A Penalty-free Infeasible Approach for a Class of Nonsmooth Optimization Problems over the Stiefel Manifold

Nachuan Xiao[*] Xin Liu[†] and Ya-xiang Yuan[‡]

March 28, 2021

Abstract

Transforming into an exact penalty function model with convex compact constraints yields efficient infeasible approaches for optimization problems with orthogonality constraints. For smooth and $\ell_{2,1}$-norm regularized cases, these infeasible approaches adopt simple and orthonormalization-free updating schemes and show high efficiency in some numerical experiments. However, to avoid orthonormalization while enforcing the feasibility of the final solution, these infeasible approaches introduce a quadratic penalty term, where an inappropriate penalty parameter can lead to numerical inefficiency. Inspired by penalty-free approaches for smooth optimization problems, we proposed a sequential linearized proximal gradient method (SLPG) for a class of optimization problems with orthogonality constraints and nonsmooth regularization term. This approach alternatively takes tangential steps and normal steps to improve the optimality and feasibility respectively. In SLPG, the orthonormalization process is invoked only once at the last step if high precision for feasibility is needed, showing that main iterations in SLPG are orthonormalization-free. Besides, both the tangential steps and normal steps do not involve the penalty parameter, and thus SLPG is penalty-free and avoids the inefficiency caused by possible inappropriate penalty parameter. We analyze the global convergence properties of SLPG where the tangential steps are inexactly computed. By inexactly computing tangential steps, for smooth cases and $\ell_{2,1}$-norm regularized cases, SLPG has a closed-form updating scheme, which leads to cheap tangential steps. Numerical experiments illustrate the advantages of SLPG when compared with existing first-order methods.

1 Introduction

1.1 Problem description

In this paper, we focus on a class of composite optimization problems with orthogonality constraints,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & f(X) + r(X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned} \tag{COS}$$

where the objective is the summation of two functions $f, r : \mathbb{R}^{n\times p} \mapsto \mathbb{R}$ satisfying the following blanket assumption.

Assumption 1 (blanket assumption). • *Function f is differentiable and $\nabla f(X)$ is locally Lipschitz continuous in $\mathbb{R}^{n\times p}$;*

[*] State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, China, (email: xnc@lsec.cc.ac.cn).

[†] State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, China (email: liuxin@lsec.cc.ac.cn).

[‡] State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, China, (email: yyx@lsec.cc.ac.cn).

- *Function r is convex and Lipschitz continuous in $\mathbb{R}^{n\times p}$;*
- *For any $X, G \in \mathbb{R}^{n\times p}$ and any $\eta > 0$, the problem*

$$\min_{D\in\mathbb{R}^{n\times p}} \quad \langle G, D\rangle + r(D) + \frac{1}{2\eta}\left\|D - X\right\|_{\mathrm{F}}^2$$

is of closed-form solution or can be solved efficiently by certain iterative approach.

The feasible region of the orthogonality constraints $X^\top X = I_p$ is the Stiefel manifold embedded in real matrix space $X \in \mathcal{S}_{n,p} := \{X \in \mathbb{R}^{n\times p} | X^\top X = I_p\}$. We also call it as the Stiefel manifold for brevity.

The optimization problems of the form (COS) have wide applications in data science and statistics. We mention a few of them in the following.

Problem 1 (**Sparse Principal Component Analysis).** *Principal component analysis (PCA) is a basic tool in data processing and dimensional reduction. It pursues the p leading eigenvectors of the empirical covariance matrix L associated with N samples in $\mathbb{R}^n$. Contemporary datasets often have a new feature that the dimension n is comparable with or even much larger than the samples N. At this point, we need to take into account the sparsity in the principal components for better representation and consistency. Mathematically, we consider the following sparse PCA model [9, 36], which admits a nonsmooth ℓ_1 norm regularization term.*

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & -\frac{1}{2}\mathrm{tr}\left(X^\top L X\right) + \gamma\left\|X\right\|_1 \\ s.\ t. \quad & X^\top X = I_p, \end{aligned} \tag{1.1}$$

where γ is a positive parameter controlling the sparsity.

Problem 2 (**$\ell_{2,1}$-norm regularized PCA).** *To pursue the sparsity in the features (variables) of the principal components, we can impose the row sparsity to the classical PCA model and arrive at the following $\ell_{2,1}$-norm regularized PCA problem [46, 9].*

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & -\frac{1}{2}\mathrm{tr}\left(X^\top L X\right) + \sum_{i=1}^{n}\gamma_i\left\|X_{i\cdot}\right\|_2 \\ s.\ t. \quad & X^\top X = I_p, \end{aligned} \tag{1.2}$$

where $X_{i\cdot}$ and γ_i are the i-th row of matrix $X \in \mathbb{R}^{n\times p}$ and a positive parameter controlling the row sparsity, respectively, for all $i = 1, ..., n$. The problem (1.2) *is also known as the Coordinate-independent Sparse Estimation [14].*

When the nonsmooth part vanishes, i.e. $r = 0$, the objective function of (COS) reduces to a smooth function. There are many applications in this scenario as well, for instance, the discretized Kohn-Sham energy minimization problem arising in material sciences.

Problem 3 (Discretized Kohn-Sham Energy Minimization)**.** *Kohn-Sham density functional theory (KSDFT) [33] is widely used in electronic structure calculation. In the last step of KSDFT, it requires to minimize the following discretized Kohn-Sham energy function over the Stiefel manifold.*

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \frac{1}{4}\mathrm{tr}\left(X^\top L X\right) + \frac{1}{2}\mathrm{tr}\left(X^\top V_{ion} X\right) + \frac{1}{4}\rho^\top L^\dagger \rho + \frac{1}{2}\rho^\top \epsilon_{\mathrm{xc}}(\rho) \\ s.t. \quad & X^\top X = I_p, \end{aligned} \tag{1.3}$$

where $L \in \mathbb{R}^{n\times n}$ and diagonal matrix $V_{\mathrm{ion}} \in \mathbb{R}^{n\times n}$ refers to the Laplace operator in the planewave basis and discretized local ionic potential, respectively, $\rho := \mathrm{diag}(XX^\top)$ denotes the charge density, and $\epsilon_{\mathrm{xc}} : \Re^n \mapsto \Re^n$ stands for the exchange correlation function.

Remark 1. *The blanket assumption is not strict at all, since it holds at all the instances we listed above. Moreover, it is the same as those imposed in [12, 30, 31].*

1.2 Existing methods

On minimizing smooth objectives over the Stiefel manifold, there exist several efficient approaches, such as gradient-based methods [37, 39, 2], conjugate gradient methods [18, 1], projection-based methods [4, 17], constraint preserving updating scheme [47, 32], Newton methods [27], trust-region methods [3], first-order methods with multipliers correction framework [19], infeasible methods [20, 49], etc. Interested readers are referred to the book [4], the survey paper [28] and the references therein. It is worth mentioning that several infeasible approaches have been proposed and show their high efficiency in solving optimization problems over the Stiefel manifold. The ALM-based approaches PLAM and PCAL[20] update the Lagrangian multipliers by an explicit expression derived by the first-order stationarity conditions. Such explicit expression involves the gradient of the objective, and hence these algorithms can only tackle the problems with smooth objective function. Gao et al. [21] provide a clear route of applying PCAL to the electronic structure calculation. Xiao et al. [49] present a novel penalty function with compact convex constraints (PenC). In the framework of PenC, they propose approximate projected gradient and Newton methods PenCF and PenCS, respectively. Hu et al. [29] propose an unconstrained penalty model for sparse dictionary learning and dual principal component pursuit.

However, most of the above-mentioned approaches can hardly be applied to the problem with nonsmooth objective function directly. The approaches for solving (COS) with $r \neq 0$ are not as many as those for smooth minimization. We review a few representative ones in the following.

The first class of approaches are based on the splitting and alternating. The splitting method for orthogonality constrained problem (SOC) [34] introduces auxiliary variables to split the objective function and the orthogonality constraints, and then applies the alternating direction method of multipliers (ADMM) to solve the equivalent splitting model. The subproblem related to the objective function lacks closed-form solution in general which is a main limit of SOC. Meanwhile, Rosman et al. [41] propose a variable splitting framework based on augmented Lagrangian method for problems on imaging processing, which can also be extended to solve optimization problems on $\mathcal{S}_{n,n}$. Besides, Chen et al. [13] propose a proximal alternating minimization approach based on augmented Lagrangian method (PAMAL). Different from SOC, PAMAL develops an equivalent model by introducing two blocks of variables to split the orthogonality constraints, smooth and nonsmooth terms apart. PAMAL invokes the augmented Lagrangian method (ALM) framework and block coordinate descent (BCD) method to solve the split model and the subproblems related to the primal variables, respectively.

The second classes of approaches apply the proximal gradient method to tackle the nonsmooth term in (COS). Chen et al. [12] propose the Riemannian proximal gradient method (ManPG) and its accelerated version, ManPG-Ada. The main iteration, which occupies the main computational cost of ManPG or ManPG-Ada, is to compute the following proximal mapping restricted to the tangent space $\mathcal{T}_{X_k} := \{\Delta \in \mathbb{R}^{n\times p} | \Delta^\top X_k + {X_k}^\top \Delta = 0\}$ of the Stiefel manifold.

$$\min_{D \in X_k + \mathcal{T}_{X_k}} \quad \langle D, \nabla f(X_k) \rangle + r(D) + \frac{1}{2\eta_k} \left\| D - X_k \right\|_{\mathrm{F}}^2, \tag{1.4}$$

where $\eta_k > 0$ is the stepsize. The subproblem (1.4) is a nonsmooth convex optimization problem without closed-form solution in general and can be solved by the semi-smooth Newton method (SSN) [43]. Their numerical experiments show that both ManPG and ManPG-Ada outperform the existing splitting and alternating based approaches SOC and PAMAL. Recently, Huang et al. present a Riemannian version of fast iterative shrinkage-thresholding algorithm with safeguard (AManPG) in [30], which exhibits the accelerated behavior over the Riemannian proximal gradient method. Nevertheless, no convergence rate analysis is presented for AManPG. They also propose a modified Riemannian proximal gradient method (RPG) and its accelerated version (ARPG), respectively, in [31]. They show the $\mathcal{O}(\frac{1}{k})$-convergence rate of RPG and ARPG. However, the proximal mapping subproblems in both RPG and ARPG are even more expensive to solve than ManPG due to their nonsmoothness and nonconvexiety. Thus, ARPG and RPG are usually slower than AManPG and ManPG-Ada in solving optimization problems on the Stiefel manifold as illustrated in [31].

The key motivation of PenC is to utilize the explicit expression of the Lagrangian multipliers at first-order stationary points, which involves the Euclidian gradient of the objective function. Hence, it can hardly be generalized to the nonsmooth case, in which the gradient of the objective function is absent. Xiao et al. [50] extend PenC to a special case of (COS) in which r takes the $\ell_{2,1}$-norm like (1.2) in Problem 2. Although the subdifferential of r in this case is set-valued, the term $X^\top \partial r(X)$ is single-valued. Based on this observation, the authors of [50] propose the corresponding PenC model and a proximal gradient method called PenCPG. In PenCPG, the proximal subproblem is of closed-form solution, which leads to its numerical superiority when compared with the existing Riemannian proximal gradient approaches in solving $\ell_{2,1}$-norm regularized problems.

However, if the nonsmooth term r is not a $\ell_{2,1}$-norm, the term $X^\top \partial r(X)$ is set-valued in general. Hence, the Lagrangian multipliers at any stationary point no longer have closed-form expression. Therefore, the PenC model does not apply to (COS) in general.

Another limitation of PenC based approaches is that their numerical performances are related to the choice of the penalty parameter, as reported in [19, 49, 50]. But slow convergence or even divergence occurs, if the penalty parameter is out of such range. The authors in [20] provide heuristic way to select the penalty parameter without theoretical guarantee.

1.3 Motivation

In order to develop an efficient infeasible approach for solving (COS) which is not sensitive to the penalty parameter, we borrow the idea from a class of sequential quadratic programming (SQP) approaches developed for solving the equality constrained smooth nonlinear optimization problems. These approaches include the inexact-restoration method proposed by Martinez [38], the nonmonotone trust-region based SQP methods proposed by Ulbrich and Ulbrich [44], Gould and Toint [23], Liu and Yuan [35], and Chen et al. [15], respectively. In particular, the authors in [23] and [35] provide inexact strategies to tackle the SQP subproblems. Besides, the approaches presented in [45] and [42] utilize the nonmonotone filter techniques. However, all of these approaches invoke the second-order oracle or use the first-order information to approximate the Hessian of the objective function or its Lagrangian. Hence, these approaches are only valid in the smooth problems. To the best of our knowledge, there are few approaches for solving the nonsmooth optimization problems such as (COS) by adopting the SQP-like techniques.

Our main idea is to reduce the objective function in the tangent space and to improve the feasibility in the normal space alternatively. We first approximate the objective function by a proximal linearized model and minimize it on an affine subspace spanned at the current iterate which is parallel to a tangent space of the Stiefel manifold. We then invoke a normal step which searches in the range space of the Jacobian of the constraints $X^\top X - I_p = 0$ to reduce the feasibility violation.

1.4 Contributions

We propose a first-order penalty-free infeasible approach, called sequential linearized proximal gradient method (SLPG), for solving a class of composite optimization problems with orthogonality constraints (COS). In each iteration, SLPG takes the tangential and the normal steps one after the other, both of which do not involve any orthonormalization procedure or updating of penalty parameters. Consequently, SLPG enjoys high scalability and avoids the numerical inefficiency from inappropriately selected penalty parameters. We discuss how to solve the tangential subproblems inexactly, which is different from the existing approaches since the iterates are no longer feasible. We provide a novel idea to conduct the normal steps which simultaneously have both low computational cost and fast convergence to the feasible region. To combine the tangential and normal steps together, the subsequence convergence as well as the worst-case complexity of SLPG can be established under mild assumptions. Furthermore, when the nonsmooth term of (COS) has a special structure, i.e. the Lagrange multipliers with respect to the orthogonality constraints are of closed-form expressions, the tangential steps of SLPG enjoy closed-form approximate solutions, and hence an inner loop to solve the tangential subproblem is waived. The efficiency and robustness of SLPG are illustrated by a set of numerical experiments on the sparse PCA, the $\ell_{2,1}$-norm regularized PCA, and the discretized Kohn-Sham energy minimization problems. SLPG visibly outperforms the state-of-the-art feasible approaches in solving those nonsmooth problems. It exhibits its prominent robustness when compared with the existing infeasible approaches.

1.5 Notations and Organization

Let $\mathbb{S}^{p\times p} := \{A \mid A \in \mathbb{R}^{p\times p},\ A = A^\top\}$ be the set containing all the real symmetric $p \times p$ matrices. We use I_p to denote the $p \times p$ identity matrix. The entry in the i-th row and the j-th column of a matrix $X \in \mathbb{R}^{n\times p}$ is denoted by X_{ij}. For brevity, we use $\|X\|_1$ to represent the component-wise ℓ_1 norm, i.e. $\|X\|_1 = \sum_{i,j} |X_{ij}|$. The Euclidean inner product of two matrices $X, Y \in \mathbb{R}^{n\times p}$ is defined as $\langle X, Y\rangle = \mathrm{tr}(X^\top Y)$, where $\mathrm{tr}(A)$ is the trace of a matrix $A \in \mathbb{R}^{p\times p}$. $\|\cdot\|_2$ and $\|\cdot\|_\mathrm{F}$ represent the 2-norm and the Frobenius norm, respectively. For a positive semi-definite matrix $A \in \mathbb{R}^{p\times p}$, $A^{\frac{1}{2}}$ refers to the unique positive semi-definite matrix satisfying $A^{\frac{1}{2}}A^{\frac{1}{2}} = A$ and $A^{-\frac{1}{2}}$ is its inverse.

The rest of this paper is organized as follows. In Section 2, we present the detailed description of SLPG , and introduce the practical implementations on computing the tangential and the normal steps. We establish the convergence analysis for SLPG in Section 3. The preliminary numerical experiments are reported in Section 4. Finally, we conclude this paper in Section 5.

2 Algorithm Description

In this section, we mainly propose the framework of SLPG. We first provide some necessary preliminary definitions. Then we present the mathematical formulations of the tangential and normal subproblems and introduce how to solve them respectively. Finally, we demonstrate the complete algorithm framework.

2.1 Preliminary

We first review the definition of Clark's subdifferential [16] for nonsmooth functions.

Definition 1 ([16, 40]). *For any Lipschitz continuous f on $\mathbb{R}^{n\times p}$, the generalized directional derivative of f in the direction $D \in \mathbb{R}^{n\times p}$ is defined by,*

$$f^o(X, D) := \limsup_{Y\to X, t\to 0^+} \frac{f(Y + tD) - f(Y)}{t}. \tag{2.1}$$

Based on generalized directional derivative of f, the Clark's subdifferential ("subdifferential" for brevity) of f is defined by,

$$\partial f(X) := \left\{W \in \mathbb{R}^{n\times p} \middle| \langle W, D\rangle \leq f^o(X, D) \text{ for any } D \in \mathbb{R}^{n\times p}\right\}. \tag{2.2}$$

Next we describe the stationarity of (COS), which is same as [12, Definition 3.3].

Definition 2. *Under the Assumption 1, we call X a first-order stationary point of* (COS) *if and only if there exists $W \in \partial r(X)$ such that*

$$\begin{cases} \nabla f(X) + W - X\Phi\left(X^\top\left(\nabla f(X) + W\right)\right) = 0, \\ X^\top X = I_p. \end{cases} \tag{2.3}$$

Definition 3. *A operator $T : \mathbb{R}^{p\times p} \mapsto \mathbb{R}^{p\times p}$ is nonexpansive if and only if there exists a constant $c \in [0, 1]$ such that*

$$\|T(\Lambda_1) - T(\Lambda_2)\|_\mathrm{F} \leq c\,\|\Lambda_1 - \Lambda_2\|_\mathrm{F}$$

holds for any $\Lambda_1, \Lambda_2 \in \mathbb{R}^{p\times p}$.

2.2 Computing the tangential step

Suppose X_k is the current iterate, we define the affine subspace $\mathcal{A}_k$ as

$$\mathcal{A}_k := \left\{ X \in \mathbb{R}^{n\times p} \,\middle|\, \Phi(X^\top X_k) = {X_k}^\top X_k \right\}.$$

Here $\Phi : \mathbb{R}^{p\times p} \to \mathbb{R}^{p\times p}, \Phi(M) = \frac{M+M^\top}{2}$ is an operator that symmetrize the square matrices in $\mathbb{R}^{p\times p}$. To reduce the function value, we minimize the following proximal linearized approximation of the objective function with stepsize η_k on the space $\mathcal{A}_k$.

$$\min_{D\in\mathcal{A}_k} \langle \nabla f(X_k), D\rangle + r\left(D\right) + \frac{1}{2\eta_k}\left\|D - X_k\right\|_{\mathrm{F}}^2. \tag{2.4}$$

We call (2.4) the tangential subproblem for convenience hereinafter. Different with the tangential step in [12], the subproblem (2.4) is constructed on an infeasible point X_k.

By simple calculations, we can obtain the following KKT condition of the convex optimization problem (2.4).

$$\begin{cases} 0 \in \nabla f(X_k) - X_k\Lambda + \partial r(D) + \dfrac{1}{\eta_k}(D - X_k), \\ \Phi(D^\top X_k) = {X_k}^\top X_k, \end{cases} \tag{2.5}$$

where $\Lambda \in \mathbb{S}^{p\times p}$ is the Lagrangian multiplier of the linear constraint $D \in \mathcal{A}_k$.

Once Λ is fixed, the first relation in (2.5) determines

$$D = \mathrm{prox}_{\eta_k}(\nabla f(X_k) - X_k\Lambda; X_k),$$

where the proximal mapping $\mathrm{prox}_{\eta_k} : \mathbb{R}^{n\times p} \otimes \mathbb{R}^{n\times p} \mapsto \mathbb{R}^{n\times p}$ is defined by

$$\mathrm{prox}_{\eta_k}(G; X_k) := \operatorname*{arg\,min}_{D\in\mathbb{R}^{n\times p}} \langle G, D\rangle + r(D) + \frac{1}{2\eta_k}\left\|D - X_k\right\|_{\mathrm{F}}^2.$$

Then it is clear that the KKT condition (2.5) is equivalent to the nonlinear equation $E(\Lambda) = 0$, where

$$E(\Lambda) := \Phi\left(\left(\mathrm{prox}_{\eta_k}(\nabla f(X_k) - X_k\Lambda; X_k) - X_k\right)^\top X_k \right).$$

This equation can be rewritten as the fixed point equation

$$\Lambda - tE(\Lambda) = \Lambda, \quad \text{where } t > 0. \tag{2.6}$$

We adopt the following Arrow-Hurwicz algorithm proposed by Beale et al. [6] to solve (2.6).

Algorithm 1 Fixed point iteration

```
Require: Input data: current iterate $X_k$, parameter $\eta_k$;
Choose initial guess $\Lambda_0 \in \mathbb{S}^{p\times p}$, set $j := 0$;
while not terminate do
    Calculate the proximal mapping: $D_j = \mathrm{prox}_{\eta_k}(\nabla f(X_k) - X_k\Lambda_j; X_k)$;
    Main update: $\Lambda_{j+1} = \Lambda_j - \frac{1}{\eta_k}E(\Lambda_j)$;
    Set $j := j + 1$;
end while
Return $Y_k := D_j$.
```

Chambolle et al. [10] have provided an $\mathcal{O}\left(\frac{1}{k}\right)$ convergence rate of the Arrow-Hurwicz algorithm locally. Later on, He et al. [24] have shown that the Arrow-Hurwicz algorithm, as a special case of their primal-dual hybrid gradient algorithm (PDHG), enjoys an $\mathcal{O}\left(\frac{1}{k}\right)$ convergence rate in the ergodic sense under mild conditions containing our case.

In our infeasible framework, we actually do not need an accurate solution to the tangential subproblem (2.4). More specifically, in Algorithm 1, we adopt the following terminating condition for the residual.

Condition 1. *There exist $C > 0$ such that $Y_k \in \mathbb{R}^{n\times p}$ returned by Algorithm 1 satisfy*

$$\left\|\Phi\left((Y_k - X_k)^\top X_k\right)\right\|_{\mathrm{F}} \leq c\eta_k \left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}. \tag{2.7}$$

2.3 A practical inexact tangential step in special cases

In [49] and [50], it is shown that for two special cases of (COS) with $r(X) = 0$ and $r(X) = \sum_{i=1}^n \gamma_i \|X_{i\cdot}\|_2$, the Lagrangian multipliers have explicit expressions

$$\Lambda(X) \;=\; \Phi(X^\top \nabla f(X)) \quad \text{and} \tag{2.8}$$

$$\Lambda(X) \;=\; \Phi(X^\top \nabla f(X)) + \sum_{i=1}^{n} \gamma_i S(X_{i\cdot}^\top), \tag{2.9}$$

respectively, at any first-order stationary point, where S is defined by

$$S(x) := \begin{cases} \frac{xx^\top}{\|x\|_2}, & \text{if } x \neq 0; \\ 0_p, & \text{otherwise,} \end{cases} \tag{2.10}$$

and 0_p is the zero vector in $\mathbb{R}^p$.

Here we can propose an alternative way to inexactly solve the tangential subproblem (2.4) other than Algorithm 1 with Condition 1 by using the expressions (2.8) and (2.9) to estimate the multipliers of (2.4) and then get the proximal mapping. Namely, we adopt the following two step algorithm.

Algorithm 2 Using explicit expressions

```
Require: Input data: current iterate $X_k$, parameter $\eta_k$;
1: If $r(X) = 0$, calculate $\Lambda$ by (2.8) with $X = X_k$;
2: If $r(X) = \sum_{i=1}^n \gamma_i \|X_{i\cdot}\|_2$, calculate $\Lambda_k$ by (2.9) with $X = X_k$;
3: Calculate the proximal mapping: $Y_k = \mathrm{prox}_{\eta_k}(\nabla f(X_k) - X_k\Lambda_k; X_k)$;
4: Return $Y_k$.
```

2.4 Computing the normal step

After obtaining Y_k, an inexact solution of the tangential subproblem, we need to consider a normal step to reduce the feasibility violation. A usual way to realize it is to pull this intermediate point back to the Stiefel manifold through certain projection, i.e. orthonormalization process, such as the QR decomposition, the polar decomposition, and so on. As we know, orthonormalization is usually unscalable and expensive when p is large. An accurate normal step usually does not help much for the overall performance as the tangential step is an inexact solution of a linear approximate model. Therefore, we consider to compute the orthonormalization inexactly to balance the accuracies of the tangential and the normal steps. A parallelizable algorithm proposed in [25] computes the polar decomposition by adopting the Padé approximant, whose main computational cost can be attributed to the inverse of a series of $p \times p$ matrices which can be realized by solving linear equations.

The Taylor expansion of $z^{-\frac{1}{2}}$ at $z = 1$ to order one is $z^{-\frac{1}{2}} = 1 - \frac{1}{2}(z-1) + \mathcal{O}((z-1)^2)$. Let $z = X^\top X$, we have

$$\left\|(X^\top X)^{-\frac{1}{2}} - \left(\frac{3}{2}I_p - \frac{1}{2}X^\top X\right)\right\|_{\mathrm{F}} = \mathcal{O}\left(\left\|X^\top X - I_p\right\|_{\mathrm{F}}^2\right).$$

Hence, the polar decomposition at the intermediate iterate $Y_k({Y_k}^\top Y_k)^{-\frac{1}{2}}$ can be approximated by the following normal step

$$X_{k+1} = Y_k \left(\frac{3}{2} I_p - \frac{1}{2} {Y_k}^\top Y_k \right). \tag{2.11}$$

Next we show how the above normal step reduce the feasibilit violation.

Lemma 1. *For any $X \in \mathbb{R}^{n\times p}$ satisfying $\left\|X^\top X - I_p\right\|_2 \leq \frac{1}{4}$, let $\hat{X} := X\left(\frac{3}{2}I_p - \frac{1}{2}X^\top X\right)$, then it holds that*

$$\left\|\hat{X}^\top \hat{X} - I_p\right\|_{\mathrm{F}} \leq \frac{13}{16}\left\|X^\top X - I_p\right\|_{\mathrm{F}}^2.$$

Proof. It directly follows from the condition $\left\|X^\top X - I_p\right\|_2 \leq \frac{1}{4}$ that

$$\left\|I_p - \frac{1}{4}X^\top X\right\|_2 = \left\|\frac{3}{4}I_p + \frac{1}{4}(I_p - X^\top X)\right\|_2 \leq \frac{3}{4} + \frac{1}{4}\left\|X^\top X - I_p\right\|_2 \leq \frac{13}{16}.$$

Together with the definition of $\hat{X}$, we have

$$\begin{aligned} \left\|\hat{X}^\top \hat{X} - I_p\right\|_{\mathrm{F}} &= \left\|X^\top X - I_p + (I_p - X^\top X)X^\top X + \frac{1}{4}X^\top X(X^\top X - I_p)^2\right\|_{\mathrm{F}} \\ &= \left\|\left(I_p - \frac{1}{4}X^\top X\right)(X^\top X - I_p)^2\right\|_{\mathrm{F}} \leq \frac{13}{16}\left\|(X^\top X - I_p)^2\right\|_{\mathrm{F}} \leq \frac{13}{16}\left\|X^\top X - I_p\right\|_{\mathrm{F}}^2. \end{aligned} \tag{2.12}$$

□

2.5 Algorithm

Now, we are ready to present the framework of our SLPG algorithm which alternatively takes the tangential and the normal steps introduced in the Subsections 2.2, 2.4, respectively.

Algorithm 3 Sequential Linearized Proximal Gradient method (SLPG)

```
Require: Input data: functions $f$ and $r$;
 1: Choose initial guess $X_0$, set $k := 0$;
 2: while not terminate do
 3:     Choose parameter $\eta_k$;
 4:     Solve the tangential subproblem (2.4) inexactly to make Condition 1 hold by Algorithm 1, and
        obtain $Y_k$;
 5:     Compute the normal step (2.11), and obtain $X_{k+1}$;
 6:     Set $k := k + 1$;
 7: end while
 8: if need post-process then
 9:     Compute $X_k := U_k V_k^\top$, where $X_k = U_k \Sigma_k V_k^\top$ is the singular value decomposition (SVD) of $X_k$
        in economic size;
10: end if
11: Return $X_k$.
```

The post-process stated in Line 9 of Algorithm 3 pursues an accurate feasible solution if necessary. As shown later in theoretical and numerical analyses, such post-process does not affect the substationarity much. In addition, it can further reduce the function value while decreasing the feasibility violation to machine precision.

Remark 2. *It is also worth mentioning that if the normal step in SLPG takes the orthonormalization process, the sequence $\{X_k\}$ generated by SLPG is on the Stiefel manifold. In addition, the tangential subproblem is strictly on the tangent space of the Stiefel manifold at X_k. By choosing suitable parameter c, SLPG reduces to the existing approach ManPG [12]. In other word, ManPG can be regarded as a special variant of SLPG in which both of the tangential and the normal steps are computed more precisely.*

3 Global Convergence of SLPG

In this section, we first establish the global convergence of SLPG without the post-process by constructing a merit function and evaluating the sufficient function value reduction. For convenience, when we mention Algorithm 3 in the first two subsections in this section, the post-process is switched off. Then we demonstrate that the post-process provides further function value reduction. For convenience, we define the following constants at the very beginning:

$$\begin{aligned} L_f &:= \sup_{X,Y\in\Omega_{1/2}} \frac{|f(X)-f(Y)|}{\|X-Y\|_{\mathrm{F}}} = \sup_{X\in\Omega_{1/2}} ||\nabla f(X)||_{\mathrm{F}}, \\ L_{f'} &:= \sup_{X,Y\in\Omega_{1/2}} \frac{\|\nabla f(X)-\nabla f(Y)\|_{\mathrm{F}}}{\|X-Y\|_{\mathrm{F}}}, \\ L_r &:= \sup_{X,Y\in\Omega_{1/2}} \frac{|r(X)-r(Y)|}{\|X-Y\|_{\mathrm{F}}}, \end{aligned}$$

where $\Omega_r := \{X \mid \|X^\top X - I_p\|_{\mathrm{F}} \leq r\}$, for any given $r \geq 0$. We also introduce a new assumption on the parameter sequence $\{\eta_k\}$.

Assumption 2. *Assume that the parameters in Algorithm 3 satisfies $\eta_k \in \left[\frac{\tilde{\eta}}{2}, \tilde{\eta}\right]$ for all $k = 0, 1, ...,$ where*

$$\tilde{\eta} = \min\left\{\frac{1}{6(1+c)(L_f+L_r+c+1)}, \frac{1}{2L_{f'}+8(L_f+L_r)+3}\right\},$$

and c is defined in Condition 1.

3.1 Preliminary properties of the iterate sequences

We first demonstrate some properties of the iterate sequences $\{X_k\}$ and $\{Y_k\}$ generated by SLPG including the boundedness and the reduction on the feasibility under a mild assumption on the initial guess.

Lemma 2. *Suppose the sequences $\{X_k\}$ and $\{Y_k\}$ are generated by Algorithm 3. Then, it holds that*

$$\left\|{Y_k}^\top Y_k - I_p\right\|_{\mathrm{F}} \leq (1+2c\eta_k)\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} + \|Y_k - X_k\|_{\mathrm{F}}^2. \tag{3.1}$$

Proof. Let $\tilde{D} := Y_k - X_k$, we have

$$\begin{aligned} & \left\|{Y_k}^\top Y_k - I_p\right\|_{\mathrm{F}} = \left\|{X_k}^\top X_k + 2\Phi(\tilde{D}^\top X_k) + \tilde{D}^\top\tilde{D} - I_p\right\|_{\mathrm{F}} \\ \leq\ & \left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} + 2\left\|\Phi(\tilde{D}^\top X_k)\right\|_{\mathrm{F}} + \left\|\tilde{D}^\top\tilde{D}\right\|_{\mathrm{F}} \\ \leq\ & (1+2c\eta_k)\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} + \|Y_k - X_k\|_{\mathrm{F}}^2. \end{aligned}$$

Here the last inequality follows from Condition 1. □

Lemma 2 shows that the tangential step may increase the feasibility violation. But fortunately, it can be controlled in some senses. Next, we investigate the boundedness of the iterate sequences with a suitable initial guess.

Lemma 3. *Suppose that Assumption 2 holds and $X_k \in \Omega_\tau$ with*

$$\tau = \frac{1}{4(1+c)^2} \tag{3.2}$$

Then, we have

$$\begin{aligned}&\langle Y_k, \nabla f(X_k)\rangle + r(Y_k) + \frac{1}{2\eta_k}\left\|Y_k - X_k\right\|_{\mathrm{F}}^2\\ \leq\ &\langle X_k, \nabla f(X_k)\rangle + r(X_k) + \frac{6c(5L_f + 5L_r + c)\eta_k}{25}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}.\end{aligned} \tag{3.3}$$

Proof. For convenience, we denote $p_k(D) := \langle D, \nabla f(X_k)\rangle + r(D) + \frac{1}{2\eta_k}\left\|D - X_k\right\|_{\mathrm{F}}^2$. It directly follows from the fact $X_k \in \Omega_\tau$ that ${X_k}^\top X_k$ is non-singular and hence we can define $Z_k := X_k + X_k({X_k}^\top X_k)^{-1}\Phi((Y_k - X_k)^\top X_k)$. By the definition of Z_k, we first obtain

$$\Phi((Z_k - X_k)^\top X_k) = \Phi((Y_k - X_k)^\top X_k).$$

Besides, together with Condition 1, the distance between Z_k and X_k can be estimated by

$$\left\|Z_k - X_k\right\|_{\mathrm{F}} \leq \left\|X_k({X_k}^\top X_k)^{-1}\right\|_2 \left\|\Phi((Y_k - X_k)^\top X_k)\right\|_{\mathrm{F}} \leq \frac{6c\eta_k}{5}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}},$$

where the last inequality results from the fact that $\left\|X_k({X_k}^\top X_k)^{-1}\right\|_2 \leq \frac{2\sqrt{3}}{3} < 6/5$ which is implied by the inclusion $X_k \in \Omega_\tau$.

Then we have

$$\begin{aligned}&\left|p_k(Z_k) - p_k(X_k)\right| \leq \left|\langle Z_k - X_k, \nabla f(X_k)\rangle\right| + \left|r(Z_k) - r(X_k)\right| + \frac{1}{2\eta_k}\left\|Z_k - X_k\right\|_{\mathrm{F}}^2\\ \leq\ &(L_f + L_r)\left\|Z_k - X_k\right\|_{\mathrm{F}} + \frac{1}{2\eta_k}\left\|Z_k - X_k\right\|_{\mathrm{F}}^2\\ \leq\ &\frac{6c(L_f + L_r)\eta_k}{5}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} + \frac{18c^2\eta_k}{25}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}^2.\end{aligned} \tag{3.4}$$

On the other hand, we consider the following optimization problem,

$$\begin{aligned}\min_{D\in\mathbb{R}^{n\times p}}\ &\langle D, \nabla f(X_k)\rangle + r(D) + \frac{1}{2\eta_k}\left\|D - X_k\right\|_{\mathrm{F}}^2\\ \text{s.t. }&\Phi({X_k}^\top(D - X_k)) = \Phi({X_k}^\top(Y_k - X_k)).\end{aligned} \tag{3.5}$$

As described in Algorithm 1, there exists a symmetric matrix Λ_k such that $Y_k = \mathrm{prox}_{\eta_k}(\nabla f(X_k) - X_k\Lambda_k; X_k)$. Then by Rockafellar and Wets [40], we have

$$0 \in \nabla f(X_k) - X_k\Lambda_k + \frac{1}{\eta_k}(Y_k - X_k) + \partial r(Y_k) = \partial p_k(Y_k) - X_k\Lambda_k. \tag{3.6}$$

Therefore, there exists $W_k \in \partial p_k(Y_k)$ such that $W - X_k\Lambda_k = 0$. Therefore, for any feasible D in (3.5), we have

$$\begin{aligned}&\langle W, D - Y_k\rangle = \langle X_k\Lambda_k, D - Y_k\rangle = \left\langle \Lambda_k, {X_k}^\top(D - Y_k)\right\rangle = \left\langle \Lambda_k, \Phi({X_k}^\top(D - Y_k))\right\rangle\\ =\ &\left\langle \Lambda_k, \Phi({X_k}^\top(D - X_k))\right\rangle - \left\langle \Lambda_k, \Phi({X_k}^\top(Y_k - X_k))\right\rangle = 0 - 0 = 0.\end{aligned} \tag{3.7}$$

Then together with [26, Theorem 1.1.1], we can conclude that Y_k is the global minimizer of (3.5), and hence $p_k(Y_k) \leq p_k(Z_k)$. Recall the inequality (3.4), we arrive at

$$\begin{aligned}&p_k(Y_k) - p_k(X_k) \leq p_k(Z_k) - p_k(X_k) \leq \left|p_k(Z_k) - p_k(X_k)\right|\\ \leq\ &\frac{6\eta_k c(L_f + L_r)}{5}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} + \frac{18c^2\eta_k}{25}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}^2\\ \leq\ &\frac{6c(5L_f + 5L_r + c)\eta_k}{25}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}.\end{aligned}$$

Here the last inequality follows the the inclusion $X_k \in \Omega_\tau \subset \Omega_{1/3}$. Then we complete the proof. □

Lemma 4. *Suppose that Assumption 2 holds and the iterate sequences $\{X_k\}$ and $\{Y_k\}$ are generated by Algorithm 3 initiated from X_0 satisfying $X_0 \in \Omega_\tau$ with τ defined in* (3.2). *Then for any $k = 0, 1, ...,$ it holds*

$$\max\left\{\|Y_k - X_k\|_{\mathrm{F}}, \left\|{Y_k}^\top Y_k - I_p\right\|_{\mathrm{F}}\right\} \leq \tfrac{1}{2(1+c)}, \quad \left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} \leq \tfrac{1}{4(1+c)^2}, \tag{3.8}$$

$$\left\|{X_{k+1}}^\top X_{k+1} - I_p\right\|_{\mathrm{F}} \leq \tfrac{13}{32}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} + \tfrac{13}{32}\|Y_k - X_k\|_{\mathrm{F}}^2. \tag{3.9}$$

Proof. We use mathematical induction. Clearly $\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} \leq \frac{1}{4(1+c)^2}$ holds for $k = 0$. From Lemma 3, we have

$$\frac{1}{2\eta_k}\|Y_k - X_k\|_{\mathrm{F}}^2 \leq (L_f + L_r)\|Y_k - X_k\|_{\mathrm{F}} + \frac{6\eta_k c(5L_f + 5L_r + c)}{25}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}. \tag{3.10}$$

Suppose that $\|Y_k - X_k\|_{\mathrm{F}} > 3\eta_k(L_f + L_r + c + 1)$, we have

$$\begin{aligned} &\frac{1}{2\eta_k}\|Y_k - X_k\|_{\mathrm{F}}^2 = \frac{1}{3\eta_k}\|Y_k - X_k\|_{\mathrm{F}}^2 + \frac{1}{6\eta_k}\|Y_k - X_k\|_{\mathrm{F}}^2 \\ > \quad &(L_f + L_r)\|Y_k - X_k\|_{\mathrm{F}} + \frac{3}{2}(L_f + L_r + c + 1)^2\eta_k \\ > \quad &(L_f + L_r)\|Y_k - X_k\|_{\mathrm{F}} + \frac{6c(5L_f + 5L_r + c)\eta_k}{25}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}, \end{aligned}$$

where the last inequality results from the inclusion $X_k \in \Omega_\tau$. Clearly, this statement contradicts the inequality (3.10). Therefore, we have

$$\|Y_k - X_k\|_{\mathrm{F}} \leq 3\eta_k(L_f + L_r + c + 1) \leq \frac{1}{2(1+c)},$$

where the last inequality follows from Assumption 2.

On the other hand, by recalling Lemmas 1, 2 and the Cauchy-Schwartz inequality, we obtain

$$\begin{aligned} &\left\|{X_{k+1}}^\top X_{k+1} - I_p\right\|_{\mathrm{F}} \leq \frac{13}{16}\left\|{Y_k}^\top Y_k - I_p\right\|_{\mathrm{F}}^2 \leq \frac{13}{16}\left((1 + 2\eta_k c)\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} + \|Y_k - X_k\|_{\mathrm{F}}^2\right)^2 \\ \leq \quad &\frac{13}{8}\left((1 + 2\eta_k c)^2\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}^2 + \|Y_k - X_k\|_{\mathrm{F}}^4\right) \leq \frac{13}{32}\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}} + \frac{13}{32}\|Y_k - X_k\|_{\mathrm{F}}^2. \end{aligned}$$

Finally, by Lemma 1 we have

$$\left\|{X_{k+1}}^\top X_{k+1} - I_p\right\|_{\mathrm{F}} \leq \frac{13}{16}\left\|{Y_k}^\top Y_k - I_p\right\|_{\mathrm{F}}^2 < \frac{1}{4(1+c)^2}. \tag{3.11}$$

Thus, we can conclude the proof by using the mathematical induction.

□

3.2 Global convergence

Before presenting the main convergence theorem of SLPG, we first estimate certain sufficient function value reduction.

Lemma 5. *Suppose that Assumption 2 holds and the iterate sequences $\{X_k\}$ and $\{Y_k\}$ are generated by Algorithm 3 initiated from X_0 satisfying $X_0 \in \Omega_\tau$ with τ defined in* (3.2). *Then for any $k = 0, 1, ...,$ it holds*

$$\begin{aligned} f(X_{k+1}) + r(X_{k+1}) \leq f(X_k) + r(X_k) &+ \left(-\frac{1}{2\eta_k} + \frac{L_{f'}}{2} + L_f + L_r\right)\|Y_k - X_k\|_{\mathrm{F}}^2 \\ &+ \left(L_f + L_r + \frac{3c}{4+4c}\right)\left\|{X_k}^\top X_k - I_p\right\|_{\mathrm{F}}. \end{aligned} \tag{3.12}$$

Proof. Recalling the inequality (3.3) and the Taylor expansion of the objective function of (COS), we can obtain

$$\begin{aligned}
&(f(Y_k) + r(Y_k)) - (f(X_k) + r(X_k)) \\
\leq{}& \langle Y_k - X_k, \nabla f(X_k)\rangle + \frac{L_{f'}}{2} \left\| Y_k - X_k \right\|_{\mathrm{F}}^2 + r(Y_k) - r(X_k) \\
&+ \frac{6\eta_k c(5L_f + 5L_r + 3c)}{25} \left\| {X_k}^\top X_k - I_p \right\|_{\mathrm{F}}. \\
\leq{}& \left(-\frac{1}{2\eta_k} + \frac{L_{f'}}{2} \right) \left\| Y_k - X_k \right\|_{\mathrm{F}}^2 + \frac{c}{4+4c} \left\| {X_k}^\top X_k - I_p \right\|_{\mathrm{F}}.
\end{aligned} \tag{3.13}$$

Here the last inequality follows the upper-bound for η_k described in Assumption 2.

On the other hand, the assertion (3.8) of Lemma 4 directly implies that $\left\| Y_k \right\|_2 \leq 2$, which together with the normal step (2.11) lead to the fact that

$$\left\| X_{k+1} - Y_k \right\|_{\mathrm{F}} = \left\| Y_k \left(\frac{3}{2} I_p - \frac{1}{2} {Y_k}^\top Y_k \right) - Y_k \right\|_{\mathrm{F}} = \frac{1}{2} \left\| Y_k ({Y_k}^\top Y_k - I_p) \right\|_{\mathrm{F}} \leq \left\| {Y_k}^\top Y_k - I_p \right\|_{\mathrm{F}}.$$

Then by the Lipschitz continuity of f and r, Lemma 2 and Assumption 2, we arrive at

$$\begin{aligned}
&(f(X_{k+1}) + r(X_{k+1})) - (f(Y_k) + r(Y_k)) \\
\leq{}& (L_f + L_r) \left\| X_{k+1} - Y_k \right\|_{\mathrm{F}} \leq (L_f + L_r) \left\| {Y_k}^\top Y_k - I_p \right\|_{\mathrm{F}} \\
\leq{}& (1 + 2\eta_k c) (L_f + L_r) \left\| {X_k}^\top X_k - I_p \right\|_{\mathrm{F}} + (L_f + L_r) \left\| Y_k - X_k \right\|_{\mathrm{F}}^2 \\
\leq{}& \left(L_f + L_r + \frac{c}{2+2c} \right) \left\| {X_k}^\top X_k - I_p \right\|_{\mathrm{F}} + (L_f + L_r) \left\| Y_k - X_k \right\|_{\mathrm{F}}^2.
\end{aligned} \tag{3.14}$$

After summing up the inequalities (3.13) and (3.14) together, we complete the proof. □

In the next step, we need to evaluate the sufficient reduction of the following merit function.

$$h(X) := f(X) + r(X) + \left(2L_f + 2L_r + \frac{3}{2} \right) \left\| X^\top X - I_p \right\|_{\mathrm{F}}. \tag{3.15}$$

Lemma 6. *Suppose that Assumption 2 holds and the iterate sequences $\{X_k\}$ and $\{Y_k\}$ are generated by Algorithm 3 initiated from X_0 satisfying $X_0 \in \Omega_\tau$ with τ defined in* (3.2). *Then for any $k = 0, 1, ...,$ it holds*

$$h(X_{k+1}) - h(X_k) \leq -\frac{1}{4\eta_k} \left\| Y_k - X_k \right\|_{\mathrm{F}}^2 - \frac{3}{16} \left(L_f + L_r + 1 \right) \left\| {X_k}^\top X_k - I_p \right\|_{\mathrm{F}}. \tag{3.16}$$

Proof. This is a direct corollary of inequalities (3.9), (3.12) and (3.15). □

Theorem 1. *Suppose that Assumption 2 holds and the iterate sequences $\{X_k\}$ and $\{Y_k\}$ are generated by Algorithm 3 initiated from X_0 satisfying satisfying $X_0 \in \Omega_\tau$ with τ defined in* (3.2). *Then the sequence $\{X_k\}$ exists at least one accumulation point which must be a first-order stationary point of* (COS).

Moreover,

$$\min_{0 \leq i \leq k} \frac{1}{\eta_i} \left\| Y_i - X_i \right\|_{\mathrm{F}} \leq \sqrt{\frac{26L_f + 26L_r + 6}{(k+1)\tilde{\eta}}}. \tag{3.17}$$

and

$$\min_{0 \leq i \leq k} \left\| {X_i}^\top X_i - I_p \right\|_{\mathrm{F}} \leq \frac{52}{3(k+1)}. \tag{3.18}$$

Proof. Summing up the inequality (3.16) from $k = 0$ to $+\infty$, we obtain

$$\begin{aligned}
&\sum_{k=0}^{+\infty} \left[\frac{1}{4\eta_k} \|Y_k - X_k\|_{\mathrm{F}}^2 + \frac{3}{16} \left(L_f + L_r + 1 \right) \left\| {X_k}^\top X_k - I_p \right\|_{\mathrm{F}} \right] \leq h(X_0) - \lim_{N \to +\infty} h(X_N) \\
\leq\ & \sup_{N \to +\infty} \left(L_f + L_r \right) \|X_N - X_0\|_{\mathrm{F}} + \left(2L_f + 2L_r + \frac{3}{2} \right) \cdot 2 \cdot \frac{1}{4(1+c)^2} \qquad (3.19) \\
\leq\ & \left(\frac{13}{4} L_f + \frac{13}{4} L_r + \frac{3}{4} \right),
\end{aligned}$$

where the last inequality uses the fact $||X_k||_{\mathrm{F}} \leq \frac{\sqrt{5}}{2}$ which is implied by the second inequality of (3.8). Thus, it holds that

$$\lim_{k \to +\infty} \|Y_k - X_k\|_{\mathrm{F}} = 0, \quad \text{and} \quad \lim_{k \to +\infty} \left\| {X_k}^\top X_k - I_p \right\|_{\mathrm{F}} = 0.$$

On the other hand, by the boundedness of $\{X_k\}$, we know that this sequence exists accumulation point, and denote it by $\bar{X}$. Recalling the boundedness of $\{X_k\}$, $\{Y_k\}$ and $\{\eta_k\}$, without loss of generality, we can assume that there exists a subsequence $\{k_j\}_{j=1,2,\dots}$ such that $X_{k_j} \to \bar{X}$ and meanwhile it holds that $\eta_{k_j} \to \bar{\eta}$. It can be easily verified that

$$\left\| Y_{k_j} - X_{k_j} \right\|_{\mathrm{F}} \to 0 \quad \text{and} \quad \left\| {X_{k_j}}^\top X_{k_j} - I_p \right\|_{\mathrm{F}} \to 0, \tag{3.20}$$

which imply $\bar{X}^\top \bar{X} - I_p = 0$.

For convenience, we invoke the Maximum Theorem stated in [7, p.116] without proof. The Maximum Theorem tells us that the Lipschitz continuities of ∇f and r lead to the fact that the global minimizer of (2.4) is continuous with respect to X_k. We define $\bar{Y}$ as

$$\bar{Y} = \operatorname*{arg\,min}_{\Phi(D^\top \bar{X}) = \Phi(\bar{X}^\top \bar{X})} \langle \nabla f(\bar{X}), D \rangle + r(D) + \frac{1}{2\bar{\eta}} \|D - \bar{X}\|_{\mathrm{F}}^2 .$$

Combining the definition of the tangential step, the orthonormalization of $\bar{X}$, the relation (3.20), we have $\bar{Y} = \bar{X}$. By simple calculation, we can conclude that $\bar{X}$ satisfies (2.3) and hence is a first-order stationary point of (COS).

By summing up the inequality (3.16) from $i = 0$ to k, and using the same deduction in (3.19), we obtain

$$\begin{aligned}
&\sum_{i=0}^{k} \tfrac{1}{4{\eta_i}^2} \|Y_i - X_i\|_{\mathrm{F}}^2 \leq \sum_{0=1}^{k} \tfrac{1}{2\bar{\eta}\eta_i} \|Y_i - X_i\|_{\mathrm{F}}^2 \leq \tfrac{13L_f + 13L_r + 3}{2\bar{\eta}}, \\
\text{and} \quad &\sum_{i=0}^{k} \tfrac{3}{16} \left(L_f + L_r + 1 \right) \left\| {X_i}^\top X_i - I_p \right\|_{\mathrm{F}} \leq \tfrac{13L_f + 13L_r + 3}{4},
\end{aligned}$$

which imply the inequalities (3.17) and (3.18) immediately.

□

3.3 Orthonormalization as post-process

In the last subsection, we present a result on how the post-process affects the value of the merit function.

Proposition 1. *Suppose X satisfying $\left\|X^\top X - I_p\right\|_{\mathrm{F}} \leq \frac{1}{4}$. Let $X = U\Sigma V^\top$ be the SVD of X in economic size for X and we set $X_{\mathrm{orth}} := UV^\top$, then it holds that*

$$h(X_{\mathrm{orth}}) \leq h(X) - \left(L_f + L_r + \frac{3}{2} \right) \left\| X^\top X - I_p \right\|_{\mathrm{F}} .$$

Proof. Firstly, by simple calculation, we have

$$\|X_{\text{orth}} - X\|_{\text{F}} = \|\Sigma - I_p\|_{\text{F}} \leq \|(\Sigma + I_p)(\Sigma - I_p)\|_{\text{F}} = \left\|X^\top X - I_p\right\|_{\text{F}}. \tag{3.21}$$

Then by the Lipschitz continuity of f and r, we obtain

$$(f(X_{\text{orth}}) + r(X_{\text{orth}})) - (f(X) + r(X)) \leq (L_f + L_r)\|X_{\text{orth}} - X\|_{\text{F}} \leq (L_f + L_r)\left\|X^\top X - I_p\right\|_{\text{F}},$$

which implies

$$\begin{aligned} & h(X_{\text{orth}}) - h(X) \\ \leq\ & (f(X_{\text{orth}}) + r(X_{\text{orth}})) - (f(X) + r(X)) - \left(2L_f + 2L_r + \frac{3}{2}\right)\left\|X^\top X - I_p\right\|_{\text{F}} \\ \leq\ & -\left(L_f + L_r + \frac{3}{2}\right)\left\|X^\top X - I_p\right\|_{\text{F}}. \end{aligned}$$

□

4 Numerical Experiments

In this section, we perform preliminary numerical experiments to illustrate the efficiency and the robustness of SLPG . We first present the test settings including how to choose the parameters in SLPG, introduce the test problems and then illustrate some observations in the numerical tests. Then we compare SLPG with some of the state-of-the-art algorithms on these test problems.

All the numerical experiments in this section are run in serial in a platform with Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz and 384GB RAM running MATLAB R2018a under Ubuntu 18.10.

4.1 Test settings

Theorem 1 has provided a range for choosing the stepsize parameter η_k with guaranteed convergence. However, such choice is too restrictive to be practically useful. In this section, we suggest to adopt the following extended version, which was first proposed in [48], of Barzilar-Borwein (BB) stepsize [5] in SLPG.

$$\eta_k = \left\langle S^k, V^k \right\rangle \Big/ \left\langle V^k, V^k \right\rangle, \tag{4.1}$$

where $S^k = X_k - X_{k-1}$ and

$$V^k = \left[\nabla f(X_k) - X_k \Phi({X_k}^\top \nabla f(X_k))\right] - \left[\nabla f(X_{k-1}) - X_{k-1}\Phi({X_{k-1}}^\top \nabla f(X_{k-1}))\right].$$

In Algorithm 1, we set the maximum iterations as 10 and choose the stepsize t as $1/\eta_k$. In SLPG, we also adopt the warm-start technique in selecting the initial guess of Algorithm 1. Namely, Λ_0 in the $k+1$-th iteration can be set as the last Λ_j in the k-th iteration. Besides, we set the constant c as 1000 in Condition 1.

In this paper, the substationarity, the feasibility violation ("feasibility" for short) and that of the tangential subproblem ("TS feasibility" for short) at the k-th iterate are estimated by

$$\|Y_k - X_k\|_{\text{F}}/\eta_k, \quad \left\|{X_k}^\top X_k - I_p\right\|_{\text{F}}, \quad \text{and} \quad \left\|\Phi({X_k}^\top (Y_k - X_k))\right\|_{\text{F}},$$

respectively. Unless otherwise stated, SLPG terminates if either the stopping criteria $\|Y_k - X_k\|_{\text{F}}/\eta_k \leq 10^{-4}$ is satisfied or the maximum number of iterations 10000 is reached.

4.2 Test Problems

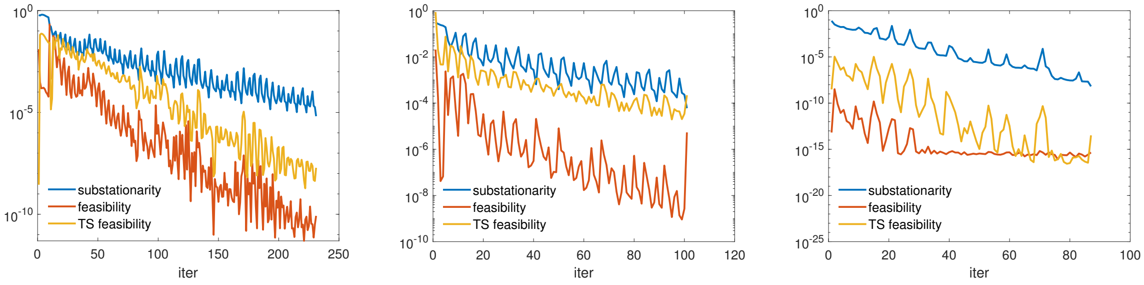
We adopt Problems 1-3 as the test problems. Unless otherwise stated, for Problems 1 and 2, we set the covariance matrix $L \in \mathbb{R}^{1000\times 1000}$ of 200 randomly generated samples $S = \mathrm{randn}(1000, 200)$ with unified normalization as the following

$$L = SS^\top / \|S\|_2^2 . \tag{4.2}$$

For Problem 3, we uses the test instances as "h2o" molecular from KSSOLV toolbox [51]. Additionally, the initial points are chosen as the leading p eigenvectors of L for Problems 1 and 2, or generated by the build-in function "getX0" in KSSOLV toolbox [51] for Problem 3.

4.3 Observations in testing SLPG

We first investigate how the substationarity, feasibility and TS feasibility vary in the running of SLPG without post-process in solving Problems 1 and 2 with randomly generated data. We put the numerical results in Figure 1. The blue, red and yellow lines represent the substationarity, feasibility and TS feasibility, respectively. The problem parameters are listed below the subfigures. We can learn from Figure 1(a)-1(c) that the feasibility violation of SLPG is actually a high order infinitesimal of the substationarity, which coincides with the theoretical results in Lemma 1. It is worthy of mentioning that SLPG decreases the feasibility violation much faster than those existing infeasible first-order approaches, such as PLAM, PCAL in [20], and PenCF from [49], in solving (COS). Although we can only theoretically establish the global sublinear convergence rate for SLPG, in Figure 1, we have observed its local linear convergence rate in solving Problems 1-3.



(a) Problem 1, $(p, \gamma) = (5, 0.007)$

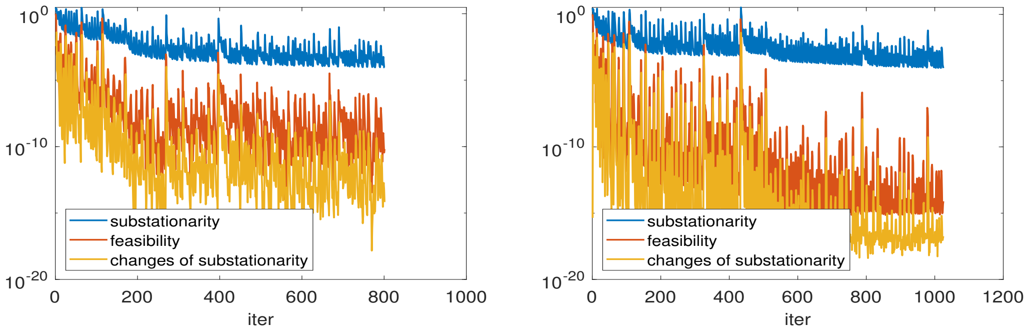
(b) Problem 2, $(p, \gamma) = (5, 0.001)$

(c) Problem 3, $(n, p) = (2013, 7)$

Figure 1: The substationarity, feasibility and TS feasibility of SLPG.

Next we investigate how the post-process of SLPG affects the substationarity by testing SLPG in solving Problems 1 and 2 with randomly generated data. We display the substationarity and the feasibility of SLPG without the post-process, and the difference on the substationarity of SLPG after imposing the post-process as the blue, red and yellow lines, respectively, in Figure 2. The problem parameters are listed below the subfigures. We can learn from Subfigures 2(b) and 2(a) that the post-process only affects the substationarity a little. In fact, the difference is a high order infinitesimal of the substationarity, which can partly be explained as the feasibility violation itself is a high order infinitesimal of the substationarity.

We note that our original problem (COS) is nonconvex, hence it is expected to have multi-stationary points. Therefore, it is meaningful to check how the initial guesses affect the performance of SLPG. We generate two data sets by (4.2) for Problems 1 and 2, respectively. Then we fix these two data sets and run SLPG for 1000 times with different randomly generated initial points $X_0 = \mathrm{qr}(\mathrm{randn}(n, p))$ for each problem. To achieve high precision in function value, we set the stopping criteria as $\|Y_k - X_k\|_\mathrm{F} / \eta_k \leq 10^{-10}$ here. We regard the function values varying in a range less than 10^{-7} as one value due to the possible numerical error. We study the function value distribution in the 1000 runs for each problem

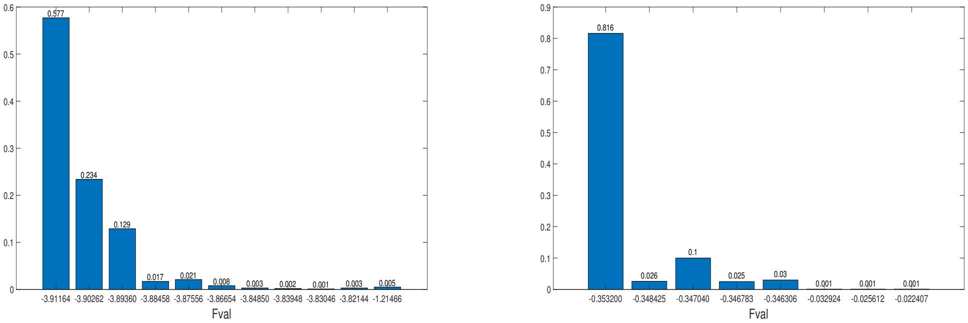


(a) Problem 1, $(p, \gamma) = (5, 0.007)$

(b) Problem 2, $(p, \gamma) = (5, 0.001)$

Figure 2: The effect of the post-process.

and put the results into Figure 3. The problem parameters are listed below the subfigures. From both Subfigures 3(a) and 3(b), we can conclude that SLPG has high probability to reach the lowest function values, which could be regarded as good estimates of the global minimizers of Problems 1 and 2, respectively, with certain probability.



(a) Problem 1, $(p, \gamma) = (5, 0.05)$

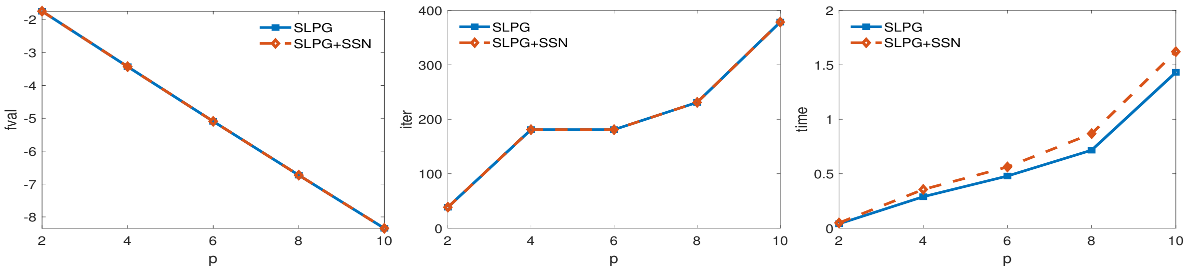
(b) Problem 2, $(p, \gamma) = (5, 0.001)$

Figure 3: Function value distributions of SLPG .

Finally, we investigate how the inner solver affects the overall performance of SLPG. We compare our fixed point iteration Algorithm 1 with the frequently used semi-smooth Newton methods in solving Problem 1. In our numerical examples, SLPG refers to SLPG where the subproblem is solved by Algorithm 1 while SLPG+SSN refer to the algorithm where the the subproblems are solved by semi-smooth Newton methods. The parameters of the semi-smooth Newton method adopted in SLPG+SSN are fixed as its default setting as stated in [30]. Figure 4 illustrates the performance of SLPG and SLPG+SSN under different column size with (n, γ) fixed as $(4000, 0.07)$. From subfigures 4(b)-4(c) we can learn that SLPG reaches the same function value in same number of iterations, but requires slightly less CPU time than SLPG+SSN. That is the reason we use Algorithm 1 as the default inner solver in SLPG.

4.4 $\ell_{2,1}$-norm regularized PCA

In this subsection, we first compare SLPG with some of the state-of-the-art algorithms including ManPG-Ada and PenCPG. Then, we further investigate the robustness brought by our penalty-free scheme. The first algorithm in comparison is ManPG-Ada, which is an accelerated version of ManPG



(a) $(n, \gamma) = (4000, 0.007)$ (b) $(n, \gamma) = (4000, 0.007)$ (c) $(n, \gamma) = (4000, 0.007)$

Figure 4: The comparison of inner solver.

[11]. The second one is PenCPG, which is an infeasible proximal gradient method based on the closed-form expression of the multipliers. In our experiments, all three algorithms are run in their default settings. As suggested in [22], the penalty parameter of Problem 2 is set as $\gamma = b\sqrt{p + \log(n)}$, where parameter b is used to control the sparsity.

Figure 5 illustrates the performance of the three algorithms in comparison in solving Problem 2 with different combinations of n, p, b. The detailed problem parameters are listed below the subfigures. As illustrated in Figure 5, all of these three algorithm reach the same function values. SLPG takes fewer iterations than the other two, meanwhile it takes much less CPU time than ManPG-Ada. Since PenCPG does not have any subproblem to solve, it has the lowest computational cost in each iteration among the three. Finally, it only takes slightly less CPU time than SLPG . We can conclude that SLPG is superior to the other two algorithms in the testing problems.

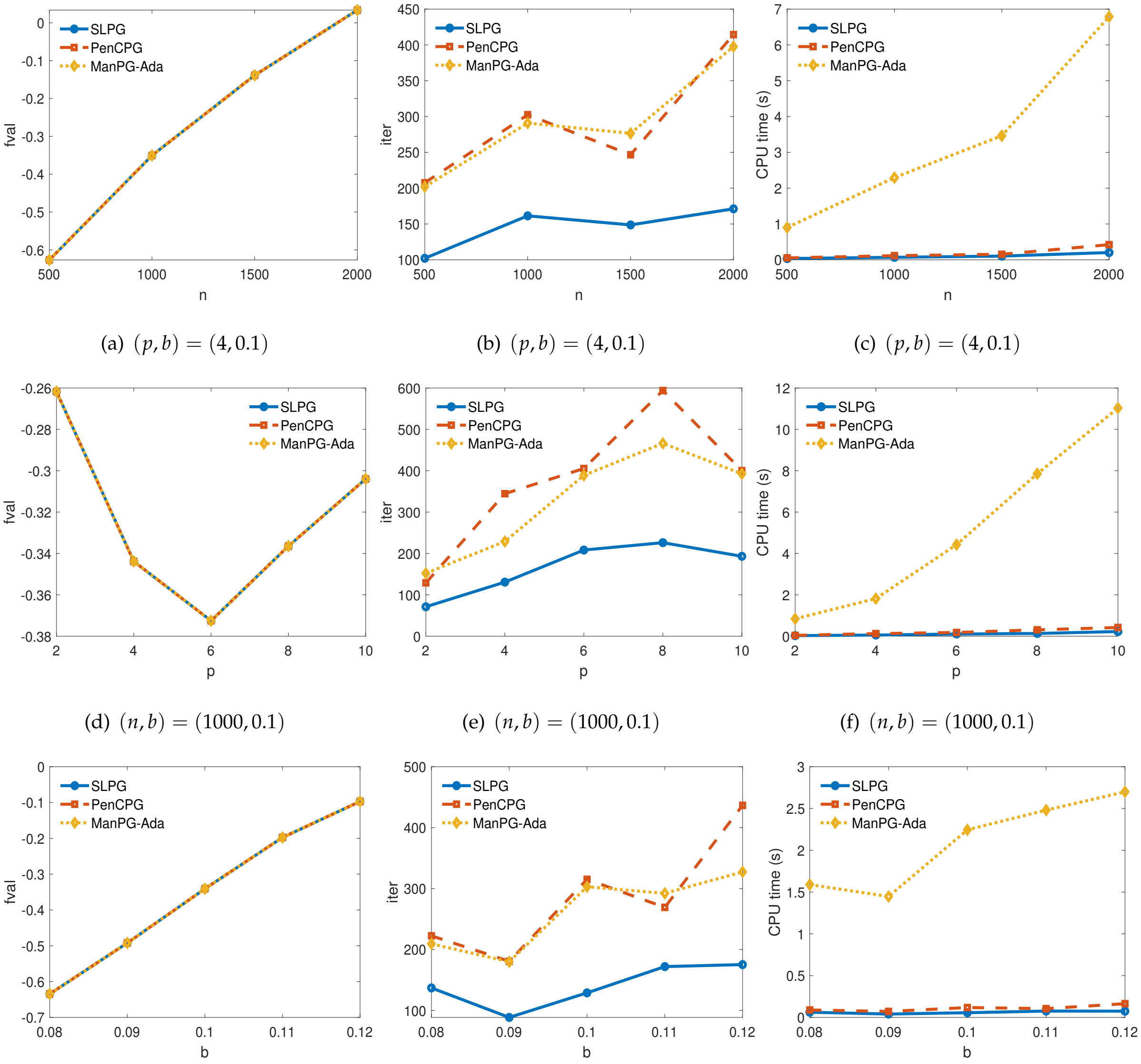
From the above experiment, we notice that PenCPG is comparable with SLPG in the aspect of CPU time. However, we notice that PenCPG requires to tune a penalty parameter β while SLPG does not have one. In the following experiment, we compare SLPG with PenCPG equipped with different choices of β. We still use Problem 2 with data set generated randomly as stated in (4.2). We present the results in Figure 6. The detailed problem settings are listed below the subfigures. We can learn for Figure 6 that the performance of PenCPG is sensitive to the penalty parameter, meanwhile, SLPG can always outperforms PenCPG with the best choice of β.

4.5 Sparse PCA

In this subsection, we compare SLPG with two state-of-the-art algorithms including ManPG-Ada [12] and AManPG [30] in solving sparse PCA problem. In our experiments, all the three algorithms are run in their default settings. Figure 5 illustrates the performance of the three algorithms in comparison in solving Problem 1 with different combinations of n, p, γ. The detailed problem parameters are listed below the subfigures. We can learn from Figure 5 that all of these three algorithms reach the same function values. SLPG takes much fewer iterations than ManPG-Ada and slightly fewer iterations than AManPG. Meanwhile, it takes much less CPU time than the other two algorithms.

4.6 Kohn-Sham total energy minimization

In this subsection, we compare our algorithm with some state-of-the-art approaches in solving Problem 3. The test problems are selected from the Kohn-Sham total energy minimization platform KSSOLV [51], which is a MATLAB toolbox designed for electronic structure calculation. The algorithms in comparison include PCAL [20] and PenCF [49]. We compare all these algorithms in their default settings. We first study the numerical performance of SLPG and compare it with PCAL and PenCF equipped with different penalty parameter β. The performances of these algorithms are demonstrated in Figure 8. We can learn that the performances of PCAL and PenCF are sensitive to the penalty parameter β. Meanwhile, SLPGis penalty-paramter-free and has comparable performs with the other two algorithms equipped with fine-tuned penalty parameters.



(a) $(p, b) = (4, 0.1)$

(b) $(p, b) = (4, 0.1)$

(c) $(p, b) = (4, 0.1)$

(d) $(n, b) = (1000, 0.1)$

(e) $(n, b) = (1000, 0.1)$

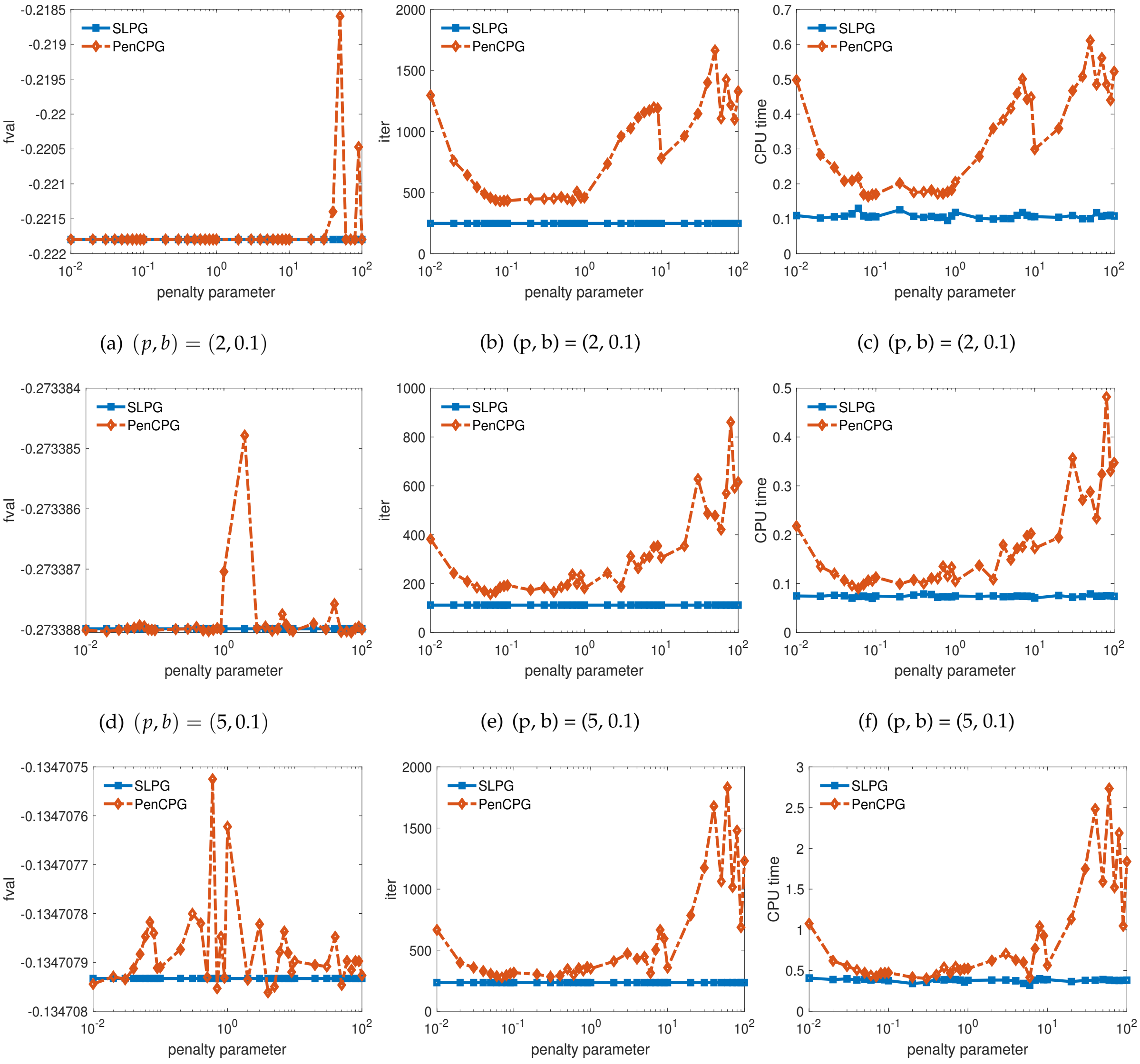
(f) $(n, b) = (1000, 0.1)$

(g) $(n, p) = (1000, 4)$

(h) $(n, p) = (1000, 4)$

(i) $(n, p) = (1000, 4)$

Figure 5: A comparison among SLPG, PenCPG and ManPG-Ada in solving $\ell_{2,1}$-norm regularized PCA problems.



(a) $(p, b) = (2, 0.1)$

(b) (p, b) = (2, 0.1)

(c) (p, b) = (2, 0.1)

(d) $(p, b) = (5, 0.1)$

(e) (p, b) = (5, 0.1)

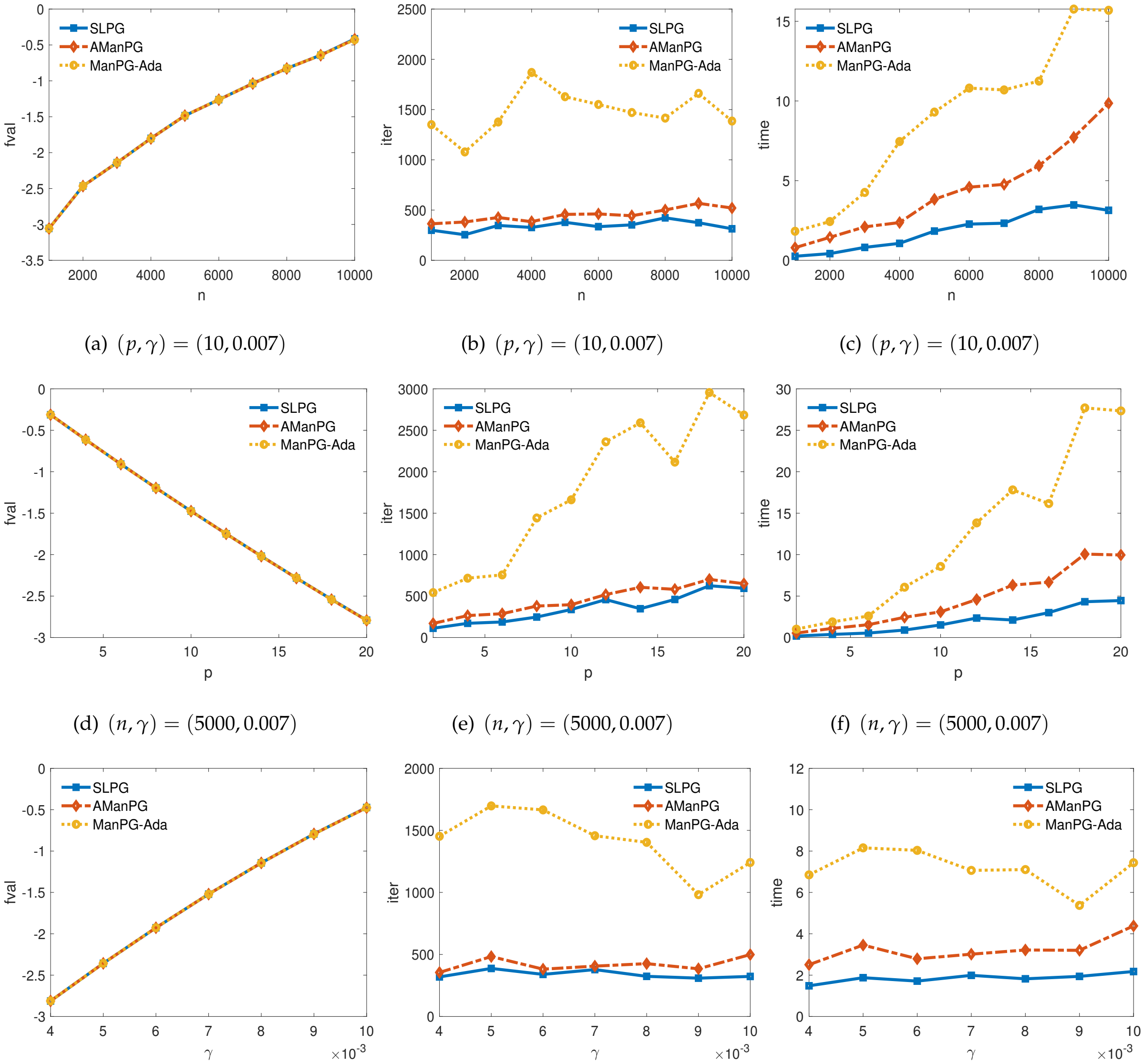
(f) (p, b) = (5, 0.1)

(g) (p, b) = (10, 0.1)

(h) (p, b) = (10, 0.1)

(i) (p, b) = (10, 0.1)

Figure 6: A comparison between SLPG and PenCPG with different penalty parameters.



(a) $(p, \gamma) = (10, 0.007)$ (b) $(p, \gamma) = (10, 0.007)$ (c) $(p, \gamma) = (10, 0.007)$

(d) $(n, \gamma) = (5000, 0.007)$ (e) $(n, \gamma) = (5000, 0.007)$ (f) $(n, \gamma) = (5000, 0.007)$

(g) $(n, p) = (5000, 10)$ (h) $(n, p) = (5000, 10)$ (i) $(n, p) = (5000, 10)$

Figure 7: A comparison among SLPG, AManPG and ManPG-Ada in solving sparse PCA problems.

Finally, we comprehensively compare the performance of SLPG with more state-of-the-art algorithms, including the projection-based feasible method with QR factorization as retraction ("ManOptQR" for short) from Manopt toolbox [4, 8], OptM proposed by Wen and Yin [47], PCAL and PenCF. In this experiment, all the algorithms are run in their default settings. We set the stopping criteria and the maximum number of iterations as $\left\|\nabla f(X_k) - X_k \Phi(X_k{}^\top \nabla f(X_k))\right\|_\mathrm{F} \leq 10^{-7}$ and 1000, respectively. Table 1 illustrates the performance of these algorithms on 8 test problems with respect to different molecules. The terms "E_{tot}", "Substationarity", "Iteration", "Feasibility violation" and "CPU time" stand for the function value, $\left\|\nabla f(X) - X\Lambda(X)\right\|_\mathrm{F}$, the number of iterations, $\left\|X^\top X - I_p\right\|_\mathrm{F}$, and the wall-clock running time, respectively. We can learn from Table 1 that SLPG is comparable with these state-of-the-art algorithms in the aspect of iterations and CPU time in solving all the test problems.

To sum up, from the above numerical experiments, we can conclude that SLPG exhibits its robustness and efficiency comparing with the existing algorithms in solving both smooth and nonsmooth minimization over the Stiefel manifold.

Solver	E_{tot}	Substationarity	Iteration	Feasibility violation	CPU time(s)
alanine, $(n, p) = (12671, 18)$					
ManOptQR	-6.11e+01	9.88e-08	80	2.01e-15	24.16
OptM	-6.11e+01	2.15e-08	87	4.22e-14	26.35
PCAL	-6.11e+01	7.16e-08	97	2.80e-15	30.56
PenCF	-6.11e+01	2.83e-08	83	1.90e-15	24.10
SLPG	-6.11e+01	7.15e-08	73	6.65e-16	21.27
benzene, $(n, p) = (8407, 15)$					
ManOptQR	-3.72e+01	8.48e-08	163	2.07e-15	27.52
OptM	-3.72e+01	1.19e-08	82	2.46e-14	14.91
PCAL	-3.72e+01	6.88e-08	67	2.35e-15	13.17
PenCF	-3.72e+01	7.44e-08	67	2.57e-15	12.15
SLPG	-3.72e+01	1.68e-08	66	8.66e-16	11.83
c12h26, $(n, p) = (5709, 37)$					
ManOptQR	-8.15e+01	8.85e-08	439	5.06e-15	131.12
OptM	-8.15e+01	2.49e-08	105	8.50e-14	33.71
PCAL	-8.15e+01	8.46e-08	66	4.68e-15	25.23
PenCF	-8.15e+01	7.70e-08	81	4.72e-15	25.92
SLPG	-8.15e+01	8.83e-08	72	1.34e-15	24.26
ctube661, $(n, p) = (12599, 48)$					
ManOptQR	2.51e+01	2.45e+01	1000	4.22e-15	743.02
OptM	-1.34e+02	8.64e-09	108	4.89e-15	90.88
PCAL	-1.34e+02	9.54e-08	73	4.80e-15	68.42
PenCF	-1.34e+02	5.04e-08	76	5.04e-15	60.32
SLPG	-1.34e+02	7.67e-08	69	1.36e-15	56.94
glutamine, $(n, p) = (16517, 29)$					
ManOptQR	-9.18e+01	7.03e-08	180	3.20e-15	138.08
OptM	-9.18e+01	1.39e-08	129	3.29e-15	102.58
PCAL	-9.18e+01	6.67e-08	108	3.08e-15	88.68
PenCF	-9.18e+01	9.35e-08	109	3.16e-15	83.01
SLPG	-9.18e+01	9.06e-08	104	9.50e-16	78.98
graphene16, $(n, p) = (3071, 37)$					
ManOptQR	-9.40e+01	8.74e-08	326	4.24e-15	67.13
OptM	-9.40e+01	2.34e-08	313	4.34e-15	66.44

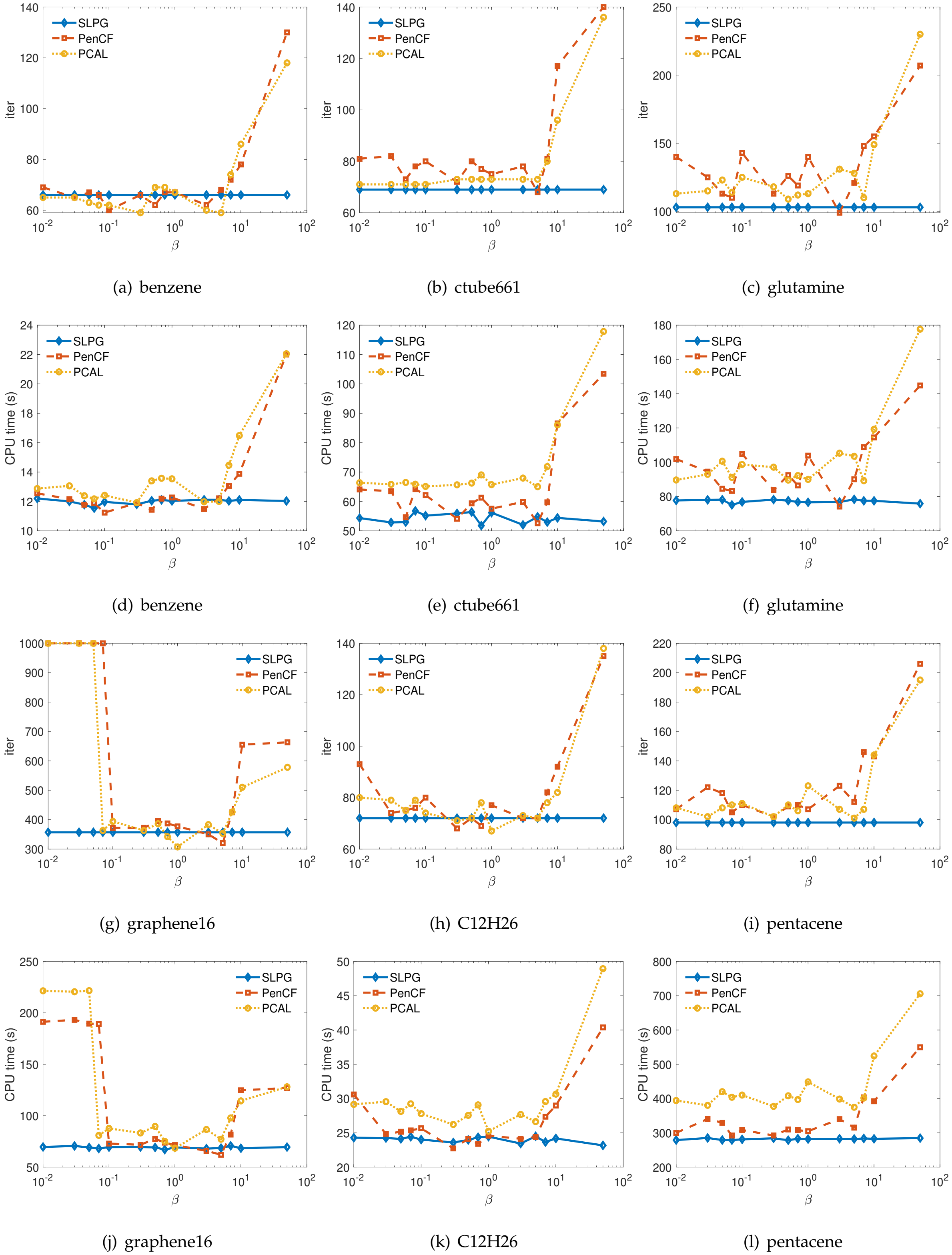
Table 1 continued from previous page

PCAL	-9.40e+01	4.81e-08	416	4.41e-15	94.70
PenCF	-9.40e+01	3.74e-08	327	4.18e-15	65.00
SLPG	-9.40e+01	9.53e-08	286	1.22e-15	56.71
pentacene, $(n, p) = (44791, 51)$					
ManOptQR	-1.31e+02	9.20e-08	150	4.79e-15	425.62
OptM	-1.31e+02	2.37e-08	126	4.52e-15	374.43
PCAL	-1.31e+02	8.62e-08	111	4.06e-15	365.80
PenCF	-1.31e+02	6.40e-08	109	4.56e-15	301.13
SLPG	-1.31e+02	9.58e-08	113	1.25e-15	308.88
ptnio, $(n, p) = (4069, 43)$					
ManOptQR	-2.26e+02	8.62e-08	661	3.90e-15	166.45
OptM	-2.26e+02	2.49e-08	662	3.96e-15	171.86
PCAL	-2.26e+02	9.53e-08	596	3.75e-15	169.96
PenCF	-2.26e+02	8.54e-08	508	3.98e-15	123.19
SLPG	-2.26e+02	6.70e-08	706	1.19e-15	169.87
si64, $(n, p) = (6451, 128)$					
ManOptQR	1.58e+02	2.88e+01	1000	8.02e-15	2413.09
OptM	-2.53e+02	2.18e-08	124	1.03e-14	328.57
PCAL	-2.53e+02	9.97e-08	74	1.02e-14	227.23
PenCF	-2.53e+02	7.48e-08	68	1.03e-14	179.30
SLPG	-2.53e+02	9.73e-08	74	2.25e-15	194.01
si8, $(n, p) = (799, 16)$					
ManOptQR	-3.13e+01	9.39e-08	394	2.33e-15	38.27
OptM	-3.13e+01	2.20e-08	182	1.73e-15	18.66
PCAL	-3.13e+01	8.23e-08	73	2.16e-15	8.27
PenCF	-3.13e+01	9.14e-08	90	1.73e-15	9.22
SLPG	-3.13e+01	8.54e-08	72	5.81e-16	7.60

Table 1: The results in Kohn-Sham total energy minimization

5 Conclusion

In this paper, we have presented a penalty-free infeasible approach called SLPG for solving optimization problems over the Stiefel manifold with possibly nonsmooth objective functions. Our SLPG has two main steps. The first step is to solve a linearized proximal approximation in an affine subspace, which reduces to a tangent space of the Stiefel manifold if the iterate is feasible. We suggest to adopt a fixed point iteration to solve this tangential subproblem. Particularly, when the objective function is smooth or of $\ell_{2,1}$ regularization term, we can adopt an empirical direct approach to inexactly solve the tangential subproblem instead of the fixed point iteration. The other step is to approximate the orthonormalization procedure by a cheap normal step, which is inspired from the Taylor expansion of the polar decomposition. The main advantages of our approach lie in the following three aspects. Firstly, we adopt an infeasible framework which is of better scalability than those manifold-based approaches. Secondly, compared with the existing infeasible approaches, SLPG does not invoke any penalty function, and hence the sensitivity of the performance to the choice of penalty parameters is naturally eliminated. Thirdly, numerical experiments demonstrate the great potential of SLPG in solving (COS) with both smooth and nonsmooth objective functions. In addition, we have established the global convergence results for SLPG.



(a) benzene

(b) ctube661

(c) glutamine

(d) benzene

(e) ctube661

(f) glutamine

(g) graphene16

(h) C12H26

(i) pentacene

(j) graphene16

(k) C12H26

(l) pentacene

Figure 8: A detailed comparison on the iterations and CPU time taken by SLPG, PenCF and PCAL in KSSOLV.

References

[1] Traian Abrudan, Jan Eriksson, and Visa Koivunen. Conjugate gradient algorithm for optimization under unitary matrix constraint. *Signal Processing*, 89(9):1704–1714, 2009.

[2] Traian E Abrudan, Jan Eriksson, and Visa Koivunen. Steepest descent algorithms for optimization under unitary matrix constraint. *IEEE Transactions on Signal Processing*, 56(3):1134–1147, 2008.

[3] P-A Absil, Christopher G Baker, and Kyle A Gallivan. Trust-region methods on riemannian manifolds. *Foundations of Computational Mathematics*, 7(3):303–330, 2007.

[4] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[5] Jonathan Barzilai and Jonathan M Borwein. Two-point step size gradient methods. *IMA journal of numerical analysis*, 8(1):141–148, 1988.

[6] E. M. L. Beale, Kenneth J. Arrow, Leonid Hurwicz, Hirofumi Uzawa, Hollis B. Chenery, Selmer M. Johnson, Samuel Karlin, Thomas Marschak, and Robert M. Solow. Studies in linear and non-linear programming. *Journal of the Royal Statistical Society. Series A (General)*, 122(3): 381, 1959.

[7] Claude Berge. Topological spaces, oliver and boyed, edinburg-london. *1st English edition*, 1963.

[8] Nicolas Boumal, Bamdev Mishra, P-A Absil, and Rodolphe Sepulchre. Manopt, a matlab toolbox for optimization on manifolds. *The Journal of Machine Learning Research*, 15(1):1455–1459, 2014.

[9] T Tony Cai, Zongming Ma, Yihong Wu, et al. Sparse pca: Optimal rates and adaptive estimation. *The Annals of Statistics*, 41(6):3074–3110, 2013.

[10] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.

[11] Shixiang Chen, Zengde Deng, Shiqian Ma, and Anthony Man-Cho So. Manifold proximal point algorithms for dual principal component pursuit and orthogonal dictionary learning. In *Asilomar Conference on Signals, Systems, and Computers*, 2019.

[12] Shixiang Chen, Shiqian Ma, Anthony Man-Cho So, and Tong Zhang. Proximal gradient method for nonsmooth optimization over the stiefel manifold. *SIAM Journal on Optimization*, 30(1):210–239, 2020.

[13] Weiqiang Chen, Hui Ji, and Yanfei You. An augmented lagrangian method for 1-regularized optimization problems with orthogonality constraints. *SIAM Journal on Scientific Computing*, 38 (4):B570–B592, 2016.

[14] Xin Chen, Changliang Zou, R Dennis Cook, et al. Coordinate-independent sparse sufficient dimension reduction and variable selection. *The Annals of Statistics*, 38(6):3696–3723, 2010.

[15] Zhongwen Chen, Yu-Hong Dai, and Jiangyan Liu. A penalty-free method with superlinear convergence for equality constrained optimization. *Computational Optimization and Applications*, pages 1–33, 2019.

[16] Frank H Clarke. *Optimization and nonsmooth analysis*, volume 5. Siam, 1990.

[17] Xiaoying Dai, Liwei Zhang, and Aihui Zhou. Adaptive step size strategy for orthogonality constrained line search methods. *arXiv preprint arXiv:1906.02883*, 2019.

[18] Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.

[19] Bin Gao, Xin Liu, Xiaojun Chen, and Ya-xiang Yuan. A new first-order algorithmic framework for optimization problems with orthogonality constraints. *SIAM Journal on Optimization*, 28(1): 302–332, 2018.

[20] Bin Gao, Xin Liu, and Ya-xiang Yuan. Parallelizable algorithms for optimization problems with orthogonality constraints. *SIAM Journal on Scientific Computing*, 41(3):A1949–A1983, 2019.

[21] Bin Gao, Guanghui Hu, Yang Kuang, and Xin Liu. An orthogonalization-free parallelizable framework for all-electron calculations in density funcitonal theory. *arXiv preprint arXiv:2007.14228*, 2020.

[22] Chao Gao, Zongming Ma, Harrison H Zhou, et al. Sparse cca: Adaptive estimation and computational barriers. *The Annals of Statistics*, 45(5):2074–2101, 2017.

[23] N. I. M. Gould and Ph. L. Toint. Nonlinear programming without a penalty function or a filter. *Mathematical Programming*, 122(1):155–196, 2010.

[24] Bingsheng He, Yanfei You, and Xiaoming Yuan. On the convergence of primal-dual hybrid gradient algorithm. *Siam Journal on Imaging Sciences*, 7(4):2526–2537, 2014.

[25] Nicholas J. Higham and Pythagoras Papadimitriou. A parallel algorithm for computing the polar decomposition. *Parallel Computing*, 20:1161–1173, 1994.

[26] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Convex analysis and minimization algorithms I: Fundamentals*, volume 305. Springer science & business media, 2013.

[27] Jiang Hu, Andre Milzarek, Zaiwen Wen, and Yaxiang Yuan. Adaptive quadratically regularized newton method for riemannian optimization. *SIAM Journal on Matrix Analysis and Applications*, 39(3):1181–1207, 2018.

[28] Jiang Hu, Xin Liu, Zaiwen Wen, and Ya-xiang Yuan. A brief introduction to manifold optimization. *Journal of the Operations Research Society of China*, (8):199–248, 2020.

[29] Xiaoyin Hu and Xin Liu. An efficient orthonormalization-free approach for sparse dictionary learning and dual principal component pursuit. *Sensors*, 20(3041), 2020.

[30] Wen Huang and Ke Wei. Extending fista to riemannian optimization for sparse pca. *arXiv preprint arXiv:1909.05485*, 2019.

[31] Wen Huang and Ke Wei. Riemannian proximal gradient methods. *arXiv preprint arXiv:1909.06065*, 2019.

[32] Bo Jiang and Yu-Hong Dai. A framework of constraint preserving update schemes for optimization on stiefel manifold. *Mathematical Programming*, 153(2):535–575, 2015.

[33] Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical review*, 140(4A):A1133, 1965.

[34] Rongjie Lai and Stanley Osher. A splitting method for orthogonality constrained problems. *Journal of Scientific Computing*, 58(2):431–449, 2014.

[35] Xinwei Liu and Yaxiang Yuan. A sequential quadratic programming method without a penalty function or a filter for nonlinear equality constrained optimization. *Siam Journal on Optimization*, 21(2):545–571, 2011.

[36] Zongming Ma et al. Sparse principal component analysis and iterative thresholding. *The Annals of Statistics*, 41(2):772–801, 2013.

[37] Jonathan H Manton. Optimization algorithms exploiting unitary constraints. *IEEE Transactions on Signal Processing*, 50(3):635–650, 2002.

[38] J. M. Martinez. Inexact-restoration method with lagrangian tangent decrease and new merit function for nonlinear programming. *Journal of Optimization Theory and Applications*, 111(1):39–58, 2001.

[39] Yasunori Nishimori and Shotaro Akaho. Learning algorithms utilizing quasi-geodesic flows on the stiefel manifold. *Neurocomputing*, 67:106–135, 2005.

[40] R Tyrrell Rockafellar and Roger J-B Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

[41] Guy Rosman, Xuecheng Tai, Ron Kimmel, and Alfred M Bruckstein. Augmented-lagrangian regularization of matrix-valued maps. *Methods and applications of analysis*, 21(1):105–122, 2014.

[42] Chungen Shen, Sven Leyffer, and Roger Fletcher. A nonmonotone filter method for nonlinear optimization. *Computational Optimization and Applications*, 52(3):583–607, 2012.

[43] Defeng Sun and Jie Sun. Semismooth matrix-valued functions. *Mathematics of Operations Research*, 27:150–169, 2002.

[44] Michael Ulbrich and Stefan Ulbrich. Non-monotone trust region methods for nonlinear equality constrained optimization without a penalty function. *Mathematical Programming*, 95(1):103–135, 2003.

[45] Stefan Ulbrich. On the superlinear local convergence of a filter-sqp method. *Mathematical Programming*, 100(1):217–245, 2004.

[46] Magnus O Ulfarsson and Victor Solo. Sparse variable pca using geodesic steepest descent. *IEEE Transactions on Signal Processing*, 56(12):5823–5832, 2008.

[47] Zaiwen Wen and Wotao Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434, 2013.

[48] Zaiwen Wen, Wotao Yin, Donald Goldfarb, and Yin Zhang. A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization, and continuation. *SIAM Journal on Scientific Computing*, 32(4):1832–1857, 2010.

[49] Nachuan Xiao, Xin Liu, and Ya-xiang Yuan. A class of smooth exact penalty function methods for optimization problems with orthogonality constraints. *Optimization Methods and Software*, 2020.

[50] Nachuan Xiao, Xin Liu, and Ya-xiang Yuan. Exact penalty function for $\ell_{2,1}$ norm minimization with orthogonality constraints. *Optimization Online preprint:2020/07/7908*, 2020.

[51] Chao Yang, Juan C Meza, Byounghak Lee, and Lin-Wang Wang. Kssolv—a matlab toolbox for solving the kohn-sham equations. *ACM Transactions on Mathematical Software (TOMS)*, 36(2):10, 2009.