Primal-Dual Optimization Algorithms over Riemannian Manifolds: an Iteration Complexity Analysis

Junyu Zhang* Shiqian Ma† Shuzhong Zhang‡

October 5, 2017

Abstract

In this paper we study nonconvex and nonsmooth multi-block optimization over Riemannian manifolds with coupled linear constraints. Such optimization problems naturally arise from machine learning, statistical learning, compressive sensing, image processing, and tensor PCA, among others. We develop an ADMM-like primal-dual approach based on decoupled solvable subroutines such as linearized proximal mappings. First, we introduce the optimality conditions for the afore-mentioned optimization models. Then, the notion of ϵ-stationary solutions is introduced as a result. The main part of the paper is to show that the proposed algorithms enjoy an iteration complexity of $O(1/\epsilon^2)$ to reach an ϵ-stationary solution. For prohibitively large-size tensor or machine learning models, we present a sampling-based stochastic algorithm with the same iteration complexity bound in expectation. In case the subproblems are not analytically solvable, a feasible curvilinear line-search variant of the algorithm based on retraction operators is proposed. Finally, we show specifically how the algorithms can be implemented to solve a variety of practical problems such as the NP-hard maximum bisection problem, the ℓ_q regularized sparse tensor principal component analysis and the community detection problem. Our preliminary numerical results show great potentials of the proposed methods.

Keywords: nonconvex and nonsmooth optimization, Riemannian manifold, ϵ-stationary solution, ADMM, iteration complexity.

*Department of Industrial and System Engineering, University of Minnesota (zhan4393@umn.edu).
†Department of Mathematics, UC Davis (sqma@math.ucdavis.edu).
‡Department of Industrial and System Engineering, University of Minnesota (zhangs@umn.edu).

1 Introduction

Multi-block nonconvex optimization with nonsmooth regularization functions has recently found important applications in statistics, computer vision, machine learning, and image processing. In this paper, we aim to solve a class of *constrained* nonconvex and nonsmooth optimization models. To get a sense of the problems at hand, let us consider the following *Multilinear (Tensor) Principal Component Analysis* (MPCA) model, which has applications in 3-D object recognition, music genre classification, and subspace learning (see e.g. [39, 46]). Details of the model will be discussed in Section 5. It pays to highlight here that a sparse optimization version of the model is as follows:

$$\begin{array}{ll} \min_{C,U,V,Y} & \sum_{i=1}^N \|T^{(i)} - C^{(i)} \times_1 U_1 \times \cdots \times_d U_d\|_F^2 + \alpha_1 \sum_{i=1}^N \|C^{(i)}\|_p^p + \alpha_2 \sum_{j=1}^d \|V_j\|_q^q + \frac{\mu}{2} \sum_{j=1}^d \|Y_j\|^2 \\ \text{s.t.} & C^{(i)} \in \mathbb{R}^{m_1 \times \cdots \times m_d},\ i = 1, ..., N \\ & U_j \in \mathbb{R}^{n_j \times m_j},\ U_j^\top U_j = I, j = 1, ..., d \\ & V_j - U_j + Y_j = 0,\ j = 1, ..., d, \end{array}$$

where $T^{(i)} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, $0 < p < 1$, $0 < q < 1$, $\alpha_1, \alpha_2, \mu > 0$ are weighting parameters. Essentially, one aims to find a Tucker decomposition of a given tensor in such a way that the orthogonal matrices are sparse. This can be naturally dealt with by a consensus-variable approach; see for example [33]. The factor matrices are introduced both as U_j and V_j. While U_j's are orthogonal (hence constrained to the Stiefel manifolds) and V_j's are sparse, they are forced to agree with each other. This way of variable splitting is a useful modeling technique. Note that a slack variable Y_j is introduced to relax this requirement. We penalize the norm of Y_j in the objective so that U_j and V_j do not need to exactly equal to each other. Notice that the objective function involves sparsity-promoting nonconvex ℓ_q $(0 < q < 1)$ loss functions. Therefore, the overall model is noncovex and nonsmooth because of the sparsity promoting objective function, in addition to the manifolds constraints. As we shall see from more examples later, such formulations are found to be common for many applications.

In general, we consider the following model:

$$\begin{array}{ll} \min & f(x_1, \cdots, x_N) + \sum_{i=1}^{N-1} r_i(x_i) \\ \text{s.t.} & \sum_{i=1}^N A_i x_i = b, \text{ with } A_N = I, \\ & x_N \in \mathbb{R}^{n_N}, \\ & x_i \in \mathcal{M}_i,\ i = 1, ..., N-1, \\ & x_i \in X_i,\ i = 1, ..., N-1, \end{array} \tag{1}$$

where f is a smooth function with L-Lipschitz continuous gradient, but is possibly nonconvex; the functions $r_i(x_i)$ are convex but are possibly nonsmooth; $\mathcal{M}_i$'s are Riemannian manifolds, not necessarily compact, embedded in Euclidean spaces; the additional constraint sets X_i are assumed to be some closed convex sets. As we shall see later, the restrictions on r_i being convex and A_N

being identity can all be relaxed, after a reformulation. For the time being however, let us focus on (1).

1.1 Related literature

On the modeling front, nonsmooth/nonconvex regularization such as the ℓ_1 or ℓ_q $(0 < q < 1)$ penalties are key ingredients in promoting sparsity in models such as the basis pursuit [7, 12], LASSO [15, 51, 66], robust principal component analysis (RPCA) [6] and sparse coding [35]. Another important source for nonconvex modeling can be attributed to decomposition problems, e.g. tensor decomposition problems [10, 31, 44], low-rank and/or nonnegative matrix completion or decomposition [11, 26, 49]. Yet, another main source for nonconvex modeling is associated with the Riemannian manifold constraints, such as sphere, product of spheres, the Stiefel manifold, the Grassmann manifold, and the low-rank elliptope are often encountered; see [3, 13, 42, 50, 55].

There has been a recent intensive research interest in studying optimization over a Riemannian manifold:

$$\min_{x \in \mathcal{M}} f(x),$$

where f is smooth; see [1,2,5,25,40,48] and the references therein. Note that viewed *within* manifold itself, the problem is essentially *unconstrained*. Alongside deterministic algorithms, the stochastic gradient descent method (SGD) and the stochastic variance reduced gradient method (SVRG) have also been extended to optimization over Riemannian manifold; see e.g. [28,30,38,61,62]. Compared to all these approaches, our proposed methods allow a nonsmooth objective, a constraint $x_i \in X_i$, as well as the coupling affine constraints. A key feature deviating from the traditional Riemannian optimization is that we take advantage of the global solutions for decoupled proximal mappings instead of relying on a retraction mapping, although if retraction mapping is available then it can be incorporated as well.

Alternating Direction Method of Multipliers (ADMM) has attracted much research attention in the past few years. Convergence and iteration complexity results have been thoroughly studied in the convex setting, and recently results have been extended to various nonconvex settings as well; see [22,23,27,37,52,54,58]. Among these results, [37,52,54,58] show the convergence to a stationary point without any iteration complexity guarantee. A closely related paper is [65], where the authors consider a multi-block nonconvex nonsmooth optimization problem on the Stiefel manifold with coupling linear constraints. An approximate augmented Lagrangian method is proposed to solve the problem and convergence to the KKT point is analyzed, but no iteration complexity result is given. Another related paper is [32], where the authors solve various manifold optimization problems with affine constraints by a two-block ADMM algorithm, without convergence assurance though. The current investigation is inspired by our previous work [27], which requires the convexity of the constraint sets. In the current paper, we drop this restriction and extend the result to stochastic setting and allow Riemannian manifold constraints. Speaking of nonconvex optimization, recent progress can be found under the name *nonsmooth and nonconvex composite optimization*;

see [16–18, 47]. However, in that case, the nonsmooth part of the objective and the constraint set are assumed to be convex, while these can be dropped in our approach as we noted earlier.

Finally, we remark that for large-scale optimization such as tensor decomposition [10, 31, 44], black box tensor approximation problems [4, 45] and the worst-case input models estimation problems [19, 20], the costs for function or gradient evaluation are prohibitively expensive. Our stochastic approach considerably alleviates the computational burden.

1.2 Our contributions

The contributions of this paper can be summarized as follows:

(i) We define the ϵ-stationary solution for problem (1) with Riemmanian manifold constraints.

(ii) We propose a nonconvex proximal gradient-based ADMM algorithm and its linearized variant, and analyze their iteration complexity to reach an ϵ-stationary solution.

(iii) We propose a stochastic variant of the nonconvex linearized proximal gradient-based ADMM with mini-batches, and establish its iteration complexity in the sense of expectation.

(iv) We propose a feasible curvilinear line-search variant of the nonconvex proximal gradient-based ADMM algorithm, where the exact minimization subroutine is replaced by a line-search procedure using a retraction operator. The iteration complexity of the method is established.

(v) We present a number of extensions to the basic method, including relaxing the convexity of nonsmooth component of the objective, and relaxing the condition on the last block matrix A_N. We also extend our analysis from Gauss-Seidel updating to Jacobi updating to enable parallel computing.

1.3 Organization of the paper

The rest of the paper is organized as follows. In Section 2, we review some basics of Riemannian manifold. In the same section we derive the necessary optimality condition for a stationary point and the corresponding ϵ-stationary solution for our optimization problem over Riemannian manifold. In Section 3, we propose a nonconvex proximal gradient-based ADMM and its three variants with iteration complexity bounds. In Section 4, we present extensions of our basic model. In Section 5, we present the implementations of our approach to nonnegative sparse tensor decomposition, the maximum bisection problem, and sparse MPCA. Finally, in Section 6 we present the results of numerical experiments. For the ease of presentation, the proofs of technical lemmas are delegated to the appendix.

2 Optimality over Manifolds

In this section, we shall introduce the basics of optimization over manifolds. The discussion is intended as background information for our purpose; thorough treatments on the topic can be found in, e.g. [3, 36]. We then extend the first-order optimality condition for constrained optimization on manifold established in [59] to our constrained model (1). Based on the optimality condition, we introduce the notion of ϵ-stationary solution, and ϵ-stationary solution in expectation (for the stochastic setting) respectively.

Suppose $\mathcal{M}$ is a differentiable manifold, then for any $x \in \mathcal{M}$, there exists a *chart* (U, ψ) in which U is an open set with $x \in U \subset \mathcal{M}$ and ψ is a homeomorphism between U and an open set $\psi(U)$ in Euclidean space. This coordinate transform enables us to locally treat a Riemannian manifold as a Euclidean space. Denote the tangent space $\mathcal{M}$ at point $x \in \mathcal{M}$ by $\mathcal{T}_x\mathcal{M}$, then $\mathcal{M}$ is a Riemannian manifold if it is equipped with a metric on the tangent space $\mathcal{T}_x\mathcal{M}$ which is continuous in x.

Definition 2.1 (Tangent Space) *Consider a Riemannian manifold $\mathcal{M}$ embedded in a Euclidean space. For any $x \in \mathcal{M}$, the tangent space $\mathcal{T}_x\mathcal{M}$ at x is a linear subspace consists of the derivatives of all smooth curves on $\mathcal{M}$ passing x; that is*

$$\mathcal{T}_x\mathcal{M} = \left\{\gamma'(0) : \gamma(0) = x, \gamma([-\delta, \delta]) \subset \mathcal{M}, \text{ for some } \delta > 0, \gamma \text{ is smooth}\right\}. \tag{2}$$

The Riemannian metric, i.e., the inner product between $u, v \in \mathcal{T}_x\mathcal{M}$, is defined to be $\langle u, v\rangle_x := \langle u, v\rangle$, where the latter is the Euclidean inner product.

Define the set of all functions differentiable at point x to be $\mathcal{F}_x$. An alternative but more general way of defining tangent space is by viewing a tangent vector $v \in \mathcal{T}_x\mathcal{M}$ as an operator mapping $f \in \mathcal{F}_x$ to $v[f] \in \mathbb{R}$ which satisfies the following property: For any given $f \in \mathcal{F}_x$, there exists a smooth curve γ on $\mathcal{M}$ with $\gamma(0) = x$ and $v[f] = \frac{d(f(\gamma(t)))}{dt}\Big|_{t=0}$. For manifolds embedded in Euclidean spaces, we can obtain Definition 2.1 by defining $v = \gamma'(0)$ and $v[f] = \langle \gamma'(0), \nabla f(x)\rangle$.

For example, when $\mathcal{M}$ is a sphere, $\mathcal{T}_x\mathcal{M}$ is the tangent plane at x with a proper translation such that the origin is included. When $\mathcal{M} = \mathbb{R}^n$, then $\mathcal{T}_x\mathcal{M} = \mathbb{R}^n = \mathcal{M}$.

Definition 2.2 (Riemannian Gradient) *For $f \in \mathcal{F}_x$, the Riemannian gradient* $\operatorname{grad} f(x)$ *is a tangent vector in $\mathcal{T}_x\mathcal{M}$ satisfying $v[f] = \langle v, \operatorname{grad} f(x)\rangle_x$ for any $v \in \mathcal{T}_x\mathcal{M}$.*

If $\mathcal{M}$ is an embedded submanifold of a Euclidean space, we have

$$\operatorname{grad} f(x) = \operatorname{Proj}_{\mathcal{T}_x\mathcal{M}}(\nabla f(x)),$$

where $\operatorname{Proj}_{\mathcal{T}_x\mathcal{M}}$ is the Euclidean projection operator onto the subspace $\mathcal{T}_x\mathcal{M}$, which is a nonexpansive linear transformation.

Definition 2.3 (Differential) *Let $F : \mathcal{M} \to \mathcal{N}$ be a smooth mapping between two Riemannian manifolds $\mathcal{M}$ and $\mathcal{N}$. The differential (or push-forward) of F at x is a mapping $\mathbf{D}F(x) : \mathcal{T}_x\mathcal{M} \to \mathcal{T}_{F(x)}\mathcal{N}$ defined by*

$$(\mathbf{D}F(x)[v])[f] = v[f \circ F], \text{ for all } v \in \mathcal{T}_x\mathcal{M}, \text{ and } \forall f \in \mathcal{F}_{F(x)}.$$

Suppose $\mathcal{M}$ is an m-dimensional embedded Riemannian submanifold of $\mathbb{R}^n, m \leq n$, and let (U, ψ) be a chart at point $x \in \mathcal{M}$, then ψ is a smooth mapping from $U \subset \mathcal{M}$ to $\psi(U) \subset \mathcal{N} = \mathbb{R}^m$. Under a proper set of basis $\{\mathbf{a}_i\}_{i=1}^m$ of $\mathcal{T}_x\mathcal{M}$ and suppose $v = \sum_{i=1}^m v_i\mathbf{a}_i$, then

$$\hat{v} := \mathbf{D}\psi(x)[v] = (v_1, ..., v_m).$$

Clearly, this establishes a bijection between the tangent space $\mathcal{T}_x\mathcal{M}$ and the tangent space of $\mathcal{T}_{\psi(x)}\psi(U) = \mathbb{R}^m$. Following the notation in [59], we use $\hat{o}$ to denote the Euclidean counterpart of an object o in $\mathcal{M}$; e.g.,

$$\hat{f} = f \circ \psi^{-1}, \ \hat{v} = \mathbf{D}\psi(x)[v], \ \hat{x} = \psi(x).$$

Finally, if we define the Gram matrix $G_x(i, j) = \langle \mathbf{a}_i, \mathbf{a}_j \rangle_x$, which is also known as the Riemannian metric, then $\langle u, v \rangle_x = \hat{u}^\top G_x \hat{v}$.

Next, we shall present a few optimization concepts generalized to the manifold case. Let C be a subset in $\mathbb{R}^n$ and $x \in C$, the tangent cone $T_C(x)$ and the normal cone $N_C(x)$ of C at x are defined in accordance with that in [43]. Suppose S is a closed subset on the Riemannian manifold $\mathcal{M}$, (U, ψ) is a chart at point $x \in S$, then by using coordinate transform (see also [41, 59]), the Riemannian tangent cone can be defined as

$$\mathcal{T}_S(x) := [\mathbf{D}\psi(x)]^{-1}[T_{\psi(S \cap U)}(\psi(x))]. \tag{3}$$

Consequently, the Riemannian normal cone can be defined as

$$\mathcal{N}_S(x) := \{u \in \mathcal{T}_x\mathcal{M} : \langle u, v \rangle_x \leq 0, \forall v \in \mathcal{T}_S(x)\}. \tag{4}$$

By a rather standard argument (see [59]), the following proposition can be shown:

Proposition 2.4 $\mathcal{N}_S(x) = [\mathbf{D}\psi(x)]^{-1}[G_x^{-1}N_{\psi(U \cap S)}(\psi(x))]$.

A function f is said to be locally Lipschitz on $\mathcal{M}$ if for any $x \in \mathcal{M}$, there exists some $L > 0$ such that in a neighborhood of x, f is L-Lipschitz in the sense of Riemannian distance. When $\mathcal{M}$ is a compact manifold, a global L exists. When $\mathcal{M}$ is an embedded submanifold of $\mathbb{R}^n$ and f is a locally Lipschitz on $\mathbb{R}^n$, let $f|_{\mathcal{M}}$ be the function f restricted to $\mathcal{M}$, then $f|_{\mathcal{M}}$ is also locally Lipschitz on $\mathcal{M}$.

Definition 2.5 (The Clarke subdifferential on Riemannian manifold [24, 59]) *For a locally Lipschitz continuous function f on $\mathcal{M}$, the* Riemannian generalized directional derivative *of f at*

$x \in \mathcal{M}$ *in direction* $v \in \mathcal{T}_x\mathcal{M}$ *is defined as*

$$f^\circ(x; v) = \limsup_{y \to x, t \downarrow 0} \frac{f \circ \psi^{-1}(\psi(y) + t\mathbf{D}\psi(y)[v]) - f \circ \psi^{-1}}{t}. \tag{5}$$

Then the Clarke subdifferential is defined as

$$\partial f(x) = \{\xi \in \mathcal{T}_x\mathcal{M} : \langle \xi, v\rangle \leq f^\circ(x; v), \forall v \in \mathcal{T}_x\mathcal{M}\}. \tag{6}$$

There are several remarks for the notion of Riemannian Clarke subdifferentials. If $\mathcal{M} = \mathbb{R}^n$ and $\psi = id$, then the above notion reduces to the original Clarke subdifferential [9]. In this case, suppose f is differentiable and r is locally Lipschitz, then we have

$$\partial(f + r)(x) = \nabla f(x) + \partial r(x), \tag{7}$$

where $\partial r(x)$ is the Clarke subdifferential. Furthermore, if we have additional manifold constraints and r is convex, from [59] we have

$$\partial(f + r)|_{\mathcal{M}}(x) = \text{Proj}_{\mathcal{T}_x\mathcal{M}}(\nabla f(x) + \partial r(x)). \tag{8}$$

The convexity of r is crucial in this property. If the nonsmooth part $r_i(x_i)$ in our problem is also nonconvex, then we will have to use additional variables and consensus constraints to decouple r_i, the manifold constraint and smooth component f, which will be discussed in Section 4. More importantly, we have the following result (see [59]):

Proposition 2.6 *Suppose* f *is locally Lipschitz continuous in a neighborhood of* x*, and* (U, ψ) *is a chart at* x*. It holds that*

$$\partial f(x) = [\mathbf{D}\psi(x)]^{-1}[G_x^{-1}\partial(f \circ \psi^{-1})(\psi(x))].$$

2.1 Optimality condition and the ϵ-stationary solution

Consider the following optimization problem over manifold:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & x \in S \subset \mathcal{M}. \end{aligned} \tag{9}$$

Suppose that x^* is a local minimum, and that (U, ψ) is a chart at x^*. Then, $\hat{x}^* := \psi(x^*)$ must also be a local minimum for the problem

$$\begin{aligned} \min \quad & \hat{f}(\hat{x}) \\ \text{s.t.} \quad & \hat{x} \in \psi(S \cap U). \end{aligned} \tag{10}$$

Therefore, problem (9) is transformed into a standard nonlinear programming problem (10) in Euclidean space. We will then find the optimality condition via (10) and map it back to that of (9) by using the differential operator.

Assume that both $\hat{f}$ and f are locally Lipschitz. The optimality of $\hat{x}^*$ yields (cf. [9])

$$0 \in \partial \hat{f}(\hat{x}^*) + N_{\psi(U \cap S)}(\hat{x}^*).$$

Apply the bijection $[\mathbf{D}\psi(x)]^{-1} \circ G_x^{-1}$ on both sides, and by Propositions 2.6 and 2.4, the first-order optimality condition for problem (9) follows as a result:

$$0 \in \partial f(x^*) + \mathcal{N}_S(x^*). \tag{11}$$

If f is differentiable, then (11) reduces to

$$-\text{grad}\, f(x^*) \in \mathcal{N}_S(x^*).$$

To specify the set S in problem (1), let us consider an equality constrained problem

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & c_i(x) = 0, i = 1, ..., m, \\ & x \in \mathcal{M} \cap X. \end{array} \tag{12}$$

Note that in the case of (1), the above constraints $c_i(x) = 0$, $i = 1, 2, ..., m$, represent the linear equality constraints. Define $\Omega := \{x \in \mathcal{M} : c_i(x) = 0,\ i = 1, ..., m\}$, and $S := \Omega \cap X$. By assuming the so-called Linear Independent Constraint Qualification (LICQ) condition on the Jacobian of $c(x)$ at x^*, Corollary 4.1 in [59] implies

$$\mathcal{N}_\Omega(x^*) = \left\{ \sum_{i=1}^{m} \lambda_i \,\text{grad}\, c_i(x^*) \,\middle|\, \lambda \in \mathbb{R}^m \right\} = -(\mathcal{T}_\Omega(x^*))^\star, \tag{13}$$

where $\mathcal{K}^\star$ indicates the dual of cone $\mathcal{K}$. Therefore, (11) implies

$$\partial f(x^*) \cap (-\mathcal{N}_S(x^*)) \neq \emptyset.$$

We have

$$\begin{array}{rcl} -(\mathcal{N}_\Omega(x^*) + \mathcal{N}_X(x^*)) & = & (\mathcal{T}_\Omega(x^*))^\star + (\mathcal{T}_X(x^*))^\star \\ & \subseteqq & \text{cl}\,((\mathcal{T}_\Omega(x^*))^\star + (\mathcal{T}_X(x^*))^\star) \\ & = & (\mathcal{T}_\Omega(x^*) \cap \mathcal{T}_X(x^*))^\star \\ & \subseteqq & (\mathcal{T}_{\Omega \cap X}(x^*))^\star. \end{array}$$

The optimality condition is established as:

Proposition 2.7 *Suppose that $x^* \in \mathcal{M} \cap X$ and $c_i(x^*) = 0, i = 1, ..., m$. If*

$$\partial f(x^*) \cap (-\mathcal{N}_\Omega(x^*) - \mathcal{N}_X(x^*)) \neq \emptyset,$$

then x^* is a stationary solution for problem (12).

By specifying the optimality condition in Proposition 2.7 to (1), we have:

Theorem 2.8 *Consider problem* (1) *where f is smooth with Lipschitz gradient and r_i's are convex and locally Lipschitz continuous. If there exists a Lagrange multiplier λ^* such that*

$$\begin{cases} \nabla_N f(x^*) - A_N^\top \lambda^* = 0, \\ \sum_{i=1}^N A_i x_i^* - b = 0, \\ \text{Proj}\,_{\mathcal{T}_{x_i^*}\mathcal{M}_i}\left(\nabla_i f(x^*) - A_i^\top \lambda^* + \partial r_i(x_i^*)\right) + \mathcal{N}_{X_i \cap \mathcal{M}_i}(x_i^*) \ni 0, i = 1, ..., N-1, \end{cases} \tag{14}$$

then x^* is a stationary solution for problem (1).

Hence, an ϵ-stationary solution of problem (1) can be naturally defined as:

Definition 2.9 (ϵ-stationary solution) *Consider problem* (1) *where f is smooth with Lipschitz gradient and r_i are convex and locally Lipschitz continuous. Solution x^* is said to be an ϵ-stationary solution if there exists a multiplier λ^* such that*

$$\begin{cases} \|\nabla_N f(x^*) - A_N^\top \lambda^*\| \leq \epsilon, \\ \|\sum_{i=1}^N A_i x_i^* - b\| \leq \epsilon, \\ \text{dist}\left(\text{Proj}\,_{\mathcal{T}_{x_i^*}\mathcal{M}_i}\left(-\nabla_i f(x^*) + A_i^\top \lambda^* - \partial r_i(x_i^*)\right), \mathcal{N}_{X_i \cap \mathcal{M}_i}(x_i^*)\right) \leq \epsilon, i = 1, ..., N-1. \end{cases}$$

In the case that x^* is a vector generated by some randomized algorithm, the following adaptation is appropriate.

Definition 2.10 (ϵ-stationary solution in expectation) *Suppose that x^* and λ^* are generated by some randomized process. Then, we call x^* and λ^* to be ϵ-stationary solution for problem* (1) *in expectation if the following holds*

$$\begin{cases} \mathsf{E}\left[\|\nabla_N f(x^*) - A_N^\top \lambda^*\|\right] \leq \epsilon, \\ \mathsf{E}\left[\|\sum_{i=1}^N A_i x_i^* - b\|\right] \leq \epsilon, \\ \mathsf{E}\left[\text{dist}\left(\text{Proj}\,_{\mathcal{T}_{x_i^*}\mathcal{M}_i}\left(-\nabla_i f(x^*) + A_i^\top \lambda^* - \partial r_i(x_i^*)\right), \mathcal{N}_{X_i \cap \mathcal{M}_i}(x_i^*)\right)\right] \leq \epsilon, i = 1, ..., N-1. \end{cases}$$

3 Proximal Gradient ADMM and Its Variants

In [27], Jiang, Lin, Ma and Zhang proposed a proximal gradient-based variant of ADMM for nonconvex and nonsmooth optimization model with convex constraints. In this paper, we extend the analysis to include nonconvex Riemannian manifold constraints, motivated by the vast array of potential applications. Moreover, we propose to linearize the nonconvex function f, which significantly broadens the applicability and enables us to utilize the stochastic gradient-based method to reduce computational costs for large-scale problems. As it turns out, the convergence result for this variant remains intact.

Concerning problem (1), we first make some assumptions on f and r_i's.

Assumption 3.1 *f and $r_i, i = 1, ..., N-1$, are all bounded from bellow in the feasible region. We denote the lower bounds by $r_i^* = \min_{x_i \in \mathcal{M}_i \cap X_i} r_i(x_i), i = 1, ..., N-1$ and*

$$f^* = \min_{x_i \in \mathcal{M}_i \cap X_i, i=1,...,N-1, x_N \in \mathbb{R}^{n_N}} f(x_1, \cdots, x_N).$$

Assumption 3.2 *f is a smooth function with L-Lipschitz continuous gradient; i.e.*

$$\|\nabla f(x_1, \ldots, x_N) - \nabla f(\hat{x}_1, \ldots, \hat{x}_N)\|_2 \leq L\|(x_1 - \hat{x}_1, \ldots, x_N - \hat{x}_N)\|_2, \ \ \forall x, \hat{x}. \tag{15}$$

Assumption 3.3 *The proximal mappings required at Step 1 of Algorithms 1, 2 and 3 are all computable. (As we will see in Section 5, this assumption holds true for many practical applications).*

3.1 Nonconvex proximal gradient-based ADMM

The augmented Lagrangian function for problem (1) is

$$\mathcal{L}_\beta(x_1, x_2, \cdots, x_N, \lambda) = f(x_1, \cdots, x_N) + \sum_{i=1}^{N-1} r_i(x_i) - \left\langle \sum_{i=1}^{N} A_i x_i - b, \lambda \right\rangle + \frac{\beta}{2} \left\| \sum_{i=1}^{N} A_i x_i - b \right\|^2, \tag{16}$$

where λ is the Lagrange multiplier, $\beta > 0$ is a penalty parameter. Our proximal gradient-based ADMM for solving (1) is described in Algorithm 1.

Algorithm 1: Nonconvex Proximal Gradient-Based ADMM on Riemannian Manifold

Given $(x_1^0, x_2^0, \cdots, x_N^0) \in (\mathcal{M}_1 \cap X_1) \times (\mathcal{M}_2 \cap X_2) \times \cdots \times (\mathcal{M}_{N-1} \cap X_{N-1}) \times \mathbb{R}^{n_N}$, $\lambda^0 \in \mathbb{R}^m$, $\beta > 0$, $\gamma > 0$, $H_i \succ 0, i = 1, \ldots, N-1$.

for $k = 0, 1, ...$ **do**

- [Step 1] For $i = 1, 2, ..., N-1$, and positive semi-definite matrix H_i, compute $x_i^{k+1} := \operatorname{argmin}_{x_i \in \mathcal{M}_i \cap X_i} \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \cdots, x_N^k, \lambda^k) + \frac{1}{2}\|x_i - x_i^k\|_{H_i}^2$;
- [Step 2] $x_N^{k+1} := x_N^k - \gamma \nabla_N \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k)$;
- [Step 3] $\lambda^{k+1} := \lambda^k - \beta(\sum_{i=1}^N A_i x_i^{k+1} - b)$.

Before we give the main convergence result of Algorithm 1, we need the following lemmas. Lemmas 3.4 and 3.6 are from [27]; and the proof of Lemma 3.5 is in the appendix.

Lemma 3.4 *(Lemma 3.9 in [27]) Suppose that the sequence $\{x_1^k, ..., x_N^k, \lambda^k\}$ is generated by Algorithm 1. Then,*

$$\begin{aligned}\|\lambda^{k+1} - \lambda^k\|^2 \;\leq\; & 3\left(\beta - \frac{1}{\gamma}\right)^2 \|x_N^k - x_N^{k+1}\|^2 + 3\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right] \|x_N^{k-1} - x_N^k\|^2 \\ & + 3L^2 \sum_{i=1}^{N-1} \|x_i^k - x_i^{k+1}\|^2. \end{aligned} \tag{17}$$

Since Steps 2 and 3 in Algorithm 1 are the same as those in [27], this lemma remains valid here. Specially, Step 2 and Step 3 directly result in

$$\lambda^{k+1} = \left(\beta - \frac{1}{\gamma}\right)(x_N^k - x_N^{k+1}) + \nabla_N f(x_1^{k+1}, \ldots, x_{N-1}^{k+1}, x_N^k). \tag{18}$$

We define a potential function

$$\Psi_G(x_1, \cdots, x_N, \lambda, \bar{x}) = \mathcal{L}_\beta(x_1, \cdots, x_N, \lambda) + \frac{3}{\beta}\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right] \|\bar{x} - x_N\|^2. \tag{19}$$

With Lemma 3.4, the following monotonicity property can be established.

Lemma 3.5 *Suppose the sequence $\{(x_1^k, \cdots, x_N^k, \lambda_k)\}$ is generated by Algorithm 1. Assume that*

$$\beta > \left(\frac{6 + 18\sqrt{3}}{13}\right) L \approx 2.860L \text{ and } H_i \succ \frac{6L^2}{\beta} I, i = 1, ..., N-1. \tag{20}$$

Then $\Psi_G(x_1^{k+1}, \cdots, x_N^{k+1}, \lambda^{k+1}, x_N^k)$ is monotonically decreasing over k if γ lies in the following interval:

$$\gamma \in \left(\frac{12}{13\beta + \sqrt{13\beta^2 - 12\beta L - 72L^2}}, \frac{12}{13\beta - \sqrt{13\beta^2 - 12\beta L - 72L^2}}\right). \tag{21}$$

More specifically, we have

$$\begin{aligned} & \Psi_G(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^{k+1}, \lambda^{k+1}, x_N^k) - \Psi_G(x_1^k, \cdots, x_{N-1}^k, x_N^k, \lambda^k, x_N^{k-1}) \\ \leq\; & \left[\frac{\beta + L}{2} - \frac{1}{\gamma} + \frac{6}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{3L^2}{\beta}\right] \|x_N^k - x_N^{k+1}\|^2 \\ & - \sum_{i=1}^{N-1} \|x_i^k - x_i^{k+1}\|^2_{\frac{1}{2}H_i - \frac{3L^2}{\beta}I} < 0. \end{aligned} \tag{22}$$

Lemma 3.6 *(Lemma 3.11 in [27]) Suppose that the sequence $\{x_1^k, \cdots, x_N^k, \lambda^k\}$ is generated by Algorithm 1. It holds that*

$$\Psi_G(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^{k+1}, \lambda^{k+1}, x_N^k) \geq \sum_{i=1}^{N-1} r_i^* + f^*, \tag{23}$$

where $r_i^*, i = 1, ..., N-1$ and f^* are defined in Assumption 3.1.

Denote $\sigma_{\min}(M)$ as the smallest singular value of a matrix M. Now we are ready to present the main convergence result of Algorithm 1.

Theorem 3.7 *Suppose that the sequence $\{x_1^k, ..., x_N^k, \lambda^k\}$ is generated by Algorithm 1, and the parameters β and γ satisfy (20) and (21) respectively. Define $\kappa_1 := \frac{3}{\beta^2}\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right]$, $\kappa_2 := \left(|\beta - \frac{1}{\gamma}| + L\right)^2$, $\kappa_3 := \left(L + \beta\sqrt{N}\max_{1\leq i\leq N}\|A_i\|_2^2 + \max_{1\leq i\leq N-1}\|H_i\|_2\right)^2$ and $\tau := \min\left\{-\left[\frac{\beta+L}{2} - \frac{1}{\gamma} + \frac{6}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{3L^2}{\beta}\right], \min_{i=1,...,N-1}\left[-\left(\frac{3L^2}{\beta} - \frac{\sigma_{\min}(H_i)}{2}\right)\right]\right\}$. Assuming $H_i \succ \frac{6L^2}{\beta}I$ and letting*

$$K := \left\lceil \frac{2\max\{\kappa_1, \kappa_2, \kappa_3\}}{\tau\epsilon^2}\left(\Psi_G(x_1^1, ..., x_N^1, \lambda^1, x_N^0) - \sum_{i=1}^{N-1} r_i^* - f^*\right)\right\rceil, \tag{24}$$

and $k^* := \operatorname{argmin}_{2\leq k\leq K+1}\sum_{i=1}^N(\|x_i^k - x_i^{k+1}\|^2 + \|x_i^{k-1} - x_i^k\|^2)$, it follows that $(x_1^{k^*+1}, \cdots, x_N^{k^*+1}, \lambda^{k^*+1})$ is an ϵ-stationary solution of (1) defined in Definition 2.9.

Proof. For the ease of presentation, we denote

$$\theta_k := \sum_{i=1}^N(\|x_i^k - x_i^{k+1}\|^2 + \|x_i^{k-1} - x_i^k\|^2). \tag{25}$$

Summing (22) over $k = 1, \ldots, K$ yields

$$\Psi_G(x_1^1, \cdots, x_N^1, \lambda^1, x_N^0) - \Psi_G(x_1^{K+1}, \cdots, x_N^{K+1}, \lambda^{K+1}, x_N^K) \geq \tau\sum_{k=1}^K\sum_{i=1}^N\|x_i^k - x_i^{k+1}\|^2, \tag{26}$$

which implies

$$\begin{aligned}
&\min_{2\leq k\leq K+1}\theta_k \\
\leq\ & \frac{1}{\tau K}\left[2\Psi_G(x_1^1, \cdots, x_N^1, \lambda^1, x_N^0) - \Psi_G(x_1^{K+1}, \ldots, x_N^{K+1}, \lambda^{K+1}, x_N^K) - \Psi_G(x_1^{K+2}, \ldots, x_N^{K+2}, \lambda^{K+2}, x_N^{K+1})\right] \\
\leq\ & \frac{2}{\tau K}\left[\Psi_G(x_1^1, \cdots, x_N^1, \lambda^1, x_N^0) - f^* - \sum_{i=1}^{N-1} r_i^*\right].
\end{aligned} \tag{27}$$

By (18) we have

$$\begin{aligned}
&\|\lambda^{k+1} - \nabla_N f(x_1^{k+1}, \cdots, x_N^{k+1})\|^2 \\
\leq\ & \left(\left|\beta - \frac{1}{\gamma}\right| \|x_N^k - x_N^{k+1}\| + \|\nabla_N f(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k) - \nabla_N f(x_1^{k+1}, \cdots, x_N^{k+1})\|\right)^2 \\
\leq\ & \left(\left|\beta - \frac{1}{\gamma}\right| + L\right)^2 \|x_N^k - x_N^{k+1}\|^2 \\
\leq\ & \kappa_2 \theta_k. \qquad (28)
\end{aligned}$$

From Step 3 of Algorithm 1 and (17), we have

$$\begin{aligned}
&\left\|\sum_{i=1}^{N-1} A_i x_i^{k+1} + x_N^{k+1} - b\right\|^2 \\
=\ & \frac{1}{\beta^2}\|\lambda^k - \lambda^{k+1}\|^2 \\
\leq\ & \frac{3}{\beta^2}\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right]\|x_N^{k-1} - x_N^k\|^2 + \frac{3}{\beta^2}\left(\beta - \frac{1}{\gamma}\right)^2 \|x_N^{k+1} - x_N^k\|^2 + \frac{3L^2}{\beta^2}\sum_{i=1}^{N-1}\|x_i^{k+1} - x_i^k\|^2 \\
\leq\ & \kappa_1 \theta_k. \qquad (29)
\end{aligned}$$

By the optimality conditions (e.g., (11)) for the subproblems in Step 1 of Algorithm 1, and using (8) and Step 3 of Algorithm 1, we can get

$$\begin{aligned}
\mathrm{Proj}_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\Bigg\{\nabla_i f(x_1^{k+1}, \cdots, x_i^{k+1}, x_{i+1}^k, \cdots, x_N^k) - A_i^\top \lambda^{k+1} + \beta A_i^\top \left(\sum_{j=i+1}^{N} A_j(x_j^k - x_j^{k+1})\right) \\
+ H_i(x_i^{k+1} - x_i^k) + g_i(x_i^{k+1})\Bigg\} + q_i(x_i^{k+1}) = 0, \quad (30)
\end{aligned}$$

for some $g_i(x_i^{k+1}) \in \partial r_i(x_i^{k+1})$, $q_i(x_i^{k+1}) \in \mathcal{N}_{X_i}(x_i^{k+1})$. Therefore,

$$\begin{aligned}
&\mathrm{dist}\left(\mathrm{Proj}_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\left\{-\nabla_i f(x^{k+1}) + A_i^\top \lambda^{k+1} - \partial r_i(x_i^{k+1})\right\}, \mathcal{N}_{X_i}(x_i^{k+1})\right) \\
\leq\ & \left\|\mathrm{Proj}_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\left\{-\nabla_i f(x^{k+1}) + A_i^T \lambda^{k+1} - g_i(x_i^{k+1}) - q_i(x_i^{k+1})\right\}\right\| \\
=\ & \Bigg\|\mathrm{Proj}_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\Bigg\{-\nabla_i f(x^{k+1}) + \nabla_i f(x_1^{k+1}, \cdots, x_i^{k+1}, x_{i+1}^k, \cdots, x_N^k) \\
& + \beta A_i^\top\Big(\sum_{j=i+1}^{N} A_j(x_j^k - x_j^{k+1})\Big) + H_i(x_i^{k+1} - x_i^k)\Bigg\}\Bigg\|
\end{aligned}$$

$$
\begin{aligned}
&\leq \| -\nabla_i f(x^{k+1}) + \nabla_i f(x_1^{k+1}, \cdots, x_i^{k+1}, x_{i+1}^k, \cdots, x_N^k) - H_i(x_i^{k+1} - x_i^k) \\
&\quad + \beta A_i^\top \Big(\sum_{j=i+1}^N A_j(x_j^{k+1} - x_j^k) \Big) \| \\
&\leq \|\nabla_i f(x^{k+1}) - \nabla_i f(x_1^{k+1}, \cdots, x_i^{k+1}, x_{i+1}^k, \cdots, x_N^k)\| + \|H_i(x_i^{k+1} - x_i^k)\| \\
&\quad + \|\beta A_i^\top \Big(\sum_{j=i+1}^N A_j(x_j^{k+1} - x_j^k) \Big) \| \\
&\leq \left(L + \beta \max_{1\leq j\leq N} \|A_j\|_2^2 \sqrt{N} \right) \sqrt{\sum_{j=i+1}^N \|x_j^{k+1} - x_j^k\|^2} + \max_{1\leq j\leq N-1} \|H_j\|_2 \|x_i^k - x_i^{k+1}\| \\
&\leq \sqrt{\kappa_3 \theta_k}. \qquad (31)
\end{aligned}
$$

Combining (28), (29), (31) and (54) yields the desired result. □

3.2 Nonconvex linearized proximal gradient-based ADMM

When modeling nonconvex and nonsmooth optimization with manifold constraints, it is often the case that computing proximal mapping (in the presence of f) may be difficult, while optimizing with a quadratic objective is still possible. This leads to a variant of ADMM which linearizes the f function. In particular, we define the following approximation to the augmented Lagrangian function:

$$
\begin{aligned}
\hat{\mathcal{L}}_\beta^i(x_i; \hat{x}_1, \cdots, \hat{x}_N, \lambda) \;:=\; & f(\hat{x}_1, \cdots, \hat{x}_N) + \langle \nabla_i f(\hat{x}_1, \cdots, \hat{x}_N), x_i - \hat{x}_i \rangle + r_i(x_i) \\
& - \Big\langle \sum_{j=1, j\neq i}^N A_j \hat{x}_j + A_i x_i - b, \lambda \Big\rangle + \frac{\beta}{2} \Big\| \sum_{j=1, j\neq i}^N A_j \hat{x}_j + A_i x_i - b \Big\|^2, \qquad (32)
\end{aligned}
$$

where λ is the Lagrange multiplier and $\beta > 0$ is a penalty parameter. It is worth noting that this approximation is defined with respect to a particular block of variable x_i. The linearized proximal gradient-based ADMM algorithm is described as in Algorithm 2.

Algorithm 2: Nonconvex Linearized Proximal Gradient-Based ADMM

Given $(x_1^0, x_2^0, \cdots, x_N^0) \in (\mathcal{M}_1 \cap X_1) \times (\mathcal{M}_2 \cap X_2) \times \cdots \times (\mathcal{M}_{N-1} \cap X_{N-1}) \times \mathbb{R}^{n_N}$, $\lambda^0 \in \mathbb{R}^m$, $\beta > 0$, $\gamma > 0$, $H_i \succ 0, i = 1, \ldots, N-1$.

for $k = 0, 1, \ldots$ **do**

- [Step 1] For $i = 1, 2, ..., N-1$ and positive semi-definite matrix H_i, compute $x_i^{k+1} := \operatorname{argmin}_{x_i \in \mathcal{M}_i \cap X_i} \hat{\mathcal{L}}_\beta^i(x_i; x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i^k, \cdots, x_N^k, \lambda^k) + \frac{1}{2}\|x_i - x_i^k\|_{H_i}^2$,
- [Step 2] $x_N^{k+1} := x_N^k - \gamma \nabla_N \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k)$,
- [Step 3] $\lambda^{k+1} := \lambda^k - \beta(\sum_{i=1}^N A_i x_i^{k+1} - b)$.

Essentially, instead of solving the subproblem involving the exact augmented Lagrangian defined by (16), we use the linearized approximation defined in (32). It is also noted that the Steps 2 and 3 of Algorithm 2 are the same as the ones in Algorithm 1, and thus Lemmas 3.4 and 3.6 still hold, as they do not depend on Step 1 of the algorithms. As a result, we only need to present the following lemma, which is a counterpart of Lemma 3.5, and the proof is given in the appendix.

Lemma 3.8 *Suppose that the sequence $(x_i^k, \cdots, x_N^k, \lambda^k)$ is generated by Algorithm 2. Let the parameters β and γ be defined according to* (20) *and* (21)*, and $\Psi_G(x_1, \cdots, x_N, \lambda, \bar{x})$ be defined according to* (19)*. If we choose*

$$H_i \succ \left(\frac{6L^2}{\beta} + L\right) I, \ for \ i = 1, ..., N-1,$$

then $\Psi_G(x_1^{k+1}, \cdots, x_N^{k+1}, \lambda^{k+1}, x_N^k)$ monotonically decreases. More specifically, we have

$$\begin{aligned} & \Psi_G(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^{k+1}, \lambda^{k+1}, x_N^k) - \Psi_G(x_1^k, \cdots, x_{N-1}^k, x_N^k, \lambda^k, x_N^{k-1}) \\ \leq & \left[\frac{\beta + L}{2} - \frac{1}{\gamma} + \frac{6}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{3L^2}{\beta}\right] \|x_N^k - x_N^{k+1}\|^2 \\ & - \sum_{i=1}^{N-1} \|x_i^k - x_i^{k+1}\|_{\frac{1}{2}H_i - \frac{L}{2}I - \frac{3L^2}{\beta}I}. \end{aligned} \tag{33}$$

Note that the right hand side of (33) *is negative under the above conditions.*

We are now ready to present the main complexity result for Algorithm 2, and the proof is omitted because it is very similar to that of Theorem 3.7.

Theorem 3.9 *Suppose the sequence $\{x_1^k, \cdots, x_N^k, \lambda^k\}$ is generated by Algorithm 2. Let the parameters β and γ satisfy* (20) *and* (21) *respectively. Define $\kappa_1, \kappa_2, \kappa_3$ same as that in Theorem 3.7. Define*

$$\tau := \min\left\{-\left[\frac{\beta + L}{2} - \frac{1}{\gamma} + \frac{6}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{3L^2}{\beta}\right], \min_{i=1,...,N-1}\left\{-\left(\frac{3L^2}{\beta} + \frac{L}{2} - \frac{\sigma_{\min}(H_i)}{2}\right)\right\}\right\}.$$

Assume $H_i \succ \left(\frac{6L^2}{\beta} + L\right) I$, and let

$$K = \left\lceil \frac{2\max\{\kappa_1, \kappa_2, \kappa_3\}}{\tau\epsilon^2}\left(\Psi_G(x_1^1, \cdots, x_N^1, \lambda^1, x_N^0) - \sum_{i=1}^{N-1} r_i^* - f^*\right)\right\rceil,$$

and $k^* = \text{argmin}_{2 \leq k \leq K+1} \sum_{i=1}^N (\|x_i^k - x_i^{k+1}\|^2 + \|x_i^{k-1} - x_i^k\|^2)$. Then, $(x_1^{k^*+1}, \cdots, x_N^{k^*+1}, \lambda^{k^*+1})$ is an ϵ-stationary solution defined in Definition 2.9.

3.3 Stochastic linearized proximal ADMM

In machine learning applications, the objective is often in the form of

$$f(x_1, \cdots, x_N) = \frac{1}{m}\sum_{i=1}^{m} f_i(x_1, \cdots, x_N),$$

where f_i corresponds to the loss function of the ith training data, and the sample size m can be a very large number. In rank-1 CP tensor decomposition problem, people aim to find the best rank-1 CP approximation of an order-d tensor $\mathbf{T} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$. With proper transformation, the objective function f is

$$f(x_1, \cdots, x_N) = \langle \mathbf{T}, \otimes_{i=1}^{d} x_i \rangle,$$

where complete description of $\mathbf{T}$ is exponentially expensive. In such cases, function evaluations in Algorithm 1, and the gradient evaluations in Algorithm 2 are prohibitively expensive. In this section, we propose a nonconvex linearized stochastic proximal gradient-based ADMM with mini-batch to resolve this problem. First, let us make the following assumption.

Assumption 3.10 *For smooth f and $i = 1, \ldots, N$, there exists a stochastic first-order oracle that returns a noisy estimation to the partial gradient of f with respect to x_i, and the noisy estimation $G_i(x_1, \cdots, x_N, \xi_i)$ satisfies*

$$\mathsf{E}[G_i(x_1, \cdots, x_N, \xi_i)] = \nabla_i f(x_1, \cdots, x_N), \tag{34}$$

$$\mathsf{E}\left[\|G_i(x_1, \cdots, x_N, \xi_i) - \nabla_i f(x_1, \cdots, x_N)\|^2\right] \leq \sigma^2, \tag{35}$$

where the expectation is taken with respect to the random variable ξ_i.

Let M be the size of mini-batch, and denote

$$G_i^M(x_1, \cdots, x_N) := \frac{1}{M}\sum_{j=1}^{M} G_i(x_1, \cdots, x_N, \xi_i^j),$$

where $\xi_i^j, j = 1, ..., M$ are i.i.d. random variables. Clearly it holds that

$$\mathsf{E}[G_i^M(x_1, \cdots, x_N)] = \nabla_i f(x_1, \cdots, x_N)$$

and

$$\mathsf{E}\left[\|G_i^M(x_1, \cdots, x_N) - \nabla_i f(x_1, \cdots, x_N)\|^2\right] \leq \sigma^2/M. \tag{36}$$

Now, the stochastic linear approximation of the augmented Lagrangian function with respect to block x_i at point $(\hat{x}_1, \cdots, \hat{x}_N)$ is defined as (note that $r_N \equiv 0$):

$$\begin{aligned}\tilde{\mathcal{L}}_\beta^i(x_i; \hat{x}_1, \cdots, \hat{x}_N, \lambda; M) &= f(\hat{x}_1, \cdots, \hat{x}_N) + \langle G_i^M(\hat{x}_1, \cdots, \hat{x}_N), x_i - \hat{x}_i \rangle + r_i(x_i) \\ &\quad - \left\langle \sum_{j \neq i}^{N} A_j \hat{x}_j + A_i x_i - b, \lambda \right\rangle + \frac{\beta}{2}\left\| \sum_{j \neq i}^{N} A_j \hat{x}_j + A_i x_i - b \right\|^2, \end{aligned} \tag{37}$$

where λ and $\beta > 0$ follow the previous definitions. Compared to (32), the full partial derivative $\nabla_i f$ is replaced by the sample average of stochastic first-order oracles.

Algorithm 3: Nonconvex Linearized Stochastic Proximal Gradient-Based ADMM

Given $(x_1^0, x_2^0, \cdots, x_N^0) \in (\mathcal{M}_1 \cap X_1) \times (\mathcal{M}_2 \cap X_2) \times \cdots \times (\mathcal{M}_{N-1} \cap X_{N-1}) \times \mathbb{R}^{n_N}$, $\lambda^0 \in \mathbb{R}^m$, $\beta > 0$, $\gamma > 0$, $H_i \succ 0, i = 1, \ldots, N-1$, and the batch-size M.

for $k = 0, 1, ...$ **do**

[Step 1] For $i = 1, 2, ..., N-1$, and positive semi-definite matrix H_i, compute $x_i^{k+1} = \text{argmin}_{x_i \in \mathcal{M}_i \cap X_i} \tilde{\mathcal{L}}_\beta^i(x_i; x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i^k, \cdots, x_N^k, \lambda^k; M) + \frac{1}{2}\|x_i - x_i^k\|_{H_i}^2$;

[Step 2] $x_N^{k+1} = x_N^k - \gamma \nabla_N \tilde{\mathcal{L}}_\beta^N(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k)$;

[Step 3] $\lambda^{k+1} = \lambda^k - \beta(\sum_{i=1}^N A_i x_i^{k+1} - b)$.

The convergence analysis of this algorithm follows the similar logic as that of the previous two algorithms. The proofs of these lemmas can be found in the appendix.

Lemma 3.11 *The following inequality holds:*

$$\begin{aligned}\mathsf{E}[\|\lambda^{k+1} - \lambda^k\|^2] \;\leq\; & 4\left(\beta - \frac{1}{\gamma}\right)^2 \mathsf{E}[\|x_N^k - x_N^{k+1}\|^2] + 4\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right] \mathsf{E}[\|x_N^{k-1} - x_N^k\|^2] \\ & + 4L^2 \sum_{i=1}^{N-1} \mathsf{E}[\|x_i^k - x_i^{k+1}\|^2] + \frac{8}{M}\sigma^2. \end{aligned} \tag{38}$$

In the stochastic setting, define the new potential function

$$\Psi_S(x_1, \cdots, x_N, \lambda, \bar{x}) = \mathcal{L}_\beta(x_1, \cdots, x_N, \lambda) + \frac{4}{\beta}\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right] \|\bar{x} - x_N\|^2. \tag{39}$$

Lemma 3.12 *Suppose the sequence $\{(x_1^k, ..., x_N^k, \lambda_k)\}$ is generated by Algorithm 3. Define $\Delta = 17\beta^2 - 16(L+1)\beta - 128L^2$, and assume that*

$$\beta \in \left(\frac{8(L+1) + 8\sqrt{(L+1)^2 + 34L^2}}{17}, +\infty\right), H_i \succ \left(\frac{8L^2}{\beta} + L + 1\right) I,\ i = 1, \ldots, N-1, \tag{40}$$

$$\gamma \in \left(\frac{16}{17\beta + \sqrt{\Delta}}, \frac{16}{17\beta - \sqrt{\Delta}}\right). \tag{41}$$

Then it holds that

$$\begin{aligned} & \mathsf{E}[\Psi_S(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^{k+1}, \lambda^{k+1}, x_N^k)] - \mathsf{E}[\Psi_S(x_1^k, \cdots, x_{N-1}^k, x_N^k, \lambda^k, x_N^{k-1})] \\ \leq\; & \left[\frac{\beta + L}{2} - \frac{1}{\gamma} + \frac{8}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{4L^2}{\beta} + \frac{1}{2}\right] \mathsf{E}[\|x_N^{k+1} - x_N^k\|^2] \\ & - \sum_{i=1}^{N-1} \mathsf{E}\left[\|x_i^k - x_i^{k+1}\|^2_{\frac{1}{2}H_i - \frac{4L^2}{\beta}I - \frac{L+1}{2}I}\right] + \left(\frac{8}{\beta} + \frac{N}{2}\right)\frac{\sigma^2}{M}, \end{aligned} \tag{42}$$

and the coefficient in front of $\mathsf{E}[\|x_N^{k+1} - x_N^k\|^2]$ *is negative.*

Lemma 3.13 *Suppose the sequence* $\{x_1^k, \cdots, x_N^k, \lambda^k\}$ *is generated by Algorithm 3. It holds that*

$$\mathsf{E}[\Psi_S(x_1^{k+1}, \cdots, x_N^{k+1}, \lambda^{k+1}, x_N^k)] \;\geq\; \sum_{i=1}^{N-1} r_i^* + f^* - \frac{2\sigma^2}{\beta M} \geq \sum_{i=1}^{N-1} r_i^* + f^* - \frac{2\sigma^2}{\beta}. \tag{43}$$

We are now ready to present the iteration complexity result for Algorithm 3.

Theorem 3.14 *Suppose that the sequence* $\{x_1^k, ..., x_N^k, \lambda^k\}$ *is generated by Algorithm 3. Let the parameters* β *and* γ *satisfy* (40) *and* (41) *respectively. Define* $\kappa_1 := \frac{4}{\beta^2}\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right]$, $\kappa_2 := 3\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right]$, $\kappa_3 := 2\left(L + \beta\sqrt{N}\max_{1\leq i\leq N}\{\|A_i\|_2^2\} + \max_{1\leq i\leq N-1}\|H_i\|_2\right)^2$, $\kappa_4 = \frac{2}{\tau}\left(\frac{8}{\beta} + \frac{N}{2}\right)$ *with*
$\tau := \min\left\{-\left(\frac{\beta+L}{2} - \frac{1}{\gamma} + \frac{8}{\beta}\left[\beta - \frac{1}{\gamma}\right]^2 + \frac{4L^2}{\beta} + \frac{1}{2}\right), \min_{i=1,...,N-1}\left\{-\left(\frac{4L^2}{\beta} + \frac{L+1}{2} - \frac{\sigma_{\min}(H_i)}{2}\right)\right\}\right\}$. *Assume* $H_i \succ (\frac{8L^2}{\beta} + L + 1)I$ *and let*

$$M \geq \frac{2\sigma^2}{\epsilon^2}\max\{\kappa_1\kappa_4 + \frac{8}{\beta^2}, \kappa_2\kappa_4 + 3, \kappa_3\kappa_4 + 2\},$$

$$K = \left\lceil \frac{4\max\{\kappa_1, \kappa_2, \kappa_3\}}{\tau\epsilon^2}\left(\mathsf{E}[\Psi_G(x_1^1, ..., x_N^1, \lambda^1, x_N^0)] - \sum_{i=1}^{N-1} r_i^* - f^* + \frac{2\sigma^2}{\beta}\right)\right\rceil.$$

Let $k^* = \operatorname{argmin}_{2\leq k\leq K+1}\sum_{i=1}^N(\|x_i^k - x_i^{k+1}\|^2 + \|x_i^{k-1} - x_i^k\|^2)$, *then* $(x_1^{k^*+1}, \cdots, x_N^{k^*+1}, \lambda^{k^*+1})$*is an* ϵ*-stationary solution in accordance of Definition 2.10.*

Proof. Most parts of the proof are similar to that of Theorem 3.7, the only difference is that we need to carry the stochastic errors throughout the process. For simplicity, we shall highlight the key differences. First, we define θ_k according to (25) and then bound $\mathsf{E}[\theta_{k^*}]$ by

$$\begin{aligned}\mathsf{E}[\theta_{k^*}] \;&\leq\; \min_{k=2,...,K+1}\mathsf{E}[\theta_k] \qquad (44)\\ &\leq\; \frac{2}{\tau K}\left(\mathsf{E}[\Psi_S(x_1^1, ..., x_N^1, \lambda^1, x_N^0)] - \sum_{i=1}^{N-1} r_i^* - f^* + \frac{2\sigma^2}{\beta}\right) + \kappa_4\frac{\sigma^2}{M}.\end{aligned}$$

Second, we have

$$\mathsf{E}\left[\|\lambda^{k+1} - \nabla_N f(x_1^{k+1}, ..., x_N^{k+1})\|^2\right] \leq \kappa_2\mathsf{E}[\theta_k] + \frac{3\sigma^2}{M}, \tag{45}$$

$$\mathsf{E}\left[\|\sum_{i=1}^{N-1} A_i x_i^{k+1} + x_N^{k+1} - b\|^2\right] \leq \kappa_1\mathsf{E}[\theta_k] + \frac{8\sigma^2}{\beta^2 M}, \tag{46}$$

and

$$\mathsf{E}\left[\operatorname{dist}\left(\operatorname{Proj}_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\left(-\nabla_i f(x^{k+1})+A_i^\top\lambda^{k+1}-\partial r_i(x_i^{k+1})\right),\mathcal{N}_{X_i}(x_i^{k+1})\right)^2\right]\leq\kappa_3\mathsf{E}[\theta_k]+\frac{2\sigma^2}{M}. \quad (47)$$

Finally, apply Jensen's inequality $\mathsf{E}_\xi[\sqrt{\xi}]\leq\sqrt{\mathsf{E}_\xi[\xi]}$ to the above bounds (44), (45) and (46), and choose K as defined, the ϵ-stationary solution defined in (2.10) holds in expectation. □

3.4 A feasible curvilinear line-search variant of ADMM

We remark that the efficacy of the previous algorithms rely on the solvability of the subproblems at Step 1. Though the subproblems may be easy computable as we shall see from application examples in Section 5, there are also examples where such solutions are not available for many manifolds even when the objective is linearized. As a remedy we present in this subsection a feasible curvilinear line-search based variant of the ADMM. First let us make a few additional assumptions.

Assumption 3.15 *In problem* (1)*, the manifolds* $\mathcal{M}_i, i=1,\ldots,N-1$ *are compact. The nonsmooth regularizing functions* $r_i(x_i)$ *vanish, and the constraint sets* $X_i=\mathbb{R}^{n_i}$*, for* $i=1,...,N-1$*.*

Accordingly, the third part of the optimality condition (14) is simplified to

$$\operatorname{Proj}_{\mathcal{T}_{x_i^*}\mathcal{M}_i}\left(\nabla_i f(x^*)-A_i^\top\lambda^*\right)=0, i=1,...,N-1. \quad (48)$$

Let $R_i(\bar{x}_i,tg)$ be a retraction operator at point $\bar{x}_i\in\mathcal{M}_i$ in direction $g\in\mathcal{T}_{\bar{x}_i}\mathcal{M}_i$. Then a parameterized curve $Y_i(t)=R_i(\bar{x}_i,tg)$ is defined on $\mathcal{M}_i$. In particular, it satisfies

$$Y_i(0)=\bar{x}_i \text{ and } Y_i'(0)=g. \quad (49)$$

Proposition 3.16 *For retractions* $Y_i(t)=R_i(\bar{x}_i,tg), i=1,...,N-1$*, there exist* $L_1,L_2>0$ *such that*

$$\begin{aligned}\|Y_i(t)-Y_i(0)\| &\leq L_1 t\|Y_i'(0)\|, &(50)\\ \|Y_i(t)-Y_i(0)-tY_i'(0)\| &\leq L_2 t^2\|Y_i'(0)\|^2. &(51)\end{aligned}$$

The above proposition states that the retraction curve is approximately close to a line in Euclidean space. It was proved as a byproduct of Lemma 3 in [5] and was also adopted by [28]. Let the augmented Lagrangian function be defined by (16) (without the $r_i(x_i)$ terms) and denote

$$\operatorname{grad}_{x_i}\mathcal{L}_\beta(x_1,\cdots,x_N,\lambda)=\operatorname{Proj}_{\mathcal{T}_{x_i}\mathcal{M}_i}\left\{\nabla_i\mathcal{L}_\beta(x_1,\cdots,x_N,\lambda)\right\}$$

as the Riemannian partial gradient. We present the algorithm as in Algorithm 4.

Algorithm 4: A feasible curvilinear line-search-based ADMM

Given $(x_1^0, \cdots, x_{N-1}^0, x_N^0) \in \mathcal{M}_1 \times \cdots \times \mathcal{M}_{N-1} \times \mathbb{R}^{n_N}$, $\lambda^0 \in \mathbb{R}^m$, $\beta, \gamma, \sigma > 0, s > 0$, $\alpha \in (0,1)$.

for $k = 0, 1, ...$ **do**

  [Step 1] **for** $i = 1, 2, ..., N-1$ **do**

    Compute $g_i^k = \operatorname{grad}_{x_i} \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i^k, \cdots, x_N^k, \lambda^k)$;

    Initialize with $t_i^k = s$. While

$$\begin{aligned} & \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{i-1}^{k+1}, R_i(x_i^k, -t_i^k g_i^k), x_{i+1}^k, \cdots, x_N^k, \lambda^k) \\ > \; & \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i^k, \cdots, x_N^k, \lambda^k) - \frac{\sigma}{2}(t_i^k)^2 \|g_i^k\|^2, \end{aligned}$$

    shrink t_i^k by $t_i^k \leftarrow \alpha t_i^k$;

    Set $x_i^{k+1} = R_i(x_i^k, -t_i^k g_i^k)$;

  [Step 2] $x_N^{k+1} := x_N^k - \gamma \nabla_N \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k)$;

  [Step 3] $\lambda^{k+1} := \lambda^k - \beta(\sum_{i=1}^N A_i x_i^{k+1} - b)$.

For Steps 2 and 3, Lemma 3.4 and Lemma 3.6 still hold. Further using Proposition 3.16, Lemma 3.4 becomes

Lemma 3.17 *Suppose that the sequence $\{x_1^k, ..., x_N^k, \lambda^k\}$ is generated by Algorithm 4. Then,*

$$\begin{aligned} \|\lambda^{k+1} - \lambda^k\|^2 \; \leq \; & 3\left(\beta - \frac{1}{\gamma}\right)^2 \|t_N^k g_N^k\|^2 + 3\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right] \|t_N^{k-1} g_N^{k-1}\|^2 \\ & + 3L^2 L_1^2 \sum_{i=1}^{N-1} \|t_i^k g_i^k\|^2, \end{aligned} \tag{52}$$

where we define $t_N^k = \gamma$ and $x_N^{k+1} = x_N^k + t_N^k g_N^k, \forall k \geq 0$, for simplicity. Moreover, for the definition of Ψ_G in (19), Lemma 3.5 remains true, whereas the amount of decrease becomes

$$\begin{aligned} & \Psi_G(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^{k+1}, \lambda^{k+1}, x_N^k) - \Psi_G(x_1^k, \cdots, x_{N-1}^k, x_N^k, \lambda^k, x_N^{k-1}) \\ \leq \; & \left[\frac{\beta + L}{2} - \frac{1}{\gamma} + \frac{6}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{3L^2}{\beta}\right] \|t_N^k g_N^k\|^2 - \sum_{i=1}^{N-1} \left(\frac{\sigma}{2} - \frac{3}{\beta} L^2 L_1^2\right) \|t_i^k g_i^k\|^2 < 0. \end{aligned} \tag{53}$$

Now we are in a position to present the iteration complexity result, where the detailed proof can be found in the appendix.

Theorem 3.18 *Suppose that the sequence $\{x_1^k, ..., x_N^k, \lambda^k\}$ is generated by Algorithm 4, and the parameters β and γ satisfy* (20) *and* (21) *respectively. Denote $A_{\max} = \max_{1 \leq j \leq N} \|A_j\|_2$. Define $\tau := \min\left\{-\left[\frac{\beta+L}{2} - \frac{1}{\gamma} + \frac{6}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{3L^2}{\beta}\right], \frac{\sigma}{2} - \frac{3}{\beta} L^2 L_1^2\right\}$, $\kappa_1 := \frac{3}{\beta^2}\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2 \cdot \max\{L_1^2, 1\}\right]$,*

$\kappa_2 := \left(|\beta - \frac{1}{\gamma}| + L\right)^2$, $\kappa_3 := \left((L + \sqrt{N}\beta A_{\max}^2) \cdot \max\{L_1, 1\} + \frac{\sigma + 2L_2 C + (L+\beta A_{\max}^2)L_1^2}{2\alpha} + \beta A_{\max}\sqrt{\kappa_1}\right)^2$, *where* $C > 0$ *is a constant that depends only on the first iterate and the initial point. Assume* $\sigma > \max\{\frac{6}{\beta}L^2 L_1^2, \frac{2\alpha}{s}\}$. *Define*

$$K := \left\lceil \frac{3\max\{\kappa_1, \kappa_2, \kappa_3\}}{\tau\epsilon^2} \left(\Psi_G(x_1^1, ..., x_N^1, \lambda^1, x_N^0) - f^*\right) \right\rceil, \tag{54}$$

and $k^* := \operatorname{argmin}_{2\le k\le K+1} \sum_{i=1}^N (\|t_i^{k+1} g_i^{k+1}\|^2 + \|t_i^k g_i^k\|^2 + \|t_i^{k-1} g_i^{k-1}\|^2)$. *Then* $(x_1^{k^*+1}, \cdots, x_N^{k^*+1}, \lambda^{k^*+1})$ *is an* ϵ*-stationary solution of* (1).

4 Extending the Basic Model

Recall that for our basic model (1), a number of assumptions have been made; e.g. we assumed that $r_i, i = 1, ..., N-1$ are convex, x_N is unconstrained and $A_N = I$. In this section we shall extend the model to relax these assumptions. We shall also extend our basic algorithmic model from the Gauss-Seidel updating style to allow the Jacobi style updating, to enable parallelization.

4.1 Relaxing the convexity requirement on nonsmooth regularizers

For problem (1) the nonsmooth part r_i are actually not necessarily convex. As an example, nonconvex and nonsmooth regularizations such as ℓ_q regularization with $0 < q < 1$ are very common in compressive sensing. To accommodate the change, the following adaptation is needed.

Proposition 4.1 *For problem* (1)*, where* f *is smooth with Lipchitz continuous gradient. Suppose that* $\mathcal{I}_1, \mathcal{I}_2$ *form a partition of the index set* $\{1, ..., N-1\}$*, in such a way that for* $i \in \mathcal{I}_1$*,* r_i*'s are nonsmooth but convex, and for* $i \in \mathcal{I}_2$*,* r_i*'s are nonsmooth and nonconvex but are locally Lipschitz continuous. If for blocks* $x_i, i \in \mathcal{I}_2$ *there are no manifold constraints, i.e.* $\mathcal{M}_i = \mathbb{R}^{n_i}, i \in \mathcal{I}_2$*, then Theorems 3.7, 3.9 and 3.14 remain true.*

Recall that in the proofs for (30) and (31), we required the convexity of r_i to ensure (8). However, if $\mathcal{M}_i = \mathbb{R}^{n_i}$, then we directly have (7), i.e., $\partial_i(f + r_i) = \nabla_i f + \partial r_i$ instead of (8). The only difference is that ∂r_i becomes the Clarke generalized subdifferential instead of the convex subgradient and the projection operator is no longer needed. In the subsequent complexity analysis, we just need to remove all the projection operators in (31) and (47). Hence the same convergence result follows.

Moreover, if for some blocks, r_i's are nonsmooth and nonconvex, while the constraint $x_i \in \mathcal{M}_i \neq \mathbb{R}^{n_i}$

is still imposed, then we can solve the problem via the following equivalent formulation:

$$\begin{aligned} \min \quad & f(x_1, ..., x_N) + \sum_{i \in \mathcal{I}_1 \cup \mathcal{I}_2} r_i(x_i) + \sum_{i \in \mathcal{I}_3} r_i(y_i) \\ \text{s.t.} \quad & \sum_{i=1}^{N} A_i x_i = b, \text{ with } A_N = I, \\ & x_N \in \mathbb{R}^{n_N}, \\ & x_i \in \mathcal{M}_i \cap X_i, \ i \in \mathcal{I}_1 \cup \mathcal{I}_3, \\ & x_i \in X_i, \ i \in \mathcal{I}_2, \\ & y_i = x_i, \ i \in \mathcal{I}_3, \end{aligned} \tag{55}$$

where $\mathcal{I}_1, \mathcal{I}_2$ and $\mathcal{I}_3$ form a partition for $\{1, ..., N-1\}$, with r_i convex for $i \in \mathcal{I}_1$ and nonconvex but locally Lipschitz continuous for $i \in \mathcal{I}_2 \cup \mathcal{I}_3$. The difference is that x_i is not required to satisfy Riemannian manifold constraint for $i \in \mathcal{I}_2$.

Unfortunately, the ℓ_q regularization itself is not locally Lipschitz at 0 and hence does not satisfy our requirement. But if we apply the modification of ℓ_q regularization in Remark 5.2, then we can circumvent this difficulty while making almost no change to the solution process and keeping closed form solutions. In fact, due to the limited machine precision of computer, we can directly use ℓ_q regularization and treat it as if we were working with the modified ℓ_q regularization.

4.2 Relaxing the condition on the last block variables

In the previous discussion, we limit our problem to the case where $A_N = I$ and x_N is unconstrained. Actually, for the general case

$$\begin{aligned} \min \quad & f(x_1, \cdots, x_N) + \sum_{i=1}^{N} r_i(x_i) \\ \text{s.t.} \quad & \sum_{i=1}^{N} A_i x_i = b, \\ & x_i \in \mathcal{M}_i \cap X_i, \ i = 1, ..., N, \end{aligned} \tag{56}$$

where x_N is as normal as other blocks, we can actually add an additional block x_{N+1} and modify the objective a little bit and arrive at the modified problem

$$\begin{aligned} \min \quad & f(x_1, \cdots, x_N, x_{N+1}) + \sum_{i=1}^{N} r_i(x_i) + \frac{\mu}{2}\|x_{N+1}\|^2 \\ \text{s.t.} \quad & \sum_{i=1}^{N} A_i x_i + x_{N+1} = b, \\ & x_{N+1} \in \mathbb{R}^m, \\ & x_i \in \mathcal{M}_i \cap X_i, \ i = 1, ..., N. \end{aligned} \tag{57}$$

Following a similar line of proofs of Theorem 4.1 in [27], we have the following proposition.

Proposition 4.2 *Consider the modified problem* (57) *with* $\mu = 1/\epsilon$ *for some given tolerance* $\epsilon \in (0,1)$ *and suppose the sequence* $\{(x_1^k, ..., x_{N+1}^k, \lambda^k)\}$ *is generated by Algorithm 1 (resp. Algorithm 2). Let* $(x_1^{k*+1}, ..., x_N^{k*+1}, \lambda^{k*+1})$ *be* ϵ*-stationary solution of* (57) *as defined in Theorem 3.7 (resp. Theorem 3.9). Then* $(x_1^{k*+1}, ..., x_N^{k*+1}, \lambda^{k*+1})$ *is an* ϵ*-stationary point of the original problem* (56).

Remark 4.3 *We remark here that when* $\mu = 1/\epsilon$*, the Lipschitz constant of the objective function* L *also depends on* ϵ*. As a result, the iteration complexity of Algorithms 1 and 2 becomes* $O(1/\epsilon^4)$.

4.3 The Jacobi-style updating rule

Parallel to (32), we define a new linearized approximation of the augmented Lagrangian as

$$\begin{aligned}\bar{\mathcal{L}}_\beta^i(x_i; \hat{x}_1, \cdots, \hat{x}_N, \lambda) &= \bar{f}_\beta(\hat{x}_1, \cdots, \hat{x}_N) + \langle \nabla_i \bar{f}_\beta(\hat{x}_1, \cdots, \hat{x}_N), x_i - \hat{x}_i \rangle \\ &\quad - \left\langle \sum_{j \neq i}^N A_j \hat{x}_j + A_i x_i - b, \lambda \right\rangle + r_i(x_i), \end{aligned} \tag{58}$$

where

$$\bar{f}_\beta(x_1, \cdots, x_N) = f(x_1, \cdots, x_N) + \frac{\beta}{2} \left\| \sum_{j=1}^N A_j x_j - b \right\|^2.$$

Compared with (32), in this case we linearize both the coupling smooth objective function and the augmented term.

In Step 1 of Algorithm 2, we have the Gauss-Seidel style updating rule,

$$x_i^{k+1} = \underset{x_i \in \mathcal{M}_i \cap X_i}{\operatorname{argmin}} \hat{\mathcal{L}}_\beta^i(x_i; x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i^k, \cdots, x_N^k, \lambda^k) + \frac{1}{2} \|x_i - x_i^k\|_{H_i}^2.$$

Now if we replace this with the Jacobi style updating rule,

$$x_i^{k+1} = \underset{x_i \in \mathcal{M}_i \cap X_i}{\operatorname{argmin}} \bar{\mathcal{L}}_\beta^i(x_i; x_1^k, \cdots, x_{i-1}^k, x_i^k, \cdots, x_N^k, \lambda^k) + \frac{1}{2} \|x_i - x_i^k\|_{H_i}^2, \tag{59}$$

then we end up with a new algorithm which updates all blocks parallelly instead of sequentially. When the number of blocks, namely N, is large, using the Jacobi updating rule can be beneficial because the computation can be parallelized.

To establish the convergence of this process, all we need is to establish a counterpart of (78) in this new setting, namely

$$\mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k) \leq \mathcal{L}_\beta(x_1^k, \cdots, x_N^k, \lambda^k) - \sum_{i=1}^{N-1} \|x_i^k - x_i^{k+1}\|_{\frac{H_i}{2} - \frac{\hat{L}}{2} I}^2, \tag{60}$$

for some $\hat{L} > 0$. Consequently, if we choose $H_i \succ \hat{L}I$, then the convergence and complexity analysis goes through for Algorithm 2. Moreover, Algorithm 3 can also be adapted to the Jacobi-style updates. The proof for (60) is given in the appendix.

5 Some Applications and Their Implementations

The applications of block optimization with manifold constraints are abundant. In this section we shall present some typical examples. Our choices include the NP-hard maximum bisection problem, the sparse multilinear principal component analysis, and the community detection problem.

5.1 Maximum bisection problem

The maximum bisection problem is a variant of the well known NP-hard maximum cut problem. Suppose we have a graph $G = (V, E)$ where $V = \{1, ..., n\} := [n]$ denotes the set of nodes and E denotes the set of edges, each edge $e_{ij} \in E$ is assigned with a weight $W_{ij} \geq 0$. For pair $(i, j) \notin E$, define $W_{ij} = 0$. Let a bisection $\{V_1, V_2\}$ of V be defined as

$$V_1 \cup V_2 = V, \quad V_1 \cap V_2 = \emptyset, \quad |V_1| = |V_2|.$$

The maximum bisection problem is to find the best bisection that maximize the graph cut value:

$$\begin{array}{ll} \max_{V_1, V_2} & \sum_{i \in V_1} \sum_{j \in V_2} W_{ij} \\ \text{s.t.} & V_1, V_2 \text{ is a bisection of } V. \end{array}$$

Note that if we relax the constraint $|V_1| = |V_2|$, that is, we only require $\{V_1, V_2\}$ to be a partition of V, then this problem becomes the *maximum cut* problem. In this paper, we propose to solve this problem by our method and compare our results with the two SDP relaxations proposed in [14, 60].

First, we model the bisection $\{V_1, V_2\}$ by a binary assignment matrix $U \in \{0, 1\}^{n \times 2}$. Each node i is represented by the ith row of matrix U. Denote this row by $u_i^\top$, where $u_i \in \{0, 1\}^{2 \times 1}$ is a column vector with exactly one entry equal to 1. Then $u_i^\top = (1, 0)$ or $(0, 1)$ corresponds to $i \in V_1$ or $i \in V_2$ respectively, and the objective can be represented by

$$\sum_{i \in V_1} \sum_{j \in V_2} W_{ij} = \sum_{i,j} (1 - \langle u_i, u_j \rangle) W_{i,j} = -\langle W, UU^\top \rangle + const.$$

The constraint that $|V_1| = |V_2|$ is characterized by the linear equality constraint

$$\sum_{i=1}^{n} (u_i)_1 - \sum_{i=1}^{n} (u_i)_2 = 0.$$

Consequently, we can develop the nonconvex relaxation of the maximum bisection problem as

$$\begin{array}{ll} \min_U & \langle W, UU^\top \rangle \\ \text{s.t.} & \|u_i\|^2 = 1, u_i \geq 0, \text{ for } i = 1, ..., n, \\ & \sum_{i=1}^n (u_i)_1 - \sum_{i=1}^n (u_i)_2 = 0. \end{array} \tag{61}$$

After the relaxation is solved, each row is first rounded to an integer solution

$$u_i \leftarrow \begin{cases} (1,0)^\top, & \text{if } (u_i)_1 \geq (u_i)_2, \\ (0,1)^\top, & \text{otherwise.} \end{cases}$$

Then a greedy algorithm is applied to adjust current solution to a feasible bisection solution. Note that this greedy step is necessary for our algorithm and the SDP relaxations in [14, 60] to reach a feasible bisection.

The ADMM formulation of this problem will be shown in the numerical experiment part and the algorithm realization is omitted. Here we only need to mention that all the subproblems are of the following form:

$$\begin{array}{ll} \min_x & b^\top x \\ \text{s.t.} & \|x\|^2 = 1, x \geq 0. \end{array} \tag{62}$$

This nonconvex constrained problem can actually be solved to global optimality in closed form, see the Lemma 1 in [63]. For the sake of completeness, we present the lemma bellow.

Lemma 5.1 *(Lemma 1 in [63].) Define* $b^+ = \max\{b, 0\}$, $b^- = -\min\{b, 0\}$, *where* max *and* min *are taken element-wise. Note that* $b^+ \geq 0$, $b^- \geq 0$, *and* $b = b^+ - b^-$. *The closed form solution for problem* (62) *is*

$$x^* = \begin{cases} \frac{b^-}{\|b^-\|}, & \text{if } b^- \neq 0 \\ e_i, & \text{otherwise,} \end{cases} \tag{63}$$

where e_i *is the* i*-th unit vector with* $i = \operatorname{argmin}_j b_j$.

5.2 The ℓ_q-regularized sparse tensor PCA

As we discussed at the beginning of Section 1, the tensor principal component analysis (or multilinear principal component analysis (MPCA)) has been a popular subject of study in recent years. Below, we shall discuss a sparse version of this problem.

Suppose that we are given a collection of order-d tensors $\mathbf{T}^{(1)}, \mathbf{T}^{(2)}, ..., \mathbf{T}^{(N)} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$. The sparse MPCA problem can be formulated as (see also [57]):

$$\begin{array}{ll} \min & \sum_{i=1}^{N} \|\mathbf{T}^{(i)} - \mathbf{C}^{(i)} \times_1 U_1 \times \cdots \times_d U_d\|_F^2 + \alpha_1 \sum_{i=1}^{N} \|\mathbf{C}^{(i)}\|_p^p + \alpha_2 \sum_{j=1}^{d} \|U_j\|_q^q \\ \text{s.t.} & \mathbf{C}^{(i)} \in \mathbb{R}^{m_1 \times \cdots \times m_d}, i = 1, ..., N \\ & U_j \in \mathbb{R}^{n_j \times m_j}, U_j^\top U_j = I, j = 1, ..., d. \end{array}$$

In order to apply our developed algorithms, we can consider the following variant of sparse MPCA:

$$\begin{array}{ll} \min & \sum_{i=1}^{N} \|\mathbf{T}^{(i)} - \mathbf{C}^{(i)} \times_1 U_1 \times \cdots \times_d U_d\|_F^2 + \alpha_1 \sum_{i=1}^{N} \|\mathbf{C}^{(i)}\|_p^p + \alpha_2 \sum_{j=1}^{d} \|V_j\|_q^q + \frac{\mu}{2} \sum_{j=1}^{d} \|Y_j\|^2 \\ \text{s.t.} & \mathbf{C}^{(i)} \in \mathbb{R}^{m_1 \times \cdots \times m_d}, i = 1, ..., N \\ & U_j \in \mathbb{R}^{n_j \times m_j}, U_j^\top U_j = I, j = 1, ..., d \\ & V_j - U_j + Y_j = 0, j = 1, ..., d. \end{array} \tag{64}$$

Note that this model is different from the ones used in [34, 53].

Denote $\mathbf{T}_{(j)}^{(i)}$ to be the mode-j unfolding of a tensor $\mathbf{T}^{(i)}$, and denote $\mathbf{C}$ to be the set of all tensors $\{\mathbf{C}^{(i)} : i = 1, ..., N\}$. The augmented Lagrangian function of (64) is

$$\begin{aligned} L_\beta(\mathbf{C}, U, V, Y, \Lambda) \;=\; & \sum_{i=1}^{N} \|\mathbf{T}^{(i)} - \mathbf{C}^{(i)} \times_1 U_1 \times \cdots \times_d U_d\|_F^2 + \alpha_1 \sum_{i=1}^{N} \|\mathbf{C}^{(i)}\|_p^p + \alpha_2 \sum_{j=1}^{d} \|V_j\|_q^q \\ & + \frac{\mu}{2} \sum_{j=1}^{d} \|Y_j\|^2 - \sum_{j=1}^{d} \langle U_j - V_j + Y_j, \Lambda_j \rangle + \frac{\beta}{2} \sum_{j=1}^{d} \|U_j - V_j + Y_j\|_F^2. \end{aligned}$$

An implementation of the Algorithm 1 for solving (64) is shown in Algorithm 5.

In Step 1 of Algorithm 5, the subproblem to be solved is

$$U_j = \underset{U^\top U = I}{\operatorname{argmin}} -\langle 2B, U \rangle = \underset{U^\top U = I}{\operatorname{argmin}} \|B - U\|_F^2, \tag{65}$$

which is known as the nearest orthogonal matrix problem. Suppose we have the SVD decomposition of the matrix B as $B = Q \Sigma P^\top$, then the global optimal solution is $U_j = QP^\top$. When B has full column rank, the solution is also unique.

In Steps 2 and 3 of Algorithm 5, they are actually a group of one-dimensional decoupled problems. Since no nonnegative constraints are imposed, we can apply ℓ_1 regularization for which soft-thresholding gives closed form solution to the subproblems. However, if we want to apply ℓ_q refularization for $0 < q < 1$, then the subproblem amounts to solve

$$\min f(x) = ax^2 + bx + c|x|^q, \tag{66}$$

Algorithm 5: A typical iteration of Algorithm 1 for solving (64)

[Step 1] **for** $j = 1, ..., d$ **do**
  Set $B = \sum_{i=1}^{N} \mathbf{T}^{(i)}_{(j)}(U_d \otimes \cdots \otimes U_{j+1} \otimes U_{j-1} \otimes \cdots \otimes U_1)(\mathbf{C}^{(i)}_{(j)})^\top + \frac{1}{2}\Lambda_j - \frac{\beta}{2}Y_j + \frac{\beta}{2}V_j + \frac{\sigma}{2}U_j$
  $U_j \leftarrow \mathrm{argmin}_{U^\top U = I} -\langle 2B, U\rangle$
[Step 2] **for** $j = 1, ..., d$ **do**
  For each component $V_j(s)$ where $s = (s_1, s_2)$ is a multilinear index,
  set $b = \beta Y_j(s) + \beta U_j(s) - \Lambda_j(s) + \sigma V_j(s)$.
  $V_j(s) = \mathrm{argmin}_x \frac{\beta+\sigma}{2}x^2 + \alpha_2|x|^q - bx$
[Step 3] **for** $i = 1, ..., N$ **do**
  For each component $\mathbf{C}^{(i)}(s)$, where $s = (s_1, ..., s_d)$ is a multilinear index,
  set $b = \sigma\mathbf{C}^{(i)}(s) - 2\left[(U_d^\top \otimes \cdots \otimes U_1^\top)\mathrm{vec}(\mathbf{T}^{(i)})\right](s)$.
  $\mathbf{C}^{(i)}(s) \leftarrow \mathrm{argmin}_x \frac{2+\sigma}{2}x^2 + \alpha_1|x|^q - bx$
[Step 4] **for** $j = 1, ..., d$ **do**
  $Y_j \leftarrow Y_j - \eta\left[(\beta + \mu)Y_j - \beta U_j - \beta V_j - \Lambda_j\right]$
[Step 5] **for** $j = 1, ..., d$ **do**
  $\Lambda_j \leftarrow \Lambda_j - \beta\left(U_j - V_j + Y_j\right)$

where $0 < q < 1$, $a > 0$, $c > 0$. The function is nonconvex and nonsmooth at 0 with $f(0) = 0$. For $x > 0$, we can take the derivative and set it to 0, and obtain $2ax + qcx^{q-1} + b = 0$, or equivalently

$$2ax^{2-q} + bx^{1-q} + cq = 0.$$

If $q = \frac{1}{2}$, then setting $z = \sqrt{x}$ leads to $2az^3 + bz + cq = 0$. If $q = \frac{2}{3}$, then setting $z = x^{\frac{1}{3}}$ leads to $2az^4 + bz + cq = 0$. In both cases, we have closed-form solutions. Similarly, we apply this trick to the case when $x < 0$. Suppose we find the roots $x_1, ..., x_k$ and we set $x_0 = 0$, then the solution to (66) is x_{i^*} with $i^* = \mathrm{argmin}_{0 \le j \le k} f(x_j)$.

Remark 5.2 *The ℓ_q regularization is not locally Lipschitz at 0 when $0 < q < 1$, which might cause problems. However, if we replace $\|x\|^q$ with $\min\{|x|^q, B|x|\}, B \gg 0$, then the new regularization is locally Lipschitz on $\mathbb{R}$, and it differs from the original function only on $(-\frac{1}{B^{1-q}}, +\frac{1}{B^{1-q}})$. The closed-form solution can still be obtained by comparing the objective values at $x_1^* = \mathrm{argmin}_x\, ax^2 + bx + c|x|^q$ and $x_2^* = \mathrm{argmin}_x\, ax^2 + bx + cB|x| = \left(\frac{-cB-b}{2a}\right)_+$. Actually due to the limited machine precision, the window $(-\frac{1}{B^{1-q}}, +\frac{1}{B^{1-q}})$ shrinks to a single point 0 when B is sufficiently large. Since this causes no numerical difficulties, we can just deal with ℓ_q penalties by replacing it by the modified version.*

5.3 The community detection problem

Given any undirected network, the community detection problem aims to figure out the clusters, in other words the communities, of this network; see for example [8, 29, 63, 64], etc. A viable way

to solve this problem is via the symmetric othorgonal nonnegative matrix approximation. Suppose the adjacency matrix of the network is A , then the method aims to solve

$$\min_{X\in\mathbb{R}^{n\times k}} \|A - XX^\top\|_F^2, \text{ s.t. } X^\top X = I_{k\times k},\ X \geq 0, \tag{67}$$

where n equals the number of nodes and k equals the number of communities. When the network is connected, the orthogonality and nonnegativeness of the optimal solution X^* indicate that there is exactly one positive entry in each row of X^*. Therefore we can reconstruct the community structure by letting node i belong to community j if $X^*_{ij} > 0$.

In our framework, this problem can be naturally formulated as

$$\begin{array}{ll} \min_{X,Y,Z\in\mathbb{R}^{n\times k}} & \|A - XX^\top\|_F^2 + \dfrac{\mu}{2}\|Z\|_F^2 \\ \text{s.t.} & X^\top X = I_{k\times k},\ Y \geq 0, \\ & X - Y + Z = 0, \end{array} \tag{68}$$

where the orthogonal X is forced to be equal to the nonnegative Y, while a slack variable Z is added so that they do not need to be exactly equal. In the implementation of the Algorithm 2, two subproblems for block X and Y need to be solved. For the orthogonal block X, the subproblem is still in the form of (65). For the nonnegative block Y, the subproblem can be formulated as:

$$Y^* = \arg\min_{Y\geq 0} \|Y - B\|_F^2 = B_+, \tag{69}$$

for some matrix B. The notation B_+ is defined by $B_+ = \max\{B, 0\}$, where the max is taken elementwise.

6 Numerical Results

6.1 The maximum bisection problem

We consider the following variant of maximum bisection problem to apply our proposed algorithm.

$$\begin{array}{ll} \min_{U,z,x} & \langle W, UU^\top\rangle + \frac{\mu}{2}\|z\|^2 \\ \text{s.t.} & \|u_i\|^2 = 1, u_i \geq 0, \text{ for } i = 1, ..., n, \\ & \sum_{i=1}^n u_i - x\mathbf{1} + z = 0, \\ & z \in \mathbb{R}^2 \text{ is free, } \frac{n}{2} - \nu \leq x \leq \frac{n}{2} + \nu, \end{array}$$

where $\nu \geq 0$ is a parameter that controls the tightness of the relaxation. In our experiments, we set $\nu = 1$. We choose five graphs from the maximum cut library *Biq Mac Library* [56] to test our algorithm, with the following specifics in Table 6.1.

For the three tested algorithms, we denote the SDP relaxation proposed by Frieze et al. in [14] as SDP-F, we denote the SDP relaxation proposed by Ye in [60] as SDP-Y, and we denote our low-rank relaxation as LR. The SDP relaxations are solved by the interior point method embedded in

Graph Information					
Network	g05_60.0	g05_80.0	g05_100.0	pw01_100.0	pw09_100.0
# nodes	60	80	100	100	100
# edges	885	1580	2475	495	4455

Table 6.1: The test graph information.

CVX [21]. To solve the problem by our proposed Algorithm 1, we set $\mu = 0.01$. Other parameters such as $\beta, \gamma, H_i = \sigma I$ are chosen according to our theories for given estimation of the Lipschitz constant L. For all cases, the number of iterations is set to 30. For each graph, all algorithms are tested for 20 times and then we compare their average cut values. The results are reported in Table 6.2.

Network	avg LR cut	SD	avg SDP-Y cut	ratio_1	avg SDP-F cut	ratio_2
g05_60.0	1051.3	15.9773	1033.2	1.0175	1045.4	1.0056
g05_80.0	1822.7	15.3180	1778.5	1.0249	1805.9	1.0093
g05_100.0	2810.2	19.4413	2775.7	1.0124	2799.8	1.0037
pw01_100.0	3946.8	28.5032	3889.7	1.0147	3944.3	1.0006
pw09_100.0	26863.2	102.1318	26609	1.0096	26764.1	1.0037

Table 6.2: The column *SD* contains the standard deviations of the LR cut values in 20 rounds. $\text{ratio}_1 = \frac{\text{avg LR cut}}{\text{avg SDP-Y cut}}$, and $\text{ratio}_2 = \frac{\text{avg LR cut}}{\text{avg SDP-F cut}}$.

It is interesting to see that in all tested cases, our proposed relaxation solved by Algorithm 1 outperforms the two SDP relaxations in [14, 60]. Moreover, our method is a first-order method, and it naturally enjoys computational advantages compared to the interior-point based methods for solving the SDP relaxation.

Finally, in this application we test the performance of Algorithm 2 by comparing it to Algorithm 1. We keep the parameters μ, β, γ, ν unchanged for testing Algorithm 2, but we reset $H_i = \sigma I$ according to its new bound in Theorem 3.9. For each graph, 20 instances are tested, and 30 iterations are performed for each algorithm. The objective measured is $\langle W, UU^\top \rangle$. The result is shown in Table 6.3. It can be observed that in this case, Algorithm 2 behaves similarly as Algorithm 1.

6.2 The ℓ_q regularized sparse tensor PCA

In this experiment, we synthesize a set of ground truth Tucker format tensors $\mathbf{T}_{true}^{(i)} = \mathbf{C}^{(i)} \times_1 U_1 \times_2 \cdots \times_d U_d$, where all $\mathbf{T}_{true}^{(i)}$'s share the same factors U_j while having different cores $\mathbf{C}^{(i)}$. We test our methods by two cases, the first set of tensors have mode sizes $30 \times 30 \times 30$ and core mode sizes $5 \times 5 \times 5$. The second set of tensors have mode sizes $42 \times 42 \times 42$ and core mode sizes $7 \times 7 \times 7$. For both cases, we generate 100 instances. We associate a componentwise Gaussian

Network	Algorithm 1		Algorithm 2	
	avg obj	SD	avg obj	SD
g05_60.0	724.2	13.4070	719.7	12.3164
g05_80.0	1335	9.6791	1340.7	18.8766
g05_100.0	2136.1	24.6446	2135.5	18.8275
pw01_100.0	1558.8	78.0591	1563.7	76.5748
pw09_100.0	22262.8	100.1208	22371.3	119.8688

Table 6.3: Numerical performance of Algorithm 2 for problem (61).

white noise $\mathbf{T}^{(i)}_{noise}$ with standard deviation 0.001 to each tensor. Namely, the input data are $\mathbf{T}^{(i)} = \mathbf{T}^{(i)}_{true} + \mathbf{T}^{(i)}_{noise}$, $i = 1, ..., 100$. For all cases, the core elements are generated by uniform distribution in $[-1, 1]$. The sparsity level of each core $\mathbf{C}^{(i)}$ is set to 0.3, i.e., we randomly set 70% of the elements to zero in each core. Finally, the orthogonal factors U_i are generated with sparsity level 1/6.

To solve (64), we set the regularization terms to $\ell_{2/3}$ penalties for cores and to ℓ_1 penalties for the factors. That is, $q = 2/3$ and $p = 1$ in (64). The sparse penalty parameters are set to $\alpha_1 = 0.1$ and $\alpha_2 = 0.01$. We set $\mu = 10^{-6}$, and other parameters $\beta, \gamma, H_i = \sigma I$ are chosen according to our theories for given estimation of the Lipschitz constant L.

Our numerical results show that it is indeed necessary to set different regularizations for cores and factors. In the output of the result, the matrices U_i's are definitely not sparse, but with plenty of entries very close to 0. The output V_i's are very sparse but are not orthogonal. We construct the final output from U_i by zeroing out all the entries with absolute value less than 0.001. Then the resulting matrices $\bar{U}_i$'s are sparse and are almost orthogonal. Finally, the relative error is measured using $\bar{U}_i$ and the underlying true tensor, i.e., $\frac{1}{100}\sum_{i=1}^{100} \frac{\|\mathbf{T}^{(i)}_{true} - \mathbf{T}^{(i)}_{out}\|^2}{\|\mathbf{T}^{(i)}_{true}\|^2}$, where $\mathbf{T}^{(i)}_{out}$'s are constructed from the output of the algorithms. The orthogonality violation is measured by $\frac{1}{3}\sum_{i=1}^{3} \|\bar{U}_i^\top \bar{U}_i - I\|_F$. In both cases, the iteration number is set to be 100. For each case, 10 instances are generated and we report the average performance in Table 6.4. The results are obtained from 20 randomly generated instances. The columns err_1, SD, err_2, $spars_1$, $spars_2$ denote the averaged objective relative errors, the standard deviation of the objective relative errors, the average orthogonality constraint violation, the average core sparse levels and the average factor sparse levels respectively.

$30 \times 30 \times 30$, core $5 \times 5 \times 5$					$42 \times 42 \times 42$, core $7 \times 7 \times 7$				
$avg\ err_1$	SD	err_2	$spars_1$	$spars_2$	err_1	SD	err_2	$spars_1$	$spars_2$
0.0043	0.0028	2.7×10^{-7}	0.5363	1/6	0.0803	0.0010	1.2×10^{-14}	0.5387	1/6

Table 6.4: Numerical performance of Algorithm 1 for problem (64).

6.3 The community detection problem

For this problem, we test our algorithm on three real world social networks with ground truth information. They are the American political blogs network with 1222 nodes and 2 communities specified by their political leaning, the Caltech facebook network with 597 nodes and 8 communities specified by their dorm number, and the Simmons College facebook network with 1168 nodes and 4 communities specified by their graduation years. Note that (68) is a very simple model, so we will not compare it the more sophisticated models such as [8, 63]. Instead it is compared with the state-of-the-art spectral methods SCORE [29] and OCCAM [64].

In all tests for the three networks, the parameter μ is set to be 50 and L is set to be 100. The other parameters $\beta, \gamma, H_i = \sigma I$ are chosen according to our theories for a given estimation of L. For each network, every algorithm is run for 20 times and the average error rate is reported in Table 6.5.

Network Name	Algorithm 2	SCORE	OCCAM
Polblogs	5.07%	**4.75%**	4.91%
Caltech	**23.68%**	28.66%	34.21%
Simmons	**20.61%**	22.54%	23.92%

Table 6.5: Numerical performance of Algorithm 2 for problem (68).

It can be observed from the numerical results that Algorithm 2 yields the best result in Caltech and Simmons College networks, and is only slightly outperformed in the political blogs network, which shows the effectiveness of our method for this problem.

7 Conclusions

In this paper we extend the framework studied in [27] and develop a proximal ADMM-like algorithm for nonsmooth and nonconvex multi-block optimization over Riemannian manifolds. It turns out that this model has a wide range of applications. The linearized and the stochastic as well as the curvilinear line-search-based variants of this algorithm are proposed to handle the situations where exact minimization is hard, or the function/gradient evaluation is expensive. For all the proposed algorithms, an $\mathcal{O}(1/\epsilon^2)$ iteration complexity is guaranteed. The numerical experiments show great potential of the proposed methods. It is worth noting that when the problem is not in the form of (1), then the reformulation proposed in Section 4 will in general lead to an increased iteration complexity.

References

[1] P. A. Absil, C. G. Baker, and K. A. Gallivan. Convergence analysis of Riemannian trust-region methods. *Technical Report*, 2006.

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

[3] P. A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds.* Princeton University Press, 2009.

[4] J. Ballani, L. Grasedyck, and M. Kluge. Black box approximation of tensors in hierarchical Tucker format. *Linear Algebra and its Applications*, 438(2):639–657, 2013.

[5] N. Boumal, P. A. Absil, and C. Cartis. Global rates of convergence for nonconvex optimization on manifolds. *arXiv preprint arXiv:1605.08101*, 2016.

[6] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):11, 2011.

[7] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159, 2001.

[8] Y. Chen, X. Li, and J. Xu. Convexified modularity maximization for degree-corrected stochastic block models. *arXiv preprint arXiv:1512.08425*, 2015.

[9] F. H. Clarke. Optimization and nonsmooth analysis. 5:847–853, 1983.

[10] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[11] I. S. Dhillon and S. Sra. Generalized nonnegative matrix approximations with Bregman divergences. In *NIPS*, volume 18, 2005.

[12] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

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

[14] A. Frieze and M. Jerrum. Improved approximation algorithms for maxk-cut and max bisection. *Algorithmica*, 18(1):67–81, 1997.

[15] W. J. Fu. Penalized regressions: the bridge versus the lasso. *Journal of Computational and Graphical Statistics*, 7(3):397–416, 1998.

[16] S. Ghadimi and G. Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[17] S. Ghadimi and G. Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156(1-2):59–99, 2016.

[18] S. Ghadimi, G. Lan, and H. Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1-2):267–305, 2016.

[19] S. Ghosh and H. Lam. Computing worst-case input models in stochastic simulation. *arXiv preprint arXiv:1507.05609*, 2015.

[20] S. Ghosh and H. Lam. Mirror descent stochastic approximation for computing worst-case stochastic input models. In *Winter Simulation Conference, 2015*, pages 425–436. IEEE, 2015.

[21] M. Grant, S. Boyd, and Y. Ye. Cvx: Matlab software for disciplined convex programming, 2008.

[22] M. Hong. Decomposing linearly constrained nonconvex problems by a proximal primal dual approach: Algorithms, convergence, and applications. *arXiv preprint arXiv:1604.00543*, 2016.

[23] M. Hong, Z.-Q. Luo, and M. Razaviyayn. Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems. *SIAM Journal on Optimization*, 26(1):337–364, 2016.

[24] S. Hosseini and M. R. Pouryayevali. Generalized gradients and characterization of epi-Lipschitz sets in Riemannian manifolds. *Fuel and Energy Abstracts*, 74(12):3884–3895, 2011.

[25] K. Huper and J. Trumpf. Newton-like methods for numerical optimization on manifolds. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference*, volume 1, pages 136–139. IEEE, 2004.

[26] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing*, pages 665–674. ACM, 2013.

[27] B. Jiang, T. Lin, S. Ma, and S. Zhang. Structured nonconvex and nonsmooth optimization: Algorithms and iteration complexity analysis. *arXiv preprint arXiv:1605.02408*, 2016.

[28] B. Jiang, S. Ma, A. M.-C. So, and S. Zhang. Vector transport-free SVRG with general retraction for Riemannian optimization: Complexity analysis and practical implementation. *Preprint available at https://arxiv.org/abs/1705.09059*, 2017.

[29] J. Jin. Fast community detection by score. *The Annals of Statistics*, 43(1):57–89, 2015.

[30] H. Kasai, H. Sato, and B. Mishra. Riemannian stochastic variance reduced gradient on Grassmann manifold. *arXiv preprint arXiv:1605.07367*, 2016.

[31] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[32] A. Kovnatsky, K. Glashoff, and M. Bronstein. Madmm: a generic algorithm for non-smooth optimization on manifolds. *arXiv preprint arXiv:1505.07676*, 2015.

[33] R. Lai and S. Osher. A splitting method for orthogonality constrained problems. *J. Sci. Comput.*, 58(2):431–449, 2014.

[34] Z. Lai, Y. Xu, Q. Chen, J. Yang, and D. Zhang. Multilinear sparse principal component analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 25(10):1942–1950, 2014.

[35] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. *Advances in Neural Information Processing Systems*, 19:801, 2007.

[36] J. M. Lee. *Introduction to smooth manifolds*, volume 41. 2008.

[37] G. Li and T. K. Pong. Global convergence of splitting methods for nonconvex composite optimization. *SIAM Journal on Optimization*, 25(4):2434–2460, 2015.

[38] H. Liu, W. Wu, and A. M.-C. So. Quadratic optimization with orthogonality constraints: Explicit Lojasiewicz exponent and linear convergence of line-search methods. *arXiv preprint arXiv:1510.01025*, 2015.

[39] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos. Mpca: Multilinear principal component analysis of tensor objects. *IEEE Transactions on Neural Networks*, 19(1):18–39, 2008.

[40] D. G. Luenberger. The gradient projection method along geodesics. *Management Science*, 18(11):620–631, 1972.

[41] D. Motreanu and N. H. Pavel. Quasi-tangent vectors in flow-invariance and optimization problems on Banach manifolds. *Journal of Mathematical Analysis and Applications*, 88(1):116–132, 1982.

[42] A. Nemirovski. Sums of random symmetric matrices and quadratic optimization under orthogonality constraints. *Mathematical Programming*, 109(2):283–317, 2007.

[43] J. Nocedal and S. J. Wright. Numerical optimization. *Springer*, 9(4):1556–1556, 1999.

[44] I. V. Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5):2295–2317, 2011.

[45] I. V. Oseledets and E. Tyrtyshnikov. TT-cross approximation for multidimensional arrays. *Linear Algebra and its Applications*, 432(1):70–88, 2010.

[46] Y. Panagakis, C. Kotropoulos, and G. R. Arce. Non-negative multilinear principal component analysis of auditory temporal modulations for music genre classification. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):576–588, 2010.

[47] S. J. Reddi, S. Sra, B. Poczos, and A. J. Smola. Proximal stochastic methods for nonsmooth nonconvex finite-sum optimization. In *Advances in Neural Information Processing Systems*, pages 1145–1153, 2016.

[48] S. T. Smith. Optimization techniques on Riemannian manifolds. *Fields Institute Communications*, 3(3):113–135, 1994.

[49] N. Srebro and T. Jaakkola. Weighted low-rank approximations. In *ICML*, volume 3, pages 720–727, 2003.

[50] J. Sun, Q. Qu, and J. Wright. Complete dictionary recovery over the sphere ii: Recovery by Riemannian trust-region method. *IEEE Transactions on Information Theory*, 2016.

[51] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, pages 267–288, 1996.

[52] F. Wang, W. Cao, and Z. Xu. Convergence of multi-block Bregman ADMM for nonconvex composite problems. *arXiv preprint arXiv:1505.03063*, 2015.

[53] S. Wang, M. Sun, Y. Chen, E. Pang, and C. Zhou. Stpca: sparse tensor principal component analysis for feature extraction. In *Pattern Recognition, 2012 21st International Conference*, pages 2278–2281. IEEE, 2012.

[54] Y. Wang, W. Yin, and J. Zeng. Global convergence of ADMM in nonconvex nonsmooth optimization. *arXiv preprint arXiv:1511.06324*, 2015.

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

[56] A. Wiegele. Biq mac librarya collection of max-cut and quadratic 0-1 programming instances of medium size. *Preprint*, 2007.

[57] Y. Xu. Alternating proximal gradient method for sparse nonnegative Tucker decomposition. *Mathematical Programming Computation*, 7(1):39–70, 2015.

[58] L. Yang, T. K. Pong, and X. Chen. Alternating direction method of multipliers for a class of nonconvex and nonsmooth problems with applications to background/foreground extraction. *SIAM Journal on Imaging Sciences*, 10(1):74–110, 2017.

[59] W. H. Yang, L.-H. Zhang, and R. Song. Optimality conditions for the nonlinear programming problems on Riemannian manifolds. *Pacific Journal of Optimization*, 10(2):415–434, 2014.

[60] Y. Ye. A .699-approximation algorithm for max-bisection. *Mathematical Programming*, 90(1):101–111, 2001.

[61] H. Zhang, S. J. Reddi, and S. Sra. Riemannian svrg: Fast stochastic optimization on Riemannian manifolds. In *Advances in Neural Information Processing Systems*, pages 4592–4600, 2016.

[62] H. Zhang and S. Sra. First-order methods for geodesically convex optimization. *arXiv preprint arXiv:1602.06053*, 2016.

[63] J. Zhang, H. Liu, Z. Wen, and S. Zhang. A sparse completely positive relaxation of the modularity maximization for community detection. *arXiv preprint arXiv:1708.01072*, 2017.

[64] Y. Zhang, E. Levina, and J. Zhu. Detecting overlapping communities in networks using spectral methods. *arXiv preprint arXiv:1412.3432*, 2014.

[65] H. Zhu, X. Zhang, D. Chu, and L. Liao. Nonconvex and nonsmooth optimization with generalized orthogonality constraints: An approximate augmented lagrangian method. *Journal of Scientific Computing*, pages 1–42, 2017.

[66] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B*, 67(2):301–320, 2005.

A Proofs of the technical lemmas

A.1 Proof of Lemma 3.5

Proof. By the global optimality for the subproblems in Step 1 of Algorithm 1, we have

$$\mathcal{L}_\beta(x_1^{k+1},\cdots,x_{N-1}^{k+1},x_N^k,\lambda^k)\le\mathcal{L}_\beta(x_1^k,\cdots,x_{N-1}^k,x_N^k,\lambda^k)-\frac{1}{2}\sum_{i=1}^{N-1}\|x_i^k-x_i^{k+1}\|_{H_i}^2. \tag{70}$$

By Step 2 of Algorithm 1 we have

$$\mathcal{L}_\beta(x_1^{k+1},\cdots,x_{N-1}^{k+1},x_N^{k+1},\lambda^k)\le\mathcal{L}_\beta(x_1^{k+1},\cdots,x_{N-1}^{k+1},x_N^k,\lambda^k)+\left(\frac{L+\beta}{2}-\frac{1}{\gamma}\right)\|x_N^k-x_N^{k+1}\|^2. \tag{71}$$

By Step 3, directly substitute λ^{k+1} into the augmented Lagrangian gives

$$\mathcal{L}_\beta(x_1^{k+1},\cdots,x_N^{k+1},\lambda^{k+1})=\mathcal{L}_\beta(x_1^{k+1},\cdots,x_N^{k+1},\lambda^k)+\frac{1}{\beta}\|\lambda^k-\lambda^{k+1}\|^2. \tag{72}$$

Summing up (70), (71), (72)) and apply Lemma 3.4, we obtain the following inequality,

$$\begin{aligned}&\mathcal{L}_\beta(x_1^{k+1},\cdots,x_{N-1}^{k+1},x_N^{k+1},\lambda^{k+1})-\mathcal{L}_\beta(x_1^k,\cdots,x_{N-1}^k,x_N^k,\lambda^k)\\ \le\ &\left[\frac{L+\beta}{2}-\frac{1}{\gamma}+\frac{3}{\beta}\left(\beta-\frac{1}{\gamma}\right)^2\right]\|x_N^k-x_N^{k+1}\|^2\\ &+\frac{3}{\beta}\left[\left(\beta-\frac{1}{\gamma}\right)^2+L^2\right]\|x_N^{k-1}-x_N^k\|^2-\sum_{i=1}^{N-1}\|x_i^k-x_i^{k+1}\|^2_{\frac{1}{2}H_i-\frac{3L^2}{\beta}I},\end{aligned} \tag{73}$$

which further indicates

$$\begin{aligned}
&\Psi_G(x_1^{k+1},\cdots,x_{N-1}^{k+1},x_N^{k+1},\lambda^{k+1},x_N^k)-\Psi_G(x_1^k,\cdots,x_{N-1}^k,x_N^k,\lambda^k,x_N^{k-1})\\
\leq\ &\left[\frac{\beta+L}{2}-\frac{1}{\gamma}+\frac{6}{\beta}\left(\beta-\frac{1}{\gamma}\right)^2+\frac{3L^2}{\beta}\right]\|x_N^k-x_N^{k+1}\|^2 \qquad (74)\\
&-\sum_{i=1}^{N-1}\|x_i^k-x_i^{k+1}\|^2_{\frac{1}{2}H_i-\frac{3L^2}{\beta}I}.
\end{aligned}$$

To ensure that the right hand side of (22) is negative, we need to choose $H_i \succ \frac{6L^2}{\beta}I$, and ensure that

$$\frac{\beta+L}{2}-\frac{1}{\gamma}+\frac{6}{\beta}\left(\beta-\frac{1}{\gamma}\right)^2+\frac{3L^2}{\beta}<0. \qquad (75)$$

This can be proved by first viewing it as a quadratic function of $z=\frac{1}{\gamma}$. To find some $z>0$ such that

$$p(z)=\frac{6}{\beta}z^2-13z+\left(\frac{L+\beta}{2}+6\beta+\frac{3}{\beta}L^2\right)<0,$$

we need the discriminant to be positive, i.e.

$$\Delta(\beta)=\frac{1}{\beta^2}(13\beta^2-12\beta L-72L^2)>0. \qquad (76)$$

It is easy to verify that (20) suffices to guarantee (76). Solving $p(z)=0$, we find two positive roots

$$z_1=\frac{13\beta-\sqrt{13\beta^2-12\beta L-72L^2}}{12},\ \text{and}\ z_2=\frac{13\beta+\sqrt{13\beta^2-12\beta L-72L^2}}{12}.$$

Note that γ defined in (21) satisfies $\frac{1}{z_2}<\gamma<\frac{1}{z_1}$ and thus guarantees (75). This completes the proof. □

A.2 Proof of Lemma 3.8

Proof. For the subproblem in Step 1 of Algorithm 2, since x_i^{k+1} is the global minimizer, we have

$$\begin{aligned}
&\langle\nabla_i f(x_1^{k+1},\cdots,x_{i-1}^{k+1},x_i^k,\cdots,x_N^k),x_i^{k+1}-x_i^k\rangle-\left\langle\sum_{j=1}^{i}A_jx_j^{k+1}+\sum_{j=i+1}^{N}A_jx_j^k-b,\lambda^k\right\rangle\\
&+\frac{\beta}{2}\left\|\sum_{j=1}^{i}A_jx_j^{k+1}+\sum_{j=i+1}^{N}A_jx_j^k-b\right\|^2+\sum_{j=1}^{i}r_j(x_j^{k+1})+\sum_{j=i+1}^{N-1}r_j(x_j^k)\\
\leq\ &-\left\langle\sum_{j=1}^{i-1}A_jx_j^{k+1}+\sum_{j=i}^{N}A_jx_j^k-b,\lambda^k\right\rangle+\frac{\beta}{2}\left\|\sum_{j=1}^{i-1}A_jx_j^{k+1}+\sum_{j=i}^{N}A_jx_j^k-b\right\|^2\\
&+\sum_{j=1}^{i-1}r_j(x_j^{k+1})+\sum_{j=i}^{N-1}r_j(x_j^k)-\frac{1}{2}\|x_i^{k+1}-x_i^k\|^2_{H_i}.
\end{aligned}$$

By the L-Lipschitz continuity of $\nabla_i f$, we have

$$\begin{aligned} & f(x_1^{k+1},\cdots,x_i^{k+1},x_{i+1}^k,\cdots,x_N^k) \\ \leq\ & f(x_1^{k+1},\cdots,x_{i-1}^{k+1},x_i^k,\cdots,x_N^k)+\langle\nabla_i f(x_1^{k+1},\cdots,x_{i-1}^{k+1},x_i^k,\cdots,x_N^k),x_i^{k+1}-x_i^k\rangle \\ & +\frac{L}{2}\|x_i^{k+1}-x_i^k\|^2. \end{aligned}$$

Combining the above two inequalities and using the definition of $\mathcal{L}_\beta$ in (16), we have

$$\mathcal{L}_\beta(x_1^{k+1},\cdots,x_i^{k+1},x_{i+1}^k,\cdots,x_N^k,\lambda^k)\leq\mathcal{L}_\beta(x_1^{k+1},\cdots,x_{i-1}^{k+1},x_i^k,\cdots,x_N^k,\lambda^k)-\|x_i^k-x_i^{k+1}\|^2_{\frac{H_i}{2}-\frac{L}{2}I}. \tag{77}$$

Summing (77) over $i=1,\ldots,N-1$, we have the following inequality, which is the counterpart of (70):

$$\mathcal{L}_\beta(x_1^{k+1},\cdots,x_{N-1}^{k+1},x_N^k,\lambda^k)\leq\mathcal{L}_\beta(x_1^k,\cdots,x_N^k,\lambda^k)-\sum_{i=1}^{N-1}\|x_i^k-x_i^{k+1}\|^2_{\frac{H_i}{2}-\frac{L}{2}I}. \tag{78}$$

Besides, since (71) and (72) still hold, by combining (78), (71) and (72) and applying Lemma 3.4, we establish the following two inequalities, which are respectively the counterparts of (73) and (22):

$$\begin{aligned} & \mathcal{L}_\beta(x_1^{k+1},\cdots,x_{N-1}^{k+1},x_N^{k+1},\lambda^{k+1})-\mathcal{L}_\beta(x_1^k,\cdots,x_{N-1}^k,x_N^k,\lambda^k) \\ \leq\ & \left[\frac{L+\beta}{2}-\frac{1}{\gamma}+\frac{3}{\beta}\left(\beta-\frac{1}{\gamma}\right)^2\right]\|x_N^k-x_N^{k+1}\|^2 \\ & +\frac{3}{\beta}\left[\left(\beta-\frac{1}{\gamma}\right)^2+L^2\right]\|x_N^{k-1}-x_N^k\|^2-\sum_{i=1}^{N-1}\|x_i^k-x_i^{k+1}\|^2_{\frac{1}{2}H_i-\frac{L}{2}I-\frac{3L^2}{\beta}I}, \end{aligned} \tag{79}$$

and

$$\begin{aligned} & \Psi_G(x_1^{k+1},\cdots,x_{N-1}^{k+1},x_N^{k+1},\lambda^{k+1},x_N^k)-\Psi_G(x_1^k,\cdots,x_{N-1}^k,x_N^k,\lambda^k,x_N^{k-1}) \\ \leq\ & \left[\frac{\beta+L}{2}-\frac{1}{\gamma}+\frac{6}{\beta}\left(\beta-\frac{1}{\gamma}\right)^2+\frac{3L^2}{\beta}\right]\|x_N^k-x_N^{k+1}\|^2 \\ & -\sum_{i=1}^{N-1}\|x_i^k-x_i^{k+1}\|^2_{\frac{1}{2}H_i-\frac{L}{2}I-\frac{3L^2}{\beta}I}. \end{aligned}$$

From the proof of Lemma 3.5, it is easy to see that the right hand side of the above inequality is negative, if $H_i\succ\left(\frac{6L^2}{\beta}+L\right)I$ and β and γ are chosen according to (20) and (21). □

A.3 Proof of Lemma 3.11

Proof. For the ease of notation, we denote

$$G_i^M(x_1^{k+1},\ldots,x_{i-1}^{k+1},x_i^k,\ldots,x_N^k)=\nabla_i f(x_1^{k+1},\ldots,x_{i-1}^{k+1},x_i^k,\ldots,x_N^k)+\delta_i^k. \tag{80}$$

Note that δ_i^k is a zero-mean random variable. By Steps 2 and 3 of Algorithm 3 we obtain

$$\lambda^{k+1} = \left(\beta - \frac{1}{\gamma}\right)(x_N^k - x_N^{k+1}) + \nabla_N f(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k) + \delta_N^k. \tag{81}$$

Applying (81) for k and $k+1$, and using (81), we get

$$\begin{aligned}
\|\lambda^{k+1} - \lambda^k\|^2 &= \left\| \left(\beta - \frac{1}{\gamma}\right)(x_N^k - x_N^{k+1}) - \left(\beta - \frac{1}{\gamma}\right)(x_N^{k-1} - x_N^k) + (\delta_N^k - \delta_N^{k-1}) \right. \\
&\quad \left. +(\nabla_N f(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k) - \nabla_N f(x_1^k, \cdots, x_{N-1}^k, x_N^{k-1}) \right\|^2 \\
&\leq 4\left(\beta - \frac{1}{\gamma}\right)^2 \|x_N^k - x_N^{k+1}\|^2 + 4\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right] \|x_N^{k-1} - x_N^k\|^2 \\
&\quad +4L^2 \sum_{i=1}^{N-1} \|x_i^k - x_i^{k+1}\|^2 + 4\|\delta_N^k - \delta_N^{k-1}\|^2.
\end{aligned}$$

Taking expectation with respect to all random variables on both sides and using $\mathsf{E}[\langle \delta_N^k, \delta_N^{k-1} \rangle] = 0$ completes the proof. □

A.4 Proof of Lemma 3.12

Proof. Similar as (77), by further incorporating (80), we have

$$\begin{aligned}
& \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_i^{k+1}, x_{i+1}^k, \cdots, x_N^k, \lambda^k) - \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i^k, \cdots, x_N^k, \lambda^k) \\
\leq\ & -\|x_i^k - x_i^{k+1}\|^2_{\frac{H_i}{2} - \frac{L}{2}I} + \langle \delta_i^k, x_i^{k+1} - x_i^k \rangle \\
\leq\ & -\|x_i^k - x_i^{k+1}\|^2_{\frac{H_i}{2} - \frac{L}{2}I} + \frac{1}{2}\|\delta_i^k\|^2 + \frac{1}{2}\|x_i^{k+1} - x_i^k\|^2.
\end{aligned}$$

Taking expectation with respect to all random variables on both sides and summing over $i = 1, \ldots, N-1$, and using (36), we obtain

$$\begin{aligned}
& \mathsf{E}[\mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k)] - \mathsf{E}[\mathcal{L}_\beta(x_1^k, \cdots, x_N^k, \lambda^k)] \\
\leq\ & -\sum_{i=1}^{N-1} \mathsf{E}\left[\|x_i^{k+1} - x_i^k\|^2_{\frac{1}{2}H_i - \frac{L+1}{2}I}\right] + \frac{N-1}{2M}\sigma^2.
\end{aligned} \tag{82}$$

Note that by the Step 2 of Algorithm 3 and the descent lemma we have

$$\begin{aligned}
0 &= \left\langle x_N^k - x_N^{k+1}, \nabla_N f(x_1^{k+1}, ..., x_{N-1}^{k+1}, x_N^k) + \delta_N^k - \lambda^k + \beta\left(\sum_{j=1}^{N-1} A_j x_j^{k+1} + x_N^k - b\right) - \frac{1}{\gamma}(x_N^k - x_N^{k+1})\right\rangle \\
&\leq f(x_1^{k+1}, ..., x_{N-1}^{k+1}, x_N^k) - f(x^{k+1}) + \left(\frac{L+\beta}{2} - \frac{1}{\gamma}\right)\|x_N^{k+1} - x_N^k\|^2 - \langle \lambda^k, x_N^k - x_N^{k+1}\rangle \\
&\quad + \frac{\beta}{2}\|\sum_{j=1}^{N-1} A_j x_j^{k+1} + x_N^k - b\|^2 - \frac{\beta}{2}\|\sum_{j=1}^{N-1} A_j x_j^{k+1} + x_N^{k+1} - b\|^2 + \langle \delta_N^k, x_N^k - x_N^{k+1}\rangle \\
&\leq \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k) - \mathcal{L}_\beta(x^{k+1}, \lambda^k) + \left(\frac{L+\beta}{2} - \frac{1}{\gamma} + \frac{1}{2}\right)\|x_N^k - x_N^{k+1}\|^2 + \frac{1}{2}\|\delta_N^k\|^2.
\end{aligned}$$

Taking the expectation with respect to all random variables yields

$$\begin{aligned}
&\mathsf{E}[\mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^{k+1}, \lambda^k)] - \mathsf{E}[\mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k)] \\
\leq\ & \left(\frac{L+\beta}{2} - \frac{1}{\gamma} + \frac{1}{2}\right)\mathsf{E}[\|x_N^k - x_N^{k+1}\|^2] + \frac{1}{2M}\sigma^2. \qquad (83)
\end{aligned}$$

The following equality holds trivially from Step 3 of Algorithm 3:

$$\mathsf{E}[\mathcal{L}_\beta(x_1^{k+1}, \cdots, x_N^{k+1}, \lambda^{k+1})] - \mathsf{E}[\mathcal{L}_\beta(x_1^{k+1}, \cdots, x_N^{k+1}, \lambda^k)] = \frac{1}{\beta}\mathsf{E}[\|\lambda^k - \lambda^{k+1}\|^2]. \qquad (84)$$

Combining (82), (83), (84) and (38), we obtain

$$\begin{aligned}
&\mathsf{E}[\Psi_S(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^{k+1}, \lambda^{k+1}, x_N^k)] - \mathsf{E}[\Psi_S(x_1^k, \cdots, x_{N-1}^k, x_N^k, \lambda^k, x_N^{k-1})] \qquad (85) \\
\leq\ & \left[\frac{\beta+L}{2} - \frac{1}{\gamma} + \frac{8}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{4L^2}{\beta} + \frac{1}{2}\right]\mathsf{E}[\|x_N^k - x_N^{k+1}\|^2] \\
& - \sum_{i=1}^{N-1}\mathsf{E}\left[\|x_i^k - x_i^{k+1}\|^2_{\frac{1}{2}H_i - \frac{4L^2}{\beta}I - \frac{L+1}{2}I}\right] + \left(\frac{8}{\beta} + \frac{1}{2} + \frac{N-1}{2}\right)\frac{\sigma^2}{M}.
\end{aligned}$$

Choosing β and γ according to (40) and (41), and using the similar arguments in the proof of Lemma 3.5, it is easy to verify that

$$\left[\frac{\beta+L}{2} - \frac{1}{\gamma} + \frac{8}{\beta}\left(\beta - \frac{1}{\gamma}\right)^2 + \frac{4L^2}{\beta} + \frac{1}{2}\right] < 0.$$

By further choosing $H_i \succ \left(\frac{8L^2}{\beta} + L + 1\right)I$, we know that the right hand side of (85) is negative, and this completes the proof. □

A.5 Proof of Lemma 3.13

Proof. From (81) and (15), we have that

$$\begin{aligned}
& \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_N^{k+1}, \lambda^{k+1}) \\
= & \sum_{i=1}^{N-1} r_i(x_i^{k+1}) + f(x^{k+1}) - \left\langle \sum_{i=1}^{N} A_i x_i^{k+1} - b, \nabla_N f(x^{k+1}) + \left(\beta - \frac{1}{\gamma}\right)(x_N^k - x_N^{k+1}) \right. \\
& \left. + \nabla_N f(x_1^{k+1}, ..., x_{N-1}^{k+1}, x_N^k) - \nabla_N f(x^{k+1}) + \delta_N^k \right\rangle + \frac{\beta}{2}\left\| \sum_{i=1}^{N} A_i x_i^{k+1} - b \right\|^2 \\
\geq & \sum_{i=1}^{N-1} r_i(x_i^{k+1}) + f(x_1^{k+1}, ..., x_{N-1}^{k+1}, b - \sum_{i=1}^{N-1} A_i x_i^{k+1}) - \frac{4}{\beta}\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right]\|x_N^k - x_N^{k+1}\|^2 \\
& + \left(\frac{\beta}{2} - \frac{\beta}{8} - \frac{\beta}{8} - \frac{L}{2}\right)\left\| \sum_{i=1}^{N} A_i x_i^{k+1} - b \right\|^2 - \frac{2}{\beta}\|\delta_N^k\|^2 \\
\geq & \sum_{i=1}^{N-1} r_i^* + f^* - \frac{4}{\beta}\left[\left(\beta - \frac{1}{\gamma}\right)^2 + L^2\right]\|x_N^k - x_N^{k+1}\|^2 - \frac{2}{\beta}\|\delta_N^k\|^2
\end{aligned}$$

where the first inequality is obtained by applying $\langle a, b\rangle \leq \frac{1}{2}(\frac{1}{\eta}\|a\|^2 + \eta\|b\|^2)$ to terms $\langle \sum_{i=1}^N A_i x_i^{k+1} - b, \left(\beta - \frac{1}{\gamma}\right)(x_N^k - x_N^{k+1})\rangle$, $\langle \sum_{i=1}^N A_i x_i^{k+1} - b, \nabla_N f(x_1^{k+1}, ..., x_{N-1}^{k+1}, x_N^k) - \nabla_N f(x^{k+1})\rangle$ and $\langle \sum_{i=1}^N A_i x_i^{k+1} - b, \delta_N^k\rangle$ respectively with $\eta = \frac{8}{\beta}, \frac{8}{\beta}$ and $\frac{4}{\beta}$. Note that $\beta > 2L$ according to (40), thus $(\frac{\beta}{2} - \frac{\beta}{8} - \frac{\beta}{8} - \frac{L}{2}) > 0$ and the last inequality holds. By rearranging the terms and taking expectation with respect to all random variables completes the proof. □

A.6 Proof for Theorem 3.18

Proof. Through similar argument, one can easily obtain

$$\|\lambda^{k+1} - \nabla_N f(x_1^{k+1}, \cdots, x_N^{k+1})\|^2 \leq \kappa_2 \theta_k \quad \text{and} \quad \left\| \sum_{i=1}^{N-1} A_i x_i^{k+1} + x_N^{k+1} - b \right\|^2 \leq \kappa_1 \theta_k.$$

The only remaining task is to guarantee an ϵ version of (48). First let us prove that

$$\|g_i^{k+1}\| \leq \frac{\sigma + 2L_2 C + (L + \beta A_{\max}^2) L_1^2}{2\alpha} \sqrt{\theta_k}. \tag{86}$$

Denote $h_i(x_i) = \mathcal{L}_\beta(x_1^{k+2}, ..., x_{i-1}^{k+2}, x_i, x_{i+1}^{k+1}, ..., x_N^{k+1}, \lambda^{k+1})$ and $Y_i(t) = R(x_i^{k+1}, -t g_i^{k+1})$, then it is not hard to see that $\nabla h_i(x_i)$ is Lipschitz continuous with parameter $L + \beta\|A_i\|_2^2 \leq L_3 := L + \beta A_{\max}^2$.

Consequently, it yields

$$\begin{aligned} h_i(Y_i(t)) &\leq h_i(Y_i(0)) + \langle \nabla h_i(Y_i(0)), Y_i(t) - Y_i(0) - tY_i'(0) + tY_i'(0)\rangle + \frac{L_3}{2}\|Y_i(t) - Y_i(0)\|^2 \\ &\leq h_i(Y_i(0)) + t\langle \nabla h_i(Y_i(0)), Y_i'(0)\rangle + L_2 t^2 \|\nabla h_i(Y_i(0))\|\|Y_i'(0)\|^2 + \frac{L_3 L_1^2}{2} t^2 \|Y_i'(0)\|^2 \\ &= h_i(Y_i(0)) - \left(t - L_2 t^2 \|\nabla h_i(Y_i(0))\| - \frac{L_3 L_1^2}{2} t^2 \right) \|Y_i'(0)\|^2, \end{aligned}$$

where the last equality is due to $\langle \nabla h_i(Y_i(0)), Y_i'(0)\rangle = -\langle Y_i'(0), Y_i'(0)\rangle$. Also note the relationship

$$\|Y_i'(0)\| = \|g_i^{k+1}\| = \|\mathrm{Proj}\,_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\{\nabla h_i(Y_i(0))\}\| \leq \|\nabla h_i(Y_i(0))\|.$$

Note that $\left\|\sum_{i=1}^{N-1} A_i x_i^{k+1} + x_N^{k+1} - b\right\| \leq \sqrt{\kappa_1 \theta_k} \leq \sqrt{\frac{\kappa_1}{\tau}(\Psi_G(x_1^1, ..., x_N^1, \lambda^1, x_N^0) - f^*)}$. Because $\mathcal{M}_i, i = 1, ..., N-1$ are all compact manifolds, $x_i^{k+1}, i = 1, ..., N-1$ are all bounded. Hence the whole sequence $\{x_N^k\}$ is also bounded. By (18) (which also holds in this case),

$$\|\lambda^{k+1}\| \leq |\beta - \frac{1}{\gamma}|\sqrt{\theta_k} + \|\nabla_N f(x_1^{k+1}, \ldots, x_{N-1}^{k+1}, x_N^k)\|.$$

By the boundedness of $\{(x_1^k, \ldots, x_N^k)\}$ and the continuity of $\nabla f(\cdot)$, the second term is bounded. Combining the boundedness of $\{\theta_k\}$, we know that whole sequence $\{\lambda^k\}$ is bounded. Consequently, there exists a constant $C > 0$ such that $\|\nabla h_i(Y_i(0))\| \leq C$, where

$$\nabla h_i(Y_i(0)) = \nabla_i f(x_1^{k+2}, ..., x_{i-1}^{k+2}, x_i^{k+1}, ..., x_N^{k+1}) - A_i^\top \lambda^{k+1} + \beta A_i^\top \left(\sum_{j=1}^{i-1} A_j x_j^{k+2} + \sum_{j=i}^{N} A_j x_j^{k+1} - b \right)\Big).$$

Note that this constant C depends only on the first two iterates $\{x_1^t, ..., x_N^t, \lambda^t\}, t = 0, 1$, except for the absolute constants such as $\|A_i\|_2, i = 1, ..., N$. Therefore, when

$$t \leq \frac{2}{2L_2 C + \sigma + L_3 L_1^2} \leq \frac{2}{2L_2\|\nabla h_i(Y_i(0))\| + \sigma + L_3 L_1^2},$$

it holds that

$$h_i(Y_i(t)) \leq h_i(x_i^{k+1}) - \frac{\sigma}{2} t^2 \|g_i^{k+1}\|^2.$$

Note that $\sigma > \frac{2\alpha}{s}$, by the terminating rule of the line-search step, we have

$$t_i^k \geq \min\left\{ s, \frac{2\alpha}{2L_2 C + \sigma + L_3 L_1^2} \right\} = \frac{2\alpha}{2L_2 C + \sigma + L_3 L_1^2}.$$

Then by noting

$$\frac{2\alpha\|g_i^{k+1}\|}{2L_2 C + \sigma + L_3 L_1^2} \leq t_i^{k+1}\|g_i^{k+1}\| \leq \sqrt{\theta_k},$$

we have (86).

Now let us discuss the issue of (48). By definition,

$$g_i^{k+1} = \mathrm{Proj}_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\Big\{\nabla_i f(x_1^{k+2},...,x_{i-1}^{k+2},x_i^{k+1},...,x_N^{k+1}) - A_i^\top\lambda^{k+1} + \beta A_i^\top\Big(\sum_{j=1}^{i-1}A_j x_j^{k+2} + \sum_{j=i}^{N}A_j x_j^{k+1} - b\Big)\Big\}.$$

Consequently, we obtain

$$\begin{aligned}
&\left\|\mathrm{Proj}_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\Big\{\nabla_i f(x^{k+1}) - A_i^\top\lambda^{k+1}\Big\}\right\| \\
= &\left\|\mathrm{Proj}_{\mathcal{T}_{x_i^{k+1}}\mathcal{M}_i}\Big\{\nabla_i f(x^{k+1}) - \nabla_i f(x_1^{k+2},\cdots,x_{i-1}^{k+2},x_i^{k+1},\cdots,x_N^{k+1}) + g_i^{k+1}\right. \\
&\left.-\beta A_i\left(\sum_{j=1}^{N}A_j x_j^{k+1} - b\right) + \beta A_i^\top\left(\sum_{j=1}^{i-1}A_j(x_j^{k+1} - x_j^{k+2})\right)\Big\}\right\| \\
\le &\|\nabla_i f(x^{k+1}) - \nabla_i f(x_1^{k+2},\cdots,x_{i-1}^{k+2},x_i^{k+1},\cdots,x_N^{k+1})\| + \|\beta A_i(\sum_{j=1}^{N}A_j x_j^{k+1} - b)\| \\
&+\|g_i^{k+1}\| + \|\beta A_i^\top(\sum_{j=i+1}^{N}A_j(x_j^{k+1} - x_j^k))\| \\
\le &\left(L + \sqrt{N}\beta A_{\max}^2\right)\max\{L_1,1\}\sqrt{\theta_k} + \frac{\sigma + 2L_2C + (L+\beta A_{\max}^2)L_1^2}{2\alpha}\sqrt{\theta_k} + \beta\|A_i\|_2\sqrt{\kappa_1\theta_k} \\
\le &\sqrt{\kappa_3\theta_k}.
\end{aligned}$$

□

A.7 Proof for inequality (60)

Proof. First, we need to figure out the Lipschitz constant of $\bar{f}_\beta$.

$$\begin{aligned}
&\|\nabla\bar{f}_\beta(x) - \nabla\bar{f}_\beta(y)\| \\
\le &L\|x-y\| + \beta\left\|\left[\left(\sum_{j=1}^{N}A_j(x_j-y_j)\right)^\top A_1,\cdots,\left(\sum_{j=1}^{N}A_j(x_j-y_j)\right)^\top A_N\right]\right\| \qquad (87)\\
\le &L\|x-y\| + \beta\sqrt{N}\max_{1\le i\le N}\|A_i\|_2\left\|\sum_{j=1}^{N}A_j(x_j-y_j)\right\| \\
\le &\left(L + \beta N\max_{1\le i\le N}\|A_i\|_2^2\right)\|x-y\|.
\end{aligned}$$

So we define $\hat{L} = L + \beta N \max_{1\le i\le N} \|A_i\|_2^2$ as the Lipschitz constant for function $\bar{f}_\beta$. The global optimality of the subproblem (59) yields

$$\langle \nabla_i \bar{f}_\beta(x_1^k, \cdots, x_N^k), x_i^{k+1} - x_i^k\rangle - \langle \lambda^k, A_i x_i^{k+1}\rangle + r_i(x_i^{k+1}) + \frac{1}{2}\|x_i^{k+1} - x_i^k\|_{H_i}^2 \le r_i(x_i^k) - \langle \lambda^k, A_i x_i^k\rangle.$$

By the descent lemma we have

$$\begin{aligned}
& \mathcal{L}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k, \lambda^k) \\
= \;& \bar{f}_\beta(x_1^{k+1}, \cdots, x_{N-1}^{k+1}, x_N^k) - \langle \lambda^k, \sum_{i=1}^N A_i x_i^{k+1} - b\rangle + \sum_{i=1}^{N-1} r_i(x_i^{k+1}) \\
\le \;& \bar{f}_\beta(x_1^k, \cdots, x_{N-1}^k, x_N^k) + \langle \nabla \bar{f}_\beta(x_1^k, \cdots, x_{N-1}^k, x_N^k), x^{k+1} - x^k\rangle \\
& \frac{\hat{L}}{2}\|x^{k+1} - x^k\|^2 - \langle \lambda^k, \sum_{i=1}^N A_i x_i^{k+1} - b\rangle + \sum_{i=1}^{N-1} r_i(x_i^{k+1}).
\end{aligned}$$

Combining the above two inequalities yields (60). □