Department of Industrial Engineering and Management

Technical Report

No. 2013-12

Successive projection method for well-conditioned matrix approximation problems

Mirai Tanaka and Kazuhide Nakata

October, 2013

Tokyo Institute of Technology

2-12-1 Ookayama, Meguro-ku, Tokyo 152-8552, JAPAN
http://www.me.titech.ac.jp/index-e.html

SUCCESSIVE PROJECTION METHOD FOR WELL-CONDITIONED MATRIX APPROXIMATION PROBLEMS

MIRAI TANAKA AND KAZUHIDE NAKATA

Abstract. Matrices are often required to be well-conditioned in a wide variety of areas including signal processing. Problems to find the nearest positive definite matrix or the nearest correlation matrix that simultaneously satisfy the condition number constraint and sign constraints are presented in this paper. Both problems can be regarded as those to find a projection to the intersection of the closed convex cone corresponding to the condition number constraint and the convex polyhedron corresponding to the other constraints. Thus, we can apply a successive projection method, which is a classical algorithm for finding the projection to the intersection of multiple convex sets, to these problems. The numerical results demonstrated that the algorithm effectively solved the problems.

1. Introduction: Well-conditioned matrix approximation

A matrix is called *well-conditioned* if its condition number (the ratio of the largest to the smallest singular value in this paper) is close to unity. Otherwise, it is called *ill-conditioned*. Well-conditioned matrices are often required in many fields of science and engineering, including signal processing and finance, as the following basic example indicates:

Example (Error of perturbed linear equation)**.** Let us see how numerical error affects the solution to a linear equation. Let $\boldsymbol{x}^*$ be the solution to the linear equation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ and $\boldsymbol{x}^* + \Delta\boldsymbol{x}^*$ be the solution to *perturbed* linear equation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} + \Delta\boldsymbol{b}$. This is the case where only $\boldsymbol{b}$ includes numerical error. Since $\boldsymbol{x}^*$ is the solution to the original linear equation, $\Delta\boldsymbol{x}^*$ can be regarded as error in the solution caused by the perturbation. In such a situation, it is known that the relative error is known to be bounded in such situation with the condition number of the coefficient matrix as:

$$\frac{\|\Delta\boldsymbol{x}^*\|_2}{\|\boldsymbol{x}^*\|_2} \leq \mathrm{cond}(\boldsymbol{A}) \frac{\|\Delta\boldsymbol{b}\|_2}{\|\boldsymbol{b}\|_2}.$$

See Horn and Johnson [11, Section 5.8] for analysis for a case where $\boldsymbol{A}$ was also perturbed.

In fact, some researchers [1, 16] on signal processing have studied the maximum likelihood estimators of covariance matrices under a condition number constraint. They have derived analytical solutions to the estimators under the multivariate Gaussian distribution.

We generally cannot always obtain such matrices easily. It is natural to approximate a given ill-conditioned matrix with the nearest well-conditioned one in such cases in view of optimization. Tanaka and Nakata [15] proposed ways of obtaining the nearest well-conditioned positive definite matrix by solving the following optimization problem on space $\mathcal{S}^n$ of n-dimensional symmetric matrices:

$$(1) \qquad \left| \begin{array}{ll} \text{minimize} & \|\boldsymbol{X} - \widehat{\boldsymbol{X}}\| \\ \text{subject to} & \boldsymbol{X} \in \mathcal{S}^n_+, \\ & \mathrm{cond}(\boldsymbol{X}) \leq \kappa, \end{array} \right.$$

where $\widehat{\boldsymbol{X}} \in \mathcal{S}^n$ is a given matrix, $\boldsymbol{X} \in \mathcal{S}^n$ is a decision variable, and $\mathcal{S}^n_+ \subset \mathcal{S}^n$ is the cone of n-dimensional positive semidefinite matrices. Here, $\mathrm{cond}(\boldsymbol{X}) = \sigma_{\max}(\boldsymbol{X})/\sigma_{\min}(\boldsymbol{X})$ is the condition number of $\boldsymbol{X}$, and $\kappa \geq 1$ is a given upper bound for the condition number of $\boldsymbol{X}$. They proved that (1) is solvable in $O(n^3)$ computational time by using some commonly used norms.

We often modify some entries of a covariance matrix whose sign is counter-intuitive in various areas including finance [3, Section 10.2]. However, the sign of X_{ij} has not been taken into consideration in (1). Thus, here we consider the following problem:

$$(2) \qquad \left| \begin{array}{lll} \text{minimize} & \|\boldsymbol{X} - \widehat{\boldsymbol{X}}\| & \\ \text{subject to} & \boldsymbol{X} \in \mathcal{S}^n_+, & \\ & \mathrm{cond}(\boldsymbol{X}) \leq \kappa, & \\ & X_{ij} \geq 0 & ((i,j) \in P), \\ & X_{ij} \leq 0 & ((i,j) \in N), \end{array} \right.$$

where $P, N \subset \{1, \ldots, n\}^2$ are given sets of indices corresponding to the sign constraints.

This problem has a relationship with sparse inverse covariance estimation [4, 5, 8]. Let us consider a multivariate Gaussian distribution with covariance matrix $\boldsymbol{\Sigma}$. It is known that $[\boldsymbol{\Sigma}^{-1}]_{ij} = 0$ if and only if variables i and j are conditionally independent. Thus, if we know set I_0 of pairs of variables that are conditionally independent *a priori*, we can approximate sample covariance matrix $\widehat{\boldsymbol{\Sigma}}$ with positive definite matrix $\boldsymbol{\Sigma}$ that satisfies $[\boldsymbol{\Sigma}^{-1}]_{ij} = 0$ for all $(i, j) \in I_0$ and $\mathrm{cond}(\boldsymbol{\Sigma}) \le c$ by solving (2) with $\widehat{\boldsymbol{X}} = \widehat{\boldsymbol{\Sigma}}^{-1}$, $\kappa = c$, and $P = N = I_0$ and taking $\boldsymbol{\Sigma} = \boldsymbol{X}^{-1}$.

In addition, we also consider the following problem to approximate a correlation matrix:

$$
\left|\begin{array}{lll}
\text{minimize} & \|\boldsymbol{X} - \widehat{\boldsymbol{X}}\| & \\
\text{subject to} & \boldsymbol{X} \in \mathcal{S}^n_+, & \\
 & \mathrm{cond}(\boldsymbol{X}) \le \kappa, & \\
 & \boldsymbol{X}_{ii} = 1 & (i = 1, \dots, n), \\
 & X_{ij} \ge 0 & ((i, j) \in P), \\
 & X_{ij} \le 0 & ((i, j) \in N).
\end{array}\right. \tag{3}
$$

When $P = N = \emptyset$, (3) corresponds to the nearest correlation matrix problem with the condition number constraint. Although the nearest correlation matrix problem has been extensively studied [10, 13], there are few researchers who have considered the condition number of the correlation matrix simultaneously.

Since (2) and (3) can be formulated as a symmetric cone optimization problem, they are solvable in polynomial time with an interior-point method However, it is still difficult to solve large-scale instances at reasonable computational cost.

We propose an efficient algorithm for (2) and (3). We employed the *successive projection method* [2, 7, 9], which is a classical algorithm for finding the projection to the intersection of multiple convex sets. We solve (1) by utilizing the binary search proposed by Tanaka and Nakata [15] as a subroutine in this algorithm.

The remainder of this paper is organized as follows. Section 2 is devoted to introducing the successive projection method for a general problem and after that for (2) and (3) with the Frobenius norm. Section 3 reports the numerical results that demonstrate the effectiveness of our approach. Section 4 provides some concluding remarks.

In what follows, we can assume $(i, i) \notin N$ for all i without loss of generality. The reason for this is that when $(i, i) \in N$ for some i, (2) has no feasible solution other than $\boldsymbol{X} = \boldsymbol{O}$ (we define $\mathrm{cond}(\boldsymbol{O}) = 1$ in this paper) and (3) becomes infeasible.

2. Successive projection method

First, we consider the more general problem below:

$$
\left|\begin{array}{lll}
\text{minimize} & \|\boldsymbol{x} - \widehat{\boldsymbol{x}}\| & \\
\text{subject to} & \boldsymbol{x} \in \mathcal{C}_i & (i = 1, \dots, m),
\end{array}\right. \tag{4}
$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is a decision variable, $\|\cdot\|$ is a norm induced by an inner product on $\mathbb{R}^n$, and $\mathcal{C}_1, \dots, \mathcal{C}_m \subset \mathbb{R}^n$ are given closed convex sets. An optimal solution to (4) can be considered as a projection of $\widehat{\boldsymbol{x}}$ to $\bigcap_{i=1}^m \mathcal{C}_i$.

We can solve (4) with a successive projection method when the computation of each projection to $\mathcal{C}_i$ is easy, *i.e.*, for each i we can obtain an optimal solution to the following problem at modest computational cost:

$$
\left|\begin{array}{ll}
\text{minimize} & \|\boldsymbol{x} - \widehat{\boldsymbol{x}}\| \\
\text{subject to} & \boldsymbol{x} \in \mathcal{C}_i.
\end{array}\right. \tag{5}
$$

In what follows, $P_i(\widehat{\boldsymbol{x}})$ denotes an optimal solution to (5). The successive projection method is a classical algorithm to solve (4) by successively projecting a point to each set. The pseudo-code of this algorithm is quite simple as seen in Algorithm 1. This algorithm was first proposed by Dykstra [7] for a family of closed convex cones. Boyle and Dykstra [2] then extended this algorithm to general closed convex sets in Hilbert space. Thus, this algorithm is also called *Dykstra's algorithm.* Han [9] has given a precise description of the extended algorithm in Euclidean space. The following theorem implies the global convergence of this algorithm.

Algorithm 1 Successive projection method for (4)

$\boldsymbol{x}_m^{(0)} := \widehat{\boldsymbol{x}}$, $\boldsymbol{y}_1^{(0)}, \dots, \boldsymbol{y}_m^{(0)} := \boldsymbol{0}$, and $k = 0$
while $\boldsymbol{x}_m^{(k)}$ is not approximate optimal **do**
  $\boldsymbol{x}_0^{(k+1)} := \boldsymbol{x}_m^{(k)}$
  for $i = 1, \dots, m$ **do**
    $\boldsymbol{x}_i^{(k+1)} := P_i(\boldsymbol{x}_{i-1}^{(k+1)} + \boldsymbol{y}_i^{(k)})$
    $\boldsymbol{y}_i^{(k+1)} := \boldsymbol{x}_{i-1}^{(k+1)} + \boldsymbol{y}_i^{(k)} - \boldsymbol{x}_i^{(k+1)}$
  end for
  $k := k + 1$
end while

Theorem 1 (Han [9, Theorem 4.8]). *Let $\mathcal{C}_1, \dots, \mathcal{C}_p$ be convex polyhedra and $\mathcal{C}_{p+1}, \dots, \mathcal{C}_m$ be closed convex sets such that $(\bigcap_{i=1}^{p} \mathcal{C}_i) \cap (\bigcap_{i=p+1}^{m} \operatorname{int} \mathcal{C}_i) \neq \emptyset$, then sequence $\{\boldsymbol{x}_m^{(k)}\}$ generated by Algorithm 1 converges to an optimal solution to* (4) *as $k \longrightarrow \infty$.*

Next, we will consider applying this algorithm to our problems. Let us define the following closed convex sets for (2):

$$\mathcal{C}_{\text{poly}} = \{\boldsymbol{X} \in \mathcal{S}^n : X_{ij} \geq 0 \ ((i,j) \in P), X_{ij} \leq 0 \ ((i,j) \in N)\},$$
$$\mathcal{C}_{\text{cond}} = \{\boldsymbol{X} \in \mathcal{S}_+^n : \operatorname{cond}(\boldsymbol{X}) \leq \kappa\}$$

and for (3) we replace $\mathcal{C}_{\text{poly}}$ with

$$\mathcal{C}_{\text{poly}} = \{\boldsymbol{X} \in \mathcal{S}^n : X_{ii} = 1 \ (i = 1, \dots, n), X_{ij} \geq 0 \ ((i,j) \in P), X_{ij} \leq 0 \ ((i,j) \in N)\}.$$

For both problems, it is easy to see $\mathcal{C}_{\text{poly}}$ is a closed convex set. The following lemma guarantees that $\mathcal{C}_{\text{cond}}$ is also closed convex set.

Lemma 2. *$\mathcal{C}_{\text{cond}}$ is a closed convex set.*

Proof. First, we will prove closedness. Since we defined $\operatorname{cond}(\boldsymbol{O}) = 1$ in this paper, we can prove that:

$$\mathcal{C}_{\text{cond}} = \{\boldsymbol{X} \in \mathcal{S}_+^n : \lambda_{\max}(\boldsymbol{X}) \leq \kappa \lambda_{\min}(\boldsymbol{X})\},$$

where $\lambda_{\min}(\boldsymbol{X})$ and $\lambda_{\max}(\boldsymbol{X})$ denote the smallest and largest eigenvalues of $\boldsymbol{X}$. By using the continuity of $\lambda_{\min}(\cdot)$ and $\lambda_{\max}(\cdot)$ [11, Theorem 2.4.9.2], we can find closedness.

Next, we will prove convexity. We arbitrarily take $\boldsymbol{A}, \boldsymbol{B} \in \mathcal{C}_{\text{cond}}$ and $\alpha \in [0, 1]$. In addition, let $\boldsymbol{v}_{\min}$ and $\boldsymbol{v}_{\max}$ be eigenvectors of $\boldsymbol{C} = (1-\alpha)\boldsymbol{A} + \alpha\boldsymbol{B}$ with $\|\boldsymbol{v}_{\min}\| = \|\boldsymbol{v}_{\max}\| = 1$ corresponding to $\lambda_{\min}(\boldsymbol{C})$ and $\lambda_{\max}(\boldsymbol{C})$. Then, we obtain the following inequality:

$$\lambda_{\min}(\boldsymbol{C}) = \boldsymbol{v}_{\min}^\top \boldsymbol{C} \boldsymbol{v}_{\min} = (1-\alpha)\boldsymbol{v}_{\min}^\top \boldsymbol{A} \boldsymbol{v}_{\min} + \alpha \boldsymbol{v}_{\min}^\top \boldsymbol{B} \boldsymbol{v}_{\min} \geq (1-\alpha)\lambda_{\min}(\boldsymbol{A}) + \alpha \lambda_{\min}(\boldsymbol{B}).$$

Similarly, we can prove the inequality below:

$$\lambda_{\max}(\boldsymbol{C}) \leq (1-\alpha)\lambda_{\max}(\boldsymbol{A}) + \alpha\lambda_{\max}(\boldsymbol{B}).$$

Since $\lambda_{\max}(\boldsymbol{A}) \leq \kappa\lambda_{\min}(\boldsymbol{A})$ and $\lambda_{\max}(\boldsymbol{B}) \leq \kappa\lambda_{\min}(\boldsymbol{B})$, we find the following inequality:

$$\operatorname{cond}(\boldsymbol{C}) = \frac{\lambda_{\max}(\boldsymbol{C})}{\lambda_{\min}(\boldsymbol{C})} \leq \frac{(1-\alpha)\lambda_{\max}(\boldsymbol{A}) + \alpha\lambda_{\max}(\boldsymbol{B})}{(1-\alpha)\lambda_{\min}(\boldsymbol{A}) + \alpha\lambda_{\min}(\boldsymbol{B})} \leq \kappa,$$

which implies convexity. □

We also define $P_{\text{poly}}(\cdot)$ and $P_{\text{cond}}(\cdot)$ as projectors to the corresponding cone. Since the corresponding optimization problem is separable, the computation of $P_{\text{poly}}(\cdot)$ becomes quite easy as:

$$[P_{\text{poly}}(\boldsymbol{X})]_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise if } (i,j) \in P \text{ with } X_{ij} < 0 \text{ or } (i,j) \in N \text{ with } X_{ij} > 0, \\ X_{ij} & \text{otherwise.} \end{cases}$$

In addition, the computation of $P_{\text{cond}}(\cdot)$ is computed in $O(n^3)$ of computational time with a binary search [15]. Thus, we can efficiently apply the successive projection method to our problem. We present our successive projection method for (2) and (3) in Algorithm 2.

Algorithm 2 Successive projection method for (2) and (3)

$\boldsymbol{Y}_{\text{poly}}^{(0)}, \boldsymbol{Y}_{\text{cond}}^{(0)} := \boldsymbol{O}$, $\boldsymbol{X}_{\text{poly}}^{(1)} := P_{\text{poly}}(\widehat{\boldsymbol{X}})$, and $k = 1$
while $\boldsymbol{X}_{\text{poly}}^{(k)}$ is not approximate optimal **do**
  $\boldsymbol{X}_{\text{cond}}^{(k+1)} := P_{\text{cond}}(\boldsymbol{X}_{\text{poly}}^{(k)} + \boldsymbol{Y}_{\text{cond}}^{(k)})$
  $\boldsymbol{Y}_{\text{cond}}^{(k+1)} := \boldsymbol{X}_{\text{poly}}^{(k)} + \boldsymbol{Y}_{\text{cond}}^{(k)} - \boldsymbol{X}_{\text{cond}}^{(k+1)}$
  $\boldsymbol{X}_{\text{poly}}^{(k+1)} := P_{\text{poly}}(\boldsymbol{X}_{\text{cond}}^{(k+1)} + \boldsymbol{Y}_{\text{poly}}^{(k)})$
  $\boldsymbol{Y}_{\text{poly}}^{(k+1)} := \boldsymbol{X}_{\text{cond}}^{(k+1)} + \boldsymbol{Y}_{\text{poly}}^{(k)} - \boldsymbol{X}_{\text{poly}}^{(k+1)}$
  $k := k + 1$
end while

The following theorem implies global convergence of our algorithm for (2) and (3).

Theorem 3. *A sequence, $\{\boldsymbol{X}_{\text{poly}}^{(k)}\}$, generated by Algorithm 2 converges to an optimal solution to* (2) *and* (3) *as $k \longrightarrow \infty$.*

Proof. The $\mathcal{C}_{\text{poly}}$ is virtually convex polyhedral for each problem. From Lemma 2, we can also see the convexity of $\mathcal{C}_{\text{cond}}$. Moreover, $\boldsymbol{I} \in \mathcal{C}_{\text{poly}} \cap \operatorname{int} \mathcal{C}_{\text{cond}}$ holds. Thus, by using Theorem 1, we can see that sequence $\{\boldsymbol{X}_{\text{poly}}^{(k)}\}$ converges to an optimal solution to each problem. □

Remark. Algorithm 2 can be applied to the following well-conditioned matrix approximation problem with linear equality constraints:

$$\left|\begin{array}{ll} \text{minimize} & \|\boldsymbol{X} - \widehat{\boldsymbol{X}}\| \\ \text{subject to} & \boldsymbol{X} \in \mathcal{S}_+^n, \\ & \operatorname{cond}(\boldsymbol{X}) \leq \kappa, \\ & \mathcal{A}\boldsymbol{X} = \boldsymbol{b}, \end{array}\right.$$

where $\mathcal{A} : \mathcal{S}^n \longrightarrow \mathbb{R}^m$ is a linear mapping and $\boldsymbol{b} \in \mathbb{R}^m$, by defining

$$\mathcal{C}_{\text{poly}} = \{\boldsymbol{X} \in \mathcal{S}^n : \mathcal{A}\boldsymbol{X} = \boldsymbol{b}\}.$$

Letting $\mathcal{A}^\top : \mathbb{R}^m \longrightarrow \mathcal{S}^n$ be the adjoint of $\mathcal{A}$, we can prove that

$$P_{\text{poly}}(\widehat{\boldsymbol{X}}) = \widehat{\boldsymbol{X}} + \mathcal{A}^\top(\mathcal{A}\mathcal{A}^\top)^{-1}(\boldsymbol{b} - \mathcal{A}\widehat{\boldsymbol{X}}).$$

Note that $\mathcal{A}$ does not change at each iteration. Thus, Algorithm 2 works well when $\mathcal{A}\mathcal{A}^\top$ has the following nice structures, *i.e.*, when the number m of equality constraints is small and The closed form of $(\mathcal{A}\mathcal{A}^\top)^{-1}$ is available.

3. Numerical results

We solved multiple instances with our algorithm and an interior-point method and compared the results To verify that it was effective. We implemented our algorithm on MATLAB 7.14. We stopped our algorithm when $\boldsymbol{X}_{\text{poly}}^{(k)} \in \mathcal{S}_+^n$ and $\operatorname{cond}(\boldsymbol{X}_{\text{poly}}^{(k)}) \leq (1 + 10^{-6})\kappa$ hold simultaneously. We modeled (2) and (3) with YALMIP 3 [12] and solved the resulting symmetric cone optimization problem with SeDuMi 1.3 [14] to compare the results with our algorithm with those with an interior-point method.

We generated instances as follows. We sampled every entry of matrix $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ from the uniform distribution on $[-1, +1)$ and generated matrix $\widehat{\boldsymbol{X}} = \boldsymbol{U} + \boldsymbol{U}^\top$. We also set P and N to the set of indices corresponding to the smallest and the largest off-diagonal $2n$ entries of $\widehat{\boldsymbol{X}}$. We also set $\kappa = 10^6$.

The results for small-scale instances of (2) and (3) are listed in Tables 1 and 2. The sizes of instances are shown in "n" columns, the elapsed time in seconds for the interior-point method and the successive projection method are in the "time (IPM) [s]" columns for the former and "time (SPM) [s]" columns for the latter, and the numbers of iterations for the successive projection method are in the "iter (SPM)" columns. We can see the following from these tables: As the size of instances became large, the elapsed time for the interior-point method rapidly increased. However, the elapsed time for our successive projection method gradually increased.

Table 1. Results for small-scale instances of (2)

n	time (IPM) [s]	time (SPM) [s]	iter (SPM)
20	3.5×10^{-1}	5.6×10^{-2}	98
30	$1.1 \times 10^{+0}$	9.7×10^{-2}	133
40	$3.1 \times 10^{+0}$	9.7×10^{-2}	94
50	$9.1 \times 10^{+0}$	1.1×10^{-1}	75
60	$2.6 \times 10^{+1}$	1.3×10^{-1}	79
70	$1.0 \times 10^{+2}$	1.8×10^{-1}	78
80	$3.1 \times 10^{+2}$	1.7×10^{-1}	70
90	$7.4 \times 10^{+2}$	2.3×10^{-1}	69
100	$1.6 \times 10^{+3}$	3.0×10^{-1}	71

Table 2. Results for small-scale instances of (3)

n	time (IPM) [s]	time (SPM) [s]	iter (SPM)
20	3.7×10^{-1}	5.5×10^{-2}	97
30	9.7×10^{-1}	9.9×10^{-2}	130
40	$3.3 \times 10^{+0}$	1.7×10^{-1}	154
50	$8.8 \times 10^{+0}$	2.3×10^{-1}	176
60	$2.7 \times 10^{+1}$	2.9×10^{-1}	186
70	$1.0 \times 10^{+2}$	3.7×10^{-1}	195
80	$3.1 \times 10^{+2}$	4.8×10^{-1}	211
90	$7.9 \times 10^{+2}$	7.1×10^{-1}	219
100	$1.4 \times 10^{+3}$	9.8×10^{-1}	238

We also solved large-scale instances with our algorithm. The results are summarized in Tables 3 and 4. We can see that our algorithm could quickly solve large-scale instances that the interior-point method could not during the modest computational time. In addition, we can see that the size of the matrix for (2) contributed less to the number of iterations, although it certainly did for (3).

Let us loot at how the sequence, $\{\boldsymbol{X}_{\text{poly}}^{(k)}\}$, generated by our algorithm converged to the approximate optimal solution, $\boldsymbol{X}^*$, generated by our successive projection algorithm for the instance of $n = 100$. The behavior of the error norms, $\|\boldsymbol{X}_{\text{poly}}^{(k)} - \boldsymbol{X}^*\|$, are plotted in Figure 1. This figure implies that $\{\boldsymbol{X}_{\text{poly}}^{(k)}\}$ converged to $\boldsymbol{X}^*$ *linearly.* The sequence

TABLE 3. Results for large-scale instances of (2)

n	time (SPM) [s]	iter (SPM)
128	4.3×10^{-1}	60
256	$1.4 \times 10^{+0}$	49
512	$5.1 \times 10^{+0}$	45
1024	$2.4 \times 10^{+1}$	42
2048	$1.5 \times 10^{+2}$	38
4096	$1.1 \times 10^{+3}$	36
8192	$8.4 \times 10^{+3}$	35

TABLE 4. Results for large-scale instances of (3)

n	time (SPM) [s]	iter (SPM)
128	$1.1 \times 10^{+0}$	237
256	$6.0 \times 10^{+0}$	308
512	$4.5 \times 10^{+1}$	402
1024	$2.7 \times 10^{+2}$	481
2048	$2.5 \times 10^{+3}$	654
4096	$2.5 \times 10^{+4}$	826
8192	$2.7 \times 10^{+5}$	1115

generated by our algorithm behaved similarly for the other instances. Similar results (Figure 2) were also obtained for (3).

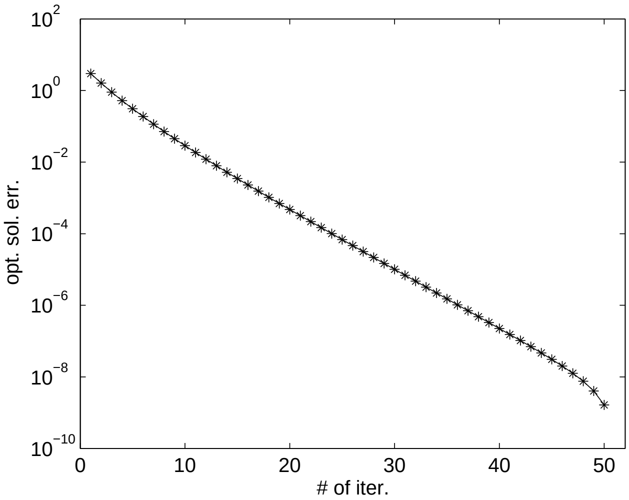

FIGURE 1. Behavior of error norms $\|\boldsymbol{X}^{(k)}_{\text{poly}} - \boldsymbol{X}^*\|_{\text{F}}$ for (2).

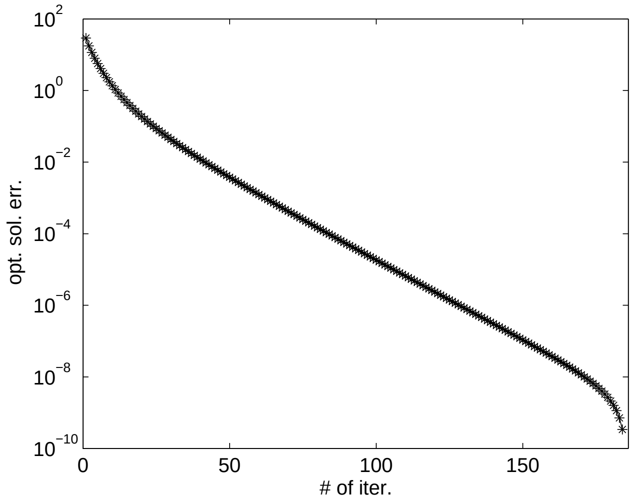

FIGURE 2. Behavior of error norms $\|\boldsymbol{X}^{(k)}_{\text{poly}} - \boldsymbol{X}^*\|_{\text{F}}$ for (3).

4. Concluding remarks

We proposed a successive projection method for well-conditioned positive definite matrix approximation problems in this paper. Our algorithm was based on projections to the closed convex cone of well-conditioned matrices and that to the convex polyhedron of matrices that satisfied sign constraints (and some linear equality constraints). Our algorithm had two main advantages. The first is simplicity, which enabled us easily implement our successive projection method. The second advantage was *practical* linear convergence. Our numerical results suggested linear convergence with our algorithm.

We were not able to prove linear convergence with our algorithm in this research. Higham [10] also mentioned that the successive projection method (Algorithm 1) linearly converges to an optimal solution *at best* for the nearest correlation matrix problem, which is a simpler one than our problems. Deutsch and Hundal [6] proved linear convergence when the convex sets were all subspaces. However, a necessary and sufficient condition to linear convergence by the algorithm has never been known. A *worst-case* analysis of the algorithm should be studied. Our future work also includes extending to the use of different norms. The confidence of entries in an estimator may not be uniform to approximate a covariance matrix. In such cases weighted norms like $\|\boldsymbol{H} \circ \boldsymbol{X}\|_{\text{F}} = \sqrt{\sum_{i,j} H_{ij}^2 X_{ij}^2}$ should be used instead of the (unweighted) Frobenius norm $\|\boldsymbol{X}\|_{\text{F}} = \sqrt{\sum_{i,j} X_{ij}^2}$. However, weighted norms are generally not unitary similarity invariant. Hence, $P_{\text{cond}}(\cdot)$ may not be able to be computed easily to use of them since we cannot simply use a binary search [15].

References

[1] A. Aubry, A. De Maio, L. Pallotta, and A. Farina. Maximum likelihood estimation of a structured covariance matrix with a condition number constraint. *IEEE Trans. Signal Process.,* 60:3004–3021, 2012.

[2] J.P. Boyle and R.L. Dykstra. A method for finding projections onto the intersection of convex sets in Hilbert spaces. in *Advances in Order Restricted Statistical Inference,* R. Dykstra, T. Robertson, and F.T. Wright (eds.), 1985, pp. 28–47.

[3] G. Cornuejols and R. Tütüncü. *Optimization Methods in Finance.* Cambridge University Press, 2007.
[4] A. d'Aspremont, O. Banerjee, and L. El Ghaoui. First-order methods for sparse covariance selection. *SIAM J. Matrix Anal. and Appl.,* 30:56–66, 2008.
[5] A.P. Dempster. Covariance selection. *Biometrics,* 28:157–175, 1972.
[6] F. Deutsch and H. Hundal. The rate of convergence for the method of alternating projections II. *J. of Math. Anal. and Appl.,* 205:381–405, 1997.
[7] R.L. Dykstra. An algorithm for restricted least squares regressions. *J. Amer. Stat. Assoc.,* 78:837–842, 1983.
[8] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostat.,* 9:432–441, 2008.
[9] S.-P. Han. A successive projection method. *Math. Prog.,* 40:1–14, 1988.
[10] N.J. Higham. Computing the nearest correlation matrix—a problem from finance. *IMA J. of Numer. Anal.,* 22:392–343, 2002.
[11] R.A. Horn and C.R. Johnson. *Matrix Analysis (Second Edition).* Cambridge University Press, 2012.
[12] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. in *Proc. of the CACSD Conf.,* 2004, pp. 284–289.
[13] H. Qi and D. Sun. A quadratically convergent newton method for computing the nearest correlation matrix. *SIAM J. Matrix Anal. and Appl.,* 28:360–385, 2006.
[14] J.F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optim. Methods Softw.,* 11:625–653, 1999.
[15] M. Tanaka and K. Nakata. Positive definite matrix approximation with condition number constraint. *Optim. Lett.,* 2013, DOI: 10.1007/s11590-013-0632-7.
[16] J.H. Won and S.-J. Kim. Maximum likelihood covariance estimation with a condition number constraint. in *Proc. 40th Asilomar Conf. Signals, Systems, and Computers,* 2006, pp. 1445–1449.

(M. Tanaka) Graduate School of Decision Science and Technology, Tokyo Institute of Technology, 2-12-1-W9-60, Ookayama, Meguro-ku, Tokyo, 152-8552, JAPAN
E-mail address: `tanaka.m.aa@m.titech.ac.jp`

(K. Nakata) Graduate School of Decision Science and Technology, Tokyo Institute of Technology, 2-12-1-W9-60, Ookayama, Meguro-ku, Tokyo, 152-8552, JAPAN
E-mail address: `nakata.k.ac@m.titech.ac.jp`