Decrease in Computational Load and Increase in Accuracy for Filtering of Random Signals

Pablo Soto-Quiros *, Anatoli Torokhti *, Phil Howlett

Posted Date: 21 March 2025

doi: 10.20944/preprints202503.1571.v1

Keywords: large covariance matrices; least squares linear estimate; singular value decomposition; error minimization

 doi:10.20944/preprints202503.1571.v1

Article

Decrease in Computational Load and Increase in Accuracy for Filtering of Random Signals

Phil Howlett [1], Anatoli Torokhti [1] and Pablo Soto-Quiros [2]*

[1] STEM discipline, University of South Australia, Mawson Lakes, SA, Australia
[2] Escuela de Matemática, Instituto Tecnológico de Costa Rica, Cartago 30101, Costa Rica
* Correspondence: jusoto@tec.ac.cr

Abstract: This paper describes methods for optimal filtering of random signals that involve large matrices. We develop a procedure that allows us to significantly decrease the computational load needed to numerically realize the associated filter and increase the associated accuracy. The procedure is based on the reduction of a large covariance matrix to a collection of smaller matrices. It is done in such a way that the filter equation with large matrices is equivalently represented by a set of equations with smaller matrices. The filter F_p we develop is represented by $F_p(\mathbf{v}_1,\dots\mathbf{v}_p)=\sum_{j=1}^{p} M_j\mathbf{v}_j$ and minimizes the associated error over all matrices $M_1,\dots,M_p$. As a result, the proposed optimal filter has two degrees of freedom to increase the associated accuracy. They are associated, first, with the optimal determination of matrices $M_1,\dots,M_p$ and second, with the increase in the number p of components in the filter F_p. The error analysis and results of numerical simulations are provided.

Keywords: large covariance matrices; least squares linear estimate; singular value decomposition; error minimization

1. Introduction

1.1. Preliminaries

Although there is an extensive literature on studying methods of filtering of random signals, the problem we consider seems to be different from the known ones. The objective of this paper is to formulate and solve this problem under scenarios which relate to a reduction of computation complexity, for the case of large covariance matrices, and an increase in the filtering accuracy.

Let Ω be a set of outcomes in probability space (Ω,Σ,μ) for which Σ is a σ–algebra of measurable subsets of Ω and $\mu:\Sigma\to[0,1]$ is an associated probability measure. Let $\mathbf{x}\in L^2(\Omega,\mathbb{R}^m)$ be a signal of interest and $\mathbf{y}\in L^2(\Omega,\mathbb{R}^n)$ be an observable signal.. Here $L^2(\Omega,\mathbb{R}^m)$ is the space of square-integrable functions defined on Ω with values in $\mathbb{R}^m$, i.e., such that $\int_\Omega\sum_{i=1}^{m}[\mathbf{x}_{(i)}(\omega)]^2d\mu(\omega)<\infty$.

Further, let us write $\mathbf{x}=[\mathbf{x}_{(1)},\dots,\mathbf{x}_{(m)}]^T$ where $\mathbf{x}_{(i)}\in L^2(\Omega,\mathbb{R})$, for $i=1,\dots,m$. We assume without loss of generality that all random vectors are of the zero mean. Let us denote

$$\|\mathbf{x}\|_\Omega^2=\int_\Omega\sum_{i=1}^{m}[\mathbf{x}_{(i)}(\omega)]^2d\mu(\omega),\ E_{xy}=\int_\Omega\mathbf{x}(\omega)[\mathbf{y}(\omega)]^Td\mu(\omega), \tag{1}$$

where E_{xy} is the covariance matrix formed from $\mathbf{x}$ and $\mathbf{y}$.

Let filter F be represented by $F(\mathbf{y})=M\mathbf{y}$ where $M\in\mathbb{R}^{m\times n}$. Note that each matrix $M\in\mathbb{R}^{m\times n}$ defines a bounded linear transformation $\mathcal{M}:L^2(\Omega,\mathbb{R}^n)\to L^2(\Omega,\mathbb{R}^m)$. It is customary to write M rather then $\mathcal{M}$ since $[\mathcal{M}(\mathbf{y})](\omega)=M[\mathbf{y}(\omega)]$, for each $\omega\in\Omega$.

The known problem of finding optimal filter F is formulated as follows. Given large covariance matrices E_{xy} and E_{yy}, find M that solves

$$\min_M\|\mathbf{x}-M\mathbf{y}\|_\Omega^2. \tag{2}$$

 doi:10.20944/preprints202503.1571.v1

The minimal Frobenius norm solution to (2) is given by (see, for example, [1–5])

$$M = E_{xy}E_{yy}^{\dagger}, \tag{3}$$

where $E_{yy}^{\dagger}$ is the Moor-Penrose pseudo-inverse of E_{yy}.

1.2. Motivations

In 3, M contains $m \times n$ entries. In other words, the filter defined by 3 contains $m \times n$ parameters to optimize the associated accuracy.

Motivation 1: Increase in accuracy. Since M in 3 is optimal there is no other linear filter that can provide a better associated accuracy. At the same time, it may happen that the accuracy is not satisfactory. Then a logical way to increase the accuracy is an increase in the number of parameters to optimize. Since m is fixed then only n can be varied; n can be increased to, say, q where $q > n$, if $\mathbf{y}$ is replaced with another vector $\mathbf{v} \in L^2(\Omega, \mathbb{R}^q)$ which is unknown. In this case, we need to determine both M and $\mathbf{v}$. In particular, $\mathbf{v}$ can be represented as $\mathbf{v} = [\mathbf{v}_1^T, \dots, \mathbf{v}_p^T]^T$ where, for $j = 1, \dots, p$, $\mathbf{v}_j \in L^2(\Omega, \mathbb{R}^{q_j})$ and $q = q_1 + \dots + q_p$. Then the associated filter F_p is represented by the equation

$$F_p(\mathbf{v}_1, \dots \mathbf{v}_p) = M\mathbf{v} = [M_1, \dots, M_p][\mathbf{v}_1^T, \dots, \mathbf{v}_p^T]^T = \sum_{j=1}^{p} M_j\mathbf{v}_j, \tag{4}$$

where, for $j = 1, \dots, p$, $M_j \in \mathbb{R}^{m \times q_j}$. We call *$p$ the degree of filter F_p*.

Therefore, the *preliminary* statement of the problem considered here and in [6] is as follows. Given E_{xv} and E_{vv}, find $M_1, \dots, M_p$ and $\mathbf{v}_1, \dots, \mathbf{v}_p$ that solve

$$\min_{\substack{M_1,\dots,M_p \\ \mathbf{v}_1,\dots,\mathbf{v}_p}} \left\| \mathbf{x} - \sum_{j=1}^{p} M_j\mathbf{v}_j \right\|_{\Omega}^2. \tag{5}$$

Note, since E_{xv} and E_{vv} are given then, for $i, j = 1, \dots, p$, E_{xv_j} and $E_{v_iv_j}$ are known as blocks of E_{xv} and E_{vv}.

Motivation 2: Decrease in computational load. In 2 and 3, dimensions of $\mathbf{x}$ and $\mathbf{y}$, m and n, respectively, can be very large. In a number of important applied problems, such as those in biology, ecology, finance, sociology and medicine (see e.g., [7,8]), $m, n = O(10^3)$ and greater. For example, measurements of gene expression can contain tens of hundreds of samples. Each sample can contain tens of thousands of genes. As a result, in such cases, associated covariance matrices matrices in 3 are very large. Further, in (4) and (5), dimension q_j of $\mathbf{v}_j$ might be of the same order as that of observable signal $\mathbf{y}$. For instance, one of $\mathbf{v}_j$ can be equal to $\mathbf{y}$. Based on 2–(5), it is natural to predict that a solution of problem (5) contains $q \times q$ pseudo-inverse $E_{vv}^{\dagger}$ and $m \times q$ pseudo-inverse $M^{\dagger}$. It is indeed so, as shown in Section 4.1.6 and [6]. Matrices $E_{vv}^{\dagger}$ and $M^{\dagger}$ are usually larger than $E_{yy}^{\dagger}$. Then computation of a large pseudo-inverse matrix leads to a quite slow computing procedure up to the cases when the computer runs out of computer memory. In particular, the associated well-known phenomenon of the "curse of dimensionality" [9] states that many problems become exponentially difficult in high dimensions.

Of course, in the case of high dimensional signals, some well known dimensionality reduction techniques can be used (as those, for example, in [2,3,5]) as an intermediate step for the filtering. At the same time, those techniques also involve large pseudo-inverse covariance matrices of the same size as that in 3. Additionally, in applying the dimensionality reduction procedure, intrinsic associated errors appear. The errors may involve, in particular, an undesirable distortion of the data. Thus, an application of the dimensionality reduction techniques to the problem under consideration does not allow us to avoid a computation of large pseudo-inverse covariance matrices. Therefore, we wish to develop techniques that reduce computation of large pseudo-inverses $E_{vv}^{\dagger}$ and $M^{\dagger}$ without involving any additional associated error.

 doi:10.20944/preprints202503.1571.v1

The following example illustrates Motivation 2. Simulations in this example and also in 2 of 4.1.7 were run on a Dell Latitude 7400 Business Laptop (manufactured in 2020).

Example 1. *Let $E_{vv}^{\dagger} \in \mathbb{R}^{q\times q}$ be a matrix whose entries are normally distributed with mean* 0 *and variance* 1. *In Table 1, we provide the time (in seconds) needed for* MATLAB *to compute matrix $E_{vv}^{\dagger}$ for different values of q.*

Table 1. Time needed for MATLAB to compute $E_{vv}^{\dagger} \in \mathbb{R}^{q\times q}$ for different dimensions q.

q	500	1000	2000	3000	4000	5000	6000
Time	0.01	0.4	4	14	33	63	115

We observe that computation of, e.g., 5000×5000 *pseudo-inverse matrix requires* 63 *seconds while computation of the five pseudo-inverse matrices of sizes* 1000×1000 *requires about* $2 = 5 \times 0.4$ *seconds. Thus, a procedure that would allow us to reduce the computation of $E_{vv}^{\dagger} \in \mathbb{R}^{5000\times 5000}$ to a computation of, say, five pseudo-inverse matrices of sizes* 1000×1000 *would be faster. At the same time, such a procedure would additionally contain associated matrix multiplications which are time consuming as well. This observation is further elaborated in 4.1.2 and 2 of 4.1.7.*

2. Statement of the Problem

The solution of the problem in (5) is divided into two parts. In this paper, we consider the first part of the solution. It mainly concerns a technique for the decrease in a computational load associated with computation of $E_{vv}^{\dagger}$. The second part of the solution of problem (5) is considered in [6]. It represents an extension of the technique considered here to the solution of the original problem in (5). More precisely, here, we consider

$$\min_{M_1,\dots,M_p} \left\| \mathbf{x} - \sum_{j=1}^{p} M_j \mathbf{v}_j \right\|_{\Omega}^2. \tag{6}$$

In terms of notation (4), the known minimal Frobenius norm solution of (6) is given by (see, e.g., [1–5])

$$M = M = E_{xv} E_{vv}^{\dagger}. \tag{7}$$

As mentioned in Sections 1.2, in (7) (and in (8) below), $q \times q$ matrix E_{vv} might be very large. Then the pseudo-inverse $E_{vv}^{\dagger}$ may be difficult to compute. In this case, formula in (7) is not useful in practice.

Therefore, the problem is as follows. Given *large* E_{xv} and E_{vv}, find a computational procedure for solving (6) which significantly decreases the computational complexity for the $E_{vv}^{\dagger}$ evaluation compared to that needed by the filter represented by (7).

While problem (6) arises in the solution provided in [6] it is also important in its own right. In particular, if $\mathbf{x} = [\mathbf{x}_1^T, \dots, \mathbf{x}_p^T]^T$ where, for $j = 1, \dots, p$, $\mathbf{x}_j \in L^2(\Omega, \mathbb{R}^{m_j})$ and $m = m_1 + \cdots + m_p$, then (6) represents the problem of finding the optimal multi-input multi-output (MIMO) filter. In this case, $\mathbf{v}_j$ and $\mathbf{x}_j$, for $j = 1, \dots, p$, are an input signal and output signal, respectively.

The problem in (6) can also be interpreted as the system identification problem [10–13]. In this case, $\mathbf{v}_1, \dots, \mathbf{v}_p$ are the inputs of the system, $M = [M_1, \dots, M_p]$ is the input-output map of the system, and $\mathbf{x}$ is the system output. For example, in environmental monitoring, $\mathbf{v}_1, \dots, \mathbf{v}_p$ may represent random data coming from nodes measuring temperature, light or pressure variations, and $\mathbf{x}$ may represent a random signal obtained after a merging of the received data in order to estimate required data parameters within a prescribed accuracy. Similar scenarios occur, e.g., in target localization and tracking [14].

As mentioned above, in all problems considered in this paper, covariance matrices are assumed to be known. This assumption is a well-known and significant limitation in problems dealing with the optimal filtering of random signals. See [15–18] in this regard. The covariances can be estimated in various ways and particular techniques are given in many papers. We cite [19–24] as the examples. Estimation of covariance matrices is an important problem which is not considered in this paper.

 doi:10.20944/preprints202503.1571.v1

3. Contribution and Novelty

The main conceptual novelty of the approach developed in this paper is in the technique that allows us to decrease the computational load needed for the numerical realization of the proposed filter. The procedure is based on the reduction of a large E_{vv} matrix to a collection of smaller matrices. It is done so that the filter equation with large matrices is equivalently represented by the set of equations with smaller matrices. As a result, as shown in 4.1.7, the proposed p-th degree filter requires computation of p pseudo-inverse matrices with sizes smaller that the size of the large matrix E_{vv}. In this regard, see 4.1.2, and 1 and 2, for more details. The associated MATLAB code is given in 4.1.7.

In 2, 3 and 4, we also show that the accuracy of the proposed filter increases with increases in the degree of the filter. Details are provided in 4.1 and more specifically, in 4.1.2. In particular, the algorithm for the optimal determination of $M_1, \dots, M_p$ is given in 4.1.7. Note that the algorithm is easy to implement numerically. The error analysis associated with the filter under consideration is provided in 4.1.3, 4.1.5 and 4.1.8.

Remark 1. *Note that for every outcome $\omega \in \Omega$, realizations $\mathbf{v}_1(\omega), \dots, \mathbf{v}_p(\omega)$ and $\mathbf{x}(\omega)$ of signals $\mathbf{v}_1, \dots, \mathbf{v}_p$ and $\mathbf{x}$, respectively, occur with certain probabilities. Thus, random signals $\mathbf{v}_j$, for $j = 1, \dots, p$, and $\mathbf{x}$ are associated with infinite sets of realizations $V_j = \{\mathbf{v}_j(\omega) | \omega \in \Omega\} \subseteq \mathbb{R}^n$ and $X = \{\mathbf{x}(\omega) | \omega \in \Omega\} \subseteq \mathbb{R}^m$, respectively. For different ω, the values $\mathbf{v}_j(\omega)$, for $j = 1, \dots, p$, and $\mathbf{x}(\omega)$ are, in general, different, and in many cases will span the entire spaces $\mathbb{R}^q_j$ and $\mathbb{R}^m$, respectively. Therefore, for each $\mathbf{x}$ and $\mathbf{v}_j$, for $j = 1, \dots, p$, filter given by (4) can be interpreted as map $\mathbb{R}^{q_1} \times \dots \mathbb{R}^{q_p} \to \mathbb{R}^m$ where probabilities are assigned to each column of $\mathbb{R}^{q_j}$ and $\mathbb{R}^m$. Importantly, the filter is invariant with respect to outcome $\omega \in \Omega$, and therefore, is the same for all $\mathbf{v}_1(\omega) \dots, \mathbf{v}_p(\omega)$ and $\mathbf{x}(\omega)$ with different outcomes $\omega \in \Omega$.*

4. Solution of the Problem

We wish that in (5) and (6), $\sum_{j=1}^{p} M_j \mathbf{v}_j$ would never been the zero vector. To this end, we need to extend the known definition of the linear independence of vectors as follows.

Let $M^{(1)} \in \mathbb{R}^{m \times q_1}, ..., M^{(p)} \in \mathbb{R}^{m \times q_p}$ be some matrices. For $j = 1, \dots, p$, let $\mathcal{N}(M^{(j)})$ be a null space of matrix $M^{(j)}$.

A linear combination in the generalized sense of vectors $\mathbf{v}_1, \dots, \mathbf{v}_p$ is a vector

$$\widehat{\mathbf{v}} = M^{(1)}\mathbf{v}_1(\omega) + \dots + M^{(p)}\mathbf{v}_p(\omega).$$

We say that the linear combination in generalized sense is non-trivial if $\mathbf{v}_j(\omega) \notin \mathcal{N}(M^{(j)})$, for at least one of $j = 1, \dots, p$. Note that if $M^{(j)} \neq \mathbb{O}$ where $\mathbb{O}$ is the zero matrix, then it is still, of course, possible that $\mathbf{v}_j(\omega) \in \mathcal{N}(M^{(j)})$. Therefore, condition $\mathbf{v}_j(\omega) \notin \mathcal{N}(M^{(j)})$ is more general than condition $M^{(j)} \neq \mathbb{O}$.

Definition 1. *Random vectors $\mathbf{v}_1, \dots, \mathbf{v}_p$ are called* linearly independent in the generalized sense *if there is no non-trivial linear combination in the generalized sense of these vectors equal to the zero vector; in other words, if*

$$M^{(1)}\mathbf{v}_1(\omega) + \dots + M^{(p)}\mathbf{v}_p(\omega) = \mathbb{O},$$

for almost all $\omega \in \Omega$, implies that $\mathbf{v}_j(\omega) \in \mathcal{N}(M^{(j)})$, for each $j = 1, \dots, p$, and almost all $\omega \in \Omega$.

All random vectors considered below are assumed to be linearly independent in the generalized sense.

4.1. Solution of Problem in (6)

In (7), matrix M is the minimum Frobenius norm solution of the equation

$$ME_{vv} = E_{xv}. \tag{8}$$

 doi:10.20944/preprints202503.1571.v1

Here, we exploit the specific structure of the original system (8) to develop a special block elimination procedure that allows us to reduce the original system of equations (8) to a collection of independent smaller subsystems. Their solution implies less computational load than that needed for (7).

To this end, let us represent E_{vv} and E_{xv} in blocked forms as follows:

$$E_{vv} = \begin{bmatrix} E_{11} & \dots & E_{1p} \\ \vdots & \vdots & \vdots \\ E_{p1} & \dots & E_{pp} \end{bmatrix} \quad \text{and} \quad E_{xv} = [E_1 \dots E_p], \tag{9}$$

where $E_{ij} = E_{v_i v_j} \in \mathbb{R}^{q_j \times q_j}$ and $E_j = E_{x v_j} \in \mathbb{R}^{m \times q_j}$, for $j, i = 1, \dots, p$. Then (8) implies

$$[M_1 \dots M_p] \begin{bmatrix} E_{11} & \dots & E_{1p} \\ \vdots & \vdots & \vdots \\ E_{p1} & \dots & E_{pp} \end{bmatrix} = [E_1 \dots E_p]. \tag{10}$$

Thus, the solution of the problem in (8) is equivalent to the solution of equation (10).

We provide a specific solution of equation (10) in the following way. First, in 4.1.1, a generic basis for the solution is given. In 4.1.4, the solution of equation (10) is considered for $p = 3$. This is a preliminary step for the solution of equation (10) for an arbitrary p which is provided in 4.1.6.

4.1.1. Generic Basis for the Solution of Problem (8): Case $p = 2$ in (10). Second Degree Filter

To provide a generic basis for the solution of equation (10), let us consider the case $p = 2$ in (4) and (6), i.e., the second degree filter. Then (10) becomes

$$[M_1 \;\; M_2] \begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix} = [E_1 \;\; E_2]. \tag{11}$$

We denote

$$E_{22}^{(1)} = E_{22} - E_{21} E_{11}^{\dagger} E_{12} \quad \text{and} \quad E_2^{(1)} = E_2 - E_1 E_{11}^{\dagger} E_{12}. \tag{12}$$

Recall, it has been shown in [4] that for any random vectors $\boldsymbol{g}$ and $\boldsymbol{h}$,

$$E_{gh} = E_{gh} E_{hh}^{\dagger} E_{hh}. \tag{13}$$

Lemma 1. *Let M_1 and M_2 satisfy equation (11). Then (11) decomposes into equations*

$$M_1 E_{11} = E_1 - M_2 E_{21} \quad \textit{and} \quad M_2 E_{22}^{(1)} = E_2^{(1)} \tag{14}$$

Proof. Let us denote $\bar{M}_1 = M_1 + M_2 E_{21} E_{11}^{\dagger}$. Then

$$[\bar{M}_1 \;\; M_2] = [M_1 \;\; M_2] \begin{bmatrix} I & \mathbb{O} \\ E_{21} E_{11}^{\dagger} & I \end{bmatrix}$$

and

$$[M_1 \;\; M_2] = [\bar{M}_1 \;\; M_2] \begin{bmatrix} I & \mathbb{O} \\ -E_{21} E_{11}^{\dagger} & I \end{bmatrix}, \tag{15}$$

where by (13),

$$E_{21} E_{11}^{\dagger} E_{11} = E_{21}. \tag{16}$$

 doi:10.20944/preprints202503.1571.v1

Thus, (11) is represented as

$$[\bar{M}_1 \;\; M_2]\begin{bmatrix} I & \mathbb{O} \\ -E_{21}E_{11}^{\dagger} & I \end{bmatrix}\begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix} = [E_1 \;\; E_2]$$

which implies

$$[\bar{M}_1 \;\; M_2]\begin{bmatrix} I & \mathbb{O} \\ -E_{21}E_{11}^{\dagger} & I \end{bmatrix}\begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix}\begin{bmatrix} I & -E_{11}^{\dagger}E_{12} \\ \mathbb{O} & I \end{bmatrix}$$
$$= [E_1 \;\; E_2]\begin{bmatrix} I & -E_{11}^{\dagger}E_{12} \\ \mathbb{O} & I \end{bmatrix}. \tag{17}$$

On the basis of (17), we see that (8), for $p = 2$, reduces to the equation

$$[\bar{M}_1 \;\; M_2]\begin{bmatrix} E_{11} & \mathbb{O} \\ \mathbb{O} & E_{22}^{(1)} \end{bmatrix} = [E_1 \;\; E_2^{(1)}]. \tag{18}$$

The latter implies 14. ■

Remark 2. *The above proof is based on equation* (15) *which implies the equivalence of equations* (11) *and* 14. *In turn,* (1) *allows us to equivalently obtain* 14 *as follows. Let us multiply* (11) *from the right by* $N_{21} = \begin{bmatrix} I & -{E_{11}}^{\dagger}E_{12} \\ \mathbb{O} & I \end{bmatrix}$. *Then*

$$[M_1 \;\; M_2]\begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix}N_{21} = [E_1 \;\; E_2]N_{21}, \tag{19}$$

i.e.,

$$[M_1 \;\; M_2]\begin{bmatrix} E_{11} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} \end{bmatrix} = [E_1 \;\; E_2^{(1)}]. \tag{20}$$

Then 14 *follows from* (20).

Theorem 1. *For* $p = 2$, *the minimal Frobenius norm solution of problem* (8) *is represented by*

$$M_2 = \tilde{M}_2 = E_2^{(1)}(E_{22}^{(1)})^{\dagger}, M_1 = \tilde{M}_1 = (E_1 - \tilde{M}_2E_{21})E_{11}^{\dagger} \tag{21}$$

Proof. By (1), matrices M_1 and M_2 that solve (11) satisfy the equations in 14. By [2,3,17,18], their minimal Frobenius norm solutions are given by (21). ■

It follows from (21) that the second degree filter requires computation of two pseudo-inverse matrices, $E_{11}^{\dagger} \in \mathbb{R}^{q_1 \times q_1}$ and $(E_{22}^{(1)})^{\dagger} \in \mathbb{R}^{q_2 \times q_2}$, of sizes that are smaller than the size of $E_{vv}^{\dagger} \in \mathbb{R}^{q \times q}$ (if, of course, $q_1 \neq 0$ and $q_2 \neq 0$).

4.1.2. Decrease in Computational Load

Here, we wish to compare the computational load needed for a numerical realization of the second degree filter given, for $p = 2$, by (4) and (21), and that of the filter represented by (7).

By [25] (p. 254), the product of $m \times n$ and $n \times q$ matrices requires, for large m, n and q, $2mnq$ flops. For large q, the Golub-Reinsch SVD used for computation of the $q \times q$ matrix pseudo-inverse requires $21q^3$ flops (see [25], p. 254).

In (12), for $q_1 = q_2 = q/2$, the evaluation of $E_{22}^{(1)}$ implies the evaluation of $E_{11}^{\dagger}$, matrix multiplications in $E_{21}E_{11}^{\dagger}E_{12}$ and matrix subtraction. These operations require $21(q/2)^3$ flops, $2(q/2)^3 + 2(q/2)^3$ flops and $q^2/4$ flops, respectively. Similarly, computation of $E_2^{(1)}$ requires $2mq^2/4 + 2q^3/8 + mq/2$

 doi:10.20944/preprints202503.1571.v1

flops. In (21), computation of M_2 and M_1 imply $21q^3/8 + 2mq^2/4$ flops and $2mq^2/4 + 2mq^2/4 + mq/2$ flops, respectively.

Let us denote by $C(A)$ a number of flops required to compute an operation A. In total, the second degree filter requires $C(E_{22}^{(1)}) + C(E_2^{(1)}) + C(M_2) + C(M_1)$ flops where

$$C(E_{22}^{(1)}) = q^2/4 + 25q^3/8,$$

$$C(E_2^{(1)}) = mq/2 + 2mq^2/4 + 2q^3/8,$$

$$C(M_2) = 2mq^2/4 + 21q^3/8,$$

$$C(M_1) = mq/2 + mq^2.$$

Thus, the number of flops needed for the numerical realization of the second degree filter is equal to $mq + q^2/4 + 2mq^2 + 6q^3$.

The computational load required by the filter given by (7) is $2mq^2 + 21q^3$ flops. It is customary to choose $m \leq q$. In this case, clearly, $mq + q^2/4 + 2mq^2 + 6q^3 < 2mq^2 + 21q^3$. In particular, if $m = q$ then the latter inequality becomes $5q^2/4 + 8q^3 < 23q^3$. That is, for sufficiently large m and q, the proposed second degree filter requires, roughly, up to 2.8 times less flops than that by the filter given by (7). In simulations by MATLAB provided in 2 below, the second degree filter takes about two times less seconds than the filter given by (7). A command that would allow us to compute a number of flops is not available in MATLAB.

In the following 4.1.4 and 4.1.6, the procedure described in 4.1.1 is extended to the case of the filter of higher degrees. For given q, this will imply, obviously, smaller sizes of blocks of matrices E_{vv} and E_{xv} than those for the second degree filter, and as a result, a further decrease in the associated computational load.

4.1.3. Error Associated with the Second Degree Filter Determined by (4)

Now, we wish to show that the error associated with the second degree filter determined, for $p = 2$, by (4) and (21) is less than the error associated with the filter determined by 3.

Let us denote

$$\epsilon = \min_M \|\mathbf{x} - M\mathbf{y}\|_\Omega^2, \varepsilon_2 = \min_{M_1, M_2} \left\| \mathbf{x} - [M_1 M_2] \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix} \right\|_\Omega^2. \tag{22}$$

The Frobenius norm of matrix M is denoted by $\|M\|$.

Theorem 2. *Let* $\mathbf{v}_1$ *and* $\mathbf{v}_2$ *be such that*

$$\sum_{j=1}^{2} \left\| E_j^{(j-1)} [(E_{jj}^{(j-1)})^{1/2}]^\dagger \right\|^2 > \|E_{xy} E_{yy}^{1/2}\|^2, \tag{23}$$

where $E_1 = E_1^{(0)}$ *and* $E_{11} = E_{11}^{(0)}$. *Then*

$$\varepsilon_2 < \epsilon. \tag{24}$$

Proof. It is known (see, for example, [1–5]) that, for M determined by 3, and for $\tilde{M}_1$ and $\tilde{M}_2$ determined by (21),

$$\epsilon = \|E_{xx}^{1/2}\|^2 - \mathrm{tr}\{E_{xy} E_{yy}^\dagger E_{yx}\}, \ \varepsilon_2 = \|E_{xx}^{1/2}\|^2 - \mathrm{tr}\{E_{xv} E_{vv}^\dagger E_{vx}\}, \tag{25}$$

where $\mathbf{v} = \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix}$ and, bearing in mind (2), $[\tilde{M}_1 \tilde{M}_2] = E_{xv} E_{vv}^\dagger$. Therefore, by (21),

$$\mathrm{tr}\{E_{xv} E_{vv}^\dagger E_{vx}\} = \mathrm{tr}\{[\tilde{M}_1 \tilde{M}_2] E_{vx}\} = \mathrm{tr}\{\tilde{M}_1 E_1^T + \tilde{M}_2 E_2^T\}$$

 doi:10.20944/preprints202503.1571.v1

$$
\begin{aligned}
&= \mathrm{tr}\{(E_1E_{11}^{\dagger} - \tilde{M}_2E_{21}E_{11}^{\dagger})E_1^T + \tilde{M}_2E_2^T\} \\
&= \mathrm{tr}\{E_1E_{11}^{\dagger}E_1^T\} + \mathrm{tr}\{\tilde{M}_2(E_2^T - E_{21}E_{11}^{\dagger}E_1^T)\} \\
&= \mathrm{tr}\{E_1E_{11}^{\dagger}E_1^T\} \\
&+\mathrm{tr}\{(E_2 - E_1E_{11}^{\dagger}E_{12})(E_{22} - E_{21}E_{11}^{\dagger}E_{12})^{\dagger}(E_2^T - E_{21}E_{11}^{\dagger}E_1^T)\} \\
&= \|E_1E_{11}^{1/2}\|^2 + \|E_2^{(1)}[(E_{22}^{(1)})^{1/2}]^{\dagger}\|^2. \qquad (26)
\end{aligned}
$$

Then

$$\varepsilon_2 = \|E_{xx}^{1/2}\|^2 - (\|E_1E_{11}^{1/2}\|^2 + \|E_2^{(1)}[(E_{22}^{(1)})^{1/2}]^{\dagger}\|^2). \qquad (27)$$

At the same time,

$$\epsilon = \|E_{xx}^{1/2}\|^2 - \|E_{xy}E_{yy}^{1/2}\|^2. \qquad (28)$$

As a result, (24) follows from (23), (27) and (28). ■

Remark 3. *The condition in* (23) *can practically be satisfied by, in particular, a suitable choice of* $\mathbf{v}_1$ *and* $\mathbf{v}_2$*, and/or their dimensions. The following 1 demonstrates such a particular choice of* $\mathbf{v}_1$*.*

Corollary 1. *Let* $\mathbf{v}_1 = \mathbf{y}$ *and* $\mathbf{v}_2$ *be a nonzero signal. Then* (24) *is always true.*

Proof. The proof follows directly from (23), (27) and (28). ■

In general, the proposed solution of equation ((10)) is based on a combination of (1) and (2) with the extension of the idea of Gaussian elimination [25] to the case of matrices with specific blocks E_{ij} and E_j. This allows us to build the special procedure considered below in 4.1.4 and 4.1.6.

4.1.4. Solution of Equation (10) for $p = 3$

To clarify the solution device of equation (10) for an arbitrary p, let us now consider a particular case with $p = 3$. The solution is based on the development of the procedure represented by (19) and (20).

For $p = 3$, equation (10) is as follows:

$$[M_1 \;\; M_2 \;\; M_3]\begin{bmatrix} E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{bmatrix} = [E_1 \;\; E_2 \;\; E_3]. \qquad (29)$$

First, consider a reduction of (29) to the equation with *the block-lower triangular matrix.*

Step 1. By (13), for $i = 2, 3$, $E_{i1} = E_{i1}E_{11}^{\dagger}E_{11}$. Define

$$N_{3,1} = \begin{bmatrix} I & -E_{11}{}^{\dagger}E_{12} & -E_{11}{}^{\dagger}E_{13} \\ \mathbb{O} & I & \mathbb{O} \\ \mathbb{O} & \mathbb{O} & I \end{bmatrix}.$$

Now, in (29), for $i = 1, 2, 3$ and $j = 2, 3$, update block-matrices E_{ij} and E_j to $E_{ij}^{(1)}$ and $E_j^{(1)}$, respectively, so that

$$\begin{bmatrix} E_{11} & \mathbb{O} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & E_{23}^{(1)} \\ E_{31} & E_{32}^{(1)} & E_{33}^{(1)} \end{bmatrix} = \begin{bmatrix} E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{bmatrix} N_{3,1} \qquad (30)$$

and

$$[E_1 \;\; E_2^{(1)} \;\; E_3^{(1)}] = [E_1 \;\; E_2 \;\; E_3]N_{3,1} \qquad (31)$$

 doi:10.20944/preprints202503.1571.v1

where $E_{kj}^{(1)} = E_{kj} - E_{k1}E_{11}^{\dagger}E_{1j}$, for $k = 2,3$ and $j = 2,3$, and $E_{j}^{(1)} = E_{j} - E_{1}E_{11}^{\dagger}E_{1j}$, for $j = 2,3$. Note that $E_{12}^{(1)} = E_{13}^{(1)} = \mathbb{O}$ by (13).

Step 2. Set

$$N_{3,2} = \begin{bmatrix} I & \mathbb{O} & \mathbb{O} \\ \mathbb{O} & I & -{E_{22}^{(1)}}^{\dagger}E_{23}^{(1)} \\ \mathbb{O} & \mathbb{O} & I \end{bmatrix}.$$

By (13), $E_{32}^{(1)} = E_{32}^{(1)}{E_{22}^{(1)}}^{\dagger}E_{22}^{(1)}$. Taking the latter into account, update block-matrices $E_{ij}^{(1)}$ and $E_{j}^{(1)}$ in the LHS of (31) and 31, for $i = 2,3$ and $j = 3$, so that

$$\begin{bmatrix} E_{11} & \mathbb{O} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & \mathbb{O} \\ E_{31} & E_{32}^{(1)} & E_{33}^{(2)} \end{bmatrix} = \begin{bmatrix} E_{11} & \mathbb{O} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & E_{23}^{(1)} \\ E_{31} & E_{32}^{(1)} & E_{33}^{(1)} \end{bmatrix} N_{3,2} \tag{32}$$

and

$$[E_1 \;\; E_2^{(1)} \;\; E_3^{(2)}] = [E_1 \;\; E_2^{(1)} \;\; E_3^{(1)}]N_{3,2} \tag{33}$$

where $E_{33}^{(2)} = E_{33}^{(1)} - E_{32}^{(1)}{E_{22}^{(1)}}^{\dagger}E_{23}^{(1)}$ and $E_{3}^{(2)} = E_{3}^{(1)} - E_{2}^{(1)}{E_{22}^{(1)}}^{\dagger}E_{23}^{(1)}$.

As a result, we finally obtain the updated matrix equation with the block-lower triangular matrix

$$[M_1 \;\; M_2 \;\; M_3] \begin{bmatrix} E_{11} & \mathbb{O} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & \mathbb{O} \\ E_{31} & E_{32}^{(1)} & E_{33}^{(2)} \end{bmatrix} = [E_1 \;\; E_2^{(1)} \;\; E_3^{(2)}]. \tag{34}$$

Solution of equation (34). Using a back substitution, equation (34) is represented by the three equations as follows:

$$M_3E_{33}^{(2)} = E_3^{(2)}, \tag{35}$$

$$M_2E_{22}^{(1)} + M_3E_{32}^{(1)} = E_2^{(1)}, \tag{36}$$

$$M_1E_{11} + M_2E_{21} + M_3E_{31} = E_1, \tag{37}$$

where E_{11}, E_{21}, E_{31} and E_1 are the original blocks in (29).

It follows from (35)–(37) that the solution of equation (10), for $p = 3$, is represented by the minimal Frobenius norm solutions to the problems

$$\min_{M_3} \|E_3^{(2)} - M_3E_{33}^{(2)}\|^2, \tag{38}$$

$$\min_{M_2} \|(E_2^{(1)} - \tilde{M}_3E_{32}^{(1)}) - M_2E_{22}^{(1)}\|^2, \tag{39}$$

$$\min_{M_1} \|(E_1 - \sum_{j=2}^{3} \tilde{M}_jE_{j1}) - M_1E_{11}\|^2, \tag{40}$$

where $\tilde{M}_3$ and $\tilde{M}_2$ are solutions to the problems in (38) and (39), respectively. Matrices $\tilde{M}_j$, for $j = 1,2,3$, that solve (38)-(40) are as follows:

$$\tilde{M}_3 = E_3^{(2)}[E_{33}^{(2)}]^{\dagger}, \tag{41}$$

$$\tilde{M}_2 = (E_2^{(1)} - \tilde{M}_3E_{32}^{(1)})(E_{22}^{(1)})^{\dagger}, \tag{42}$$

$$\tilde{M}_1 = (E_1 - \sum_{j=2}^{3} \tilde{M}_jE_{j1})(E_{11})^{\dagger}. \tag{43}$$

 doi:10.20944/preprints202503.1571.v1

Thus, the third degree filter requires computation of three pseudo-inverse matrices, $E_{11}^{\dagger} \in \mathbb{R}^{q_1 \times q_1}$, $(E_{22}^{(1)})^{\dagger} \in \mathbb{R}^{q_2 \times q_2}$ and $[E_{33}^{(2)}]^{\dagger} \in \mathbb{R}^{q_3 \times q_3}$.

4.1.5. Error Associated with the Third Degree Filter Determined by (41)–(43)

We wish to show that the error associated with the filter determined by (41)–(43), is less than that associated with the filter determined by 3, ϵ.

To this end, let us denote

$$\varepsilon_3 = \min_{M_1,M_2,M_3} \|\mathbf{x} - [M_1 M_2 M_3]\mathbf{v}\|_{\Omega}^2, \tag{44}$$

where $\mathbf{v} = [\mathbf{v}_1^T, \mathbf{v}_2^T, \mathbf{v}_3^T]^T$. As before, $E_1 = E_1^{(0)}$ and $E_{11} = E_{11}^{(0)}$.

Theorem 3. *Let* $\mathbf{v}_1$, $\mathbf{v}_2$ *and* $\mathbf{v}_3$ *be such that*

$$\sum_{j=1}^{3} \left\| E_j^{(j-1)} [(E_{jj}^{(j-1)})^{1/2}]^{\dagger} \right\|^2 > \|E_{xy} E_{yy}^{1/2}\|^2. \tag{45}$$

Then

$$\varepsilon_3 < \epsilon. \tag{46}$$

Proof. We have

$$\varepsilon_3 = \|E_{xx}^{1/2}\|^2 - \mathrm{tr}\{E_{xv} E_{vv}^{\dagger} E_{vx}\}, \tag{47}$$

and, bearing in mind an obvious extension of Remark (2), for $p = 3$, $[\tilde{M}_1 \tilde{M}_2 \tilde{M}_3] = E_{xv} E_{vv}^{\dagger}$. Therefore, by ((41))–((43)),

$$\begin{aligned}
&\mathrm{tr}\{E_{xv} E_{vv}^{\dagger} E_{vx}\} \\
&= \mathrm{tr}\{[\tilde{M}_1 \tilde{M}_2 \tilde{M}_3] E_{vx}\} \\
&= \mathrm{tr}\{\tilde{M}_1 E_1^T + \tilde{M}_2 E_2^T + \tilde{M}_3 E_3^T\} \\
&= \mathrm{tr}\{(E_1 E_{11}^{\dagger} - [\tilde{M}_2 E_{21} + \tilde{M}_3 E_{31}] E_{11}^{\dagger}) E_1^T + \tilde{M}_2 E_2^T + \tilde{M}_3 E_3^T\} \\
&= \mathrm{tr}\{E_1 E_{11}^{\dagger} E_1^T\} + \mathrm{tr}\{\tilde{M}_2 (E_2^T - E_{21} E_{11}^{\dagger} E_1^T) \\
&\quad + \tilde{M}_3 (E_3^T - E_{31} E_{11}^{\dagger} E_1^T)\} \\
&= \mathrm{tr}\{E_1 E_{11}^{\dagger} E_1^T\} + \mathrm{tr}\{(E_2^{(1)} (E_{22}^{(1)})^{\dagger} - \tilde{M}_3 E_{32}^{(1)} (E_{22}^{(1)})^{\dagger}) (E_2^{(1)})^T \\
&\quad + \tilde{M}_3 (E_3^T - E_{31} E_{11}^{\dagger} E_1^T)\} \\
&= \mathrm{tr}\{E_1 E_{11}^{\dagger} E_1^T + E_2^{(1)} (E_{22}^{(1)})^{\dagger} (E_2^{(1)})^T \\
&\quad + \tilde{M}_3 [E_3^T - E_{31} E_{11}^{\dagger} E_1^T - E_{32}^{(1)} (E_{22}^{(1)})^{\dagger} + (E_2^{(1)})^T]\} \\
&= \|E_1 E_{11}^{1/2}\|^2 + \|E_2 E_{22}^{1/2}\|^2 + \mathrm{tr}\{E_3^{(2)} (E_{33}^{(2)})^{\dagger} (E_3^{(2)})^T\} \\
&= \|E_1 E_{11}^{1/2}\|^2 + \|E_2^{(1)} [(E_{22}^{(1)})^{1/2}]^{\dagger}\|^2 + \|E_3^{(2)} [(E_{33}^{(2)})^{1/2}]^{\dagger}\|^2.
\end{aligned} \tag{48}$$

Thus, (45)–(46) follows from ϵ in (22), (47) and (48). ■

Corollary 2. *If* $\mathbf{v}_1 = \mathbf{y}$, *and at least one of* $\mathbf{v}_2$ *or* $\mathbf{v}_3$ *is a nonzero vector then* (46) *is always true.*

Proof. The proof follows from (45), (48) and (28). ■

4.1.6. Solution of Equation (10) for Arbitrary p

Here, we extend the procedure considered in (4.1.4) to the case of an arbitrary p. First, we reduce the equation in (10) to an equation with *the block-lower triangular matrix.* We do it by the following steps where each step is justified by an obvious extension of (1).

 doi:10.20944/preprints202503.1571.v1

Step 1. First, in (10), for $k = 1, \ldots, p$ and $j = 2, \ldots, p$, we wish to update blocks E_{kj} and E_k to $E_{kj}^{(1)}$ and $E_k^{(1)}$, respectively. To this end, define

$$N_{p,1} = \begin{bmatrix} I & -{E_{11}}^{\dagger} E_{12} & \ldots & -{E_{11}}^{\dagger} E_{1p} \\ \mathbb{O} & I & \ldots & \mathbb{O} \\ \ldots & \ldots & \ldots & \ldots \\ \mathbb{O} & \ldots & I & \mathbb{O} \\ \mathbb{O} & \ldots & \mathbb{O} & I \end{bmatrix}.$$

Bearing in mind that by (13), $E_{i1} = E_{i1}E_{11}^{\dagger}E_{11}$, for $i = 2, \ldots, p$, we obtain

$$\begin{bmatrix} E_{11} & \mathbb{O} & \ldots & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & \ldots & E_{2p}^{(1)} \\ \ldots & \ldots & \ldots & \ldots \\ E_{p-1,1} & E_{p-1,2}^{(1)} & \ldots & E_{p-1,p}^{(1)} \\ E_{p1} & E_{p2}^{(1)} & \ldots & E_{pp}^{(1)} \end{bmatrix}$$

$$= \begin{bmatrix} E_{11} & E_{12} & \ldots & E_{1p} \\ E_{21} & E_{22} & \ldots & E_{2p} \\ \ldots & \ldots & \ldots & \ldots \\ E_{p-1,1} & E_{p-1,2} & \ldots & E_{p-1,p} \\ E_{p1} & E_{p2} & \ldots & E_{pp} \end{bmatrix} N_{p,1} \tag{49}$$

and

$$[E_1 \;\; E_2^{(1)} \;\; \ldots \;\; E_p^{(1)}] = [E_1 \;\; E_2 \;\; \ldots \;\; E_p]N_{p,1}, \tag{50}$$

where, for $k = 1, \ldots, p$ and $j = 2, \ldots, p$,

$$E_{kj}^{(1)} = E_{kj} - E_{k1}E_{11}^{\dagger}E_{1j} \tag{51}$$

and

$$E_j^{(1)} = E_j - E_1E_{11}^{\dagger}E_{1j}. \tag{52}$$

Step 2. Now we wish to update blocks $E_{kj}^{(1)}$ and $E_k^{(1)}$, for $k = 2, \ldots, p$ and $j = 3, \ldots, p$, of the matrices in the LHS of (49) and (50), respectively.

Set

$$N_{p,2} = \begin{bmatrix} I & \mathbb{O} & \mathbb{O} & \ldots & \mathbb{O} \\ \mathbb{O} & I & -{E_{22}^{(1)}}^{\dagger} E_{23}^{(1)} & \ldots & -{E_{22}^{(1)}}^{\dagger} E_{2p}^{(1)} \\ \mathbb{O} & \mathbb{O} & I & \ldots & \mathbb{O} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \mathbb{O} & \mathbb{O} & \mathbb{O} & \ldots & I \end{bmatrix},$$

Because by (13), $E_{i2}^{(1)} = E_{i2}^{(1)}{E_{22}^{(1)}}^{\dagger}E_{22}^{(1)}$, for $i = 3, \ldots, p$, we have

$$\begin{bmatrix} E_{11} & \mathbb{O} & \mathbb{O} & \ldots & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & \mathbb{O} & \ldots & \mathbb{O} \\ E_{31} & E_{32}^{(1)} & E_{33}^{(2)} & \ldots & E_{3p}^{(2)} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ E_{p1} & E_{p2}^{(1)} & E_{p3}^{(2)} & \ldots & E_{pp}^{(2)} \end{bmatrix}$$

 doi:10.20944/preprints202503.1571.v1

$$= \begin{bmatrix} E_{11} & \mathbb{O} & \mathbb{O} & \dots & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & E_{23}^{(1)} & \dots & E_{2p}^{(1)} \\ E_{31} & E_{32}^{(1)} & E_{33}^{(1)} & \dots & E_{3p}^{(1)} \\ \dots & \dots & \dots & \dots & \dots \\ E_{p1} & E_{p2}^{(1)} & E_{p3}^{(1)} & \dots & E_{pp}^{(1)} \end{bmatrix} N_{p,2} \tag{53}$$

and

$$[E_1 \;\; E_2^{(1)} \;\; E_3^{(2)} \;\; \dots \;\; E_p^{(2)}] = [E_1 \;\; E_2^{(1)} \;\; E_3^{(1)} \;\; \dots \;\; E_p^{(1)}] N_{p,2}, \tag{54}$$

where, for $k = 2, \dots, p$ and $j = 3, \dots, p$,

$$E_{kj}^{(2)} = E_{kj}^{(1)} - E_{k2}^{(1)} {E_{22}^{(1)}}^{\dagger} E_{2j}^{(1)} \tag{55}$$

and

$$E_j^{(2)} = E_j^{(1)} - E_2^{(1)} {E_{22}^{(1)}}^{\dagger} E_{2j}^{(1)}. \tag{56}$$

We continue this updating procedure up to the final Step $(p-1)$. In Step $(p-2)$, the following matrices are obtained:

$$\begin{bmatrix} E_{11} & \mathbb{O} & \dots & \mathbb{O} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & \dots & \mathbb{O} & \mathbb{O} \\ \dots & \dots & \dots & \dots & \dots \\ E_{p-1,1} & E_{p-1,2}^{(1)} & \dots & E_{p-1,p-1}^{(p-2)} & E_{p-1,p}^{(p-2)} \\ E_{p1} & E_{p2}^{(1)} & \dots & E_{p,p-1}^{(p-2)} & E_{pp}^{(p-2)} \end{bmatrix} \tag{57}$$

and

$$[E_1 \;\; E_2^{(1)} \;\; \dots \;\; E_{p-2}^{(p-3)} \;\; E_{p-1}^{(p-2)} \;\; E_p^{(p-2)}]. \tag{58}$$

Step $(p-1)$. By (13),

$$E_{p-1,p}^{(p-2)} = E_{p-1,p}^{(p-2)} (E_{p-1,p-1}^{(2)})^{\dagger} E_{p-1,p-1}^{(p-2)}.$$

In 57 and 58, we now update only two blocks $E_{p-1,p}^{(p-2)}$ and $E_{pp}^{(p-2)}$, and the single block $E_p^{(p-2)}$. To this end, define

$$N_{p,p-1} = \begin{bmatrix} I & \mathbb{O} & \dots & \mathbb{O} & \mathbb{O} \\ \mathbb{O} & I & \dots & \mathbb{O} & \mathbb{O} \\ \dots & \dots & \dots & \dots & \dots \\ \mathbb{O} & \mathbb{O} & \dots & I & -{E_{p-1,p-1}^{(p-2)}}^{\dagger} E_{p-1,p}^{(p-2)} \\ \mathbb{O} & \mathbb{O} & \dots & \mathbb{O} & I \end{bmatrix},$$

and obtain

$$\begin{bmatrix} E_{11} & \mathbb{O} & \dots & \mathbb{O} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & \dots & \mathbb{O} & \mathbb{O} \\ \dots & \dots & \dots & \dots & \dots \\ E_{p-1,1} & E_{p-1,2}^{(1)} & \dots & E_{p-1,p-1}^{(p-2)} & \mathbb{O} \\ E_{p1} & E_{p2}^{(1)} & \dots & E_{p,p-1}^{(p-2)} & E_{pp}^{(p-1)} \end{bmatrix}$$

$$= \begin{bmatrix} E_{11} & \mathbb{O} & \dots & \mathbb{O} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & \dots & \mathbb{O} & \mathbb{O} \\ \dots & \dots & \dots & \dots & \dots \\ E_{p-1,1} & E_{p-1,2}^{(1)} & \dots & E_{p-1,p-1}^{(p-2)} & E_{p-1,p}^{(p-2)} \\ E_{p1} & E_{p2}^{(1)} & \dots & E_{p,p-1}^{(p-2)} & E_{pp}^{(p-2)} \end{bmatrix} N_{p,p-1} \tag{59}$$

and

$$[E_1 \;\; E_2^{(1)} \;\; \dots \;\; E_{p-2}^{(p-3)} \;\; E_{p-1}^{(p-2)} \;\; E_p^{(p-1)}]$$

 doi:10.20944/preprints202503.1571.v1

$$= [E_1 \;\; E_2^{(1)} \;\; \ldots \;\; E_{p-2}^{(p-3)} \;\; E_{p-1}^{(p-2)} \;\; E_p^{(p-2)}] N_{p,p-1}, \tag{60}$$

where

$$E_{pp}^{(p-1)} = E_{pp}^{(p-2)} - E_{p,p-1}^{(p-2)} (E_{p-1,p-1}^{(p-2)})^{\dagger} E_{p-1,p}^{(p-2)} \tag{61}$$

and

$$E_p^{(p-1)} = E_p^{(p-2)} - E_{p-1}^{(p-2)} (E_{p-1,p-1}^{(p-2)})^{\dagger} E_{p-1,p}^{(p-2)}. \tag{62}$$

As a result, on the basis of (59)–(62), equation (10) is reduced to the equation with the the block-lower triangular matrix as follows

$$\widehat{M} \begin{bmatrix} E_{11} & \mathbb{O} & \ldots & \mathbb{O} & \mathbb{O} \\ E_{21} & E_{22}^{(1)} & \ldots & \mathbb{O} & \mathbb{O} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ E_{p-1,1} & E_{p-1,2}^{(1)} & \ldots & E_{p-1,p-1}^{(p-2)} & \mathbb{O} \\ E_{p1} & E_{p2}^{(1)} & \ldots & E_{p,p-1}^{(p-2)} & E_{pp}^{(p-1)} \end{bmatrix} = \widehat{E}, \tag{63}$$

where

$$\widehat{M} = [M_1 \;\; M_2 \;\; \ldots \;\; M_{p-1} \;\; M_p]$$

and

$$\widehat{E} = [E_1 \;\; E_2^{(1)} \;\; \ldots \;\; E_{p-1}^{(p-2)} \;\; E_p^{(p-1)}].$$

4.1.7. Solution of Equation (10)

Using a back substitution, (63) is represented by p equations as follows:

$$\begin{aligned} M_p E_{pp}^{(p-1)} &= E_p^{(p-1)}, \\ M_{p-1} E_{p-1,p-1}^{(p-2)} + M_p E_{p,p-1}^{(p-2)} &= E_{p-1}^{(p-2)}, \\ &\vdots \\ M_2 E_{22}^{(1)} + \ldots + M_{p-1} E_{p-1,2}^{(1)} + M_p E_{p2}^{(1)} &= E_2^{(1)}, \\ M_1 E_{11} + \ldots + M_{p-1} E_{p-1,1} + M_p E_{p1} &= E_1. \end{aligned}$$

Therefore, the solution of equation (10) is represented by solutions to the problems

$$\min_{M_p} \|E_p^{(p-1)} - M_p E_{pp}^{(p-1)}\|^2, \tag{64}$$

$$\min_{M_{p-1}} \|(E_{p-1}^{(p-2)} - \tilde{M}_p E_{p,p-1}^{(p-2)}) - M_{p-1} E_{p-1,p-1}^{(p-2)}\|^2, \tag{65}$$

$$\vdots$$

$$\min_{M_2} \|(E_2^{(1)} - \sum_{j=3}^{p} \tilde{M}_j E_{j2}^{(1)}) - M_2 E_{22}^{(1)}\|^2, \tag{66}$$

$$\min_{M_1} \|(E_1 - \sum_{j=2}^{p} \tilde{M}_j E_{j1}) - M_1 E_{11}\|^2. \tag{67}$$

In (65)–(67), $\tilde{M}_p, \ldots, \tilde{M}_2$ are solutions to the problems in (64)–(66), respectively. On the basis of [26], the minimal Frobenius norm solutions $\tilde{M}_p, \ldots, \tilde{M}_1$ to (64)–(67) are given by

$$\tilde{M}_p = E_p^{(p-1)} (E_{pp}^{(p-1)})^{\dagger}, \tag{68}$$

 doi:10.20944/preprints202503.1571.v1

$$\tilde{M}_{p-1} = (E_{p-1}^{(p-2)} - \tilde{M}_p E_{p,p-1}^{(p-2)})(E_{p-1,p-1}^{(p-2)})^\dagger, \tag{69}$$

$$\vdots$$

$$\tilde{M}_2 = (E_2^{(1)} - \sum_{j=3}^{p} \tilde{M}_j E_{j2}^{(1)})(E_{22}^{(1)})^\dagger, \tag{70}$$

$$\tilde{M}_1 = (E_1 - \sum_{j=2}^{p} \tilde{M}_j E_{j1}) E_{11}^\dagger. \tag{71}$$

Thus, the p-th degree filter requires computation of p pseudo-inverse matrices, $E_{11}^\dagger \in \mathbb{R}^{q_1 \times q_1}$, $(E_{22}^{(1)})^\dagger \in \mathbb{R}^{q_2 \times q_2}, \ldots, (E_{pp}^{(p-1)})^\dagger \in \mathbb{R}^{q_p \times q_p}$.

The algorithm that follows is based on formulas (51), (52), (55), (56), (61), (62) and (68)–(71).

Algorithm 1: Solution of equation (10)

Input: p, E_{kj} for $k, j = 1, \ldots, p$, and E_k for $k = 1, \ldots, p$.
Output: M_j for $j = 1, \ldots, p$.

1 Set $s \leftarrow 1$;
2 **while** $s < p - 1$ **do**
3 **for** $k \leftarrow s$ ***to*** p **do**
4 **for** $j \leftarrow k + 1$ ***to*** p **do**
5 $E_{kj} \leftarrow E_{kj} - E_{ks} E_{ss}^\dagger E_{sj}$;
6 $E_k \leftarrow E_k - E_s E_{ss}^\dagger E_{sk}$;
7 $s \leftarrow s + 1$;
8 $M_s \leftarrow E_s E_{ss}^\dagger$;
9 **while** $s > 0$ **do**
10 $M_s \leftarrow (E_s - \sum_{j=s+1}^{p} M_j E_{js}) E_{ss}^\dagger$;
11 $s \leftarrow s - 1$;

Note that the algorithm is quite simple. Its output is constructed from the successive computation of matrices E_{kj} and E_k.

- In line 2, the algorithm updates matrices E_{kj} and E_k similar to those for the particular cases given by (51), (52), (55), (56), (61), (62).
- In line 5, matrix M_p is calculated as in (68).
- In line 7, matrices $M_{p-1}, \ldots, M_1$ are calculated as in (69)–(71).

Importantly, $E_{ss}^\dagger$ is the $q_s \times q_s$ matrix with $q_s < q$, therefore, it requires less computation than $q \times q$ pseudo-inverse $E_{vv}^\dagger$ in (7).

In the MATLAB code below, m, q, p are as above and `Evv.mat`, `Exv.mat` are files containing matrices E_{vv} and E_{xv}, respectively.

Listing 1. MATLAB Code for Solving Equation (10).

```
% Load data
load Evv.mat, Exv.mat  % Evv is q x q, Exv is m x q

% Generate blocks E(:,:,k,j) = E_{j,k} from E_{vv}
qj = q/p;
E = zeros(qj, qj, p, p);
Eorgnl = zeros(qj, qj, p, p);
for k = 1:p
    for j = 1:p
        E(:,:,k,j) = Evv((k-1)*qj+1:k*qj, (j-1)*qj+1:j*qj);
```

 doi:10.20944/preprints202503.1571.v1


```
        Eorgnl(:,:,k,j) = E(:,:,k,j); % Used to compute M_1 below
    end
end

% Generate blocks H(:,:,k) = E_k from Exv
H = zeros(m, qj, p);
Horgnl = zeros(m, qj, p);
for k = 1:p
    H(:,:,k) = Exv(:,(k-1)*qj+1:k*qj);
    Horgnl(:,:,k) = H(:,:,k); % Used to compute M_1 below
end

% Perform algorithm steps
for s = 1:p-1
    for k = s:p
        for j = s+1:p
            pE(:,:,s,s) = pinv(E(:,:,s,s)); % Compute pseudo-inverse

            % Update matrix E_{k,j}
            E(:,:,k,j) = E(:,:,k,j) - E(:,:,k,s) * pE(:,:,s,s) * E(:,:,s,j);

            % Update matrix E_{k}
            H(:,:,k) = H(:,:,k) - H(:,:,s) * pE(:,:,s,s) * E(:,:,s,k);
        end
    end
end

% Compute M(:,:,s) = M_s
M = zeros(m, qj, p);
M(:,:,p) = H(:,:,p) * pinv(E(:,:,p,p));
for s = p-1:-1:2
    Sum = zeros(m, qj);
    for j = s+1:p
        Sum = Sum + M(:,:,j) * E(:,:,j,s);
    end
    M(:,:,s) = (H(:,:,s) - Sum) * pE(:,:,s,s);
end

% Compute M(:,:,1) = M_1 using original blocks
Sum = zeros(m, qj);
for j = 2:p
    Sum = Sum + M(:,:,j) * Eorgnl(:,:,j,1);
end
M(:,:,1) = (Horgnl(:,:,1) - Sum) * pinv(Eorgnl(:,:,1,1));
```

Example 2. *Here, we wish to numerically illustrate the above algorithm and* MATLAB *code. To this end, in equation* (10)*, we consider matrices* $E_{vv} \in \mathbb{R}^{q\times q}$ *and* $E_{xv} \in \mathbb{R}^{m\times q}$ *whose entries are normally distributed with mean* 0 *and variance* 1*. We choose* $m = 1000$, $q = 1000, \ldots, 5000$, $p = 1, 2, 4, 5$ *and* $q_j = q/p$, *for* $j = 1, \ldots, p$.

In Table 2 *and Figure* 1*, for* $p = 1, 2, 4, 5$, *diagrams of the time needed by the above algorithm versus different values of dimension* q *are presented. Recall, the simulations carry out by the Dell Latitude 7400 Business Laptop (manufactured in 2020). The time decreases with the increase in* p*. In other words, the computational load needed for the numerical realization of the proposed* p*-th degree filer decreases with the increase in degree* p*. Note that the proposed* p*-th degree filter requires computation of* p *pseudo-inverse matrices and also matrix multiplications determined by the above algorithm. In particular, for* $q = 5000$ *and* $p = 5$, *the proposed filter is around four times faster than the filter defined by* (7).

 doi:10.20944/preprints202503.1571.v1

Table 2

	Dimension q					
	1000	2000	3000	4000	5000	6000
	Time in seconds					
$p=1$	1	4	14	33	63.5	116
$p=2$	1	2	7	17	35	58
$p=4$	0.7	1.8	4	9	19	37
$p=5$	0.5	1.5	3.7	8	15	28

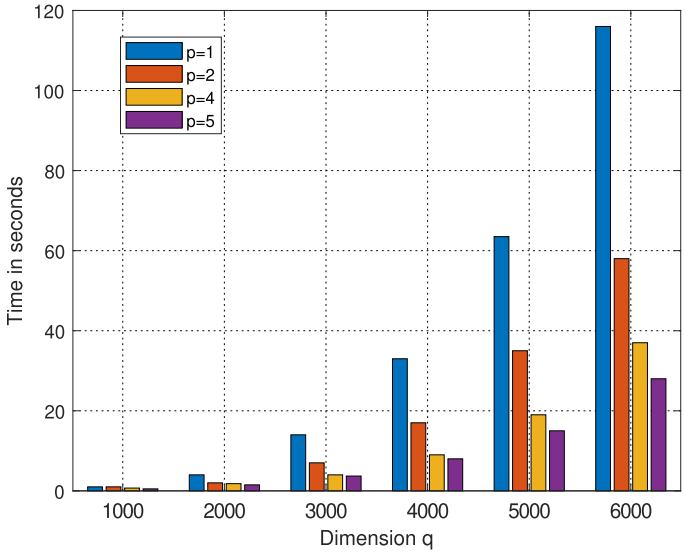

Figure 1. Time needed by the above algorithm, for $p = 1, 2, 4, 5$, versus dimension q.

4.1.8. The Error Associated with the Filter Represented by (4) and (68)–(71)

Let us denote by

$$\varepsilon_p = \min_{M_1,\ldots,M_p} \|\mathbf{x} - [M_1, \ldots, M_p]\mathbf{v}\|_\Omega^2 \tag{72}$$

the error associated with the filter determined by (4) and (68)–(71). We wish to analyze the error.

Theorem 4. *The error associated with the filter determined by* (4) *and* (68)–(71) *is given by*

$$\varepsilon_p = \|E_{xx}^{1/2}\|^2 - \sum_{j=1}^{p} \left\| E_j^{(j-1)} [(E_{jj}^{(j-1)})^{1/2}]^\dagger \right\|^2. \tag{73}$$

Let $\mathbf{v}_p$ be a nonzero vector. Then the error decreases if the degree of the proposed filter p increases, i.e.,

$$\varepsilon_p < \varepsilon_{p-1}, \quad \text{for } p = 2, 3, \ldots. \tag{74}$$

In particular, if $\mathbf{v}_1, \ldots, \mathbf{v}_p$ are such that

$$\sum_{j=1}^{p} \left\| E_j^{(j-1)} [(E_{jj}^{(j-1)})^{1/2}]^\dagger \right\|^2 > \|E_{xy} E_{yy}^{1/2}\|^2 \tag{75}$$

then ε_p is less than ϵ, the error associated with the filter given by (3),

$$\varepsilon_p < \epsilon. \tag{76}$$

Proof. We have

$$\varepsilon_p = \|E_{xx}^{1/2}\|^2 - \mathrm{tr}\{E_{xv} E_{vv}^\dagger E_{vx}\}, \tag{77}$$

 doi:10.20944/preprints202503.1571.v1

where similar to the proofs of (2) and (3), $[\tilde{M}_1 \dots \tilde{M}_p] = E_{xv}E_{vv}^{\dagger}$. Note that, as before, $E_1 = E_1^{(0)}$ and $E_{11} = E_{11}^{(0)}$. Therefore, by (68)–(71),

$$\begin{aligned}
&\mathrm{tr}\{E_{xv}E_{vv}^{\dagger}E_{vx}\}\\
&= \mathrm{tr}\{[\tilde{M}_1 \dots \tilde{M}_p]E_{vx}\}\\
&= \mathrm{tr}\{\tilde{M}_1E_1^T + \dots + \tilde{M}_pE_p^T\}\\
&= \mathrm{tr}\{(E_1E_{11}^{\dagger} - [\tilde{M}_2E_{21} + \dots + \tilde{M}_pE_{p1}]E_{11}^{\dagger})E_1^T\\
&\quad +\tilde{M}_2E_2^T + \dots + \tilde{M}_pE_p^T\}\\
&= \mathrm{tr}\{E_1E_{11}^{\dagger}E_1^T\} + \mathrm{tr}\{\tilde{M}_2(E_2^T - E_{21}E_{11}^{\dagger}E_1^T) + \dots\\
&\quad +\tilde{M}_p(E_p^T - E_{p1}E_{11}^{\dagger}E_1^T)\} + \mathrm{tr}\{[E_2^{(1)}(E_{22}^{(1)})^{\dagger} - \tilde{M}_3E_{32}^{(1)}(E_{22}^{(1)})^{\dagger} -\\
&\quad \dots - \tilde{M}_pE_{p2}^{(1)}(E_{22}^{(1)})^{\dagger}](E_2^{(1)})^T + \dots + \tilde{M}_p(E_p^T - E_{p1}E_{11}^{\dagger}E_1^T)\}\\
&= \dots\\
&= \|E_1E_{11}^{1/2}\|^2 + \|E_2E_{22}^{1/2}\|^2 + \dots + \|E_{p-1}E_{p-1,p-1}^{1/2}\|^2\\
&\quad +\mathrm{tr}\{E_p^{(2)}(E_{pp}^{(2)})^{\dagger}(E_p^{(2)})^T\}\\
&= \|E_1E_{11}^{1/2}\|^2 + \|E_2^{(1)}[(E_{22}^{(1)})^{1/2}]^{\dagger}\|^2 + \dots + \|E_p^{(2)}[(E_{pp}^{(2)})^{1/2}]^{\dagger}\|^2.
\end{aligned} \tag{78}$$

As a result, (77) and (78) imply (73). In turn, (73) implies (74). Inequalities (23) and (76) follow from (28), (77) and (78). ■

Remark 4. *Of course, in practice, the condition in* (23) *can easily be satisfied by a variation of the dimensions of vectors* $\mathbf{v}_1, \dots, \mathbf{v}_p$ *and/or by their choice. The following* (3) *illustrates this observation.*

Corollary 3. *If* $\mathbf{v}_1 = \mathbf{y}$ *and at least one of* $\mathbf{v}_2, \dots, \mathbf{v}_p$ *is not the zero vector then* (76) *is always true.*

Proof. The proof follows directly from (28) and (73). ■

5. Conclusion

In a number of important applications, such as those in biology and medicine, associated random signals are high-dimensional. Therefore, covariance matrices formed from those signals are high-dimensional as well. Large matrices also appear because of the filter structure. As a result, the numerical realization of the filter that processes high-dimensional signals might be very time consuming, and in some cases run out of computer memory. In this paper, we have presented a method of optimal filter determination which targets the case of high-dimensional random signals. In particular, we have developed a procedure that significantly decreases the computational load needed to numerically realize the filter (see (4.1.7) and (2)). The procedure is based on the reduction of the filter equation with large matrices to its equivalent representation by the set of equations with smaller matrices. The associated algorithm has been provided. The algorithm is easy to implement numerically. We have also shown that the filter accuracy is improved with an increase in filter degree, i.e. in the number of filter parameters (see (4)).

The provided results demonstrate that in high-dimensional random signal settings, a number of existing applications may benefit from using the proposed technique. The MATLAB code is given in (4.1.7). In [6], the technique provided here is extended to the case of filtering which is optimal with respect to minimization of both matrices $M_1, \dots, M_p$ and random vectors $\mathbf{v}_1, \dots, \mathbf{v}_p$.

Author Contributions: Conceptualization, P.H, A.T. and P.S.Q.; methodology, P.H. and A.T.; software, A.T. and P.S.Q; validation, P.H. and A.T.; writing – original draft, A.T.; visualization, P.S.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

 doi:10.20944/preprints202503.1571.v1

Acknowledgments: This work was financially supported by *Vicerrectoría de Investigación y Extensión* from *Instituto Tecnológico de Costa Rica* (Research #1440054).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Fomin, V.N.; Ruzhansky, M.V. Abstract Optimal Linear Filtering. *SIAM Journal on Control and Optimization* **2000**, *38*, 1334 – 1352.
2. Hua, Y.; Nikpour, M.; Stoica, P. Optimal reduced-rank estimation and filtering. *IEEE Transactions on Signal Processing* **2001**, *49*, 457–469.
3. Brillinger, D.R. *Time Series: Data Analysis and Theory*; 2001.
4. Torokhti, A.; Howlett, P. An Optimal Filter of the Second Order. *IEEE Transactions on Signal Processing* **2001**, *49*, 1044–048.
5. Scharf, L. The SVD and reduced rank signal processing. *Signal Processing* **1991**, *25*, 113 – 133.
6. Howlett, P.; Torokhti, A.; Soto-Quiros, P. Decrease in computational load and increase in accuracy for filtering of random signals. Part II: Increase in accuracy. *Signal Processing (submitted)*.
7. Leclercq, M.; Vittrant, B.; Martin-Magniette, M.L.; Boyer, M.P.S.; Perin, O.; Bergeron, A.; Fradet, Y.; Droit, A. Large-Scale Automatic Feature Selection for Biomarker Discovery in High-Dimensional OMICs Data. *Frontiers in Genetics* **2019**, *10*, doi: 10.3389/fgene.2019.00452.
8. Artoni, F.; Delorme, A.; Makeig, S. Applying dimension reduction to EEG data by Principal Component Analysis reduces the quality of its subsequent Independent Component decomposition. *Neuroimage* **2018**, *175*, 176 – 187.
9. Bellman, R.E. *Dynamic Programming*, 2 ed.; Courier Corporation, 2003.
10. Stoica, P.; Jansson, M. MIMO system identification: State-space and subspace approximation versus transfer function and instrumental variables. *IEEE Transactions on Signal Processing* **2000**, *48*, 3087– 3099.
11. Billings, S.A. *Nonlinear System Identification - Narmax Methods in the Time, Frequency, and Spatio-temporal Domains*; John Wiley and Sons, Ltd., 2013.
12. Schoukens, M.; Tiels, K. Identification of Block-oriented Nonlinear Systems Starting from Linear Approximations: A Survey. *Automatica* **2017**, *85*, 272 – 292.
13. Howlett, P.G.; Torokhti, A.P.; Pearce, C.E.M. A Philosophy for the Modelling of Realistic Non-linear Systems. *Processing of Amer. Math. Soc.* **2003**, *131*, 353–363.
14. Li, D.; Wong, K.D.; Hu, Y.H.; Sayeed, A.M. Detection, classification, and tracking of targets. *IEEE Signal Processing Mag.* **2002**, *19*, 17 – 29.
15. Mathews, V.J.; Sicuranza, G.L. *Polynomial Signal Processing*; John Willey & Sons, Inc: New York, 2001.
16. Marelli, D.E.; Fu, M. Distributed weighted least-squares estimation with fast convergence for large-scale systems. *Automatica* **2015**, *51*, 27 – 39.
17. Torokhti, A.P.; Howlett, P.G. Best Operator Approximation in Modelling of Nonlinear Systems. *IEEE Trans. CAS. Part I, Fundamental theory and applications* **2002**, *49*, 1792 – 1798.
18. Torokhti, A.; Howlett, P. Best approximation of identity mapping: the case of variable memory. *Journal of Approximation Theory* **2006**, *143*, 111 – 123.
19. Perlovsky, L.; Marzetta, T. Estimating a covariance matrix from incomplete realizations of a random vector. *IEEE Transactions on Signal Processing* **1992**, *40*, 2097–2100.
20. Ledoit, O.; Wolf, M. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis* **2004**, *88*, 365 – 411.
21. Ledoit, O.; Wolf, M. Nonlinear shrinkage estimation of large-dimensional covariance matrices. *The Annals of Statistics* **2012**, *40*, 1024–1060.
22. Vershynin, R. How Close is the Sample Covariance Matrix to the Actual Covariance Matrix? *J. Th. Prob.* **2012**, *25*, 655–686.
23. Joong-Ho Won, Johan Lim, S.J.K.; Rajaratnam, B. Condition-number-regularized covariance estimation. *Journal of the Royal Statistical Society: Series B)* **2013**, *75*, 427–450.
24. Schneider, M.K.; Willsky, A.S. A Krylov Subspace Method for Covariance Approximation and Simulation of Random Processes and Fields. *Multidimensional Systems and Signal Processing* **2003**, *14*, 295–318.
25. Golub, G.; Van Loan, C. *Matrix Computations*, 4 ed.; Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, 2013.

 doi:10.20944/preprints202503.1571.v1

26. Friedland, S.; Torokhti, A. Generalized Rank-Constrained Matrix Approximations. *SIAM Journal on Matrix Analysis and Applications* **2007**, *29*, 656–659.