FAST PROXIMAL ALGORITHMS FOR SELF-CONCORDANT FUNCTION MINIMIZATION WITH APPLICATION TO SPARSE GRAPH SELECTION

Anastasios Kyrillidis and Volkan Cevher

École Politechnique Fédéral de Lausanne
{anastasios.kyrillidis,volkan.cevher}@epfl.ch

ABSTRACT

The convex ℓ_1-regularized $\log\det$ divergence criterion has been shown to produce theoretically consistent graph learning. However, this objective function is challenging since the ℓ_1-regularization is nonsmooth, the $\log\det$ objective is not *globally* Lipschitz gradient function, and the problem is high-dimensional. Using the self-concordant property of the objective, we propose a new adaptive step size selection and present the (F)PS ((F)ast Proximal algorithms for Self-concordant functions) algorithmic framework which has linear convergence and exhibits superior empirical results as compared to state-of-the-art first order methods.

Index Terms— Sparse inverse covariance estimation, self-concordance, step size selection

1. INTRODUCTION

Problem setup: Let $\mathcal{X} = \{X_1, X_2, \dots, X_n\}$ be a set of variables with joint Gaussian distribution $f(X_1, X_2, \dots, X_n) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ where $\boldsymbol{\mu} \in \mathbb{R}^n$ is assumed known and $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}, \boldsymbol{\Sigma} \succ 0$ denotes the *unknown* covariance matrix. In this setting, assume we only have access to the underlying model through a set of independent and identically distributed (iid) samples $\{\mathbf{x}_j\}_{j=1}^p$ such that $\mathbf{x}_j \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \ \forall j$. Given $\{\mathbf{x}_j\}_{j=1}^p$, we are interested in inferring any conditional dependencies among $\mathcal{X}$ by estimating $\boldsymbol{\Sigma}^{-1}$. A non-robust estimate of $\boldsymbol{\Sigma}^{-1}$ is through the sample covariance $\widehat{\boldsymbol{\Sigma}} = \frac{1}{p}\sum_{j=1}^p (\mathbf{x}_j - \widehat{\boldsymbol{\mu}})(\mathbf{x}_j - \widehat{\boldsymbol{\mu}})^T$ where $\widehat{\boldsymbol{\mu}} = \frac{1}{p}\sum_{j=1}^p \mathbf{x}_j$. Unfortunately, in many cases, we cannot afford to acquire adequate samples for accurate $\boldsymbol{\Sigma}^{-1}$ estimation via $\widehat{\boldsymbol{\Sigma}}$; for $p \ll n$, $\widehat{\boldsymbol{\Sigma}}$ is rank-deficient and the use of sophisticated estimation procedures is imperative.

Graphical models interpretation: In undirected graphical models, each variable X_i corresponds to a node in a Gaussian Markov random field (GMRF). Moreover, let $E = \{(i,j) : X_i \not\perp X_j \mid X_k$ is observed $\forall k \neq i,j\}$ be the set of edges in the graph. Under this setting, we desire to infer the graph structure given a set of observations. Due to the Gaussianity assumption, $\boldsymbol{\Sigma}_{ij}^{-1} = 0 \Leftrightarrow (i,j) \notin E$.

Optimization criteria: [1] shows that the maximum likelihood estimation $(\boldsymbol{\Sigma}^*)^{-1} = \arg\max_{\boldsymbol{\Sigma}^{-1} \succ 0} \prod_{j=1}^p f(\mathbf{x}_j)$ is equivalent to:

$$\boldsymbol{\Theta}^* = \underset{\boldsymbol{\Theta} \succ 0}{\text{argmin}} \ \Big\{ -\log\det(\boldsymbol{\Theta}) + \text{tr}(\boldsymbol{\Theta}\widehat{\boldsymbol{\Sigma}}) \Big\}, \qquad (1)$$

where $\boldsymbol{\Theta}^* = (\boldsymbol{\Sigma}^*)^{-1}$. Based on (1), developments in random matrix theory [2] divulge the poor performance of $\boldsymbol{\Theta}^*$ without regularization: the solution to (1) is usually fully dense and no inference about the graph structure is possible. Moreover, when $p \ll n$, the absence of a regularization term leads to non-robust estimates of $\boldsymbol{\Sigma}^{-1}$.

In practice though, parsimonious solutions that adequately explain the data, increase the interpretability of the results even if they lead to worse-valued loss objective values. Using ℓ_1-norm to regularize the objective, (1) can be well-approximated by:

$$\boldsymbol{\Theta}^* = \underset{\boldsymbol{\Theta} \succ 0}{\arg\min} \left\{ F(\boldsymbol{\Theta}) := f(\boldsymbol{\Theta}) + g(\boldsymbol{\Theta}) \right\}, \qquad (2)$$

where $f(\boldsymbol{\Theta}) := -\log\det(\boldsymbol{\Theta}) + \text{tr}(\widehat{\boldsymbol{\Sigma}}\boldsymbol{\Theta})$ and $g(\boldsymbol{\Theta}) := \rho\|\text{vec}(\boldsymbol{\Theta})\|_1$ with $\rho > 0$ that defines the sparsity of the graph selection.

Challenges: Within this context, the main challenges in (2) are:

- High-dimensional problems have become the norm in data analysis; thus, time- and memory-efficient schemes are crucial.
- Apart from its computational challenge, (2) is a non-trivial convex problem: $f(\boldsymbol{\Theta})$ is a strictly convex but not *globally* Lipschitz-continuous gradient function; moreover, $g(\boldsymbol{\Theta})$ is a nonsmooth regularizer. Even in simple gradient descent schemes, Lipschitz-based *optimal* step size calculation becomes infeasible and heuristics lead to slowly convergent, state-of-the-art algorithms [3]. Moreover, (2) is constrained over the set of positive-definite matrices and the choice of regularization parameter ρ is crucial [4].

Prior work: Being a special case of semidefinite programming, (2) can be solved using off-the-shelf interior point approaches [5, 6]. Though, the resulting per iteration complexity for existing interior point methods is $\mathcal{O}(n^6)$ [7]. This has led to the development of multifarious works, which can be roughly categorized into five camps: (i) first-order gradient methods [7, 8, 9], (ii) second order (Newton-based) gradient methods [10, 11], (iii) interior point-based schemes [12], (iv) Lagrangian [13, 3] and (iii) greedy approaches [14].

While many of the first-order approaches are slowly convergent and require numerous parameters to be set *apriori* (reducing their universality), recent developments on second-order methods have resulted in very fast solvers. Though, to achieve this fast performance, these approaches "sacrifice" their universality for faster implementation: one can envision complicated examples (e.g., non-modular regularization) where second-order approaches fail to use their "arsenal" (e.g., greedy heuristics) for computational superiority.

Contributions: Our contributions can be summarized as follows:

- We introduce a *new* adaptive step size for first-order methods to solve (2), based on the self-concordance property. This technique can be incorporated in mane other minimization problems with the same property. Moreover, this tool can be subsumed in many existing schemes [3] with a wide range of diverse regularization terms, decreasing their time-complexity.

This work was supported in part by the European Commission under Grant MIRG-268398, ERC Future Proof, ARO MURI W911NF0910383, and SNF 200021-132548. VC also would like to acknowledge Rice University for his Faculty Fellowship.

- To illustrate the substance of the step size selection, we propose the (F)PS ((F)ast Proximal algorithms for Self-concordant functions) framework and show its computational- and memory-efficiency. The resulting schemes have fast convergence and require the minimum number of input parameters.

2. PRELIMINARIES

Notation: We reserve lower-case and bold lower-case letters for scalar and vector representation, respectively. Upper-case letters denote matrices. The inner product between matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m\times n}$ is denoted as $\text{tr}(\mathbf{A}^T\mathbf{B})$, where $\text{tr}(\cdot)$ is the trace operator. Given a matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$, we reserve $\text{diag}\,(\mathbf{A}) \in \mathbb{R}^{n\times n}$ to denote the diagonal matrix with entries taken from the diagonal of $\mathbf{A}$.

We reserve $\mathbb{R}_{++}$ to denote the set of positive scalars. Let $\mathbb{S}^n_{++}$ denote the set of *positive* definite $n \times n$ matrices. For $p(\mathbf{X}) : \mathbb{S}^n_{++} \rightarrow \mathbb{R}$, the gradient is denoted as $\nabla p(\mathbf{X})$; for $h(x) : \mathbb{R} \rightarrow \mathbb{R}$, we use $h'(x), h''(x), h'''(x)$ to denote the first, second and, third derivative.

Definition 1 (Bregman divergence)**.** *Let $p : \mathbb{S}^n_{++} \rightarrow \mathbb{R} \cup \{+\infty\}$ be a continuously differentiable and strictly convex function. Given $\mathbf{\Theta}_1, \mathbf{\Theta}_2 \in \mathbb{R}^{n\times n}$, the Bregman divergence $\mathcal{D}_p(\cdot \parallel \cdot)$ is given by:*

$$\mathcal{D}_p(\mathbf{\Theta}_1 \parallel \mathbf{\Theta}_2) = p(\mathbf{\Theta}_1) - p(\mathbf{\Theta}_2) - tr(\nabla p(\mathbf{\Theta}_2)(\mathbf{\Theta}_1 - \mathbf{\Theta}_2)).$$

Definition 2 (Convexity bounds in gradient methods)**.** *Let $p : \mathbb{S}^n_{++} \rightarrow \mathbb{R}$ be a strongly convex function with continuous Lipschitz gradient $\nabla p(\mathbf{X})$ for $\mathbf{X} \in \mathbb{S}^n_{++}$. Then, there exist $\mu,\ L > 0$ such that, for any $\mathbf{\Theta}_1, \mathbf{\Theta}_2 \in \mathbb{S}^n_{++}$: $\frac{\mu}{2} \leqslant \frac{\mathcal{D}_p(\mathbf{\Theta}_1 \| \mathbf{\Theta}_2)}{\|\mathbf{\Theta}_1 - \mathbf{\Theta}_2\|_F^2} \leqslant \frac{L}{2}$.*

Proposition 1 (Step size selection for strongly convex gradient descent schemes)**.** *For strongly convex (unconstrained) minimization problems $\min_{\mathbf{X}} q(\mathbf{X})$ where $q : \mathbb{R}^{n\times n} \rightarrow \mathbb{R}$, $\tau^* := 2/(\mu + L)$ is the optimal step size in the gradient descent scheme $\mathbf{X}_{i+1} = \mathbf{X}_i - \tau^*\nabla q(\mathbf{X}_i)$ [15].*

Definition 3 (Second order expansion of a function)**.** *[16] Let $h : \mathbb{R} \rightarrow \mathbb{R}$ be a twice differentiable over an open sphere $\mathcal{S}$. Then, for $x, y \in \mathcal{S}$, there exists an constant $\alpha \in [0, 1]$ such that:*

$$h(x + y) = h(x) + h'(x) \cdot y + \frac{1}{2}y^2 \cdot h''(x + \alpha y). \qquad (3)$$

Definition 4 (Self-concordant functions)**.** *[17] A convex function $h : \mathbb{R} \rightarrow \mathbb{R}$ is self-concordant if $|h'''(x)| \leqslant 2h''(x)^{3/2}$, $\forall x \in \mathbb{R}$. Given two self-concordant functions h_1, h_2, $h_1 + h_2$ is self-concordant.*

Lemma 1 (Upper and lower bounds on second derivatives for self- -concordant functions)**.** *[17] Let $h : \mathbb{R} \rightarrow \mathbb{R}$ be a* strictly convex, *self-concordant function. Then, $h''(t)$ satisfies:*

$$\frac{h''(0)}{\left(1 + t\sqrt{h''(0)}\right)^2} \leqslant h''(t) \leqslant \frac{h''(0)}{\left(1 - t\sqrt{h''(0)}\right)^2},$$

where both bounds are valid for $0 \leqslant t < 1/\sqrt{h''(0)}$.

3. GRAPH SELECTION VIA PROXIMAL METHODS

Given that $F(\mathbf{\Theta}) := f(\mathbf{\Theta}) + g(\mathbf{\Theta})$ is strictly convex and provided a putative solution $\mathbf{\Theta}_i \in \mathbb{S}^n_{++}$, an iterative descent scheme follows:

$$\mathbf{\Theta}_{i+1} = \mathbf{\Theta}_i + \tau_i^*\mathbf{\Delta},$$

where $\mathbf{\Delta} \in \mathbb{R}^{n\times n}$ is a descent direction such that $F(\mathbf{\Theta}_{i+1}) < F(\mathbf{\Theta}_i)$ for $\tau_i^* > 0$. To compute $\{\mathbf{\Delta}, \tau_i^*\}$, we can form the following optimization problem:

$$\{\mathbf{\Delta}, \tau_i^*\} = \underset{\mathbf{\Delta}\in\mathbb{R}^{n\times n}, \tau>0}{\arg\min} \ \{F(\mathbf{\Theta}_i + \tau\mathbf{\Delta}) : \mathbf{\Theta}_i + \tau\mathbf{\Delta} \succ 0\}. \qquad (4)$$

While (4) is the *proper* way to compute a direction $\mathbf{\Delta}$ and a corresponding step size τ_i^*, in this paper we present an approximation scheme to (4) that introduces the notion of self-concordance in step size selection and performs extremelly well in practice; we reserve the detailed convergence analysis for an extended version.

To this end, the proposed algorithm iteratively computes a putative solution by forming a quadratic surrogate *only* for $f(\mathbf{\Theta})$ at $\mathbf{\Theta}_i \in \mathbb{S}^n_{++}$, i.e., $f(\mathbf{\Theta}) \leqslant U(\mathbf{\Theta}, \mathbf{\Theta}_i) := f(\mathbf{\Theta}_i) + \text{tr}\,(\mathbf{\Delta} \cdot (\mathbf{\Theta} - \mathbf{\Theta}_i)) + \frac{1}{2\tau_i^*}\|\mathbf{\Theta} - \mathbf{\Theta}_i\|_F^2$, for a *carefully* selected $\tau_i^* > 0$ and a direction satisfying $\mathbf{\Delta} := -\nabla f(\mathbf{\Theta}_i)$, depending *only* on $f(\cdot)$, i.e., we ignore the presence of $g(\cdot)$ in $F(\cdot)$. Then, instead of minimizing (2), we iteratively solve the following problem:

$$\mathbf{\Theta}_{i+1} = \underset{\mathbf{\Theta}\succ 0}{\arg\min}\Big\{U(\mathbf{\Theta}, \mathbf{\Theta}_i) + g(\mathbf{\Theta})\Big\}, \qquad (5)$$

which can be equivalently stated in proximity operator form [18] as:

$$\mathbf{\Theta}_{i+1} = \underset{\mathbf{\Theta}\succ 0}{\arg\min}\Big\{\frac{1}{2\tau_i^*}\|\mathbf{\Theta} - (\mathbf{\Theta}_i + \tau_i^*\mathbf{\Delta})\|_F^2 + g(\mathbf{\Theta})\Big\}. \qquad (6)$$

The recursive relation in (6) proposes an optimization recipe : given a step size τ_i^*, we perform a gradient descent step $\mathbf{\Theta}_i + \tau_i^*\mathbf{\Delta}$ where $\mathbf{\Delta} := -\nabla f(\mathbf{\Theta}_i)$ followed by a soft-thresholding operation $\mathbf{\Theta}_{i+1} = \text{Soft}\left(\mathbf{X}_i, \tau_i^*\rho\right)$ with threshold $\tau_i^*\rho$ as the closed-form solution the the proximity operator in (6). Finally, we perform a projection onto the positive definite cone using eigenvalue decomposition.

4. τ_i^* SELECTION FOR SELF-CONCORDANT FUNCTIONS

Given $\mathbf{\Delta} := -\nabla f(\mathbf{\Theta}_i)$, we perform a gradient descent step $\mathbf{X}_i = \mathbf{\Theta}_i - \tau_i^*\nabla f(\mathbf{\Theta}_i)$ where $\tau_i^* > 0$ and $\nabla f(\mathbf{\Theta}_i) := -\mathbf{\Theta}_i^{-1} + \widehat{\mathbf{\Sigma}}$. Since τ_i^* is unknown, for clarity let $\mathbf{X}_i = \mathbf{\Theta}_i - \tau\nabla f(\mathbf{\Theta}_i)$ where τ is the unknown variable step size. Then, for $\mathbf{\Theta}_1 := \mathbf{X}_i$ and $\mathbf{\Theta}_2 := \mathbf{\Theta}_i$ in Bregman divergence, we define function $\phi(\tau)$ as:

$$\begin{aligned}\phi(\tau) &:= \mathcal{D}_f(\mathbf{X}_i \parallel \mathbf{\Theta}_i) = -\log\det(\mathbf{X}_i) + \log\det(\mathbf{\Theta}_i) \\ &\quad + \text{tr}(\mathbf{\Theta}_i^{-1}(\mathbf{X}_i - \mathbf{\Theta}_i)) \\ &= -\log\det(\mathbf{\Theta}_i - \tau\nabla f(\mathbf{\Theta}_i)) + \log\det(\mathbf{\Theta}_i) \\ &\quad - \tau \cdot \text{tr}(\mathbf{\Theta}_i^{-1}\nabla f(\mathbf{\Theta}_i)). \qquad (7)\end{aligned}$$

In (7), we can rewrite the first $\log\det(\cdot)$ term as [17]:

$$-\log\det(\mathbf{\Theta}_i - \tau\nabla f(\mathbf{\Theta}_i)) = -\log\det(\mathbf{\Theta}_i) - \sum_{j=1}^n \log(1 - \tau\lambda_j),$$

where λ_j are the eigenvalues of $\mathbf{\Theta}_i^{-1/2}\nabla f(\mathbf{\Theta}_i)\mathbf{\Theta}_i^{-1/2}$. Then:

$$\phi(\tau) = -\sum_{j=1}^n \log(1 - \tau\lambda_j) - \tau \cdot \text{tr}(\mathbf{\Theta}_i^{-1}\nabla f(\mathbf{\Theta}_i)), \qquad (8)$$

which is a self-concordant function as the superposition of a self-concordant and a linear (thus self-concordant) function.

Remark 1. *In (8), we assume $1 - \tau\lambda_j \geqslant 0,\ \forall j$ by the definition of the logarithm function. Subsequently, we show that our step size selection always satisfies these conditions, $\forall j$.*

We observe that (8) is *strictly* convex as a function of τ. Applying the second order expansion (Definition 3) on $\phi(\tau)$, we have:

Lemma 2. *The function $\phi(\tau)$ satisfies: $\phi(\tau) = \frac{1}{2} \cdot \tau^2 \cdot \phi''(\hat{\tau})$, for $\hat{\tau} \in [0, \tau]$ and $\phi''(\hat{\tau}) = \sum_{j=1}^n \frac{\lambda_j^2}{(1-\hat{\tau}\lambda_j)^2}$.*

Proof. For $y := \tau$, $x := 0$ and $\alpha \cdot y := \hat{\tau}$ in Definition 3, the second order expansion of $\phi(\tau)$ satisfies according to (3):

$$\phi(\tau) = \phi(0) + \phi'(0) \cdot \tau + \frac{1}{2} \cdot \tau^2 \cdot \phi''(\hat{\tau}).$$

It is easy to verify the following: (i) $\phi(0) = 0$, (ii) $\phi''(\hat{\tau}) = \sum_{j=1}^n \frac{\lambda_j^2}{(1-\hat{\tau}\lambda_j)^2}$. Moreover, $\phi'(0) = \sum_{j=1}^n \lambda_j - \text{tr}(\mathbf{\Theta}_i^{-1} \nabla f(\mathbf{\Theta}_i))$. But $\sum_{j=1}^n \lambda_j = \text{tr}\left(\mathbf{\Theta}_i^{-1} \nabla f(\mathbf{\Theta}_i)\right)$. Therefore, $\phi'(0) = 0$. □

Let $\xi(\tau) := \frac{\phi''(0)}{\left(1+\tau\sqrt{\phi''(0)}\right)^2}$. Since $\phi(\cdot)$ is self-concordant and *strictly convex*, the following inequalities hold true for $\hat{\tau} \in (0, \tau]$:

$$\xi(\tau) \leqslant \xi(\hat{\tau}) \leqslant \phi''(\hat{\tau}) \leqslant \xi(-\hat{\tau}) \leqslant \xi(-\tau). \tag{9}$$

From Lemma 2, $\phi''(0) = \sum_{j=1}^n \lambda_j^2$. We know that $\text{tr}(\mathbf{A}^k) = \sum_{j=1}^n \xi_j^k$ for $\mathbf{A} \in \mathbb{R}^{n \times n}$ where ξ_j are the eigenvalues of $\mathbf{A}$. Thus, $\phi''(0) = \sum_{j=1}^n \lambda_j^2 = \text{tr}\left((\mathbf{\Theta}_i^{-1} \nabla f(\mathbf{\Theta}_i))^2\right)$.

Given (7), Lemma 2 and $\|\mathbf{X}_i - \mathbf{\Theta}_i\|_F^2 = \tau^2 \|\nabla f(\mathbf{Y}_i)\|_F^2$:

$$\mathcal{D}_f(\mathbf{X}_i \parallel \mathbf{\Theta}_i) = \frac{1}{2} \cdot \tau^2 \cdot \phi''(\hat{\tau}) \Rightarrow \frac{\mathcal{D}_f(\mathbf{X}_i \parallel \mathbf{\Theta}_i)}{\|\nabla f(\mathbf{\Theta}_i)\|_F^2} = \frac{\phi''(\hat{\tau})}{2\|\nabla f(\mathbf{\Theta}_i)\|_F^2}$$

Combining the above equation with (9), we *locally* have:

$$\frac{\widetilde{\mu}}{2} \leqslant \frac{\mathcal{D}_f\left(\mathbf{X}_i \| \mathbf{\Theta}_i\right)}{\|\mathbf{X}_i - \mathbf{\Theta}_i\|_F^2} \leqslant \frac{\widetilde{L}}{2} \tag{10}$$

where $\widetilde{L} = \frac{\delta}{(1-\tau\sqrt{\delta})^2\epsilon}$ and $\widetilde{\mu} = \frac{\delta}{(1+\tau\sqrt{\delta})^2\epsilon}$ for $\delta := \phi''(0)$ and $\epsilon := \|\nabla f(\mathbf{\Theta}_i)\|_F^2$.

By Definition 2, a *safe* step size selection at the i-th iteration satisfies $\tau_i^* := \tau = 2/(\widetilde{\mu} + \widetilde{L})$ which leads to the following lemma:

Lemma 3. *At the i-th iteration, the step size $\tau_i^* = 2/(\widetilde{\mu} + \widetilde{L})$ is determined as $\tau_i^* = \frac{1}{2}\left(-\frac{1}{\epsilon} \pm \sqrt{\frac{1}{\epsilon^2} + \frac{4}{\delta}}\right)$. Moreover, τ_i^* is guaranteed to satisfy $0 \leqslant \tau_i^* < \sqrt{\phi''(0)}$, $\forall i$.*

Proof. For $\tau_i^* := \tau = 2/(\widetilde{\mu} + \widetilde{L})$ we obtain:

$$\tau = \frac{2}{\frac{\delta}{(1+\tau\sqrt{\delta})\epsilon} + \frac{\delta}{(1-\tau\sqrt{\delta})\epsilon}} \Rightarrow \tau^2 + \frac{1}{\epsilon}\tau - \frac{1}{\delta} = 0 \tag{11}$$

with roots $\tau_{\text{min,max}} = \frac{1}{2}\left(-\frac{1}{\epsilon} \pm \sqrt{\frac{1}{\epsilon^2} + \frac{4}{\delta}}\right)$. To use the upper bound in (9), the solution τ must satisfy $0 \leqslant \tau < 1/\sqrt{\delta}$. We easily observe that $\tau_{\text{min}} \leqslant 0$. For $\tau_{\text{max}} = \frac{1}{2}\left(-\frac{1}{\epsilon} + \sqrt{\frac{1}{\epsilon^2} + \frac{4}{\delta}}\right)$, we have: $\tau_{\text{max}} \geqslant 0$ and $\tau_{\text{max}} \leqslant \frac{1}{2}\left(-\frac{1}{\epsilon} + \sqrt{\frac{1}{\epsilon^2}} + \sqrt{\frac{4}{\delta}}\right) = \frac{1}{\sqrt{\delta}}$. since $\frac{1}{\epsilon^2} + \frac{4}{\delta} > 0$. Thus, $\tau_i^* := \tau_{\text{max}}$ such that $\tau_i^* = 2/(\widetilde{\mu} + \widetilde{L})$ and $0 \leqslant \tau_i^* < \frac{1}{\sqrt{\phi''(0)}}$. □

Remark 2. *An alternative step size selection is computed as the* minimum *root of $\tau_i^* = 1/\widetilde{L}$. While this scheme performs well, it does not exploit the strong convexity of the smooth term.*

Algorithm 1 Proximal algorithm for Self-concordant functions

```
Input: $\widehat{\mathbf{\Sigma}} \succeq 0$, $\rho$, MaxIter, tol
Initialize: $\mathbf{\Theta}_0 = \text{diag}(\widehat{\mathbf{\Sigma}})^{-1}$
repeat
    1. $\{\tau_i^*,\ \nabla f(\mathbf{\Theta}_i)\}$ = compute_tau($\widehat{\mathbf{\Sigma}}, \mathbf{\Theta}_i$)        $\mathcal{O}(n^3)$
    2. $\mathbf{X}_i = \mathbf{\Theta}_i - \tau_i^* \nabla f(\mathbf{\Theta}_i)$                                        $\mathcal{O}(n^2)$
    3. $\mathbf{\Theta}_{i+1} = \text{Soft}\left(\mathbf{X}_i, \tau_i^* \rho\right)$                                    $\mathcal{O}(n^2)$
    4. If $\mathbf{\Theta}_{i+1} \succ 0$ then continue                                    $\mathcal{O}(1)$
    5. else repeat steps 2-3 with $\tau_i^* := \tau_i^*/2$.                   $\mathcal{O}(n^3)$
until MaxIter is reached or $\frac{\|\mathbf{\Theta}_{i+1} - \mathbf{\Theta}_i\|_F}{\|\mathbf{\Theta}_{i+1}\|_F} \leqslant$ tol
```

Proposition 2. *The step size selection proposed in Lemma 3 satisfies $1 - \tau_i^* \lambda_j \geqslant 0$, $\forall j$ in (8).*

Proof. By construction, we observe that $\tau_i^* < 1/\sqrt{\phi''(0)} = \frac{1}{(\sum_j \lambda_j^2)^{1/2}} = 1/\|\boldsymbol{\lambda}\|_2$ where $\boldsymbol{\lambda} := [\lambda_1, \ldots, \lambda_n]$. Then,

$$1 - \tau_i^* \lambda_j \begin{cases} \geqslant 0 & \forall j \text{ such that } \lambda_j \leqslant 0 \text{ since } \tau_i^* \geqslant 0, \\ \geqslant 0 & \begin{array}{l} \forall j \text{ such that } \lambda_j > 0 \text{ since} \\ 1 - \tau_i^* \lambda_j \geqslant 1 - \frac{\lambda_j}{\|\boldsymbol{\lambda}\|_2} \geqslant 1 - \frac{\|\boldsymbol{\lambda}\|_\infty}{\|\boldsymbol{\lambda}\|_2} \geqslant 0. \end{array} \end{cases}$$

□

5. BASIC PROXIMAL ALGORITHM

Algorithm 1 shows the Proximal algorithm for Self-concordant functions (PS) in detail. The per iteration complexity is $\mathcal{O}(n^3)$. The step size selection is dominated by the calculation of the gradient $\nabla f(\mathbf{\Theta}_i) = -\mathbf{\Theta}_i^{-1} + \widehat{\mathbf{\Sigma}}$; an efficient way to compute $\mathbf{\Theta}_i^{-1}$ is through Cholesky factorization with $\mathcal{O}(n^3)$ complexity. Given $\nabla f(\mathbf{\Theta}_i)$ and $\mathbf{\Theta}_i^{-1}$, the time-complexity for $\delta := \text{tr}\left((\mathbf{\Theta}_i^{-1} \nabla f(\mathbf{\Theta}_i))^2\right)$ and $\epsilon := \|\nabla f(\mathbf{\Theta}_i)\|_F^2$ is $\mathcal{O}(n^2)$ while for the quadratic form root-finding step we need $\mathcal{O}(1)$ operations. The soft-thresholding operation requires $\mathcal{O}(n^2)$ complexity.

According to (6), we require $\mathbf{\Theta}_i \succ 0$, $\forall i$. The best projection of an arbitrary matrix onto the set of positive definite $n \times n$ matrices requires an eigenvalue decomposition with $O(n^3)$ complexity; a prohibitive time-complexity that does not scale well for many applications. In practice though, the projection onto $\mathbb{S}_{++}^n$ can be avoided with a backtrack line search over τ_i^*. After soft-thresholding, we can check $\mathbf{\Theta}_{i+1} \succ 0$ via its Cholesky factorization. In case $\mathbf{\Theta}_{i+1} \not\succ 0$, we decrease the step size $\tau_i^* := \tau_i^*/2$ and repeat steps 2 and 3 with complexity $\mathcal{O}(n^2)$. Otherwise, we can reuse the Cholesky factorization of $\mathbf{\Theta}_{i+1}$ to compute $\mathbf{\Theta}_{i+1}^{-1}$ and $\nabla f(\mathbf{\Theta}_{i+1})$ in the next iteration. In practice though, we rarely need this additional operation.

6. FAST PROXIMAL ALGORITHM

To gain momentum in convergence, we can use memory in estimates as proposed by Nesterov for *strongly* convex functions [15]; the same acceleration technique has been integrated in other convex approaches and problems such as [11, 19]. Moreover, to overcome the oscillatory behaviour in the trace of the objective value due to the momentum update, we can use adaptive "restart" techniques; c.f. [20]. Algorithm 2 summarizes the FPS scheme; the main difference with Algorithm 1 is that, at each iteration, we no longer operate on the previous estimate $\mathbf{\Theta}_{i-1}$ but rather on $\mathbf{Y}_i$ which simulates an

Algorithm 2 Fast Proximal algorithm for Self-concordant functions

```
**Input:** $\widehat{\mathbf{\Sigma}} \geq 0$, $\rho$, MaxIter, tol
**Initialize:** $\mathbf{\Theta}_0 = \text{diag}(\widehat{\mathbf{\Sigma}})^{-1}$, $\mathbf{Y}_1 = \mathbf{\Theta}_0$, $\alpha_1 = 1$.
**repeat**
    1. $\{\tau_i^*, \nabla f(\mathbf{Y}_i), \widetilde{\mu}, \widetilde{L}\} = \texttt{compute\_tau}(\widehat{\mathbf{\Sigma}}, \mathbf{Y}_i)$    $\mathcal{O}(n^3)$
    2. $\mathbf{X}_i = \mathbf{Y}_i - \tau_i^* \nabla f(\mathbf{Y}_i)$    $\mathcal{O}(n^2)$
    3. $\mathbf{\Theta}_i = \text{Soft}\left(\mathbf{X}_i, \tau_i^* \rho\right)$    $\mathcal{O}(n^2)$
    4. $\mathbf{Y}_{i+1} = \mathbf{\Theta}_i + \gamma_i \left(\mathbf{\Theta}_i - \mathbf{\Theta}_{i-1}\right)$ for $\gamma_i > 0$    $\mathcal{O}(n^2)$
    5. **If** $\mathbf{Y}_{i+1} \succ 0$ **then continue**    $\mathcal{O}(1)$
    6. **else** repeat steps 2-4 with $\tau_i^* := \tau_i^*/2$.    $\mathcal{O}(n^3)$
**until** MaxIter is reached or $\frac{\|\mathbf{Y}_{i+1} - \mathbf{Y}_i\|_F}{\|\mathbf{Y}_{i+1}\|_F} \leqslant$ tol
```

additional (rough) gradient descent step using the previous two estimates $\mathbf{\Theta}_i$ and $\mathbf{\Theta}_{i-1}$. To compute $\nabla f(\mathbf{Y}_i)$ at each iteration, $\mathbf{Y}_i$'s shall satisfy the positive definiteness constraint.

We suggest two schemes for γ_i [15]: (A): $\gamma_i = \left(\frac{\alpha_i - 1}{\alpha_{i+1}}\right)$ where $\alpha_{i+1} = \frac{1+\sqrt{1+4\alpha_i^2}}{2}$ and $\alpha_1 = 1$ and, (B): $\gamma_i = \frac{1-\sqrt{\widetilde{\mu}\cdot\tau_i^*}}{1+\sqrt{\widetilde{\mu}\cdot\tau_i^*}}$. We identified that both strategies perform well in practice where scheme (A) is more stable when $\widehat{\mathbf{\Sigma}}$ is rank-deficient (non-strictly convex case).

Since we operate on $\mathbf{Y}_i$, we have to guarantee the positive definiteness of both $\mathbf{\Theta}_i$ and $\mathbf{Y}_i$ per iteration, leading to an additional Cholesky factorization calculation per iteration. A key lemma for an effcient implementation of Algorithm 2 is the following:

Lemma 4. *Given* $\mathbf{\Theta}_0 \succ 0$, $\mathbf{Y}_{i+1} \succ 0$ *implies* $\mathbf{\Theta}_i \succ 0$, $\forall i$.

Proof. If $\mathbf{Y}_{i+1} \succ 0$, then: $\mathbf{\Theta}_i + \gamma_i\left(\mathbf{\Theta}_i - \mathbf{\Theta}_{i-1}\right) \succ 0 \Rightarrow \mathbf{\Theta}_i\left(1+\gamma_i\right) \succ \gamma_i \mathbf{\Theta}_{i-1} \Rightarrow \mathbf{\Theta}_i \succ \beta_i \mathbf{\Theta}_{i-1}$, where $\beta_i := \frac{\gamma_i}{1+\gamma_i} > 0$, $\forall i$. Unfolding the recursion, we have:

$$\mathbf{\Theta}_i \succ \underbrace{\left(\min\{\beta_i, \beta_{i-1}, \dots, \beta_1\}\right)^{i-1}}_{>0} \mathbf{\Theta}_0 \succ 0, \ \forall i, \qquad \square$$

By Lemma 4, we can check the positive definiteness of $\mathbf{\Theta}_i$ through the Cholesky factorization of $\mathbf{Y}_{i+1}$.

7. EXPERIMENTS

Experimental configuration: we synthetically generate sparse inverse covariance matrices $\mathbf{\Sigma}^{-1}$, according to the simple model:

$$\mathbf{\Sigma}^{-1} = \mathbb{I} + \mathbf{\Omega}, \text{ such that } \mathbf{\Sigma}^{-1} \succ 0 \text{ and } \|\mathbf{\Sigma}^{-1}\|_0 = \kappa, \qquad (12)$$

where $\mathbf{\Omega} \in \mathbb{R}^{n \times n}$ contains random iid off-diagonal entries $\sim \mathcal{N}(0,1)$. Given $\mathbf{\Sigma}^{-1}$, we draw $\{\mathbf{x}_j\}_{j=1}^p \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ and calculate $\widehat{\mathbf{\Sigma}}$. Given the above, we consider two test settings:

(i) $n = 1000$, $p = n/2$ and, $\kappa = 2 \cdot 10^{-3} \cdot n^2$. To observe interpretable results, we set $\rho = 5 \cdot 10^{-2}$.

(ii) $n = 3000$, $p = 5n$ and, $\kappa = 10^{-3} \cdot n^2$. To observe interpretable results, we set $\rho = 4 \cdot 10^{-2}$.

Linear convergence: We empirically illustrate the convergence rate of the proposed schemes towrads a high-accuracy solution $\mathbf{\Theta}^*$ of (2); we retain a convergence analysis for an extended version. Let $n = 700, p = 5n, \rho = 2 \cdot 10^{-2}, \kappa = 0.01n^2$. Figure 1 depicts the linear convergence rate of the proposed schemes and their variants; FPSa uses an adaptive restart scheme [20]. In practice, we observe that the choice of ρ *heavily* affects the condition number of the problem and thus the convergence rate of first-order schemes.

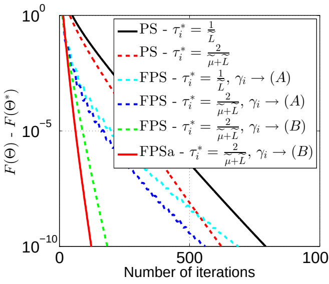

Fig. 1: *Convergence rates*

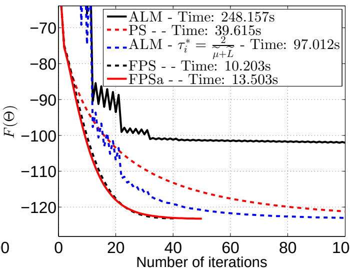

Fig. 2: *Comparison plot*

Setting (i)	ALM	PS	FPS	FPSa
$\frac{\|\mathbf{\Theta}^* - \mathbf{\Sigma}^{-1}\|_F}{\|\mathbf{\Sigma}^{-1}\|_F}$	0.44	0.414	**0.413**	**0.413**
Correct	1705	**1893**	**1893**	**1893**
Missed	291	**103**	**103**	**103**
Extra	365	232	**228**	**228**
Iterations	400	379	129	**114**
#Inversions	400	379	129	**114**
Setting (ii)	ALM	PS	FPS	FPSa
$\frac{\|\mathbf{\Theta}^* - \mathbf{\Sigma}^{-1}\|_F}{\|\mathbf{\Sigma}^{-1}\|_F}$	-	0.444	**0.43**	**0.43**
Correct	-	8710	**8725**	8724
Missed	-	290	**275**	276
Extra	-	**4**	**4**	**4**
Iterations	-	300	100	**92**
#Inversions	-	300	100	**92**

Table 1: *"Correct", "Missed" and "Extra" stand for the edges correctly identified, missed or added in the true graph, respectively. MaxIter = 400 and tol.* $= 10^{-8}$. *"-" depicts no results due to time overhead.*

List of algorithms: We compare our scheme against ALM [3], current state-of-the-art *first-order gradient method* to illustrate the effect of the step size selection. All codes are exclusively written in MATLAB.

Convergence comparison: Figure 2 summarizes the convergence performance of the aforementioned schemes. We simulate test setting (i). Here, "ALM - $\tau_i^* = \frac{2}{\widetilde{\mu}+\widetilde{L}}$"' corresponds to ALM [3] using τ_i^* in both steps of the algorithm, thus illustrating the universality of our step size selection. All algorithms use $\tau_i^* = \frac{2}{\widetilde{\mu}+\widetilde{L}}$ and $\gamma_i \rightarrow$ (B).

Sparsity pattern recovery performance: For each test setting, we record the median values over 50 Monte-Carlo realizations. Table 1 summarizes the results.

8. CONCLUSIONS

Many state-of-the-art gradient approaches for sparse inverse covariance estimation in GMRFs use heuristics to compute a step size which introduce additional "computational losses" due to matrix inversion recalculations or slow convergence. In this work, we present a first-order proximal method which, at its core, utilizes a novel adaptive step size selection procedure based on the self-concordance property of the objective value. Numerical results indicate that our methods overcome state-of-the-art first order methods. Moreover, our framework extends straightforwardly to many convex regularizers; following a simplistic avenue to solve the problem is valuable for the universal application of the algorithm to diverse problems.

9. REFERENCES

[1] J. Dahl, L. Vandenberghe, and V. Roychowdhury. Covariance selection for nonchordal graphs via chordal embedding. *Optimization Methods & Software*, 23(4):501–520, 2008.

[2] I.M. Johnstone. On the distribution of the largest eigenvalue in principal components analysis.(english. *Ann. Statist*, 29(2):295–327, 2001.

[3] K. Scheinberg, S. Ma, and D. Goldfarb. Sparse inverse covariance selection via alternating linearization methods. *arXiv preprint arXiv:1011.0097*, 2010.

[4] N. Meinshausen and P. Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010.

[5] K.C. Toh, M.J. Todd, and R.H. Tütüncü. Sdpt3a matlab software package for semidefinite programming, version 1.3. *Optimization Methods and Software*, 11(1-4):545–581, 1999.

[6] J.F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization methods and software*, 11(1-4):625–653, 1999.

[7] O. Banerjee, L. El Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research*, 9:485–516, 2008.

[8] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[9] J. Duchi, S. Gould, and D. Koller. Projected subgradient methods for learning sparse gaussians. *arXiv preprint arXiv:1206.3249*, 2012.

[10] C.J. Hsieh, M.A. Sustik, I.S. Dhillon, and P. Ravikumar. Sparse inverse covariance matrix estimation using quadratic approximation. *Advances in Neural Information Processing Systems (NIPS)*, 24, 2011.

[11] P.A. Olsen, F. Oztoprak, J. Nocedal, and S.J. Rennie. Newton-like methods for sparse inverse covariance estimation. *Optimization Online*, 2012.

[12] L. Li and K.C. Toh. An inexact interior point method for l 1-regularized sparse covariance selection. *Mathematical Programming Computation*, 2(3):291–315, 2010.

[13] X. Yuan. Alternating direction methods for sparse covariance selection. *preprint*, 2009.

[14] K. Scheinberg and I. Rish. Sinco-a greedy coordinate ascent method for sparse inverse covariance selection problem. *preprint*, 2009.

[15] Y. Nesterov. *Introductory lectures on convex optimization*. Kluwer Academic Publishers, 1996.

[16] D. Bertsekas. *Nonlinear programming*. Athena Scientific, 1995.

[17] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[18] P.L. Combettes and J.C. Pesquet. Proximal splitting methods in signal processing. *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212, 2011.

[19] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[20] B. O'Donoghue and E. Candes. Adaptive restart for accelerated gradient schemes. *arXiv preprint arXiv:1204.3982*, 2012.