Iteration complexity of Riemannian ADMM

Kangkang Deng, Jiachen Jin, Jiang Hu, Hongxia Wang

National University of Defense Technology
Tsinghua University

Outline

1. Riemannian ADMM
2. Numerical results

Problem formulation

Consider

$$\min_{x\in\mathcal{M}} F(x) := f(x) + h(\mathcal{A}x), \tag{1}$$

Assumptions:

1. The manifold $\mathcal{M}$ is a compact submanifold embedded in $\mathbb{R}^n$.
2. The function f is ℓ_f-Lipschitz coninuous and $\ell_{\nabla f}$-smooth on $\mathcal{M}$.
3. The function h is convex and ℓ_h-Lipschitz continuous.
4. The linear mapping $\mathcal{A}$ satisfies $\|\mathcal{A}\|_{op} \leq \sigma_A$.

The splitting orthogonality constraints algorithm [Lai and Osher, 2014]

Introduce $y = x$:

$$\min_{x,y} f(x) + h(\mathcal{A}x), \text{ s.t. } x = y, \; y \in \mathcal{M}.$$

The SOC algorithm is given as follows:

$$\begin{cases} x_{k+1} = \arg\min_{x} f(x) + h(\mathcal{A}x) - \langle \lambda_k, x - y_k \rangle + \frac{\rho}{2}\|x - y_k\|^2, \\ y_{k+1} = \arg\min_{y \in \mathcal{M}} \langle \lambda_k, x_{k+1} - y \rangle + \frac{\rho}{2}\|x_{k+1} - y\|^2, \\ \lambda_{k+1} = \lambda_k - \rho(x_{k+1} - y_{k+1}). \end{cases}$$

Difficult subproblem and the convergence is unclear.

The Riemannian ADMM algorithm [Kovnatsky et al., 2016]

Introduce $y = \mathcal{A}x$:

$$\min_{x,y} f(x) + h(y), \text{ s.t. } \mathcal{A}x = y, \; x \in \mathcal{M}.$$

The Riemannian ADMM is given as follows:

$$\begin{cases} x_{k+1} = \arg\min_{x \in \mathcal{M}} f(x) - \langle \lambda_k, \mathcal{A}x - y_k \rangle + \frac{\rho}{2}\|\mathcal{A}x - y_k\|^2, \\ y_{k+1} = \arg\min_{y} h(y) - \langle \lambda_k, \mathcal{A}x_{k+1} - y \rangle + \frac{\rho}{2}\|\mathcal{A}x_{k+1} - y\|^2, \\ \lambda_{k+1} = \lambda_k - \rho(\mathcal{A}x_{k+1} - y_{k+1}). \end{cases}$$

- The x-subproblem is smooth optimization on manifold and y_{k+1} is a proximal operator.
- The convergence is still unclear.

Primal-dual ADMM algorithm [Zhang et al., 2020]

Consider the following N-block nonconvex and nonsmooth optimization on manifold:

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

- They propose (linearized) proximal ADMM algorithms and establish the iteration complexity of $\mathcal{O}(\epsilon^{-2})$.
- The last variable x_N must not appear in the nonsmooth part of the objective, nor be subject to manifold constraints.

Convergence of Euclidean ADMM

The key component in analyzing convergence of ADMM is to bound:

$$\|\lambda_{k+1} - \lambda_k\|.$$

In Euclidean ADMM, this term is often bounded using the optimality condition

$$\nabla f(x_{k+1}) + \rho\mathcal{A}^*(\mathcal{A}x_{k+1} - y_k) - \mathcal{A}^*\lambda_k = 0.$$

Using the **full-rank** of $\mathcal{A}$, and taking difference between k and $k+1$:

$$\begin{aligned}
&\|\lambda_k - \lambda_{k-1}\| \\
\leq&\frac{1}{\sigma_{\min}(\mathcal{A})}\left(\|\nabla f(x_{k+1}) - \nabla f(x_k)\| + \rho\|\mathcal{A}\|_{op}^2\|x_{k+1} - x_k\| + \rho\|\mathcal{A}\|_{op}\|y_k - y_{k-1}\|\right) \\
\leq&\mathcal{O}(\|x_{k+1} - x_k\| + \|y_k - y_{k-1}\|).
\end{aligned}$$

Return to Riemannian ADMM

Recall that x_{k+1} obtained by solving

$$x_{k+1} = \arg\min_{x \in \mathcal{M}} f(x) - \langle \lambda_k, \mathcal{A}x - y_k \rangle + \frac{\rho}{2} \|\mathcal{A}x - y_k\|^2$$

The optimality condition of x-subproblem is given by

$$\mathcal{P}_{T_x\mathcal{M}} \left(\nabla f(x_{k+1}) + \rho \mathcal{A}^*(\mathcal{A}x_{k+1} - y_k) - \mathcal{A}^* \lambda_k \right) = 0.$$

The projection operator $\mathcal{P}_{T_x\mathcal{M}}$ is **not full-rank**, even though when $\mathcal{M}$ is convex set.

Smoothed Riemannian ADMM [Li et al., 2024]

One choice is consider smoothed problem:

$$\min_{x \in \mathcal{M}} f(x) + h_\mu(\mathcal{A}x),$$

The smoothed Riemannian ADMM is given as follows:

$$\begin{cases} x_{k+1} = \arg\min_{x \in \mathcal{M}} f(x) - \langle \lambda_k, \mathcal{A}x - y_k \rangle + \frac{\rho}{2} \|\mathcal{A}x - y_k\|^2, \\ y_{k+1} = \arg\min_{y} h_\mu(y) - \langle \lambda_k, \mathcal{A}x_{k+1} - y \rangle + \frac{\rho}{2} \|\mathcal{A}x_{k+1} - y\|^2, \\ \lambda_{k+1} = \lambda_k - \rho(\mathcal{A}x_{k+1} - y_{k+1}). \end{cases}$$

The iteration complexity is $\mathcal{O}(\epsilon^{-4})$ when $\mu = \mathcal{O}(\epsilon)$.

Smoothed Riemannian ADMM

Instead of using x-subproblem, one considers the optimality condition of y-subproblem:

$$0 = \nabla h_\mu(y_{k+1}) + \rho(\mathcal{A}x_k - y_{k+1}) - \lambda_k.$$

Taking difference between k and $k+1$:

$$\|\lambda_{k+1} - \lambda_k\| \leq \|\nabla h_\mu(y_{k+1}) - \nabla h_\mu(y_k)\| + \rho\|\mathcal{A}\|_{op}\|x_k - x_{k-1}\| + \rho\|y_{k+1} - y_k\|.$$

- When parameter $\mu_k = \frac{1}{k^{1/3}}$, the complexity is improved to $\mathcal{O}(\epsilon^{-3})$ [Yuan, 2024].
- The smoothed problem can also solved by Riemannian gradient descent directly with iteration complexity of $\mathcal{O}(\epsilon^{-3})$ [Peng et al., 2023, Beck and Rosset, 2023].

Motivations

- Can we design an Riemannian ADMM for original problem:

$$\min_{x \in \mathcal{M}} f(x) + h(\mathcal{A}x).$$

The Riemannian ADMM with dynamic parameters is given as follows:

$$\begin{cases} y_{k+1} = \arg\min_{y} h(y) - \langle \lambda_k, \mathcal{A}x_k - y \rangle + \frac{\rho_k}{2} \|\mathcal{A}x_k - y\|^2, \\ x_{k+1} = \arg\min_{x \in \mathcal{M}} f(x) - \langle \lambda_k, \mathcal{A}x - y_{k+1} \rangle + \frac{\rho_k}{2} \|\mathcal{A}x - y_{k+1}\|^2, \\ \lambda_{k+1} = \lambda_k - \beta_k(\mathcal{A}x_{k+1} - y_{k+1}). \end{cases}$$

We try to control the penalty ρ_k and the dual stepsize β_k to bound $\|\lambda_{k+1} - \lambda_k\|$.

Motivations

Motivated by the Moreau envelop, we find from the update of λ_{k+1}:

$$\begin{aligned}
\lambda_{k+1} - \lambda_k &= \beta_{k+1}(\mathcal{A}x_{k+1} - y_{k+1}) \\
&= \beta_{k+1}\left(\mathcal{A}x_k - y_{k+1} - \frac{\lambda_k}{\rho_k}\right) + \beta_{k+1}\mathcal{A}(x_{k+1} - x_k) + \beta_{k+1}\frac{\lambda_k}{\rho_k} \\
&\leq \beta_{k+1}\left(\mathcal{A}x_k - \frac{\lambda_k}{\rho_k} - \mathrm{prox}_{h/\rho_k}\left(\mathcal{A}x_k - \frac{\lambda_k}{\rho_k}\right)\right) + \beta_{k+1}\mathcal{A}(x_{k+1} - x_k) + \beta_{k+1}\frac{\lambda_k}{\rho_k}.
\end{aligned}$$

- The first term is controlled by the Lipschitz constant of h:

$$\mathcal{A}x_k - \frac{\lambda_k}{\rho_k} - \mathrm{prox}_{h/\rho_k}(\mathcal{A}x_k - \frac{\lambda_k}{\rho_k}) \in \partial h(\mathcal{A}x_k - \frac{\lambda_k}{\rho_k})$$

- By careful choosing β_k and ρ_k, one can bound $\|\lambda_{k+1} - \lambda_k\|$.

Contributions

Table: Comparison of the oracle complexity results of several methods in the literature to our method to produce an ϵ-stationary point

Algorithms	Manifold	Iteration	Without smoothing	Single-loop
MADMM [Kovnatsky et al., 2016]	compact	—	Yes	No
RADMM [Li et al., 2024]	compact	$\mathcal{O}(\epsilon^{-4})$	No	Yes
OADMM [Yuan, 2024]	Stiefel	$\mathcal{O}(\epsilon^{-3})$	No	Yes
this paper	compact	$\mathcal{O}(\epsilon^{-3})$	Yes	Yes

- We propose a novel adaptive RADMM, and establish an optimal iteration complexity of order $\mathcal{O}(\epsilon^{-3})$ for finding an ϵ-approximate KKT point.
- We extend the algorithm to the stochastic setting and establish the same order of iteration complexity, up to a logarithmic factor.

Stationary point

Introduce variable $y = \mathcal{A}x$ and construct Lagrangian function

$$l(x, y, \lambda) = f(x) + h(y) - \langle \lambda, \mathcal{A}x - y \rangle, \tag{2}$$

Definition 1 (ϵ-stationarity)

We say $x \in \mathcal{M}$ is an ϵ-stationary point of (1) if there exists $y, z \in \mathbb{R}^m$ such that

$$\begin{cases} \|\mathcal{P}_{T_x\mathcal{M}}\left(\nabla f(x) - \mathcal{A}^*\lambda\right)\| \leq \epsilon, \\ \mathrm{dist}(-\lambda, \partial h(y)) \leq \epsilon, \\ \|\mathcal{A}x - y\| \leq \epsilon. \end{cases} \tag{3}$$

In other words, (x, y, λ) is an ϵ-KKT point pair of (1).

Algorithm

Introduce augmented Lagrangian function:

$$\mathcal{L}_\rho(x, y, \lambda) := f(x) + h(y) - \langle \lambda, \mathcal{A}x - y \rangle + \frac{\rho}{2}\|\mathcal{A}x - y\|^2.$$

Our Riemannian ADMM algorithm is as follows:

$$\begin{cases} y_{k+1} = \arg\min_{y} \mathcal{L}_{\rho_k}(x_k, y, \lambda_k), \\ x_{k+1} \approx \arg\min_{x \in \mathcal{M}} \mathcal{L}_{\rho_k}(x, y_{k+1}, \lambda_k), \\ \lambda_{k+1} = \lambda_k - \beta_k(\mathcal{A}x_{k+1} - y_{k+1}). \end{cases} \tag{4}$$

- The dual stepsize β_k and penalty parameter ρ_k are updated by different strategies.
- Apply one gradient step to solve x-subproblem.

Algorithm

Algorithm An Novel Riemannian ADMM

Require: Initial point $x_0, y_0, \lambda_0, \rho_0, \beta_0$, parameters c_ρ, c_β.

1: **for** $k = 0, \cdots, K-1$ **do**

2: Update auxiliary variable y_{k+1} via

$$y_{k+1} = \arg\min_{y \in \mathbb{R}^d} \mathcal{L}_{\rho_k}(x_k, y, \lambda_k). \tag{5}$$

3: Denote $\Phi_k(x) := \mathcal{L}_{\rho_k}(x, y_{k+1}, \lambda_k)$ and obtain x_{k+1} by single gradient step:

$$x_{k+1} = \mathcal{R}_{x_k}(-\tau_k \mathrm{grad}\, \Phi_k(x_k)). \tag{6}$$

4: Update the dual step size β_{k+1} via

$$\beta_{k+1} = \min\left(\frac{\beta_0 \|\mathcal{A}x_0 - y_0\| \log^2 2}{\|\mathcal{A}x_{k+1} - y_{k+1}\|(k+1)^2 \log(k+2)}, \frac{c_\beta}{k^{1/3} \log^2(k+1)} \right). \tag{7}$$

5: Update the dual variable λ_{k+1} via $\lambda_{k+1} = \lambda_k - \beta_{k+1}(\mathcal{A}x_{k+1} - y_{k+1})$.

6: **end for**

Convergence analysis

Theorem 2

Let the sequence $\{x_k, y_k, \lambda_k\}_{k=1}^K$ *be generated by Algorithm 1. Let us denote* $\rho_k = c_\rho k^{1/3}$, $\tau_k = c_\tau k^{-1/3}$ *and* $\bar{\lambda}_k = \lambda_{k-1} - \rho_{k-1}(\mathcal{A}x_k - y_k)$. *Then we have that*

$$\min_{1\le k\le K} \|\mathcal{P}_{T_{x_k}\mathcal{M}}(-\mathcal{A}^*\bar{\lambda}_k) + \mathrm{grad}\, f(x_k)\| \le \frac{4\sqrt{2G}}{K^{1/3}},$$

$$\min_{1\le k\le K} \|\mathcal{A}x_k - y_k\| \le \frac{\ell_h + \lambda_{\max} + 2\sqrt{2G}\sigma_A \alpha c_\tau c_\rho}{c_\rho K^{1/3}},$$

$$\min_{1\le k\le K} dist(\bar{\lambda}_k, \partial h(y_k)) \le \frac{2\sqrt{2G}\sigma_A \alpha}{K^{1/3}}.$$

- Algorithm 1 is single-loop and the computational cost is cheap in each iteration.

Outline

1. Riemannian ADMM
2. Numerical results

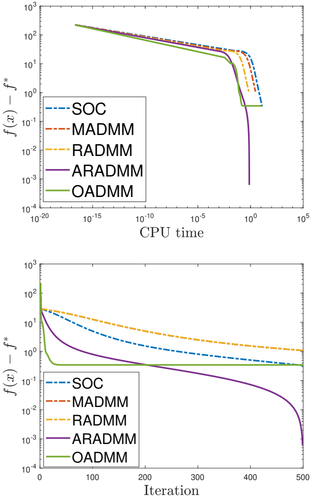
Sparse Principal Component Analysis (SPCA)

Given a data set $\{z_1, \ldots, z_m\} \in \mathbb{R}^{n \times m}$, a known formulation on the Stiefel manifold $\mathrm{St}(n, p) := \{X \in \mathbb{R}^{n \times p} : X^\top X = I_p\}$ of the SPCA problem is as follows:

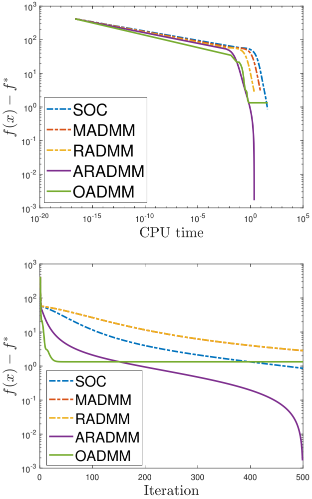
$$\min_X \|X^\top Z^\top Z X\|^2 + \mu \|X\|_1, \ \text{s.t. } X \in \mathrm{St}(n, p), \tag{8}$$

where $Z = [z_1, \ldots, z_m]^\top \in \mathbb{R}^{m \times n}$.

Comparison of SOC, MADMM, RADMM, OADMM and Algorithm 1



(a) $n = 400$, $p = 50$



(b) $n = 800$, $p = 100$

Comparison of RSG, AManPG and Algorithm 1

Settings	RSG [Li et al., 2021]		AManPG [Huang and Wei, 2022]		ARADMM	
(n, m, p)	Obj	CPU	Obj	CPU	Obj	CPU
(300, 20, 8)	-3.9517	0.7970	-4.2671	0.8555	**-4.4071**	**0.4264**
(400, 30, 10)	-5.5595	1.1105	-5.8129	1.0796	**-6.0378**	**0.5383**
(500, 40, 12)	-7.1168	1.9584	-7.3048	1.3820	**-7.6569**	**0.6888**
(600, 50, 14)	-7.2259	2.5911	-7.4608	1.7482	**-7.9029**	**0.8701**

Comparison of the ALM-type methods and Algorithm 1

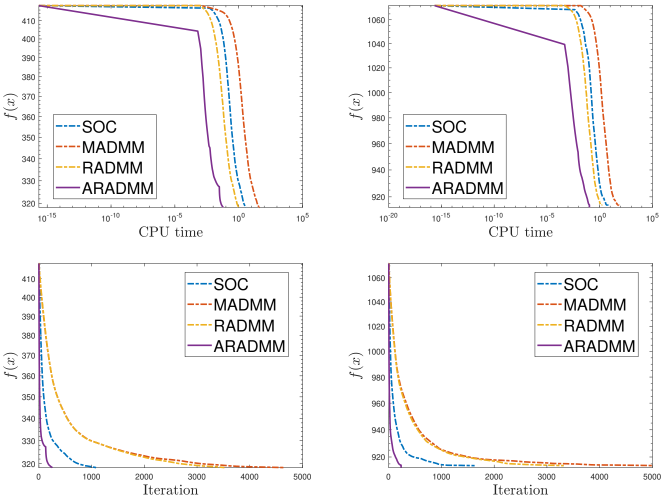
Settings	ALMSSN [Zhou et al., 2023]			ALMSRTR [Zhang et al., 2024]			ARADMM		
(n, m, p, μ)	Obj	CPU	Spa	Obj	CPU	Spa	Obj	CPU	Spa
(1500, 20, 8, 0.5)	6.0212	1.6349	93.10	4.0169	1.2679	98.38	**3.9552**	**0.9417**	**99.93**
(2000, 40, 10, 0.6)	8.9390	2.7063	94.03	5.9553	1.9281	99.85	**5.9190**	**1.3272**	**99.95**
(2500, 60, 12, 0.8)	**9.4527**	2.9387	99.61	9.5296	3.6644	99.95	9.4747	**1.8394**	**99.96**
(3000, 80, 15, 1)	15.7085	4.7114	94.65	**14.8418**	4.4791	99.96	14.8470	**2.2669**	**99.97**

Dual principal component pursuit (DPCP)

Given a dataset $Y = [\mathcal{X}, \mathcal{O}]\Gamma \in \mathbb{R}^{n\times(p_1+p_2)}$, where columns of $\mathcal{O} \in \mathbb{R}^{n\times p_2}$ represent outliers without a linear structure, columns of $\mathcal{X} \in \mathbb{R}^{n\times p_1}$ span a d-dimensional inlier subspace $\mathcal{S}$, and $\Gamma \in \mathbb{R}^{(p_1+p_2)\times(p_1+p_2)}$ is an unknown permutation matrix. The DPCP is a approach to recovery robust subspace $\mathcal{S}$, which amounts to solve the matrix optimization problem:

$$\min_{X\in\mathbb{R}^{n\times p}} F(X) := \|Y^\top X\|_1, \text{ s.t. } X^\top X = I_p. \tag{9}$$

Comparison of ADMM-type methods and Algorithm 1



(a) $p_1 = 100, p_2 = 600$ (b) $p_1 = 200, p_2 = 1600$

Beck, A. and Rosset, I. (2023).
A dynamic smoothing technique for a class of nonsmooth optimization problems on manifolds.
SIAM Journal on Optimization, 33(3):1473–1493.

Huang, W. and Wei, K. (2022).
An extension of fast iterative shrinkage-thresholding algorithm to riemannian optimization for sparse principal component analysis.
Numerical Linear Algebra with Applications, 29(1):e2409.

Kovnatsky, A., Glashoff, K., and Bronstein, M. M. (2016).
Madmm: a generic algorithm for non-smooth optimization on manifolds.
In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 680–696. Springer.

Lai, R. and Osher, S. (2014).
A splitting method for orthogonality constrained problems.
Journal of Scientific Computing, 58(2):431–449.

Li, J., Ma, S., and Srivastava, T. (2024).
A Riemannian alternating direction method of multipliers.
Mathematics of Operations Research, 0(0):1–21.

Li, X., Chen, S., Deng, Z., Qu, Q., Zhu, Z., and Man-Cho So, A. (2021).
Weakly convex optimization over Stiefel manifold using Riemannian subgradient-type methods.
SIAM Journal on Optimization, 31(3):1605–1634.

Peng, Z., Wu, W., Hu, J., and Deng, K. (2023).
Riemannian smoothing gradient type algorithms for nonsmooth optimization problem on compact Riemannian submanifold embedded in Euclidean space.
Applied Mathematics & Optimization, 88(3):1–32.

Yuan, G. (2024).
Admm for nonsmooth composite optimization under orthogonality constraints.
arXiv preprint arXiv:2405.15129.

Zhang, C., Xiao, R., Huang, W., and Jiang, R. (2024).

Riemannian trust region methods for sc 1 minimization.
Journal of Scientific Computing, 101(2):32.

Zhang, J., Ma, S., and Zhang, S. (2020).
Primal-dual optimization algorithms over Riemannian manifolds: an iteration complexity analysis.
Mathematical Programming, 184(1):445–490.

Zhou, Y., Bao, C., Ding, C., and Zhu, J. (2023).
A semismooth newton based augmented lagrangian method for nonsmooth optimization on matrix manifolds.
Mathematical Programming, 201(1):1–61.