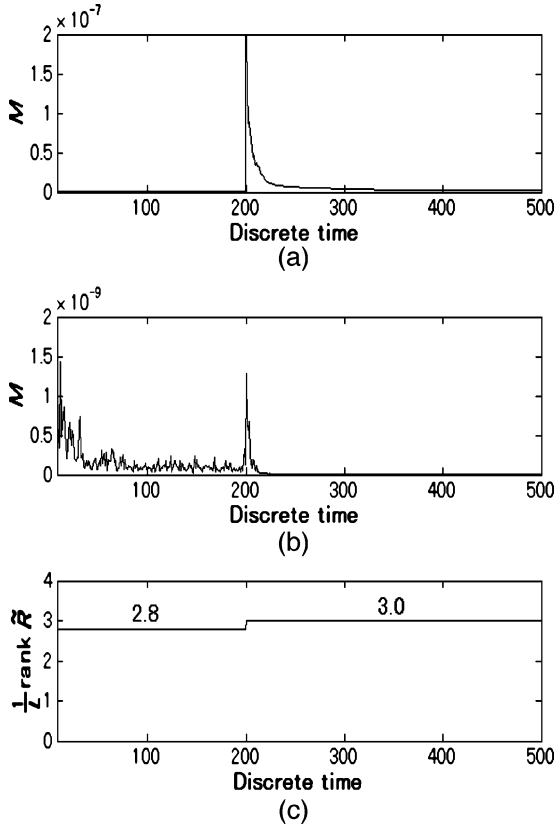
Fig. 2. Values of the M versus iterations (a) using the matrix pseudo-inversion lemma and (b) using the built-in function "pinv." (c) Values of $(1/L)\mathrm{rank}\ \tilde{\boldsymbol{R}}$ for the time-variant MIMO system.

the values of the rank of $\tilde{\boldsymbol{R}}$ normalized by the length of the deconvolver L in this case.

It can be seen from Fig. 2(a) and (b) that the accuracy of the matrix pseudo-inversion lemma is slightly worse than that of the built-in function "pinv," but the formula (10) of the matrix pseudo-inversion lemma can treat the changes of the channel.

Here, when we use (7), we should note how to calculate third term $(\boldsymbol{b}_2^H)^{\dagger}\boldsymbol{b}_2^{\dagger}$ of the right-hand side of (7). There exists a very small positive number ϵ_0 (which depends on a equipment used), such that numbers less than ϵ_0 are considered to be zero [7]. Therefore, we should set

$$\boldsymbol{b}_2^{\dagger} = \begin{cases} \frac{\boldsymbol{b}_2^H}{\|\boldsymbol{b}_2\|^2}, & \text{if } \|\boldsymbol{b}_2\|^2 > \epsilon_0 \\ 0, & \text{if } \|\boldsymbol{b}_2\|^2 \leq \epsilon_0. \end{cases} \tag{75}$$

Moreover, it can be seen from Fig. 2(c) that the rank of $\tilde{\boldsymbol{R}}$ normalized by the length of the deconvolver L changed from 2.8 to 3.0 at discrete time $t = 200$. In this simulation, the recursion formula (57) in Lemma 2 was used for calculating the pseudo-inverse of $\tilde{\boldsymbol{R}}$ except when $t = 200$. Only when $t = 200$, just the channel changed, and the recursion formula (59) in Lemma 2 was used for calculating the pseudo-inverse of $\tilde{\boldsymbol{R}}$. Therefore, we consider that the value of rank $\tilde{\boldsymbol{R}}$ normalized by the length of the deconvolver L can be used as an index for detecting the change of the channel and for changing the recursion formulas of the matrix pseudo-inversion lemma from (4)–(6) (see [14] for details).

From these results, the matrix pseudo-inversion lemma is useful to calculate the pseudo-inverse of a correlation matrix for adaptive algorithms of blind deconvolution in time-invariant MIMO systems and even time-variant MIMO systems.

V. CONCLUSION

We extended the matrix inversion lemma to the case when the matrix $\boldsymbol{A}$ in $\boldsymbol{A} + \boldsymbol{bb}^H$ is singular and presented a matrix pseudo-inversion lemma together with some illustrative examples. In order to show the usefulness of this lemma, we applied it to develop an adaptive SEA for the blind deconvolution of a MIMO system. It has been shown through computer simulations that the matrix pseudo-inversion lemma is useful for adaptive algorithms of blind deconvolution in time-invariant MIMO systems and even time-variant MIMO systems.

We have already found an explicit formula of the pseudo-inverse of an positive-semidefinite matrix $\boldsymbol{A} + \boldsymbol{BB}^H$, where $\boldsymbol{B}$ is an $n \times l$ matrix. Some details concerning this formula and its application to block-based adaptive blind deconvolution of MIMO systems can be found in [18] and [19]. However, we have not yet extended it to the case where $\boldsymbol{A}$ and $\boldsymbol{B}$ become linear bounded operators with a closed range. This type of extension is very important in applications to image processing [15] and is one of the remaining problems. Another important remaining issue is the stability analysis of the algorithm mentioned in Remark 3.

APPENDIX A
PROOF OF LEMMA 1

First, we consider the case when $\boldsymbol{b}_2 = 0$. Put

$$\boldsymbol{Q}_a := \boldsymbol{A} + \boldsymbol{b}_1\boldsymbol{b}_1^H. \tag{76}$$

Since $\boldsymbol{A}$ is Hermitian, it is unitary diagonalizable, that is, there exist a unitary matrix $\boldsymbol{U}$ and a diagonal matrix $\boldsymbol{\Lambda}$ such that

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^H \tag{77}$$

which is called the spectral decomposition (or eigenvalue decomposition) of $\boldsymbol{A}$. In (77), the diagonal elements λ_i's are arranged in decreasing order as

$$\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n) \tag{78}$$

where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$. If rank $\boldsymbol{A} = r < n$, then $\lambda_i = 0$ for $r < i \leq n$. Thus, we have $\boldsymbol{U}$ and $\boldsymbol{\Lambda}$ partitioned as

$$\boldsymbol{U} = [\boldsymbol{U}_1, \boldsymbol{U}_2], \ \ \boldsymbol{U}_1 \in \boldsymbol{C}^{n \times r}, \ \ \boldsymbol{U}_2 \in \boldsymbol{C}^{n \times (n-r)} \tag{79}$$

$$\boldsymbol{\Lambda} = \begin{bmatrix} \boldsymbol{\Lambda}_1 & 0 \\ 0 & \boldsymbol{\Lambda}_2 \end{bmatrix}, \ \ \boldsymbol{\Lambda}_1 \in \boldsymbol{C}^{r \times r}, \ \ \boldsymbol{\Lambda}_2 \in \boldsymbol{C}^{(n-r) \times (n-r)} \tag{80}$$

where $\boldsymbol{\Lambda}_2 = 0$. Therefore, from (77), we have

$$\boldsymbol{A} = [\boldsymbol{U}_1, \boldsymbol{U}_2] \begin{bmatrix} \boldsymbol{\Lambda}_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_1^H \\ \boldsymbol{U}_2^H \end{bmatrix} = \boldsymbol{U}_1\boldsymbol{\Lambda}_1\boldsymbol{U}_1^H. \tag{81}$$

Since $b_1 \in \text{Im } A$, there exists $\xi_1 \in C^r$ such that

$$b_1 = U_1 \xi_1. \tag{82}$$

From (76), (81), and (82), we have

$$
\begin{aligned}
Q_a &= U_1 \left(\Lambda_1 + \xi_1 \xi_1^H \right) U_1^H \\
&= [U_1, U_2] \left[\begin{array}{c|c} \Lambda_1 + \xi_1 \xi_1^H & 0 \\ \hline 0 & 0 \end{array} \right] \left[\begin{array}{c} U_1^H \\ U_2^H \end{array} \right].
\end{aligned} \tag{83}
$$

Here, we note the following properties of matrix pseudo-inverses [5].

P1)   If U and V are unitary, then

$$(UAV)^\dagger = V^\dagger A^\dagger U^\dagger = V^H A^\dagger U^H. \tag{84}$$

P2)   If $A = A_1 + A_2$ where $A_1^H A_2 = 0$ and $A_1 A_2^H = 0$, then

$$A^\dagger = A_1^\dagger + A_2^\dagger. \tag{85}$$

P3)   Let $r = \text{rank } A$, and let

$$A = BC \in C^{n \times m}, \quad B \in C^{n \times r}, \quad C \in C^{r \times m} \tag{86}$$

be a rank decomposition of $A \in C^{n \times m}$, that is, rank B = rank $C = r$. Then

$$A^\dagger = C^\dagger B^\dagger. \tag{87}$$

Using (83) and the above properties, we have

$$
\begin{aligned}
Q_a^\dagger &= [U_1, U_2] \left[\begin{array}{c|c} \Lambda_1 + \xi_1 \xi_1^H & 0 \\ \hline 0 & 0 \end{array} \right]^\dagger \left[\begin{array}{c} U_1^H \\ U_2^H \end{array} \right] \\
&= [U_1, U_2] \left[\begin{array}{c|c} \left(\Lambda_1 + \xi_1 \xi_1^H \right)^\dagger & 0 \\ \hline 0 & 0 \end{array} \right] \left[\begin{array}{c} U_1^H \\ U_2^H \end{array} \right] \\
&= U_1 \left(\Lambda_1 + \xi_1 \xi_1^H \right)^\dagger U_1^H \\
&= U_1 \left(\Lambda_1 + \xi_1 \xi_1^H \right)^{-1} U_1^H.
\end{aligned} \tag{88}
$$

The last equality in (88) comes from the fact that $\Lambda_1 + \xi_1 \xi_1^H$ is nonsingular, because Λ_1 is nonsingular. Using the matrix inversion lemma [3], [4], we have

$$\left(\Lambda_1 + \xi_1 \xi_1^H \right)^{-1} = \Lambda_1^{-1} - \frac{\Lambda_1^{-1} \xi_1 \xi_1^H \Lambda_1^{-1}}{1 + \xi_1^H \Lambda_1^{-1} \xi_1}. \tag{89}$$

On the other hand, from (81), (82), and P1), we have

$$U_1^H b_1 = U_1^H U_1 \xi_1 = \xi_1 \tag{90}$$
$$\xi_1^H \Lambda_1^{-1} \xi_1 = b_1^H U_1 \Lambda_1^{-1} U_1^H b_1 = b_1^H A^\dagger b_1. \tag{91}$$

Using (82) and (88)–(91), we obtain

$$
\begin{aligned}
Q_a^\dagger &= U_1 \left(\Lambda_1^{-1} - \frac{\Lambda_1^{-1} \xi_1 \xi_1^H \Lambda_1^{-1}}{1 + \xi_1^H \Lambda_1^{-1} \xi_1} \right) U_1^H \\
&= U_1 \Lambda_1^{-1} U_1^H - \frac{U_1 \Lambda_1^{-1} U_1^H U_1 \xi_1 \xi_1^H U_1^H U_1 \Lambda_1^{-1} U_1^H}{1 + b_1^H A^\dagger b_1} \\
&= A^\dagger - \frac{A^\dagger b_1 b_1^H A^\dagger}{1 + b_1^H A^\dagger b_1}.
\end{aligned} \tag{92}
$$

Equation (92) is equivalent to (4), because $Q = Q_a$ in the case when $b_2 = 0$.

Now, we consider the proof of the lemma in the general case where $b_2 \neq 0$. The proof is carried out in two stages. In the first stage, we get the pseudo-inverse of the matrix Q_b defined by

$$
\begin{aligned}
Q_b &:= A + b_1 b_1^H + b_2 b_2^H \\
&= \left(A + b_1 b_1^H \right) \oplus b_2 b_2^H \\
&= Q_a \oplus b_2 b_2^H
\end{aligned} \tag{93}
$$

where $\oplus$ denotes the direct sum of two matrices, that is, $A_1 + A_2 = A_1 \oplus A_2$ if $A_1^H A_2 = 0$ and $A_1 A_2^H = 0$. In the second stage, we obtain the pseudo-inverse of the matrix Q, which is represented as

$$Q = Q_b + [b_1, b_2] P [b_1, b_2]^H \tag{94}$$

where P is the 2×2 permutation matrix defined by

$$P := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \in R^{2 \times 2}. \tag{95}$$

The first stage of the proof is carried out as follows. Since it holds true that

$$x^\dagger = \frac{x^H}{\|x\|^2}, \quad \text{for a nonzero vector} \quad x \in C^m \tag{96}$$

we have

$$\left(b_2 b_2^H \right)^\dagger = \left(b_2^H \right)^\dagger b_2^\dagger = \left(b_2^\dagger \right)^H b_2^\dagger = \frac{b_2 b_2^H}{\|b_2\|^4} \tag{97}$$

because the matrix $b_2 b_2^H$ is represented as a rank decomposition of itself [see P3)]. Therefore, from (92), (93), and P2), we have

$$
\begin{aligned}
Q_b^\dagger &= Q_a^\dagger \oplus \left(b_2 b_2^H \right)^\dagger \\
&= Q_a^\dagger + \left(b_2^H \right)^\dagger b_2^\dagger \\
&= A^\dagger - \frac{A^\dagger b_1 b_1^H A^\dagger}{1 + b_1^H A^\dagger b_1} + \left(b_2^H \right)^\dagger b_2^\dagger.
\end{aligned} \tag{98}
$$

Equation (98) is equivalent to (5) in the case where $b_1 = 0$, because $Q = Q_b$ in this case [see (3) and (93)].

The second stage of the proof is carried out as follows: Since $\boldsymbol{b}_2 \in (\mathrm{Im}\,\boldsymbol{A})^\perp = (\mathrm{Im}\,\boldsymbol{U}_1)^\perp = \mathrm{Im}\,\boldsymbol{U}_2$, there exists $\boldsymbol{\xi}_2 \in \boldsymbol{C}^{(n-r)}$ such that

$$\boldsymbol{b}_2 = \boldsymbol{U}_2 \boldsymbol{\xi}_2. \tag{99}$$

From (82) and (99), we have

$$\boldsymbol{b} = \boldsymbol{b}_1 \oplus \boldsymbol{b}_2 = \boldsymbol{U}_1 \boldsymbol{\xi}_1 \oplus \boldsymbol{U}_2 \boldsymbol{\xi}_2 = [\boldsymbol{U}_1, \boldsymbol{U}_2]\begin{bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \end{bmatrix}. \tag{100}$$

From (3), (81), and (100), we obtain

$$\begin{aligned} \boldsymbol{Q} &= \boldsymbol{A} + \boldsymbol{b}\boldsymbol{b}^H \\ &= [\boldsymbol{U}_1, \boldsymbol{U}_2]\left\{ \begin{bmatrix} \boldsymbol{\Lambda}_1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \end{bmatrix}[\boldsymbol{\xi}_1^H, \boldsymbol{\xi}_2^H] \right\}\begin{bmatrix} \boldsymbol{U}_1^H \\ \boldsymbol{U}_2^H \end{bmatrix}. \end{aligned} \tag{101}$$

Using P1) and (101), we have

$$\boldsymbol{Q}^\dagger = [\boldsymbol{U}_1, \boldsymbol{U}_2]\tilde{\boldsymbol{Q}}^\dagger \begin{bmatrix} \boldsymbol{U}_1^H \\ \boldsymbol{U}_2^H \end{bmatrix} \tag{102}$$

where

$$\tilde{\boldsymbol{Q}} := \begin{bmatrix} \boldsymbol{\Lambda}_1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \end{bmatrix}[\boldsymbol{\xi}^H, \boldsymbol{\xi}_2^H]. \tag{103}$$

Then, $\tilde{\boldsymbol{Q}}$ becomes

$$\begin{aligned} \tilde{\boldsymbol{Q}} &= \left[\begin{array}{c|c} \boldsymbol{\Lambda}_1 + \boldsymbol{\xi}_1\boldsymbol{\xi}_1^H & 0 \\ \hline 0 & \boldsymbol{\xi}_2\boldsymbol{\xi}_2^H \end{array}\right] + \begin{bmatrix} \boldsymbol{\xi}_1 & 0 \\ 0 & \boldsymbol{\xi}_2 \end{bmatrix}\boldsymbol{P}\begin{bmatrix} \boldsymbol{\xi}_1^H & 0 \\ 0 & \boldsymbol{\xi}_2^H \end{bmatrix} \\ &= \tilde{\boldsymbol{Q}}_b + \begin{bmatrix} \boldsymbol{\xi}_1 & 0 \\ 0 & \boldsymbol{\xi}_2 \end{bmatrix}\boldsymbol{P}\begin{bmatrix} \boldsymbol{\xi}_1^H & 0 \\ 0 & \boldsymbol{\xi}_2^H \end{bmatrix} \end{aligned} \tag{104}$$

where

$$\tilde{\boldsymbol{Q}}_b := \left[\begin{array}{c|c} \boldsymbol{\Lambda}_1 + \boldsymbol{\xi}_1\boldsymbol{\xi}_1^H & 0 \\ \hline 0 & \boldsymbol{\xi}_2\boldsymbol{\xi}_2^H \end{array}\right]. \tag{105}$$

Since

$$\begin{aligned} \mathrm{rank}\,\tilde{\boldsymbol{Q}}_b &= \mathrm{rank}\left(\boldsymbol{\Lambda}_1 + \boldsymbol{\xi}_1\boldsymbol{\xi}_1^H\right) + \mathrm{rank}\left(\boldsymbol{\xi}_2\boldsymbol{\xi}_2^H\right) \\ &= r + 1 \end{aligned} \tag{106}$$

there exists a unitary matrix $\boldsymbol{V}$ and a diagonal matrix $\tilde{\boldsymbol{\Lambda}}$ such that

$$\tilde{\boldsymbol{Q}}_b = \boldsymbol{V}\tilde{\boldsymbol{\Lambda}}\boldsymbol{V}^H = [\boldsymbol{V}_1, \boldsymbol{V}_2]\begin{bmatrix} \tilde{\boldsymbol{\Lambda}}_1 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \boldsymbol{V}_1^H \\ \boldsymbol{V}_2^H \end{bmatrix} = \boldsymbol{V}_1\tilde{\boldsymbol{\Lambda}}_1\boldsymbol{V}_1^H \tag{107}$$

where $\boldsymbol{V}_1 \in \boldsymbol{C}^{n\times(r+1)}$, $\boldsymbol{V}_2 \in \boldsymbol{C}^{n\times(n-r-1)}$ and $\tilde{\boldsymbol{\Lambda}}_1$ is a nonsingular diagonal matrix belonging to $\boldsymbol{R}^{(r+1)\times(r+1)}$. Since

$$\begin{bmatrix} \boldsymbol{\xi}_1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \boldsymbol{\xi}_2 \end{bmatrix} \in \mathrm{Im}\,\tilde{\boldsymbol{Q}}_b = \mathrm{Im}\,\boldsymbol{V}_1 \tag{108}$$

there exist $\boldsymbol{\eta}_1 \in \boldsymbol{C}^{r+1}$ and $\boldsymbol{\eta}_2 \in \boldsymbol{C}^{r+1}$ such that

$$\begin{bmatrix} \boldsymbol{\xi}_1 & 0 \\ 0 & \boldsymbol{\xi}_2 \end{bmatrix} = \boldsymbol{V}_1[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]. \tag{109}$$

Therefore, from (104), (107), and (109), we have

$$\tilde{\boldsymbol{Q}} = \boldsymbol{V}_1\left\{\tilde{\boldsymbol{\Lambda}}_1 + [\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]\boldsymbol{P}[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]^H\right\}\boldsymbol{V}_1^H. \tag{110}$$

Put

$$\boldsymbol{Q}_c := \tilde{\boldsymbol{\Lambda}}_1 + [\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]\boldsymbol{P}[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]^H \in \boldsymbol{C}^{(r+1)\times(r+1)}. \tag{111}$$

Since $\boldsymbol{b}_2 \neq 0$, we have $\boldsymbol{b} \notin \mathrm{Im}\,\boldsymbol{A}$. Therefore

$$\mathrm{rank}\,\boldsymbol{Q} = \mathrm{rank}(\boldsymbol{A} + \boldsymbol{b}\boldsymbol{b}^H) = \mathrm{rank}\,\boldsymbol{A} + 1 = r + 1. \tag{112}$$

This means that, from (101) and (103), we have

$$\mathrm{rank}\,\tilde{\boldsymbol{Q}} = \mathrm{rank}(\boldsymbol{U}^H\boldsymbol{Q}\boldsymbol{U}) = \mathrm{rank}\,\boldsymbol{Q} = r + 1 \tag{113}$$

which implies from (110) and (111) that

$$\mathrm{rank}\,\boldsymbol{Q}_c = \mathrm{rank}\,\tilde{\boldsymbol{Q}} = r + 1. \tag{114}$$

Therefore, $\boldsymbol{Q}_c$ is nonsingular. By a generalized matrix inversion lemma in [4, p. 656], we have

$$\boldsymbol{Q}_c^{-1} = \tilde{\boldsymbol{\Lambda}}_1^{-1} - \tilde{\boldsymbol{\Lambda}}_1^{-1}[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]\boldsymbol{Q}_d^{-1}[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]^H\tilde{\boldsymbol{\Lambda}}_1^{-1} \tag{115}$$

where

$$\boldsymbol{Q}_d := \boldsymbol{P}^{-1} + [\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]^H\tilde{\boldsymbol{\Lambda}}_1^{-1}[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2] \in \boldsymbol{C}^{2\times2}. \tag{116}$$

Here we note that it can be shown that $\boldsymbol{Q}_d$ is also nonsingular if $\boldsymbol{Q}_c$ is nonsingular. Since $\boldsymbol{P}^{-1} = \boldsymbol{P}$, $\boldsymbol{Q}_d$ becomes

$$\begin{aligned} \boldsymbol{Q}_d &= \boldsymbol{P} + [\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]^H\tilde{\boldsymbol{\Lambda}}_1^{-1}[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2] \\ &= \left[\begin{array}{c|c} \boldsymbol{\eta}_1^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_1 & 1 + \boldsymbol{\eta}_1^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_2 \\ \hline 1 + \boldsymbol{\eta}_2^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_1 & \boldsymbol{\eta}_2^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_2 \end{array}\right] \in \boldsymbol{C}^{2\times2} \end{aligned} \tag{117}$$

which gives

$$\boldsymbol{Q}_d^{-1} = \frac{1}{l}\left[\begin{array}{c|c} -\boldsymbol{\eta}_2^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_2 & 1 + \boldsymbol{\eta}_1^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_2 \\ \hline 1 + \boldsymbol{\eta}_2^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_1 & -\boldsymbol{\eta}_1^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_1 \end{array}\right] \tag{118}$$

where

$$\begin{aligned} l &:= -\det\boldsymbol{Q}_d \\ &= \left|1 + \boldsymbol{\eta}_1^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_2\right|^2 - \boldsymbol{\eta}_2^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_2\boldsymbol{\eta}_1^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{\eta}_1 \\ &\neq 0. \end{aligned} \tag{119}$$

The proof of the nonsingularity of matrix $\boldsymbol{Q}_d$ is shown in the last part of this Appendix.

From (107)–(119) along with P3), we have

$$\begin{aligned} \tilde{\boldsymbol{Q}}^\dagger &= \left(\boldsymbol{V}_1\boldsymbol{Q}_c\boldsymbol{V}_1^H\right)^\dagger \\ &= \left(\boldsymbol{V}_1^H\right)^\dagger(\boldsymbol{V}_1\boldsymbol{Q}_c)^\dagger \\ &= \boldsymbol{V}_1\boldsymbol{Q}_c^{-1}\boldsymbol{V}_1^H \\ &= \boldsymbol{V}_1\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{V}_1^H - \boldsymbol{V}_1\tilde{\boldsymbol{\Lambda}}_1^{-1}[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]\boldsymbol{Q}_d^{-1}[\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]^H\tilde{\boldsymbol{\Lambda}}_1^{-1}\boldsymbol{V}_1^H \\ &= \tilde{\boldsymbol{Q}}_b^\dagger - \tilde{\boldsymbol{Q}}_b^\dagger\begin{bmatrix} \boldsymbol{\xi}_1 & 0 \\ 0 & \boldsymbol{\xi}_2 \end{bmatrix}\boldsymbol{Q}_d^{-1}\begin{bmatrix} \boldsymbol{\xi}_1^H & 0 \\ 0 & \boldsymbol{\xi}_2^H \end{bmatrix}\tilde{\boldsymbol{Q}}_b^\dagger \end{aligned} \tag{120}$$

where we have used from (107) and (109)

$$\tilde{Q}_b^\dagger = V_1 \tilde{\Lambda}_1^{-1} V_1^H \tag{121}$$

$$V_1 \tilde{\Lambda}_1^{-1} [\eta_1, \eta_2] = V_1 \tilde{\Lambda}_1^{-1} V_1^H V_1 [\eta_1, \eta_2]$$
$$= \tilde{Q}_b^\dagger \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix}. \tag{122}$$

Using (102) and (120) along with P3), we obtain

$$Q^\dagger = U \tilde{Q}^\dagger U^H$$
$$= U \tilde{Q}_b^\dagger U^H - U \tilde{Q}_b^\dagger \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix} Q_d^{-1} \begin{bmatrix} \xi_1^H & 0 \\ 0 & \xi_2^H \end{bmatrix} \tilde{Q}_b^\dagger U^H$$
$$= Q_b^\dagger - Q_b^\dagger [b_1, b_2] Q_d^{-1} [b_1, b_2]^H Q_b^\dagger \tag{123}$$

where we have used

$$U \tilde{Q}_b^\dagger U^H = Q_b^\dagger \tag{124}$$

$$U \tilde{Q}_b^\dagger \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix} = Q_b^\dagger [b_1, b_2]. \tag{125}$$

Equation (124) holds true because P1) holds true and

$$U \tilde{Q}_b U^H = [U_1, U_2] \left[\begin{array}{c|c} \Lambda_1 + \xi_1 \xi_1^H & 0 \\ \hline 0 & \xi_2 \xi_2^H \end{array} \right] \begin{bmatrix} U_1^H \\ U_2^H \end{bmatrix}$$
$$= U_1 \left(\Lambda_1 + \xi_1 \xi_1^H \right) U_1^H + U_2 \xi_2 \xi_2^H U_2^H$$
$$= A + b_1 b_1^H + b_2 b_2^H$$
$$= Q_b. \tag{126}$$

Equation (125) also holds true because

$$U \tilde{Q}_b^\dagger \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix} = U \tilde{Q}_b^\dagger U^H U \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix} = Q_b^\dagger [b_1, b_2]. \tag{127}$$

On the other hand, from (107) and (109), we have

$$[\eta_1, \eta_2] = V_1^H \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix}. \tag{128}$$

This gives from (121)

$$\eta_1^H \tilde{\Lambda}_1^{-1} \eta_2 = \begin{bmatrix} \xi_1^H, 0 \end{bmatrix} V_1 \tilde{\Lambda}_1^{-1} V_1^H \begin{bmatrix} 0 \\ \xi_2 \end{bmatrix}$$
$$= \begin{bmatrix} \xi_1^H, 0 \end{bmatrix} \tilde{Q}_b^\dagger \begin{bmatrix} 0 \\ \xi_2 \end{bmatrix}$$
$$= \begin{bmatrix} \xi_1^H \tilde{Q}_b^\dagger, 0 \end{bmatrix} \begin{bmatrix} 0 \\ \xi_2 \end{bmatrix}$$
$$= 0. \tag{129}$$

Similarly, we obtain

$$\eta_2^H \tilde{\Lambda}_1^{-1} \eta_1 = \begin{bmatrix} 0, \xi_2^H \end{bmatrix} V_1 \tilde{\Lambda}_1^{-1} V_1^H \begin{bmatrix} \xi_1 \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} 0, \xi_2^H \end{bmatrix} \tilde{Q}_b^\dagger \begin{bmatrix} \xi_1 \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} 0, \xi_2^H \tilde{Q}_b^\dagger \end{bmatrix} \begin{bmatrix} \xi_2 \\ 0 \end{bmatrix}$$
$$= 0. \tag{130}$$

Furthermore

$$\eta_1^H \tilde{\Lambda}_1^{-1} \eta_1 = \begin{bmatrix} \xi_1^H, 0 \end{bmatrix} V_1 \tilde{\Lambda}_1^{-1} V_1^H \begin{bmatrix} \xi_1 \\ 0 \end{bmatrix} = \begin{bmatrix} \xi_1^H, 0 \end{bmatrix} \tilde{Q}_b^\dagger \begin{bmatrix} \xi_1 \\ 0 \end{bmatrix}. \tag{131}$$

Using (100), we have

$$\begin{bmatrix} \xi_1 \\ 0 \end{bmatrix} = U^H b_1. \tag{132}$$

It follows from (124), (131), and (132) that

$$\eta_1^H \tilde{\Lambda}_1^{-1} \eta_1 = \begin{bmatrix} \xi_1^H, 0 \end{bmatrix} \tilde{Q}_b^\dagger \begin{bmatrix} \xi_1 \\ 0 \end{bmatrix} = b_1^H U \tilde{Q}_b^\dagger U^H b_1 = b_1^H Q_b^\dagger b_1. \tag{133}$$

Similarly, we obtain

$$\eta_2^H \tilde{\Lambda}_1^{-1} \eta_2 = \begin{bmatrix} 0, \xi_2^H \end{bmatrix} V_1 \tilde{\Lambda}_1^{-1} V_1^H \begin{bmatrix} 0 \\ \xi_2 \end{bmatrix}$$
$$= \begin{bmatrix} 0, \xi_2^H \end{bmatrix} \tilde{Q}_b^\dagger \begin{bmatrix} 0 \\ \xi_2 \end{bmatrix}$$
$$= b_2^H U \tilde{Q}_b^\dagger U^H b_2$$
$$= b_2^H Q_b^\dagger b_2. \tag{134}$$

Using (129), (130), (133), and (134), we have from (117)–(119)

$$Q_d^{-1} = \left(P + [b_1, b_2]^H Q_b^\dagger [b_1, b_2] \right)^{-1}$$
$$= \frac{1}{1 - b_2^H Q_b^\dagger b_2 b_1^H Q_b^\dagger b_1}$$
$$\times \left[\begin{array}{c|c} -b_2^H Q_b^\dagger b_2 & 1 \\ \hline 1 & -b_1^H Q_b^\dagger b_1 \end{array} \right]. \tag{135}$$

This completes the proof of Lemma 1.

The proof of the nonsingularity of matrix Q_d is as follows. From (77) and (79)–(81), we have

$$A^\dagger = U_1 \Lambda_1^{-1} U_1^H \tag{136}$$

$$U_1^H U_2 = 0 \tag{137}$$

$$U_2^H U_1 = 0. \tag{138}$$

From (82), (99), and (136)–(138), we have

$$A^\dagger b_2 = U_1 \Lambda_1^{-1} U_1^H U_2 \xi_2 = 0 \tag{139}$$

$$b_2^H A^\dagger = \xi_2^H U_2^H U_1 \Lambda_1^{-1} U_1^H = 0 \tag{140}$$

$$b_1^H b_2 = \xi_1^H U_1^H U_2 \xi_2 = 0 \tag{141}$$

$$b_2^H b_1 = \xi_2^H U_2^H U_1 \xi_1 = 0. \tag{142}$$

Therefore, from (98), (139), and (140), we have from (134)

$$b_2^H Q_b^\dagger b_2 = b_2^H \left[A^\dagger - \frac{A^\dagger b_1 b_1^H A^\dagger}{1 + b_1^H A^\dagger b_1} + \left(b_2^H \right)^\dagger b_2^\dagger \right] b_2$$
$$= b_2^H A^\dagger b_2 - \frac{b_2^H A^\dagger b_1 b_1^H A^\dagger b_2}{1 + b_1^H A^\dagger b_1} + b_2^H \left(b_2^H \right)^\dagger b_2^\dagger b_2$$
$$= 0 - 0 + b_2^H \left(b_2^H \right)^\dagger b_2^\dagger b_2$$
$$= 1. \tag{143}$$

Moreover, from (98), (141), and (142), we have from (133)

$$
\begin{aligned}
\boldsymbol{b}_1^H \boldsymbol{Q}_b^\dagger \boldsymbol{b}_1 &= \boldsymbol{b}_1^H \left[\boldsymbol{A}^\dagger - \frac{\boldsymbol{A}^\dagger \boldsymbol{b}_1 \boldsymbol{b}_1^H \boldsymbol{A}^\dagger}{1 + \boldsymbol{b}_1^H \boldsymbol{A}^\dagger \boldsymbol{b}_1} + \left(\boldsymbol{b}_2^H \right)^\dagger \boldsymbol{b}_2^\dagger \right] \boldsymbol{b}_1 \\
&= \boldsymbol{b}_1^H \boldsymbol{A}^\dagger \boldsymbol{b}_1 - \frac{\boldsymbol{b}_1^H \boldsymbol{A}^\dagger \boldsymbol{b}_1 \boldsymbol{b}_1^H \boldsymbol{A}^\dagger \boldsymbol{b}_1}{1 + \boldsymbol{b}_1^H \boldsymbol{A}^\dagger \boldsymbol{b}_1} + \frac{\boldsymbol{b}_1^H \boldsymbol{b}_2 \boldsymbol{b}_2^H \boldsymbol{b}_1}{\left(\boldsymbol{b}_2^H \boldsymbol{b}_2 \right)^2} \\
&= g - \frac{g^2}{1+g} + 0 \\
&= \frac{g}{1+g} > 0, \quad 0 < \boldsymbol{b}_1^H \boldsymbol{Q}_b^\dagger \boldsymbol{b}_1 < 1
\end{aligned}
\tag{144}
$$

where $g := \boldsymbol{b}_1^H \boldsymbol{A}^\dagger \boldsymbol{b}_1 > 0$.

Using (143) and (144), l defined by (119) becomes

$$
l = 1 - \boldsymbol{b}_2^H \boldsymbol{Q}_b^\dagger \boldsymbol{b}_2 \boldsymbol{b}_1^H \boldsymbol{Q}_b^\dagger \boldsymbol{b}_1 = 1 - \left(\frac{g}{1+g} \right) = \frac{1}{1+g} > 0.
\tag{145}
$$

Therefore, matrix $\boldsymbol{Q}_d$ is nonsingular. This completes the proof.

APPENDIX B
PROOF OF THEOREM 1

First, we consider case 1). If $\boldsymbol{b}_2 = 0$, then $\boldsymbol{Q}_b^\dagger$ and $\boldsymbol{Q}_d^{-1}$ are represented from (7) and (8), respectively, as follows:

$$
\begin{aligned}
\boldsymbol{Q}_b^\dagger &= \left(\boldsymbol{A} + \boldsymbol{b}_1 \boldsymbol{b}_1^H \right)^\dagger \\
&= \boldsymbol{A}^\dagger - \frac{\boldsymbol{A}^\dagger \boldsymbol{b}_1 \boldsymbol{b}_1^H \boldsymbol{A}^\dagger}{1 + \boldsymbol{b}_1^H \boldsymbol{A}^\dagger \boldsymbol{b}_1}
\end{aligned}
\tag{146}
$$

$$
\boldsymbol{Q}_d^{-1} = \left[\begin{array}{c|c} -0 & 1 \\ \hline 1 & -\boldsymbol{b}_1^H \boldsymbol{Q}_b^\dagger \boldsymbol{b}_1 \end{array} \right].
\tag{147}
$$

Then, from (6), (146), and (147), we have

$$
\begin{aligned}
\boldsymbol{Q}^\dagger &= \boldsymbol{Q}_b^\dagger - \boldsymbol{Q}_b^\dagger [\boldsymbol{b}_1, 0] \left[\begin{array}{c|c} 0 & 1 \\ \hline 1 & -\boldsymbol{b}_1^H \boldsymbol{Q}_b^\dagger \boldsymbol{b}_1 \end{array} \right] [\boldsymbol{b}_1, 0]^H \boldsymbol{Q}_b^\dagger \\
&= \boldsymbol{Q}_b^\dagger \\
&= \boldsymbol{A}^\dagger - \frac{\boldsymbol{A}^\dagger \boldsymbol{b}_1 \boldsymbol{b}_1^H \boldsymbol{A}^\dagger}{1 + \boldsymbol{b}_1^H \boldsymbol{A}^\dagger \boldsymbol{b}_1}
\end{aligned}
\tag{148}
$$

which is identical to (4).

Second, we consider case 2). If $\boldsymbol{b}_2 \neq 0$ and $\boldsymbol{b}_1 = 0$, then $\boldsymbol{Q}_b^\dagger$ and $\boldsymbol{Q}_d^{-1}$ are represented from (7) and (8), respectively, as follows:

$$
\begin{aligned}
\boldsymbol{Q}_b^\dagger &= \left(\boldsymbol{A} + \boldsymbol{b}_2 \boldsymbol{b}_2^H \right)^\dagger \\
&= \boldsymbol{A}^\dagger + \left(\boldsymbol{b}_2^H \right)^\dagger \boldsymbol{b}_2^\dagger
\end{aligned}
\tag{149}
$$

$$
\boldsymbol{Q}_d^{-1} = \left[\begin{array}{c|c} -\boldsymbol{b}_2^H \boldsymbol{Q}_b^\dagger \boldsymbol{b}_2 & 1 \\ \hline 1 & 0 \end{array} \right].
\tag{150}
$$

Then, from (6), (149), and (150), we have

$$
\begin{aligned}
\boldsymbol{Q}^\dagger &= \boldsymbol{Q}_b^\dagger - \boldsymbol{Q}_b^\dagger [0, \boldsymbol{b}_2] \left[\begin{array}{c|c} -\boldsymbol{b}_2^H \boldsymbol{Q}_b^\dagger \boldsymbol{b}_2 & 1 \\ \hline 1 & 0 \end{array} \right] [0, \boldsymbol{b}_2]^H \boldsymbol{Q}_b^\dagger \\
&= \boldsymbol{Q}_b^\dagger \\
&= \boldsymbol{A}^\dagger + \left(\boldsymbol{b}_2^H \right)^\dagger \boldsymbol{b}_2^\dagger
\end{aligned}
\tag{151}
$$

which is identical to (5). This completes the proof.

APPENDIX C
PROOF OF THEOREM 2

Substituting (48) and $\boldsymbol{P}(t) = \tilde{\boldsymbol{R}}^\dagger(t)$ into (46), we have

$$
\tilde{\boldsymbol{w}}_i(t) = \boldsymbol{P}(t) \left\{ (1 - \alpha_t) \tilde{\boldsymbol{d}}_i(t-1) + \alpha_t \tilde{\boldsymbol{y}}^*(t) \tilde{z}_i(t) \right\}.
\tag{152}
$$

On the other hand, from (44)–(47), we obtain the following relation.

$$
\begin{aligned}
\tilde{\boldsymbol{d}}_i(t-1) \\
= \tilde{\boldsymbol{R}}(t-1) \tilde{\boldsymbol{w}}_i(t-1) \\
= \frac{1}{1 - \alpha_t} \left\{ \tilde{\boldsymbol{R}}(t) - \alpha_t \tilde{\boldsymbol{y}}^*(t) \tilde{\boldsymbol{y}}^T(t) \right\} \tilde{\boldsymbol{w}}_i(t-1).
\end{aligned}
\tag{153}
$$

Substituting (153) into (152), the recursion formula for time-updating of the tap vector $\tilde{\boldsymbol{w}}_i(t)$ is obtained as follows:

$$
\begin{aligned}
\tilde{\boldsymbol{w}}_i(t) &= \boldsymbol{P}(t) \left[\left\{ \tilde{\boldsymbol{R}}(t) - \alpha_t \tilde{\boldsymbol{y}}^*(t) \tilde{\boldsymbol{y}}^T(t) \right\} \tilde{\boldsymbol{w}}_i(t-1) \right. \\
&\quad \left. + \alpha_t \tilde{\boldsymbol{y}}^*(t) \tilde{z}_i(t) \right] \\
&= \boldsymbol{P}(t) \tilde{\boldsymbol{R}}(t) \tilde{\boldsymbol{w}}_i(t-1) \\
&\quad + \alpha_t \boldsymbol{P}(t) \tilde{\boldsymbol{y}}^*(t) \left[\tilde{z}_i(t) - \hat{\boldsymbol{y}}^T(t) \tilde{\boldsymbol{w}}_i(t-1) \right] \\
&= \boldsymbol{P}(t) \tilde{\boldsymbol{R}}(t) \tilde{\boldsymbol{w}}_i(t-1) \\
&\quad + \boldsymbol{k}(t) \left[\tilde{z}_i(t) - \hat{\boldsymbol{y}}^T(t) \tilde{\boldsymbol{w}}_i(t-1) \right]
\end{aligned}
\tag{154}
$$

where

$$
\boldsymbol{k}(t) := \alpha_t \boldsymbol{P}(t) \tilde{\boldsymbol{y}}^*(t).
\tag{155}
$$

This completes the proof.

APPENDIX D
PROOF OF RELATION (69)

As in Lemma 1, the vector linear space $\boldsymbol{C}^{mL}$ can be uniquely decomposed as

$$
\boldsymbol{C}^{mL} = \operatorname{Im} \tilde{\boldsymbol{R}} \oplus (\operatorname{Im} \tilde{\boldsymbol{R}})^\perp.
\tag{156}
$$

Since $\tilde{\boldsymbol{y}}^*(t) \in \boldsymbol{C}^{mL}$, it can be decomposed as

$$
\tilde{\boldsymbol{y}}^*(t) = \tilde{\boldsymbol{y}}_1(t) \oplus \tilde{\boldsymbol{y}}_2(t),
\tag{157}
$$

where

$$
\tilde{\boldsymbol{y}}_1(t) \in \operatorname{Im} \tilde{\boldsymbol{R}} \quad \text{and} \quad \tilde{\boldsymbol{y}}_2(t) \in (\operatorname{Im} \tilde{\boldsymbol{R}})^\perp.
\tag{158}
$$

Since $\tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger$ is the orthogonal projection of $\boldsymbol{C}^{mL}$ onto the subspace $\operatorname{Im} \tilde{\boldsymbol{R}}$ and $\boldsymbol{I} - \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger$ is the orthogonal projection of $\boldsymbol{C}^{mL}$ onto the subspace $(\operatorname{Im} \tilde{\boldsymbol{R}})^\perp$, $\tilde{\boldsymbol{y}}_1(t)$ and $\tilde{\boldsymbol{y}}_2(t)$ are respectively represented as

$$
\tilde{\boldsymbol{y}}_1(t) = \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger \tilde{\boldsymbol{y}}^*(t)
\tag{159}
$$

$$
\tilde{\boldsymbol{y}}_2(t) = (\boldsymbol{I} - \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger) \tilde{\boldsymbol{y}}^*(t).
\tag{160}
$$

It follows from (42) and (160) that

$$
\begin{aligned}
E \left[\tilde{\boldsymbol{y}}_2(t) \tilde{\boldsymbol{y}}_2(t)^H \right] &= E \left[(\boldsymbol{I} - \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger) \tilde{\boldsymbol{y}}^*(t) \tilde{\boldsymbol{y}}^*(t)^H (\boldsymbol{I} - \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger)^H \right] \\
&= (\boldsymbol{I} - \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger) E \left[\tilde{\boldsymbol{y}}^*(t) \tilde{\boldsymbol{y}}^*(t)^H \right] (\boldsymbol{I} - \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger)^H \\
&= (\boldsymbol{I} - \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger) \tilde{\boldsymbol{R}} (\boldsymbol{I} - \tilde{\boldsymbol{R}} \tilde{\boldsymbol{R}}^\dagger)^H \\
&= 0
\end{aligned}
\tag{161}
$$

which means

$$E\left[\|\tilde{\boldsymbol{y}}_2(t)\|^2\right] = 0 \qquad (162)$$

because

$$
\begin{aligned}
E\left[\|\tilde{\boldsymbol{y}}_2(t)\|^2\right] &= E\left[\operatorname{tr}\left\{\tilde{\boldsymbol{y}}_2(t)\tilde{\boldsymbol{y}}_2(t)^H\right\}\right] \\
&= \operatorname{tr}\left\{E\left[\tilde{\boldsymbol{y}}_2(t)\tilde{\boldsymbol{y}}_2(t)^H\right]\right\} \\
&= 0 \qquad\qquad\qquad (163)
\end{aligned}
$$

where $\operatorname{tr}\{\boldsymbol{A}\}$ denotes the trace of a matrix $\boldsymbol{A}$.

Since $\|\tilde{\boldsymbol{y}}_2(t)\|^2 \geq 0$, (162) means that $\|\tilde{\boldsymbol{y}}_2(t)\|$ is a.s. zero, i.e.,

$$\|\tilde{\boldsymbol{y}}_2(t)\| = 0, \quad \text{a.s.} \qquad (164)$$

which implies

$$\tilde{\boldsymbol{y}}_2(t) = 0, \quad \text{a.s.} \qquad (165)$$

Therefore, we obtain from (157) and (158)

$$\tilde{\boldsymbol{y}}^*(t) = \tilde{\boldsymbol{y}}_1(t) \in \operatorname{Im} \tilde{\boldsymbol{R}}, \quad \text{a.s.} \qquad (166)$$

This completes the proof.

ACKNOWLEDGMENT

REFERENCES

[1] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ: Prentice-Hall, 1987.

[2] J. M. Mendel, *Lessons in Estimation Theory for Signal Processing, Communications, and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

[3] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 1996.

[4] T. Kailath, *Linear Systems*. New York: Prentice-Hall, 1980.

[5] R. Penrose, "A generalized inverse for matrices," *Proc. Cambridge Philops. Soc.*, vol. 51, pp. 406–413, 1955.

[6] R. Penrose, "On best approximate solutions of linear matrix equations," *Proc. Cambridge Philops. Soc.*, vol. 52, pp. 17–19, 1956.

[7] S. L. Campbell and C. D. Meyer, Jr., *Generalized Inverses of Linear Transformations*. New York: Dover, 1979.

[8] Y. Inouye and K. Tanebe, "Super-exponential algorithms for multichannel blind deconvolution," *IEEE Trans. Signal Process.*, vol. 48, no. 3, pp. 881–888, Mar. 2000.

[9] K. Kohno, Y. Inouye, M. Kawamoto, and T. Okamoto, "Adaptive super-exponential algorithms for blind deconvolution of MIMO systems," in *Proc. ISCAS*, Vancouver, ON, Canada, May 23–26, 2004, vol. 5, pp. 680–683.

[10] K. Kohno, Y. Inouye, M. Kawamoto, and T. Okamoto, "An adaptive super-exponential deflation algorithm for blind deconvolution of MIMO systems using the QR-factorization of matrix algebra," in *Proc. MWSCAS*, Hiroshima, Japan, Jul. 25–28, 2004, vol. 3, pp. 419–422.

[11] K. Kohno, Y. Inouye, and M. Kawamoto, "An adaptive super-exponential deflation algorithm for blind deconvolution of MIMO systems using the matrix pseudo-inversion lemma," in *Proc. ISCAS*, Kobe, Japan, May 23–26, 2005, pp. 5870–5873.

[12] P. Lancaster and M. Tismenetsky, *The Theory of Matrix*. San Diego, CA: Academic, 1985.

[13] O. Shalvi and E. Weinstein, "Super-exponential methods for blind deconvolution," *IEEE Trans. Inf. Theory*, vol. 39, no. 2, pp. 504–519, Mar. 1993.

[14] K. Kohno, Y. Inouye, and M. Kawamoto, "Adaptive super-exponential methods for blind deconvolution of MIMO systems using the matrix pseudo-inversion lemma," *IEEE Trans. Circuits Syst. I, Reg. Papers*, submitted for publication.

[15] H. Ogawa, "An operator pseudo-inversion lemma," *SIAM J. Appl. Math.*, vol. 48, no. 6, pp. 1527–1531, Dec. 1988.

[16] G. E. Bottomley and S. T. Alexander, "A novel approach for stabilizing recursive least squares filters," *IEEE Trans. Inf. Theory*, vol. 39, no. 8, pp. 1770–1779, Aug. 1993.

[17] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1989.

[18] K. Kohno, Y. Inouye, and M. Kawamoto, "A matrix pseudo-inversion lemma and its application to block-based adaptive blind deconvolution for MIMO systems," in *Proc. ISCAS*, New Orleans, LA, May 27–30, 2007, pp. 3490–3493.

[19] K. Kohno, Y. Inouye, and M. Kawamoto, "A block-based adaptive super-exponential deflation algorithm for blind deconvolution of MIMO systems using the matrix pseudo-inversion lemma," *Signal Process.*, to be published.

Kiyotaka Kohno (S'04–M'07) received the B.Eng. and M.Eng. degrees in electrical electronics and systems engineering from Nagaoka University of Technology, Nagaoka, Japan, in 1981 and 1983, respectively, and the D.Eng. degree in electronic functions and systems engineering from Shimane University, Matsue, Japan, in 2007.

From April 1983 to March 1990, he was a Researcher with Matsushita Electric Industrial Company, Ltd. Since April 1990, he has been with Yonago National College of Technology, Yonago, Japan, where he was an Assistant Professor from April 1990 to March 1994, an Associate Professor from April 1994 to March 2007, and is currently a Professor of Electronic Control Engineering. His current research interests are in blind signal processing, blind source separation, communication systems, and image processing.

Dr. Kohno is a member of the European Association for Signal processing (EURASIP), the Institute of Electronics, Information and Communication Engineers (IEICE) in Japan, and the Information Processing Society of Japan (IPSJ).

Yujiro Inouye received the B.Eng., M.Eng., and D.Eng. degrees in control engineering from Osaka University, Osaka, Japan, in 1966, 1968, and 1971, respectively.

From April 1971 to March 1997, he was with the Faculty of Engineering Science, Osaka University. Since April 1997, he has been with the Faculty of Science and Engineering, Shimane University, Matsue, Japan, where he is currently a Professor of Electronic and Control Systems Engineering. His major research interests are in blind signal processing, blind source separation, neural networks, and communications.

Dr. Inouye is a member of the European Association for Signal processing (EURASIP), the Institute of Electronics, Information and Communication Engineers in Japan, the Society of Instrument and Control Engineers in Japan, and the Institute of Systems, Control, and Information Engineers in Japan.
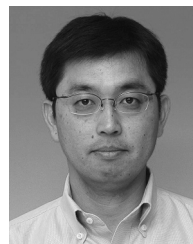
Mitsuru Kawamoto received the B.Eng., M.Eng., and D.Eng. degrees in control engineering from Kyushu Institute of Technology, Kyushu, Japan, in 1992, 1994, and 1997, respectively.

He was a Research Scientist with the Bio-Mimetic Control Research Center, Institute of Physical and Chemical Research (RIKEN), from April 1997 to December 1999. From January 2000 to March 2002, he was an Assistant Professor and then an Associate Professor, from April 2002 to March 2005, with the Department of Electronic and Control Systems Engineering, Shimane University, Matsue, Japan. From April 2004 to March 2005, he was with the Department of Electrical Engineering and Electronics, The University of Liverpool, Liverpool, U.K., as a Visiting Researcher. Since March 2005, he has been a Research Scientist with the National Institute of Advanced Industrial Science and Technology (AIST), Ibaraki, Japan. His current research interests include blind signal processing, higher order statistics, sound signal processing, and communication systems.

Dr. Kawamoto is a member of the Society of Instrument and Control Engineers, the Institute of Electronics, Information and Communication Engineers, and the Acoustical Society of Japan.