SMOOTH OPTIMIZATION APPROACH FOR SPARSE COVARIANCE SELECTION[*]

ZHAOSONG LU[†]

Abstract. In this paper we first study a smooth optimization approach for solving a class of nonsmooth *strictly* concave maximization problems whose objective functions admit smooth convex minimization reformulations. In particular, we apply Nesterov's smooth optimization technique [Y. E. Nesterov, *Dokl. Akad. Nauk SSSR*, 269 (1983), pp. 543–547; Y. E. Nesterov, *Math. Programming*, 103 (2005), pp. 127–152] to their dual counterparts that are smooth convex problems. It is shown that the resulting approach has $\mathcal{O}(1/\sqrt{\epsilon})$ iteration complexity for finding an ϵ-optimal solution to both primal and dual problems. We then discuss the application of this approach to sparse covariance selection that is approximately solved as an l_1-norm penalized maximum likelihood estimation problem, and also propose a variant of this approach which has substantially outperformed the latter one in our computational experiments. We finally compare the performance of these approaches with other first-order methods, namely, Nesterov's $\mathcal{O}(1/\epsilon)$ smooth approximation scheme and block-coordinate descent method studied in [A. d'Aspremont, O. Banerjee, and L. El Ghaoui, *SIAM J. Matrix Anal. Appl.*, 30 (2008), pp. 56–66; J. Friedman, T. Hastie, and R. Tibshirani, *Biostatistics*, 9 (2008), pp. 432–441] for sparse covariance selection on a set of randomly generated instances. It shows that our smooth optimization approach substantially outperforms the first method above, and moreover, its variant substantially outperforms both methods above.

Key words. sparse covariance selection, nonsmooth strictly concave maximization, smooth minimization

AMS subject classifications. 90C22, 90C25, 90C47, 65K05, 62J10

DOI. 10.1137/070695915

1. Introduction. In [19, 21], Nesterov proposed an efficient smooth optimization method for solving convex programming problems of the form

$$(1) \qquad \min\{f(u) : u \in U\},$$

where f is a convex function with Lipschitz continuous gradient, and U is a closed convex set. It is shown that his method has $\mathcal{O}(1/\sqrt{\epsilon})$ iteration complexity bound, where $\epsilon > 0$ is the absolute precision of the final objective function value. A proximal-point-type algorithm for (1) having the same complexity as above has also been proposed more recently by Auslender and Teboulle [2].

Motivated by [10], we are particularly interested in studying the use of a smooth optimization approach for solving a class of nonsmooth *strictly* concave maximization problems whose objective functions admit smooth convex minimization reformulations in this paper. Our key idea is to apply Nesterov's smooth optimization technique [19, 21] to their dual counterparts that are smooth convex problems. It is shown that the resulting approach has $\mathcal{O}(1/\sqrt{\epsilon})$ iteration complexity for finding an ϵ-optimal solution to both primal and dual problems.

One interesting application of the above approach is for sparse covariance selection. Given a set of random variables with Gaussian distribution for which the

---

true covariance matrix is unknown, covariance selection is a procedure used to esti-
mate true covariance from a sample covariance matrix by maximizing its likelihood
while imposing a certain sparsity on the inverse of the covariance estimation (e.g., see
[11]). Therefore, it can be applied to determine a robust estimate of the true vari-
ance matrix, and simultaneously to discover the sparse structure in the underlying
model. Despite its popularity in numerous real-world applications (e.g., see [3, 10, 25]
and the references therein), sparse covariance selection itself is a challenging NP-hard
combinatorial optimization problem. By an argument that is often used in regres-
sion techniques such as LASSO [23], Yuan and Lin [25] and d'Aspremont et al. [10]
(see also [3]) showed that it can be approximately solved as an l_1-norm penalized
maximum likelihood estimation problem. Moreover, the authors of [10] studied two
efficient first-order methods for solving this problem, that is, Nesterov's smooth ap-
proximation scheme and block-coordinate descent method. It was shown in [10] that
their first method has $\mathcal{O}(1/\epsilon)$ iteration complexity for finding an ϵ-optimal solution.
For their second method, each iterate requires solving a box constrained quadratic
programming, and it has a local linear convergence rate. However, its global iteration
complexity for finding an ϵ-optimal solution is theoretically unknown. After the first
release of our paper, Friedman, Hastie, and Tibshirani [16] studied a slight variant
of the block-coordinate descent method proposed in [10]. At each iteration of their
method, a coordinate descent approach is applied to solve a lasso (l_1-regularized)
least-squares problem, which is the dual of the box constrained quadratic program-
ming appearing in the block-coordinate descent method [10]. In contrast with these
methods, the smooth optimization approach proposed in this paper has a more at-
tractive iteration complexity that is $\mathcal{O}(1/\sqrt{\epsilon})$ for finding an ϵ-optimal solution. In
addition, we propose a variant of the smooth optimization approach which has sub-
stantially outperformed the latter one in our computational experiments. We also
compare the performance of our approaches with their methods for sparse covariance
selection on a set of randomly generated instances. It shows that our smooth opti-
mization approach substantially outperforms their first method above (i.e., Nesterov's
smooth approximation scheme) and, moreover, its variant substantially outperforms
their methods [10, 16] mentioned above.

The paper is organized as follows. In section 2, we introduce a class of nonsmooth
concave maximization problems in which we are interested and propose a smooth op-
timization approach to them. In section 3, we briefly introduce sparse covariance
selection and show that it can be approximately solved as an l_1-norm penalized max-
imum likelihood estimation problem. We also discuss the application of the smooth
optimization approach for solving this problem and propose a variant of this approach.
In section 4, we compare the performance of our smooth optimization approach and
its variant with two other first-order methods studied in [10, 16] for sparse covari-
ance selection on a set of randomly generated instances. Finally, we present some
concluding remarks in section 5.

1.1. Notation. In this paper, all vector spaces are assumed to be finite-dimen-
sional. The space of symmetric $n \times n$ matrices will be denoted by $\mathcal{S}^n$. If $X \in \mathcal{S}^n$ is
positive semidefinite, we write $X \succeq 0$. Also, we write $X \preceq Y$ to mean $Y - X \succeq 0$.
The cone of positive semidefinite (resp., definite) matrices is denoted by $\mathcal{S}^n_+$ (resp.,
$\mathcal{S}^n_{++}$). Given matrices X and Y in $\Re^{p \times q}$, the standard inner product is defined
by $\langle X, Y \rangle := \mathrm{Tr}(XY^T)$, where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. $\| \cdot \|$ denotes
the Euclidean norm and its associated operator norm unless it is explicitly stated
otherwise. The Frobenius norm of a real matrix X is defined as $\|X\|_F := \sqrt{\mathrm{Tr}(XX^T)}$.

We denote by e the vector of all ones, and by I the identity matrix. Their dimensions should be clear from the context. For a real matrix X, we denote by $\text{Card}(X)$ the cardinality of X, that is, the number of nonzero entries of X, and denote by $|X|$ the absolute value of X, that is, $|X|_{ij} = |X_{ij}|$ for all i, j. The determinant and the minimal (resp., maximal) eigenvalue of a real symmetric matrix X are denoted by $\det X$ and $\lambda_{\min}(X)$ (resp., $\lambda_{\max}(X)$), respectively. For an n-dimensional vector w, $\text{diag}(w)$ denotes the diagonal matrix whose ith diagonal element is w_i for $i = 1, \ldots, n$. We denote by $\mathcal{Z}_+$ the set of all nonnegative integers.

Let the space $\mathcal{F}$ be endowed with an arbitrary norm $\| \cdot \|$. The dual space of $\mathcal{F}$, denoted by $\mathcal{F}^*$, is the normed real vector space consisting of all linear functionals of $s : \mathcal{F} \to \Re$, endowed with the dual norm $\| \cdot \|^*$ defined as

$$\|s\|^* := \max_u \{\langle s, u \rangle : \|u\| \leq 1\} \quad \forall s \in \mathcal{F}^*,$$

where $\langle s, u \rangle := s(u)$ is the value of the linear functional s at u. Finally, given an operator $\mathcal{A} : \mathcal{F} \to \mathcal{F}^*$, we define

$$\mathcal{A}[H, H] := \langle \mathcal{A}H, H \rangle$$

for any $H \in \mathcal{F}$.

2. Smooth optimization approach. In this section, we consider a class of concave nonsmooth maximization problems:

$$(2) \qquad \max_{x \in X} \{g(x) := \min_{u \in U} \phi(x, u)\},$$

where X and U are nonempty convex compact sets in finite-dimensional real vector spaces $\mathcal{E}$ and $\mathcal{F}$, respectively, and $\phi(x, u) : X \times U \to \Re$ is a continuous function which is *strictly* concave in $x \in X$ for every fixed $u \in U$, and convex differentiable in $u \in U$ for every fixed $x \in X$. Therefore, for any $u \in U$, the function

$$(3) \qquad f(u) := \max_{x \in X} \phi(x, u)$$

is well-defined. We also easily conclude that $f(u)$ is convex differentiable on U, and its gradient is given by

$$(4) \qquad \nabla f(u) = \nabla_u \phi(x(u), u) \quad \forall u \in U,$$

where $x(u)$ denotes the unique solution of (3).

Let the space $\mathcal{F}$ be endowed with an arbitrary norm $\| \cdot \|$. We further assume that $\nabla f(u)$ is Lipschitz continuous on U with respect to $\| \cdot \|$, i.e., there exists some $L > 0$ such that

$$\|\nabla f(u) - \nabla f(\tilde{u})\|^* \leq L \|u - \tilde{u}\| \quad \forall u, \tilde{u} \in U.$$

Under the above assumptions, we easily observe that (i) problem (2) and its dual, that is,

$$(5) \qquad \min_u \{f(u) : u \in U\},$$

are both solvable and have the same optimal value; and (ii) the dual problem (5) can be suitably solved by Nesterov's smooth minimization approach [19, 21].

Denote by $d(u)$ a prox-function of the set U. We assume that $d(u)$ is continuous and strongly convex on U with modulus $\sigma > 0$. Let u_0 be the center of the set U defined as

$$(6) \qquad\qquad u_0 = \arg\min\{d(u) : \ u \in U\}.$$

Without loss of generality assume that $d(u_0) = 0$. We now describe Nesterov's smooth minimization approach [19, 21] for solving the dual problem (5), and we will show that it simultaneously solves the nonsmooth concave maximization problem (2).

SMOOTH MINIMIZATION ALGORITHM.

Let $u_0 \in U$ be given in (6). Set $x_{-1} = 0$ and $k = 0$.

(1) Compute $\nabla f(u_k)$ and $x(u_k)$. Set $x_k = \frac{k}{k+2}x_{k-1} + \frac{2}{k+2}x(u_k)$.

(2) Find $u_k^{sd} \in \operatorname{Argmin}\left\{\langle \nabla f(u_k), u - u_k\rangle + \frac{L}{2}\|u - u_k\|^2 : \ u \in U\right\}$.

(3) Find $u_k^{ag} = \operatorname{argmin}\left\{\frac{L}{\sigma}d(u) + \sum_{i=0}^{k}\frac{i+1}{2}[f(u_i) + \langle\nabla f(u_i), u - u_i\rangle] : \ u \in U\right\}$.

(4) Set $u_{k+1} = \frac{2}{k+3}u_k^{ag} + \frac{k+1}{k+3}u_k^{sd}$.

(5) Set $k \leftarrow k + 1$ and go to step 1.

end

The following property of the above algorithm is established in Theorem 2 of Nesterov [21].

THEOREM 2.1.   *Let the sequence $\{(u_k, u_k^{sd})\}_{k=0}^{\infty} \subseteq U \times U$ be generated by the smooth minimization algorithm. Then for any $k \geq 0$ we have*

$$(7)$$
$$\frac{(k+1)(k+2)}{4}f(u_k^{sd}) \leq \min\left\{\frac{L}{\sigma}d(u) + \sum_{i=0}^{k}\frac{i+1}{2}[f(u_i) + \langle\nabla f(u_i), u - u_i\rangle] : \ u \in U\right\}.$$

We are ready to establish the main convergence result of the smooth minimization algorithm for solving the nonsmooth concave maximization problem (2) and its dual (5). Its proof is a generalization of the one given in a more special context in [21].

THEOREM 2.2.   *After k iterations, the smooth minimization algorithm generates a pair of approximate solutions (u_k^{sd}, x_k) to problem (2) and its dual (5), respectively, which satisfy the following inequality:*

$$(8) \qquad\qquad 0 \leq f(u_k^{sd}) - g(x_k) \leq \frac{4LD}{\sigma(k+1)(k+2)}.$$

Thus if the termination criterion $f(u_k^{sd}) - g(x_k) \leq \epsilon$ is applied, the iteration complexity of finding an ϵ-optimal solution to problem (2) and its dual (5) by the smooth minimization algorithm does not exceed $2\sqrt{LD/(\sigma\epsilon)}$, where

$$(9) \qquad\qquad D = \max\{d(u) : \ u \in U\}.$$

Proof. In view of (3), (4), and the notation $x(u)$, we have

$$(10) \qquad f(u_i) + \langle\nabla f(u_i), u - u_i\rangle = \phi(x(u_i), u_i) + \langle\nabla_u\phi(x(u_i), u_i), u - u_i\rangle.$$

Invoking the fact that the function $\phi(x, \cdot)$ is convex on U for every fixed $x \in X$, we obtain

$$(11) \qquad\qquad \phi(x(u_i), u_i) + \langle\nabla_u\phi(x(u_i), u_i), u - u_i\rangle \leq \phi(x(u_i), u).$$

Notice that $x_{-1} = 0$, and $x_k = \frac{k}{k+2}x_{k-1} + \frac{2}{k+2}x(u_k)$ for any $k \geq 0$, which imply

$$(12) \qquad x_k = \sum_{i=0}^{k} \frac{2(i+1)}{(k+1)(k+2)}x(u_i).$$

Using (10), (11), (12), and the fact that the function $\phi(\cdot, u)$ is concave on X for every fixed $u \in U$, we have

$$\sum_{i=0}^{k}(i+1)[f(u_i) + \langle \nabla f(u_i), u - u_i \rangle] \leq \sum_{i=0}^{k}(i+1)\phi(x(u_i), u)$$

$$\leq \frac{1}{2}(k+1)(k+2)\phi(x_k, u)$$

for all $u \in U$. It follows from this relation, (7), (9), and (2) that

$$f(u_k^{sd}) \leq \frac{4LD}{\sigma(k+1)(k+2)}$$

$$+ \min_u \left\{ \sum_{i=0}^{k} \frac{2(i+1)}{(k+1)(k+2)}[f(u_i) + \langle \nabla f(u_i), u - u_i \rangle] : \ u \in U \right\}$$

$$\leq \frac{4LD}{\sigma(k+1)(k+2)} + \min_{u \in U} \phi(x_k, u) = \frac{4LD}{\sigma(k+1)(k+2)} + g(x_k),$$

and hence the inequality (8) holds. The remaining conclusion directly follows from (8). □

Remark. We shall mention that Nesterov [20] developed the excessive gap technique for solving problem (2) and its dual (5) in a special context, which enjoys the same iteration complexity as the smooth minimization algorithm described above. In addition, it is not hard to observe that the technique proposed in [20] can be extended to solve problem (2) and its dual (5) in the aforementioned general framework, provided that the subproblem

$$(13) \qquad \min_{u \in U} \phi(x, u) + \mu d(u)$$

can be suitably solved for any given $\mu > 0$ and $x \in X$. The computation of each iterate of Nesterov's excessive gap technique [20] is similar to that of the smooth minimization algorithm except that the former method requires solving a prox subproblem in the form of (13), but the latter one needs to solve the prox subproblem described in step 3 above. When the function $\phi(x, \cdot)$ is affine for every fixed $x \in X$, these two prox subproblems have the same form, and thus the computational cost of Nesterov's excessive gap technique [20] is almost the same as that of the smooth minimization algorithm; however, for a more general function $\phi(\cdot, \cdot)$, the computational cost of the former method can be more expensive than that of the latter method.

The following results will be used to develop a variant of the smooth minimization algorithm for sparse covariance selection in subsection 3.4.

LEMMA 2.3. *Problem* (2) *has a unique optimal solution, denoted by x^*. Moreover, for any $u^* \in \mathrm{Argmin}\{f(u) : \ u \in U\}$, we have*

$$(14) \qquad x^* = \arg\max_{x \in X} \phi(x, u^*).$$

Proof. We clearly know that problem (2) has an optimal solution. To prove its uniqueness, it suffices to show that $g(x)$ is strictly concave on X. Indeed, since $X \times U$ is a convex compact set and $\phi(x, u)$ is continuous on $X \times U$, it follows that for any $t \in (0, 1)$, $x^1 \neq x^2 \in X$, there exist some $\tilde{u} \in U$ such that

$$\phi(tx^1 + (1 - t)x^2, \tilde{u}) = \min_{u \in U} \phi(tx^1 + (1 - t)x^2, u).$$

Recall that $\phi(\cdot, u)$ is strictly concave on X for every fixed $u \in U$. Therefore, we have

$$\phi(tx^1 + (1 - t)x^2, \tilde{u}) > t\phi(x^1, \tilde{u}) + (1 - t)\phi(x^2, \tilde{u}),$$
$$\geq t \min_{u \in U} \phi(x^1, u) + (1 - t) \min_{u \in U} \phi(x^2, u),$$

which together with (2) implies that

$$g(tx^1 + (1 - t)x^2) > tg(x^1) + (1 - t)g(x^2)$$

for any $t \in (0, 1)$, $x^1 \neq x^2 \in X$, and hence $g(x)$ is strictly concave on X as desired.

Note that x^* is the optimal solution of problem (2). We clearly know that for any $u^* \in \mathrm{Argmin}\{f(u) : u \in U\}$, (u^*, x^*) is a saddle point for problem (2), that is,

$$\phi(x^*, u) \geq \phi(x^*, u^*) \geq \phi(x, u^*) \quad \forall (x, u) \in X \times U,$$

and hence we have

$$x^* \in \mathrm{Arg} \max_{x \in X} \phi(x, u^*).$$

This, together with the fact that $\phi(\cdot, u^*)$ is strictly concave on X, immediately yields (14). $\quad\square$

THEOREM 2.4. *Let x^* be the unique optimal solution of* (2), *and let f^* be the optimal value of problems* (2) *and* (5). *Assume that the sequences $\{u_k\}_{k=0}^{\infty}$ and $\{x(u_k)\}_{k=0}^{\infty}$ are generated by the smooth minimization algorithm. Then the following statements hold:*

(1) $f(u_k) \to f^*$, $x(u_k) \to x^*$ *as $k \to \infty$;*
(2) $f(u_k) - g(x(u_k)) \to 0$ *as $k \to \infty$.*

Proof. Recall from the smooth minimization algorithm that

$$u_{k+1} = \left(2u_k^{ag} + (k + 1)u_k^{sd}\right) / (k + 3) \quad \forall k \geq 0.$$

Since $u_k^{sd}, u_k^{ag} \in U$ for all $k \geq 0$, and U is a compact set, we have $u_{k+1} - u_k^{sd} \to 0$ as $k \to \infty$. Notice that $f(u)$ is continuous on the compact set U, and hence it is uniformly continuous on U. Then we further have $f(u_{k+1}) - f(u_k^{sd}) \to 0$ as $k \to \infty$. Also, it follows from Theorem 2.2 that $f(u_k^{sd}) \to f^*$ as $k \to \infty$. Therefore, we conclude that $f(u_k) \to f^*$ as $k \to \infty$.

Note that X is a compact set, and $x(u_k) \subseteq X$ for all $k \geq 0$. To prove that $x(u_k) \to x^*$ as $k \to \infty$, it suffices to show that every convergent subsequence of $\{x(u_k)\}_{k=0}^{\infty}$ converges to x^* as $k \to \infty$. Indeed, assume that $\{x(u_{n_k})\}_{k=0}^{\infty}$ is an arbitrary convergent subsequence, and $x(u_{n_k}) \to \tilde{x}^*$ as $k \to \infty$ for some $\tilde{x}^* \in X$. Without loss of generality, assume that the sequence $\{u_{n_k}\}_{k=0}^{\infty} \to \tilde{u}^*$ as $k \to \infty$ for some $\tilde{u}^* \in U$ (otherwise, one can consider any convergent subsequence of $\{u_{n_k}\}_{k=0}^{\infty}$). Using the result that $f(u_k) \to f^*$, we obtain that

$$\phi\left(x(u_{n_k}), u_{n_k}\right) = f(u_{n_k}) \to f^* \quad \text{as } k \to \infty.$$

Upon letting $k \to \infty$ and using the continuity of $\phi(\cdot, \cdot)$, we have $\phi(\tilde{x}^*, \tilde{u}^*) = f(\tilde{u}^*) = f^*$. Hence, it follows that

$$\tilde{u}^* \in \text{Arg} \min_{u \in U} f(u), \quad \tilde{x}^* = \arg \max_{x \in X} \phi(x, \tilde{u}^*),$$

which together with Lemma 2.3 implies that $\tilde{x}^* = x^*$. Hence, as desired, $x(u_{n_k}) \to x^*$ as $k \to \infty$.

As shown in Lemma 2.3, the function $g(x)$ is continuous on X. This result, together with statement 1, immediately implies that statement 2 holds.   □

3. Sparse covariance selection. In this section, we discuss the application of the smooth optimization approach proposed in section 2 to sparse covariance selection. More specifically, we briefly introduce sparse covariance selection in subsection 3.1 and show that it can be approximately solved as an l_1-norm penalized maximum likelihood estimation problem in subsection 3.2. In subsection 3.3, we address some implementation details of the smooth optimization approach for solving this problem and propose a variant of this approach in subsection 3.4.

3.1. Introduction of sparse covariance selection. In this subsection, we briefly introduce sparse covariance selection. For more details, see d'Aspremont, Banerjee, and El Ghaoui [10] and the references therein.

Given n variables with a Gaussian distribution $\mathcal{N}(0, C)$ for which the true covariance matrix C is unknown, we are interested in estimating C from a sample covariance matrix Σ by maximizing its likelihood while imposing a certain number of components in the inverse of the estimation of C to zero. This problem is commonly known as *sparse covariance selection* (see [11]). Since zeros in the inverse of covariance matrix correspond to conditional independence in the model, sparse covariance selection can be used to determine a robust estimate of the covariance matrix, and simultaneously discover the sparse structure in the underlying graphical model.

Several approaches have been proposed for sparse covariance selection in literature. For example, Bilmes [4] proposed a method based on choosing statistical dependencies according to conditional mutual information computed from training data. The recent works [18, 12] involve identifying the Gaussian graphical models that are best supported by the data and any available prior information on the covariance matrix. Given a sample covariance matrix $\Sigma \in \mathcal{S}_+^n$, d'Aspremont et al. [10] recently formulated sparse covariance selection as the following estimation problem:

$$\text{(15)} \qquad \begin{aligned} \max_X \quad & \log \det X - \langle \Sigma, X \rangle - \rho \text{Card}(X) \\ \text{s.t.} \quad & \tilde{\alpha} I \preceq X \preceq \tilde{\beta} I, \end{aligned}$$

where $\rho > 0$ is a parameter controlling the trade-off between likelihood and cardinality, and $0 \le \tilde{\alpha} < \tilde{\beta} \le \infty$ are the fixed bounds on the eigenvalues of the solution. For some specific choices of ρ, the formulation (15) has been used for model selection in [1, 5], and applied to speech recognition and gene network analysis (see [4, 13]).

Note that the estimation problem (15) itself is an NP-hard combinatorial problem because of the penalty term $\text{Card}(X)$. To overcome the computational difficulty, d'Aspremont et al. [10] used an argument that is often used in regression techniques (e.g., see [23, 6, 14]), where sparsity of the solution is concerned, to relax $\text{Card}(X)$ to $e^T |X| e$, and obtained the following l_1-norm penalized maximum likelihood estimation

problem:

$$\max_X \quad \log \det X - \langle \Sigma, X \rangle - \rho e^T |X| e$$

(16)

$$\text{s.t.} \quad \tilde{\alpha} I \preceq X \preceq \tilde{\beta} I.$$

Recently, Yuan and Lin [25] proposed a similar estimation problem for sparse covariance selection given as follows:

$$\max_X \quad \log \det X - \langle \Sigma, X \rangle - \rho \sum_{i \neq j} |X_{ij}|$$

(17)

$$\text{s.t.} \quad \tilde{\alpha} I \preceq X \preceq \tilde{\beta} I,$$

with $\tilde{\alpha} = 0$ and $\tilde{\beta} = \infty$. They showed that problem (17) can be suitably solved by the interior point algorithm developed in Vandenberghe, Boyd, and Wu [24]. A few other approaches have also been studied for sparse covariance selection by solving some related maximum likelihood estimation problems in the literature. For example, Huang, Liu, and Pourahmadi [17] proposed an iterative (heuristic) algorithm to minimize a nonconvex penalized likelihood. Dahl et al. [8, 7] applied Newton's method, the coordinate steepest descent method, and the conjugate gradient method for the problems for which the conditional independence structure is partially known.

As shown in d'Aspremont et al. [10] (see also [3]) and Yuan and Lin [25], the l_1-norm penalized maximum likelihood estimation problems (16) and (17) are capable of discovering effectively the sparse structure or, equivalently, the conditional independence in the underlying graphical model. Also, it is not hard to see that the estimation problem (17) becomes a special case of problem (16) if replacing Σ by $\Sigma + \rho I$ in (17). For these reasons, in the remainder of this paper we focus on problem (16) only.

3.2. Nonsmooth strictly concave maximization reformulation. In this subsection, we show that problem (16) can be reformulated as a nonsmooth strictly concave maximization problem of the form (2).

Recall from subsection 3.1 that $\Sigma \in \mathcal{S}_+^n$, and keep in mind that the notation $|\cdot|$, $\|\cdot\|$, and $\|\cdot\|_F$ is defined in subsection 1.1. We first provide some tighter bounds on the optimal solution of problem (16) for the case where $\tilde{\alpha} = 0$ and $\tilde{\beta} = \infty$.

PROPOSITION 3.1. *Assume that $\tilde{\alpha} = 0$ and $\tilde{\beta} = \infty$. Let $X^* \in \mathcal{S}_{++}^n$ be the unique optimal solution of problem (16). Then we have $\alpha I \preceq X^* \preceq \beta I$, where*

(18)

$$\alpha = \frac{1}{\|\Sigma\| + n\rho}, \quad \beta = \min\left\{ \frac{n - \alpha \text{Tr}(\Sigma)}{\rho}, \eta \right\}$$

with

$$\eta = \begin{cases} \min\left\{ e^T |\Sigma^{-1}| e, (n - \rho\sqrt{n}\alpha)\|\Sigma^{-1}\| - (n-1)\alpha \right\} & \text{if } \Sigma \text{ is invertible;} \\ 2e^T |(\Sigma + \frac{\rho}{2} I)^{-1}| e - \text{Tr}((\Sigma + \frac{\rho}{2} I)^{-1}) & \text{otherwise.} \end{cases}$$

Proof. Let

(19)

$$\mathcal{U} := \{ U \in \mathcal{S}^n : |U_{ij}| \leq 1 \quad \forall ij \}$$

and

(20)

$$L(X, U) = \log \det X - \langle \Sigma + \rho U, X \rangle \quad \forall (X, U) \in \mathcal{S}_{++}^n \times \mathcal{U}.$$

Note that $X^* \in \mathcal{S}_{++}^n$ is the optimal solution of problem (16). It can be easily shown that there exist some $U^* \in \mathcal{U}$ such that (X^*, U^*) is a saddle point of $L(\cdot, \cdot)$ on $\mathcal{S}_{++}^n \times \mathcal{U}$, that is,

$$X^* = \arg \min_{X \in \mathcal{S}_{++}^n} L(X, U^*), \quad U^* \in \text{Arg} \min_{U \in \mathcal{U}} L(X^*, U).$$

The above relations along with (19) and (20) immediately yield

$$(21) \qquad X^*(\Sigma + \rho U^*) = I, \quad \langle X^*, U^* \rangle = e^T |X^*| e.$$

Hence, we have

$$X^* = (\Sigma + \rho U^*)^{-1} \succeq \frac{1}{\|\Sigma\| + \rho\|U^*\|} I,$$

which together with (19) and the fact $U^* \in \mathcal{U}$ implies that $X^* \succeq \frac{1}{\|\Sigma\|+n\rho} I$. Thus, as desired, $X^* \succeq \alpha I$, where α is given in (18).

We next bound X^* from above. In view of (21), we have

$$(22) \qquad \langle X^*, \Sigma \rangle + \rho e^T |X^*| e = n,$$

which together with the relation $X^* \succeq \alpha I$ implies that

$$(23) \qquad e^T |X^*| e \leq \frac{n - \alpha \text{Tr}(\Sigma)}{\rho}.$$

Now let $X(t) := (\Sigma + t\rho I)^{-1}$ for $t \in (0, 1)$. By concavity of $\log \det(\cdot)$, one can easily see that $X(t)$ maximizes the function $\log \det(\cdot) - \langle \Sigma + t\rho I, \cdot \rangle$ over $\mathcal{S}_{++}^n$. Using this observation and the definition of X^*, we can have

$$\log \det X^* - \langle \Sigma + t\rho I, X^* \rangle \leq \log \det X(t) - \langle \Sigma + t\rho I, X(t) \rangle,$$

$$\log \det X(t) - \langle \Sigma, X(t) \rangle - \rho e^T |X(t)| e \leq \log \det X^* - \langle \Sigma, X^* \rangle - \rho e^T |X^*| e.$$

Adding the above two inequalities upon some algebraic simplification, we obtain that

$$e^T |X^*| e - t\text{Tr}(X^*) \leq e^T |X(t)| e - t\text{Tr}(X(t)),$$

and hence

$$(24) \qquad e^T |X^*| e \leq \frac{e^T |X(t)| e - t\text{Tr}(X(t))}{1 - t} \quad \forall t \in (0, 1).$$

If Σ is invertible, upon letting $t \downarrow 0$ on both sides of (24), we have

$$e^T |X^*| e \leq e^T |\Sigma^{-1}| e.$$

Otherwise, letting $t = 1/2$ in (24), we obtain

$$e^T |X^*| e \leq 2e^T \left| \left(\Sigma + \frac{\rho}{2} I \right)^{-1} \right| e - \text{Tr} \left(\left(\Sigma + \frac{\rho}{2} I \right)^{-1} \right).$$

Combining the above two inequalities and (23), we have

$$(25) \qquad \|X^*\| \leq \|X^*\|_F \leq e^T |X^*| e \leq \min \left\{ \frac{n - \alpha \text{Tr}(\Sigma)}{\rho}, \gamma \right\},$$

where

$$\gamma = \begin{cases} e^T|\Sigma^{-1}|e & \text{if } \Sigma \text{ is invertible;} \\ 2e^T|(\Sigma + \frac{\rho}{2}I)^{-1}|e - \text{Tr}((\Sigma + \frac{\rho}{2}I)^{-1}) & \text{otherwise.} \end{cases}$$

Further, using the relation $X^* \succeq \alpha I$, we obtain that

$$e^T|X^*|e \geq \|X^*\|_F \geq \sqrt{n}\alpha,$$

which together with (22) implies that

$$\text{Tr}(X^*\Sigma) \leq n - \rho\sqrt{n}\alpha.$$

This inequality, along with the relation $X^* \succeq \alpha I$, yields

$$\lambda_{\min}(\Sigma)((n-1)\alpha + \|X^*\|) \leq \text{Tr}(X^*\Sigma) \leq n - \rho\sqrt{n}\alpha.$$

Hence if Σ is invertible, we further have

$$\|X^*\| \leq (n - \rho\sqrt{n}\alpha)\|\Sigma^{-1}\| - (n-1)\alpha.$$

This together with (25) implies that $X^* \preceq \beta I$, where β is given in (18).   □

   Remark. Some bounds on X^* were also derived in d'Aspremont, Banerjee, and El Ghaoui [10]. In contrast with their bounds, our bounds given in (18) are tighter. Moreover, our approach for deriving the above bounds can be generalized to handle the case where $\tilde{\alpha} > 0$ and $\tilde{\beta} = \infty$, but their approach cannot. Indeed, if $\tilde{\alpha} > 0$ and $\tilde{\beta} = \infty$, we can set $\alpha = \tilde{\alpha}$ and replace the above $X(t)$ by the optimal solution of

$$\max_X \quad \log\det X - \langle \Sigma + \rho I, X \rangle$$
$$\text{s.t.} \quad \tilde{\alpha}I \preceq X,$$

which has a closed-form expression. By following a similar derivation as above, one can obtain a positive scalar β such that $X^* \preceq \beta I$. In addition, for the case where $\tilde{\alpha} = 0$ and $0 < \tilde{\beta} < \infty$, one can set $\beta = \tilde{\beta}$ and easily show that $X^* \succeq \alpha I$, where $\alpha = \beta e^{-\beta(\text{Tr}(\Sigma) + n\rho)}$.

   From the above discussion, we conclude that problem (16) is equivalent to the following problem:

$$\begin{aligned} &\max_X \quad \log\det X - \langle \Sigma, X \rangle - \rho e^T|X|e \\ (26) \quad &\text{s.t.} \quad \alpha I \preceq X \preceq \beta I \end{aligned}$$

for some $0 < \alpha < \beta < \infty$.

   We further observe that problem (26) can be rewritten as

$$(27) \quad \max_{X \in \mathcal{X}} \min_{U \in \mathcal{U}} \ \log\det X - \langle \Sigma + \rho U, X \rangle,$$

where $\mathcal{U}$ is defined in (19), and $\mathcal{X}$ is defined as follows:

$$(28) \quad \mathcal{X} := \{X \in \mathcal{S}^n : \ \alpha I \preceq X \preceq \beta I\}.$$

Therefore, we conclude that problem (16) is equivalent to (27). For the remainder of the paper, we will focus on problem (27) only.

3.3. Smooth optimization method for sparse covariance selection. In this subsection, we describe the implementation details of the smooth minimization algorithm proposed in section 2 for solving problem (27). We also compare the complexity of this algorithm with interior point methods, and two other first-order methods studied in d'Aspremont et al. [10], that is, Nesterov's smooth approximation scheme and block-coordinate descent method.

We first observe that the sets $\mathcal{X}$ and $\mathcal{U}$ both lie in the space $\mathcal{S}^n$, where $\mathcal{X}$ and $\mathcal{U}$ are defined in (28) and (19), respectively. Let $\mathcal{S}^n$ be endowed with the Frobenius norm, and let $\tilde{d}(X) = \log \det X$ for $X \in \mathcal{X}$. Then for any $X \in \mathcal{X}$, we have

$$\nabla^2 \tilde{d}(X)[H, H] = -\text{Tr}(X^{-1}HX^{-1}H) \leq -\beta^{-2}\|H\|_F^2$$

for all $H \in \mathcal{S}^n$, and hence, $\tilde{d}(X)$ is strongly concave on $\mathcal{X}$ with modulus β^{-2}. Using this result and Theorem 1 of [21], we immediately conclude that $\nabla f(U)$ is Lipschitz continuous with constant $L = \rho^2\beta^2$ on $\mathcal{U}$, where

$$(29) \qquad f(U) := \max_{X \in \mathcal{X}} \, \log \det X - \langle \Sigma + \rho U, X \rangle \quad \forall U \in \mathcal{U}.$$

Denote the unique optimal solution of problem (29) by $X(U)$. For any $U \in \mathcal{U}$, we can compute $X(U)$, $f(U)$, and $\nabla f(U)$ as follows.

Let $\Sigma + \rho U = Q\text{diag}(\gamma)Q^T$ be an eigenvalue decomposition of $\Sigma + \rho U$ such that $QQ^T = I$. For $i = 1, \ldots, n$, let

$$\lambda_i = \begin{cases} \min\{\max\{1/\gamma_i, \alpha\}, \beta\} & \text{if } \gamma_i > 0; \\ \beta & \text{otherwise.} \end{cases}$$

It is not hard to show that

$$(30) \qquad X(U) = Q\text{diag}(\lambda)Q^T, \quad f(U) = -\gamma^T\lambda + \sum_{i=1}^n \log \lambda_i, \quad \nabla f(U) = -\rho X(U).$$

From the above discussion, we see that problem (27) has exactly the same form as (2) and also satisfies all assumptions imposed on problem (2). Therefore, it can be suitably solved by the smooth minimization algorithm proposed in section 2. The implementation details of this algorithm for problem (27) are described as follows.

Given $U_0 \in \mathcal{U}$, let $d(U) = \|U - U_0\|_F^2/2$ be the proximal function on $\mathcal{U}$, which is strongly convex function with modulus $\sigma = 1$. For our specific choice of the norm and $d(U)$, we clearly see that steps 2 and 3 of the smooth minimization algorithm can be solved as a problem of the form

$$V = \arg\min_{U \in \mathcal{U}} \langle G, U \rangle + \|U\|_F^2/2$$

for some $G \in \mathcal{S}^n$. In view of (19), we see that

$$V_{ij} = \max\{\min\{-G_{ij}, 1\}, -1\}, \quad i, j = 1, \ldots, n.$$

In addition, for any $X \in \mathcal{X}$, we define

$$(31) \qquad g(X) := \log \det X - \langle \Sigma, X \rangle - \rho e^T |X| e.$$

For the ease of comparison with its latter variant, we now present a complete version of the aforementioned smooth minimization algorithm for solving problem (27) and its dual.

SMOOTH MINIMIZATION ALGORITHM FOR COVARIANCE SELECTION (SMACS).
Let $\epsilon > 0$ and $U_0 \in \mathcal{U}$ be given. Set $X_{-1} = 0$, $L = \rho^2 \beta^2$, $\sigma = 1$, and $k = 0$.
(1) Compute $\nabla f(U_k)$ and $X(U_k)$. Set $X_k = \frac{k}{k+2} X_{k-1} + \frac{2}{k+2} X(U_k)$.
(2) Find $U_k^{sd} = \text{argmin} \left\{ \langle \nabla f(U_k), U - U_k \rangle + \frac{L}{2} \|U - U_k\|_F^2 : U \in \mathcal{U} \right\}$.
(3) Find $U_k^{ag} = \text{argmin} \{ \frac{L}{2\sigma} \|U - U_0\|_F^2 + \sum_{i=0}^{k} \frac{i+1}{2} [f(U_i) + \langle \nabla f(U_i), U - U_i \rangle] : U \in \mathcal{U} \}$.
(4) Set $U_{k+1} = \frac{2}{k+3} U_k^{ag} + \frac{k+1}{k+3} U_k^{sd}$.
(5) Set $k \leftarrow k + 1$. Go to step 1 until $f(U_k^{sd}) - g(X_k) \leq \epsilon$.
end

The iteration complexity of the above algorithm for solving problem (27) is established in the following theorem.

THEOREM 3.2. *The iteration complexity performed by the algorithm SMACS for finding an ϵ-optimal solution to problem* (27) *and its dual does not exceed $\sqrt{2}\rho\beta$* $\max_{U \in \mathcal{U}} \|U - U_0\|_F / \sqrt{\epsilon}$, *and moreover, if $U_0 = 0$, it does not exceed $\sqrt{2}\rho\beta n / \sqrt{\epsilon}$.*

Proof. From the above discussion, we know that $L = \rho^2 \beta^2$, $D = \max_{U \in \mathcal{U}} \|U - U_0\|_F^2 / 2$, and $\sigma = 1$, which together with Theorem 2.2 immediately implies that the first part of the statement holds. Further, if $U_0 = 0$, we easily obtain from (19) that $D = \max_{U \in \mathcal{U}} \|U\|_F^2 / 2 = n^2 / 2$. The second part of the statement directly follows from this result and Theorem 2.2. $\square$

Remark. By the definition of $\mathcal{U}$ (see (19)), we can easily show that $\min_{U_0 \in \mathcal{U}} \max_{U \in \mathcal{U}} \|U - U_0\|_F$ has a unique minimizer $U_0 = 0$. This result together with Theorem 3.2 implies that the initial point $U_0 = 0$ gives the optimal worst-case iteration complexity for the algorithm SMACS.

Alternatively, d'Aspremont et al. [10] applied Nesterov's smooth approximation scheme [21] to solve problem (27). More specifically, let $\epsilon > 0$ be the desired accuracy, and let

$$\hat{d}(U) = \|U\|_F^2 / 2, \quad \hat{D} = \max_{U \in \mathcal{U}} \hat{d}(U) = n^2 / 2.$$

As shown in [21], the nonsmooth function $g(X)$ defined in (31) is uniformly approximated by the smooth function

$$g_\epsilon(X) = \min_{U \in \mathcal{U}} \log \det X - \langle \Sigma + \rho U, X \rangle - \frac{\epsilon}{2\hat{D}} \hat{d}(U)$$

on $\mathcal{X}$ with the error at most by $\epsilon / 2$, and, moreover, the function $g_\epsilon(X)$ has a Lipschitz continuous gradient on $\mathcal{X}$ with some constant $L(\epsilon) > 0$. Nesterov's smooth optimization technique [19, 21] is then applied to solve the perturbed problem $\max_{X \in \mathcal{X}} g_\epsilon(X)$, and problem (27) is accordingly solved. It was shown in [10] that the iteration complexity of this approach for finding an ϵ-optimal solution to problem (27) does not exceed

$$(32) \qquad \frac{2\sqrt{2}\rho\beta n^{1.5} \log \kappa}{\epsilon} + \kappa \sqrt{\frac{n \log \kappa}{\epsilon}},$$

where $\kappa := \beta / \alpha$.

In view of (32) and Theorem 3.2, we conclude that the smooth optimization approach improves upon Nesterov's smooth approximation scheme at least by a factor of $\mathcal{O}(\sqrt{n} \log \kappa / \sqrt{\epsilon})$ in terms of the iteration complexity for solving problem (27). Moreover, the computational cost per iteration of the former approach is at least as cheap as that of the latter one.

d'Aspremont et al. [10] also studied a block-coordinate descent method for solving problem (16) with $\tilde{\alpha} = 0$ and $\tilde{\beta} = \infty$. Each iterate of this method requires computing the inverse of an $(n-1) \times (n-1)$ matrix and solving a box constrained quadratic programming with $n-1$ variables. As mentioned in section 3 of [10], this method has a local linear convergence rate. However, its global iteration complexity for finding an ϵ-optimal solution is theoretically unknown. Moreover, this method is not suitable for solving problem (16) with $\tilde{\alpha} > 0$ or $\tilde{\beta} < \infty$.

In addition, we observe that problem (26) (also (16)) can be reformulated as a constrained smooth convex problem that has an explicit $\mathcal{O}(n^2)$-logarithmically homogeneous self-concordant barrier function. Thus, it can be suitably solved by interior point methods (see Nesterov and Nemirovskii [22] and Vandenberghe, Boyd, and Wu [24]). The worst-case iteration complexity of interior point methods for finding an ϵ-optimal solution to (26) is $\mathcal{O}(n \log(\epsilon_0/\epsilon))$, where ϵ_0 is an initial gap. Each iterate of interior point methods requires $\mathcal{O}(n^6)$ arithmetic cost for assembling and solving a typically dense Newton system with $\mathcal{O}(n^2)$ variables. Thus, the total worst-case arithmetic cost of interior point methods for finding an ϵ-optimal solution to (26) is $\mathcal{O}(n^7 \log(\epsilon_0/\epsilon))$. In contrast to interior point methods, the algorithm SMACS requires $\mathcal{O}(n^3)$ arithmetic cost per iteration dominated by eigenvalue decomposition and matrix multiplication of $n \times n$ matrices. Based on this observation and Theorem 3.2, we conclude that the overall worst-case arithmetic cost of the algorithm SMACS for finding an ϵ-optimal solution to (26) is $\mathcal{O}(\rho\beta n^4/\sqrt{\epsilon})$, which is substantially superior to that of interior point methods, provided that $\rho\beta$ is not too large and ϵ is not too small.

3.4. Variant of the smooth minimization algorithm. As discussed in subsection 3.3, the algorithm SMACS has a nice theoretical complexity in contrast with interior point methods, Nesterov's smooth approximation scheme, and the block-coordinate descent method. However, its practical performance is still not very attractive (see section 4). To enhance the computational performance, we propose a variant of the algorithm SMACS for solving problem (27) in this subsection.

Our first concern of the algorithm SMACS is that the eigenvalue decomposition of two $n \times n$ matrices is required per iteration. Indeed, the eigenvalue decomposition of $\Sigma + \rho U_k$ and $\Sigma + \rho U_k^{sd}$ is needed at steps 1 and 5 to compute $\nabla f(U_k)$ and $f(U_k^{sd})$, respectively. We also know that the eigenvalue decomposition is one of major computations for the algorithm SMACS. To reduce the computational cost, we now propose a new termination criterion other than $f(U_k^{sd}) - g(X_k) \le \epsilon$ that is used in the algorithm SMACS. In view of Theorem 2.4, we know that

$$f(U_k) - g(X(U_k)) \to 0 \text{ as } k \to \infty.$$

Thus, $f(U_k) - g(X(U_k)) \le \epsilon$ can be used as an alternative termination criterion. Moreover, it follows from (30) that the quantity $f(U_k) - g(X(U_k))$ is readily available in step 1 of the algorithm SMACS with almost no additional cost. We easily see that the algorithm SMACS with this new termination criterion would require only one eigenvalue decomposition per iteration. Despite this clear advantage, we shall mention that the iteration complexity of the resulting algorithm is unfortunately unknown. Nevertheless, in practice we have found that the number of iterations performed by the algorithm SMACS with the above two different termination criteria are almost same. Thus, $f(u_k) - g(x(u_k)) \le \epsilon$ is a useful practical termination criterion.

For sparse covariance selection, the penalty parameter ρ is usually small, but the parameter β can be fairly large. In view of Theorem 3.2, we know that the iteration

complexity of the algorithm SMACS for solving problem (27) is proportional to β. Therefore, when β is too large, the complexity and practical performance of this algorithm become unattractive. To overcome this drawback, we will propose one strategy to dynamically update β.

Let X^* be the unique optimal solution of problem (27). For any $\hat{\beta} \in [\lambda_{\max}(X^*), \beta]$, we easily observe that X^* is also the unique optimal solution to the following problem:

$$(33) \qquad (P_{\hat{\beta}}) \qquad\qquad \max_{X \in \mathcal{X}_{\hat{\beta}}} \min_{U \in \mathcal{U}} \ \log \det X - \langle \Sigma + \rho U, X \rangle,$$

where $\mathcal{U}$ is defined in (19) and $\mathcal{X}_{\hat{\beta}}$ is given by

$$\mathcal{X}_{\hat{\beta}} := \{X : \ \alpha I \preceq X \preceq \hat{\beta} I\}.$$

In view of Theorem 3.2, the iteration complexity of the algorithm SMACS for problem (33) is lower than that for problem (27), provided $\hat{\beta} \in [\lambda_{\max}(X^*), \beta)$. Hence, ideally we set $\hat{\beta} = \lambda_{\max}(X^*)$, which would give the lowest iteration complexity, but unfortunately $\lambda_{\max}(X^*)$ is unknown. However, we can generate a sequence $\{\hat{\beta}_k\}_{k=0}^{\infty}$ that asymptotically approaches $\lambda_{\max}(X^*)$ as the algorithm progresses. Indeed, in view of Theorem 2.4, we know that $X(U_k) \to X^*$ as $k \to \infty$, and we obtain that

$$\lambda_{\max}(X(U_k)) \to \lambda_{\max}(X^*) \text{ as } k \to \infty.$$

Therefore, we see that $\{\lambda_{\max}(X(U_k))\}_{k=0}^{\infty}$ can be used to generate a sequence $\{\hat{\beta}_k\}_{k=0}^{\infty}$ that asymptotically approaches $\lambda_{\max}(X^*)$. We next propose a strategy to generate such a sequence $\{\hat{\beta}_k\}_{k=0}^{\infty}$.

For convenience of presentation, we introduce some new notation. Given any $U \in \mathcal{U}$ and $\hat{\beta} \in [\alpha, \beta]$, we define

$$(34) \qquad\qquad X_{\hat{\beta}}(U) := \arg \max_{X \in \mathcal{X}_{\hat{\beta}}} \ \log \det X - \langle \Sigma + \rho U, X \rangle,$$

$$(35) \qquad\qquad f_{\hat{\beta}}(U) := \max_{X \in \mathcal{X}_{\hat{\beta}}} \ \log \det X - \langle \Sigma + \rho U, X \rangle.$$

DEFINITION 1. *Given any $U \in \mathcal{U}$ and $\hat{\beta} \in [\alpha, \beta]$, $X_{\hat{\beta}}(U)$ is called "active" if $\lambda_{\max}(X_{\hat{\beta}}(U)) = \hat{\beta}$ and $\hat{\beta} < \beta$; otherwise it is called "inactive."*

Let $\varsigma_1, \varsigma_2 > 1$, and $\varsigma_3 \in (0, 1)$ be given and fixed. Assume that $U_k \in \mathcal{U}$ and $\hat{\beta}_k \in [\alpha, \beta]$ are given at the beginning of the kth iteration for some $k \geq 0$. We now describe the strategy for generating the next iterate U_{k+1} and $\hat{\beta}_{k+1}$ by considering the following three different cases:

(1) If $X_{\hat{\beta}_k}(U_k)$ is active, find the smallest $s \in \mathcal{Z}_+$ such that $X_{\bar{\beta}}(U_k)$ is inactive, where $\bar{\beta} = \min\{\varsigma_1^s \hat{\beta}_k, \beta\}$. Set $\hat{\beta}_{k+1} = \bar{\beta}$, and apply the algorithm SMACS for problem $(P_{\hat{\beta}_{k+1}})$ starting with the point U_k and set its next iterate to be U_{k+1}.

(2) If $X_{\hat{\beta}_k}(U_k)$ is inactive and $\lambda_{\max}(X_{\hat{\beta}_k}(U_k)) \leq \varsigma_3 \hat{\beta}_k$, set $\hat{\beta}_{k+1} = \max\{\min\{\varsigma_2 \lambda_{\max}(X_{\hat{\beta}_k}(U_k)), \beta\}, \alpha\}$. Apply the algorithm SMACS for problem $(P_{\hat{\beta}_{k+1}})$ starting with the point U_k, and set its next iterate to be U_{k+1}.

(3) If $X_{\hat{\beta}_k}(U_k)$ is inactive and $\lambda_{\max}(X_{\hat{\beta}_k}(U_k)) > \varsigma_3 \hat{\beta}_k$, set $\hat{\beta}_{k+1} = \hat{\beta}_k$. Continue the algorithm SMACS for problem $(P_{\hat{\beta}_k})$, and set its next iterate to be U_{k+1}.

For the sequences $\{U_k\}_{k=0}^{\infty}$ and $\{\hat{\beta}_k\}_{k=0}^{\infty}$ recursively generated above, we observe that the sequence $\{X_{\hat{\beta}_{k+1}}(U_k)\}_{k=0}^{\infty}$ is always inactive. This together with (34), (35), (29), and the fact that $\hat{\beta}_k \leq \beta$ for $k \geq 0$ implies that

$$(36) \qquad f(U_k) = f_{\hat{\beta}_{k+1}}(U_k), \quad \nabla f(U_k) = \nabla f_{\hat{\beta}_{k+1}}(U_k) \;\; \forall k \geq 0.$$

Therefore, the new termination criterion $f(U_k) - g(X(U_k)) \leq \epsilon$ can be replaced by

$$(37) \qquad f_{\hat{\beta}_{k+1}}(U_k) - g(X_{\hat{\beta}_{k+1}}(U_k)) \leq \epsilon$$

accordingly.

We now incorporate into the algorithm SMACS the new termination criterion (37) and the aforementioned strategy for generating a sequence $\{\hat{\beta}_k\}_{k=0}^{\infty}$ that asymptotically approaches $\lambda_{\max}(X^*)$, and we obtain a variant of the algorithm SMACS for solving problem (27). For convenience of presentation, we omit the subscript k from $\hat{\beta}_k$.

VARIANT OF THE SMOOTH MINIMIZATION ALGORITHM FOR COVARIANCE SELECTION (VSMACS).

Let $\epsilon > 0$, $\varsigma_1, \varsigma_2 > 1$, and $\varsigma_3 \in (0,1)$ be given. Choose a $U_0 \in \mathcal{U}$. Set $\hat{\beta} = \beta$, $L = \rho^2 \beta^2$, $\sigma = 1$, and $k = 0$.

(1) Compute $X_{\hat{\beta}}(U_k)$ according to (30).

    (1a) If $X_{\hat{\beta}}(U_k)$ is active, find the smallest $s \in \mathcal{Z}_+$ such that $X_{\bar{\beta}}(U_k)$ is inactive, where $\bar{\beta} = \min\{\varsigma_1^s \hat{\beta}, \beta\}$. Set $k = 0$, $U_0 = U_k$, $\hat{\beta} = \bar{\beta}$, $L = \rho^2 \hat{\beta}^2$, and go to step 2.

    (1b) If $X_{\hat{\beta}}(U_k)$ is inactive and $\lambda_{\max}(X_{\hat{\beta}}(U_k)) \leq \varsigma_3 \hat{\beta}$, set $k = 0$, $U_0 = U_k$, $\hat{\beta} = \max\{\min\{\varsigma_2 \lambda_{\max}(X_{\hat{\beta}}(U_k)), \beta\}, \alpha\}$, and $L = \rho^2 \hat{\beta}^2$.

(2) If $f_{\hat{\beta}}(U_k) - g(X_{\hat{\beta}}(U_k)) \leq \epsilon$, terminate. Otherwise, compute $\nabla f_{\hat{\beta}}(U_k)$ according to (30).

(3) Find $U_k^{sd} = \operatorname{argmin}\{\langle \nabla f_{\hat{\beta}}(U_k), U - U_k \rangle + \frac{L}{2}\|U - U_k\|_F^2 : \; U \in \mathcal{U}\}$.

(4) Find $U_k^{ag} = \operatorname{argmin}\{\frac{L}{2\sigma}\|U - U_0\|_F^2 + \sum_{i=0}^{k} \frac{i+1}{2}[f_{\hat{\beta}}(U_i) + \langle \nabla f_{\hat{\beta}}(U_i), U - U_i \rangle] : U \in \mathcal{U}\}$.

(5) Set $U_{k+1} = \frac{2}{k+3}U_k^{ag} + \frac{k+1}{k+3}U_k^{sd}$.

(6) Set $k \leftarrow k + 1$, and go to step 1.

end

We next establish some preliminary convergence properties of the above algorithm.

PROPOSITION 3.3. *For the algorithm VSMACS, the following properties hold:*

(1) *Suppose that the algorithm VSMACS terminates at some iterate $(X_{\hat{\beta}}(U_k), U_k)$. Then $(X_{\hat{\beta}}(U_k), U_k)$ is an ϵ-optimal solution to problem (27) and its dual.*

(2) *Suppose that $\hat{\beta}$ is updated only for a finite number of times. Then the algorithm VSMACS terminates in a finite number of iterations and produces an ϵ-optimal solution to problem (27) and its dual.*

Proof. For the final iterate $(X_{\hat{\beta}}(U_k), U_k)$, we clearly know that $f_{\hat{\beta}}(U_k) - g(X_{\hat{\beta}}(U_k)) \leq \epsilon$, and $X_{\hat{\beta}}(U_k)$ is inactive. As shown in (36), $f(U_k) = f_{\hat{\beta}}(U_k)$. Hence, we have $f(U_k) - g(X_{\hat{\beta}}(U_k)) \leq \epsilon$. We also know that $U_k \in \mathcal{U}$, and $X_{\hat{\beta}}(U_k) \in \mathcal{X}$ due to $\hat{\beta} \in [\alpha, \beta]$. Thus, statement 1 immediately follows. After the last update of $\hat{\beta}$, the algorithm VSMACS behaves exactly like the algorithm SMACS as applied to solve problem $(P_{\hat{\beta}})$ except with the termination criterion $f(U_k) - g(X_{\hat{\beta}}(U_k)) \leq \epsilon$. Thus, it

follows from statement 1 and Theorem 2.4 that statement 2 holds. Thus, it follows from statement 1 and Theorem 2.4 that statement 2 holds.   ☐

4. Computational results. In this section, we compare the performance of the smooth minimization approach and its variant proposed in this paper with other first-order methods studied in [10, 16], that is, Nesterov's smooth approximation scheme and block-coordinate descent method for solving problem (16) (or, equivalently, (27)) on a set of randomly generated instances.

All instances used in this section were randomly generated in the same manner as described in d'Aspremont et al. [10]. First, we generate a sparse invertible matrix $A \in \mathcal{S}^n$ with positive diagonal entries and a density prescribed by ϱ. We then generate the matrix $B \in \mathcal{S}^n$ by

$$B = A^{-1} + \tau V,$$

where $V \in \mathcal{S}^n$ is an independent and identically distributed uniform random matrix, and τ is a small positive number. Finally, we obtain the following randomly generated sample covariance matrix:

$$\Sigma = B - \min\{\lambda_{\min}(B) - \vartheta, 0\}I,$$

where ϑ is a small positive number. In particular, we set $\varrho = 0.01$, $\tau = 0.15$, and $\vartheta = 1.0e - 4$ for generating all instances.

As discussed in section 3.3, our smooth minimization approach has much better worst-case iteration complexity than Nesterov's smooth approximation scheme studied in d'Aspremont et al. [10] for problem (27). However, it is unknown how their practical performance differs from each other. In the first experiment, we compare the practical performance of our smooth minimization approach and its variant with Nesterov's smooth approximation scheme studied in d'Aspremont et al. [10] for problem (27) with $\alpha = 0.1$, $\beta = 10$, and $\rho = 0.5$. For convenience of presentation, we label these three first-order methods as SM, VSM, and NSA, respectively. The codes for them are written in MATLAB. More specifically, the code for NSA follows the algorithm presented in d'Aspremont et al. [10], and the codes for SM and VSM are written in accordance with the algorithms SMACS and VSMACS, respectively. Moreover, we set $\varsigma_1 = \varsigma_2 = 1.05$ and $\varsigma_3 = 0.95$ for the algorithm VSMACS. These three methods terminate once the duality gap is less than $\epsilon = 0.1$. All computations are performed on an Intel Xeon 2.66 GHz machine with Red Hat Linux version 8.

The performance of the methods NSA, SM, and VSM for the randomly generated instances are presented in Table 1. The row size n of each sample covariance matrix Σ is given in column one. The numbers of iterations of NSA, SM, and VSM are given in columns two to four, the objective function values are given in columns five to seven, and the CPU times (in seconds) are given in the last three columns, respectively. From Table 1, we conclude that (i) the method SM, namely, the smooth minimization approach, outperforms substantially the method NSA, that is, Nesterov's smooth approximation scheme; and (ii) the method VSM, namely, the variant of the smooth minimization approach, substantially outperforms the other two methods. In addition, we see from this experiment that Nesterov's smooth minimization approach [19] is generally more appealing than his smooth approximation scheme [21] whenever the problem can be solved as an equivalent smooth problem. Nevertheless, we shall mention that the latter approach has a much wider field of application (e.g., see [21]), where the former approach cannot be applied.

TABLE 1
Comparison of NSA, SM and VSM.

Problem	Iter			Obj			Time		
n	NSA	SM	VSM	NSA	SM	VSM	NSA	SM	VSM
50	3657	457	20	−76.399	−76.399	−76.393	49.0	2.7	0.1
100	7629	920	27	−186.717	−186.720	−186.714	900.4	38.4	0.4
150	20358	1455	49	−318.195	−318.194	−318.184	8165.7	188.8	2.0
200	27499	2294	102	−511.246	−511.245	−511.242	26172.5	698.8	9.2
250	45122	3060	128	−3793.255	−3793.256	−3793.257	87298.9	1767.9	19.8
300	54734	3881	161	−3187.163	−3187.171	−3187.172	184798.1	3994.0	45.5
350	64641	4634	182	−2756.717	−2756.734	−2756.734	351460.7	7613.9	83.6
400	74839	5308	176	−3490.640	−3490.667	−3490.667	614237.1	13536.7	116.9

From the above experiment, we have already seen that the method VSM outperforms substantially two other first-order methods, namely, SM and NSA for solving problem (27). In the second experiment, we compare the performance of the method VSM with the block-coordinate descent methods studied in d'Aspremont et al. [10] and Friedman, Hastie, and Tibshirani [16] on relatively large-scale instances. For convenience of presentation, we label these two methods BCD1 and BCD2, respectively. The method BCD2 was developed very recently and is a slight variant of the method BCD1. In particular, each iterate of BCD1 solves a box constrained quadratic programming by means of interior point methods, but each iterate of BCD2 applies a coordinate descent approach to solving a lasso (l_1-regularized) least-squares problem, which is the dual of the box constrained quadratic programming appearing in BCD1. It is worth mentioning that the methods BCD1 and BCD2 are only applicable for solving problem (16) with $\tilde{\alpha} = 0$ and $\tilde{\beta} = \infty$. Thus, we only compare their performance with our method VSM for problem (16) with such $\tilde{\alpha}$ and $\tilde{\beta}$. As shown in subsection 3.2, problem (16) with $\tilde{\alpha} = 0$ and $\tilde{\beta} = \infty$ is equivalent to problem (27) with α and β given in (18), and hence it can be solved by applying the method VSM to the latter problem instead.

The code for the method BCD1 was written in MATLAB by d'Aspremont and El Ghaoui [9], while the code for BCD2 was written in Fortran 90 by Friedman, Hastie, and Tibshirani [15]. The methods BCD1 and VSM terminate once the duality gap is less than $\epsilon = 0.1$. The original code [15] for BCD2 uses the average absolute change in the approximate solution as the termination criterion. In particular, the average absolute change in the approximate solution is evaluated at the end of each cycle consisting of n block-coordinate descent iterations, and their code terminates once it is below a given accuracy (see [16, p. 6] for details). According to our computational experience, we found that with such a criterion BCD2 is extremely hard to terminate for relatively large-scale instances (say $n = 300$) unless a maximum number of iterations is set. Obviously, it is not easy to choose a suitable maximum number of iterations for BCD2. Thus, to be as fair as possible to BCD1 and VSM, we simply replace their termination criterion detailed in [15] for BCD2 by the one with the duality gap less than $\epsilon = 0.1$. In other words, the duality gap is computed at the end of each cycle consisting of n block-coordinate descent iterations, and BCD2 terminates once it is below $\epsilon = 0.1$. It is worth remarking that the cost for computing a duality gap is $\mathcal{O}(n^3)$ since the inverse of an $n \times n$ symmetric matrix is needed. Thus, it is reasonable to compute the duality gap once every n iterations rather than each iteration.

TABLE 2
Comparison of BCD1, BCD2, and VSM.

Problem	Iter			Obj			Time		
n	BCD1	BCD2	VSM	BCD1	BCD2	VSM	BCD1	BCD2	VSM
100	124	200	33	−186.522	−186.433	−186.522	22.3	0.1	0.5
200	531	600	109	−449.210	−449.179	−449.209	300.0	1.3	9.5
300	1530	1500	146	−767.615	−767.608	−767.614	2428.2	80.9	48.5
400	2259	2400	154	−1082.679	−1082.651	−1082.677	8402.4	298.7	112.3
500	3050	3500	154	−1402.503	−1402.457	−1402.502	22537.1	640.2	211.5
600	3705	4200	165	−1728.628	−1728.587	−1728.627	48950.4	1215.0	397.6
700	4492	4900	163	−2057.894	−2057.862	−2057.892	92052.7	1972.5	611.1
800	4958	5600	169	−2392.713	−2392.671	−2392.712	147778.9	2872.3	943.2
900	5697	6300	161	−2711.874	−2711.827	−2711.874	219644.3	3593.7	1268.5
1000	6536	7000	161	−3045.808	−3045.768	−3045.808	344687.8	6098.7	1710.0

TABLE 3
Comparison of BCD2 and VSM.

Problem	Iter		Obj		Time	
n	BCD2	VSM	BCD2	VSM	BCD2	VSM
100	200	54	−186.433	−186.435	0.1	0.77
200	1200	239	−449.119	−449.122	2.1	21.6
300	3000	310	−767.525	−767.525	32.1	104.2
400	11778400	321	−1082.592	−1082.589	72000.0	223.3
500	6997000	309	−1402.420	−1402.413	72001.0	395.5
600	4637400	318	−1728.553	−1728.538	72004.0	765.2
700	3215100	310	−2057.823	−2057.804	72005.0	1330.0
800	2307200	309	−2392.644	−2392.623	72003.0	1789.2
900	1846800	289	−2711.806	−2711.784	72024.0	2394.0
1000	1257000	283	−3045.749	−3045.718	72051.0	3115.8

All computations are performed on an Intel Xeon 2.66 GHz machine with Red Hat Linux version 8. The performance of the methods BCD1, BCD2, and VSM for the randomly generated instances are presented in Table 2. The row size n of each sample covariance matrix Σ is given in column one. The numbers of iterations of BCD1, BCD2, and VSM are given in columns two to four, the objective function values are given in columns five to eight, and the CPU times (in seconds) are given in the last three columns, respectively. From Table 2, we conclude that both BCD2 and VSM substantially outperform BCD1. We also observe that our method VSM outperforms BCD2 for almost all instances except two relatively small-scale instances.

In the above experimentation, we compared the performance of BCD2 and VSM for $\epsilon = 0.1$. We next compare their performance on the same instances as above and apply the same termination criterion as above except that we set $\epsilon = 0.01$ and an upper bound of 20 hours computation time (or 72,000 seconds) per instance for both codes. The performance of the methods BCD2 and VSM is presented in Table 3. The row size n of each sample covariance matrix Σ is given in column one. The numbers of iterations of BCD2 and VSM are given in columns two to three, the objective function

values are given in columns four to five, and the CPU times (in seconds) are given in the last two columns, respectively. It shall be mentioned that BCD2 and VSM are both feasible methods, and, moreover, (16) and (27) are maximization problems. Therefore for these two methods, the larger the objective function value, the better. From Table 3, we observe that up to accuracy $\epsilon = 0.01$, the method BCD2 cannot solve almost all instances within 20 hours except the first three relatively small-scale ones, but our method VSM does solve each of these instances in less than one hour and produces better objective function values for almost all instances except the first three relatively small-scale ones. Also, it is interesting to observe that the number of iterations for VSM nearly doubles as the accuracy parameter ϵ increases by one digit, which is even better than the theoretical estimate, which is $\sqrt{10}$ according to Theorem 3.2.

5. Concluding remarks. In this paper, we proposed a smooth optimization approach for solving a class of nonsmooth strictly concave maximization problems. We also discussed the application of this approach to sparse covariance selection and proposed a variant of this approach. The computational results showed that the variant of the smooth optimization approach substantially outperforms the latter one, as well as two other first-order methods studied in d'Aspremont et al. [10] and Friedman et al. [16].

As discussed in subsection 3.3, problem (27) has the same form as (2) and satisfies all assumptions imposed on problem (2). Moreover, its associated objective function $\phi(X, U) = \log \det X - \langle \Sigma + \rho U, X \rangle$ is affine with respect to U for every fixed $X \in \mathcal{S}_{++}^n$. In view of these facts along with the remarks made in section 2, one can observe that problem (27) can be suitably solved by Nesterov's excessive gap technique [20]. Since the iterate complexity and the computational cost per iterate of this technique is the same as those of the algorithm SMACS, we expect that the computational performance of these two methods for solving (27) is similar. It would be interesting to implement Nesterov's excessive gap technique [20] and its variant (that is, the one in a similar fashion to the algorithm VSMACS) and compare their computational performance with SMACS and VSMACS, respectively.

Though the variant of the smooth optimization approach outperforms substantially the smooth optimization approach, we are currently only able to establish some preliminary convergence properties for it. A possible direction leading to a thorough proof of its convergence would be to show that the updates on $\hat{\beta}$ in the algorithm VSMACS can occur only for a finite number of times. Given that VSMACS is a nonmonotone algorithm, it is, however, highly challenging to analyze the behavior of the sequences $\{U_k\}$ and $\{X_{\hat{\beta}}(U_k)\}$ and hence the total number of updates on $\hat{\beta}$. Interestingly, we observed in our implementation that when $\hat{\beta} > \lambda_{\max}(X^*)$, the sequence $\{X_{\hat{\beta}}(U_k)\}$ generated by the algorithm VSMACS satisfies $\lambda_{\max}(X_{\hat{\beta}}(U_k)) \in [\lambda_{\max}(X^*), \hat{\beta})$, where X^* is the optimal solution of problem (27). Nevertheless, it remains completely open whether or not this holds in general. In addition, the ideas used in the variant of the smooth optimization approach are interesting in their own right even when viewed as some heuristics. They could also be used to enhance the practical performance of Nesterov's first-order methods [19, 21] for solving some general min-max problems.

The codes for the variant of the smooth minimization approach are written in MATLAB and C, which are available online at www.math.sfu.ca/~zhaosong. The C code for this method can solve large-scale problems more efficiently, provided the LAPACK package is suitably installed. We will plan to extend these codes for solving

more general problems of the form

$$\max_X \quad \log\det X - \langle \Sigma, X \rangle - \sum_{ij} \omega_{ij}|X_{ij}|$$

$$\text{s.t.} \qquad \tilde{\alpha}I \preceq X \preceq \tilde{\beta}I,$$

$$X_{ij} = 0 \quad \forall(i,j) \in \Omega$$

for some set Ω, where $\omega_{ij} = \omega_{ji} \geq 0$ for all $i,j = 1,\ldots,n$, and $0 \leq \tilde{\alpha} < \tilde{\beta} \leq \infty$ are some fixed bounds on the eigenvalues of the solution.

REFERENCES

[1] J. AKAIKE, *Information theory and an extension of the maximum likelihood principle*, in Proceedings of the Second International Symposium on Information Theory, B. N. Petrov and F. Csaki, eds., Akedemiai Kiado, Budapest, 1973, pp. 267–281.

[2] A. AUSLENDER AND M. TEBOULLE, *Interior gradient and proximal methods for convex and conic optimization*, SIAM J. Optim., 16 (2006), pp. 697–725.

[3] O. BANERJEE, L. EL GHAOUI, A. D'ASPREMONT, AND G. NATSOULIS, *Convex optimization techniques for fitting sparse Gaussian graphical models*, in ICML '06: Proceedings of the 23rd International Conference on Machine Learning, ACM Press, New York, 2006, pp. 89–96.

[4] J. A. BILMES, *Factored sparse inverse covariance matrices*, in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2 (2000), pp. 1009–1012.

[5] K. P. BURNHAM AND R. D. ANDERSON, *Multimodel inference. Understanding AIC or BIC in model selection*, Sociol. Methods Res., 33 (2004), pp. 261–304.

[6] S. S. CHEN, D. L. DONOHO, AND M. A. SAUNDERS, *Atomic decomposition by basis pursuit*, SIAM J. Sci. Comput., 20 (1998), pp. 33–61.

[7] J. DAHL, V. ROYCHOWDHURY, AND L. VANDENBERGHE, *Maximum Likelihood Estimation of Gaussian Graphical Models: Numerical Implementation and Topology Selection*, manuscript, University of California, Los Angeles, 2004.

[8] J. DAHL, L. VANDENBERGHE, AND V. ROYCHOWDHURY, *Covariance selection for nonchordal graphs via chordal embedding*, Optim. Methods Softw., 23 (2008), pp. 501–520.

[9] A. D'ASPREMONT AND L. EL GHAOUI, *Covsel: First order methods for sparse covariance selection*, ORFE Department, Princeton University, Princeton, NJ, 2006.

[10] A. D'ASPREMONT, O. BANERJEE, AND L. EL GHAOUI, *First-order methods for sparse covariance selection*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 56–66.

[11] A. DEMPSTER, *Covariance selection*, Biometrics, 28 (1972), pp. 157–175.

[12] A. DOBRA AND M. WEST, *Bayesian Covariance Selection*, ISDS working paper, Duke University, Durham, NC, 2004.

[13] A. DOBRA, C. HANS, B. JONES, J. R. NEVINS, G. YAO, AND M. WEST, *Sparse graphical models for exploring gene expression data*, J. Multivariate Anal., 90 (2004), pp. 196–212.

[14] D. L. DONOHO AND J. TANNER, *Sparse nonnegative solutions of underdetermined linear equations by linear programming*, Proc. Natl. Acad. Sci., 102 (2005), pp. 9446–9451.

[15] J. FRIEDMAN, T. HASTIE, AND R. TIBSHIRANI, *Glasso: Graphical lasso for R*, Department of Statistics, Stanford University, Stanford, CA, 2007.

[16] J. FRIEDMAN, T. HASTIE, AND R. TIBSHIRANI, *Sparse inverse covariance estimation with the graphical lasso*, Biostatistics, 9 (2008), pp. 432–441.

[17] J. Z. HUANG, N. LIU, AND M. POURAHMADI, *Covariance matrix selection and estimation via penalised normal likelihood*, Biometrika, 93 (2006), pp. 85–98.

[18] B. JONES, C. CARVALHO, C. DOBRA, A. HANS, C. CARTER, AND M. WEST, *Experiments in stochastic computation for high-dimensional graphical models*, Statist. Sci., 20 (2005), pp. 388–400.

[19] Y. E. NESTEROV, *A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$*, Dokl. Akad. Nauk SSSR, 269 (1983), pp. 543–547 (in Russian).

[20] Y. NESTEROV, *Excessive gap technique in nonsmooth convex minimization*, SIAM J. Optim., 16 (2005), pp. 235–249.

[21] Y. E. NESTEROV, *Smooth minimization of nonsmooth functions*, Math. Programming, 103 (2005), pp. 127–152.

[22] Y. NESTEROV AND A. NEMIROVSKII, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.

[23] R. TIBSHIRANI, *Regression shrinkage and selection via the lasso*, J. Roy. Statist. Soc. Ser. B, 58 (1996), pp. 267–288.

[24] L. VANDENBERGHE, S. BOYD, AND S.-P. WU, *Determinant maximization with linear matrix inequality constraints*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 499–533.

[25] M. YUAN AND Y. LIN, *Model selection and estimation in the Gaussian graphical model*, Biometrika, 94 (2007), pp. 19–35.