Sparse and low-rank decomposition for big data systems via smoothed Riemannian optimization

Yuanming Shi
ShanghaiTech University,
Shanghai, China
shiym@shanghaitech.edu.cn

Bamdev Mishra
Amazon Development Centre India,
Bangalore, India
bamdevm@amazon.com

Abstract

We provide a unified modeling framework of sparse and low-rank decomposition to investigate the fundamental limits of communication, computation, and storage in mobile big data systems. The resulting sparse and low-rank optimization problems are highly intractable non-convex optimization problems and *conventional convex relaxation approaches are inapplicable*, for which we propose a smoothed Riemannian optimization approach. We propose novel regularized formulations that allow to exploit the Riemannian geometry of fixed-rank matrices and induce sparsity in matrices. Empirical results show the speedup, scalability, and superior performance against state-of-art algorithms across different problem instances.

1 Introduction

This paper considers the following sparse and low-rank decomposition optimization problem for mobile big data systems:

$$\mathscr{P}: \underset{\boldsymbol{X}\in\mathbb{R}^{K\times K}}{\mathsf{minimize}}\ \mathcal{L}(\boldsymbol{X})+\lambda\mathcal{R}(\boldsymbol{X}) \quad \mathsf{subject\ to}\ \ \mathrm{rank}(\boldsymbol{X})=r, \tag{1}$$

where $\mathcal{L}:\mathbb{R}^{K\times K}\to\mathbb{R}$ is a smooth *convex* loss function, $\mathcal{R}:\mathbb{R}^{K\times K}\to\mathbb{R}$ is a nonsmooth function, possibly nonconvex, and $\lambda\geq 0$ is the regularization parameter. The function $\mathcal{R}$ serves the purpose of inducing sparsity patterns in the solution of Problem $\mathscr{P}$. The fixed-rank constrained smooth optimization problem $\mathscr{P}$ commonly arises in machine learning, signal and data processing, and information theory. A non-exhaustive list of applications includes matrix completion [1], graph clustering [2], ranking [3], topological interference management [4], and index coding problems [5].

We motivate the sparse and low-rank formulation (1) in the context of investigating the fundamental limits of communication, computation, and storage in mobile big data systems. By pushing the computation and storage resource to the edge of networks, mobile big data system provides a disruptive technology to enable massive information communication, computation and storage via network densification [6, 7], mobile edge caching [8], and distributed computing [9]. Although sparse and low-rank decomposition has recently been intensively investigated for data and information processing, unique challenges arise in mobile big data systems in terms of numerical algorithms design and theoretical analysis, for which we provide the following three specific problems of interest.

Topological interference alignment. The topological interference alignment problem in the densely deployed partially connected K-user interference network is [4]

$$\mathscr{P}_1: \underset{\boldsymbol{X}\in\mathbb{R}^{K\times K}}{\mathsf{minimize}}\ \ \mathrm{rank}(\boldsymbol{X})$$

$$\mathsf{subject\ to}\ \ X_{ii}=1,\forall i=1,\ldots,K, \tag{2}$$

$$X_{ij}=0,\forall i\neq j,(i,j)\in\mathcal{V}, \tag{3}$$

9th NIPS Workshop on Optimization for Machine Learning (OPT 2016), Barcelona, Spain.

where $\mathcal{V}$ is the index set of connected transceiver pairs such that the channel coefficient from transmitter j to receiver i with $(i,j) \in \mathcal{V}$ is non-zero and zero otherwise. The interference alignment conditions (2) and (3) preserve the desired signal and cancel the interference, respectively. It should be noted that the sparsity in matrix $\boldsymbol{X}$ comes from the condition (3). The intuitive observation in the modeling formulation $\mathscr{P}_1$ is that the achievable symmetric degrees-of-freedom (DoF) equals the inverse of the rank of matrix $\boldsymbol{X}$ [4]. *The well-known nuclear norm relaxation approach [1] fails for $\mathscr{P}_1$ as it always returns the identity matrix as the optimal solution [4].* The details of the modeling framework $\mathscr{P}_1$ are in [4]. The framework provides a principled way to design communication-efficient schemes for mobile edge caching networks [8] and distributed computing systems [10, 11].

Network topology control. Given the DoF allocations, i.e., a fixed-rank matrix $\boldsymbol{X}$, the network topology control problem in the partially connected K-user interference network is [5]

$$\begin{aligned} \mathscr{P}_2 : \underset{\boldsymbol{X}\in\mathbb{R}^{K\times K}}{\mathsf{minimize}} \quad & \|\boldsymbol{X}\|_0 \\ \mathsf{subject\ to} \quad & X_{ii} = 1, \forall i = 1, \dots, K, \\ & \mathrm{rank}(\boldsymbol{X}) = r, \end{aligned}$$

where $\|\boldsymbol{X}\|_0$ is the count of non-zero entries in $\boldsymbol{X}$. It should be noted that $\|\boldsymbol{X}\|_0 = K^2 - |\mathcal{V}|$ represents the number of non-connected interference links. We, thus, aim at finding the network topologies with the maximum number of allowed connected interference links, while satisfying the DoF requirements (the rank constraint). However, due to the challenges of the ℓ_0 objective function and non-convex fixed-rank constraint, solving Problem $\mathscr{P}_2$ turns out to be highly intractable. *The widely used mixed ℓ_1-norm and nuclear norm convex penalty relaxation approach is inapplicable, as this approach always yields the identity matrix as the solution [5].* The details on the formulation are in [5]. This model also provides a novel way to minimize the storage overhead in caching networks [12, 8] and minimize the computation load in distributed computing systems [10, 11].

User admission control. The user admission control problem in the partially connected K-user interference network is [13]

$$\begin{aligned} \mathscr{P}_3 : \underset{\boldsymbol{X}\in\mathbb{R}^{K\times K}}{\mathsf{maximize}} \quad & \|\mathrm{diag}(\boldsymbol{X})\|_0 \\ \mathsf{subject\ to} \quad & X_{ij} = 0, \forall i \neq j, (i,j) \in \mathcal{V}, & (4) \\ & \mathrm{rank}(\boldsymbol{X}) = r, & (5) \end{aligned}$$

where $\mathrm{diag}(\cdot)$ extracts the diagonal entries of a matrix. Here, $\|\mathrm{diag}(\boldsymbol{X})\|_0$ equals the number of admitted users. Problem $\mathscr{P}_3$ aims at finding the maximal number of admitted users while satisfying the interference alignment condition (4) and DoF requirements (5). Due to the ℓ_0-norm *maximization* objective and non-convex fixed-rank constraint, Problem $\mathscr{P}_3$ reveals unique challenges. A simple ℓ_1-norm relaxation approach yields the objective in Problem $\mathscr{P}_3$ *unbounded* and *non-convex*. The details on the formulation $\mathscr{P}_3$ are in [13]. This model also provides a new perspective to maximize the user capacity in caching networks [12, 8] and distributed computing systems [9, 10, 11].

Overall, Problems $\mathscr{P}_1$, $\mathscr{P}_2$ and $\mathscr{P}_3$ exploit rank and sparsity constraints to characterize the conditions on the design parameters r, $\mathcal{V}$, and K for achieving the feasibility of topological interference alignment. Based on the relationships among the topological interference alignment [6], index coding [14], caching network [12] and distributed computing systems [10], we provide a unified approach to deal with such structured sparse and low-rank decomposition problems for mobile big data systems design to efficiently use the communication, computation and storage resources.

2 Smooth optimization approach

We propose to solve $\mathscr{P}_1$, $\mathscr{P}_2$, and $\mathscr{P}_3$ in two steps. In the first step, we find a good sparsity pattern by considering a *smoothed regularized* version of the problems as a smooth fixed-rank optimization problem. This is handled by using a smooth approximation $\mathcal{R}_\epsilon$ of $\mathcal{R}$, where ϵ is the smoothing parameter. In the second step, we refine the estimate obtained in the first step. The second step is equivalent to a certain fixed-rank matrix completion problem. Both the steps involve fixed-rank optimization of form Problem $\mathscr{P}$ for which we exploit the Riemannian structure of fixed-rank matrices [15, 16, 17]. Specifically, we use the *Riemannian trust-region* algorithm that is well implemented in the Matlab toolbox Manopt [18]. The Riemannian trust-region algorithm is globally convergent, i.e., it converges to a critical point starting from any random initialization [15, Chapter 7].

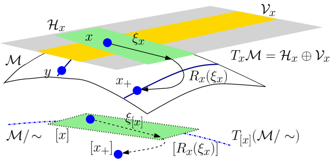


Figure 1: The Riemannian optimization machinery on a quotient manifold. The dashed lines represent abstract objects and the solid lines are their matrix representations. The points $x = (\boldsymbol{U}, \boldsymbol{V})$ and $y = (\boldsymbol{U}\boldsymbol{M}^{-1}, \boldsymbol{V}\boldsymbol{M}^T)$ in the total (computational) space $\mathcal{M}$ belong to the same equivalence class and they represent a single point $[x] = [(\boldsymbol{U}, \boldsymbol{V})] := \{(\boldsymbol{U}\boldsymbol{M}^{-1}, \boldsymbol{V}\boldsymbol{M}^T) : \boldsymbol{M} \in \mathrm{GL}(r)\}$ in the quotient space $\mathcal{M}/\sim$. An algorithm is implemented in $\mathcal{M} := \mathbb{R}^{K\times r} \times \mathbb{R}^{K\times r}$, but conceptually, the search is on the quotient manifold $\mathcal{M}/\mathrm{GL}(r)$. The search is only along the horizontal space $\mathcal{H}_x$ (a subspace of $T_x\mathcal{M}$) that is a mathematical representation of $T_{[x]}(\mathcal{M}/\sim)$. Given a search direction ξ_x along $\mathcal{H}_x$, the retraction mapping R_x maps it onto an element in $\mathcal{M}$ [15].

2.1 Proposed approach

To address the computational challenges in Problems $\mathscr{P}_1$, $\mathscr{P}_2$, and $\mathscr{P}_3$, we propose the following regularized formulations that are cast as $\mathscr{P}$. Below we show the corresponding objective functions.

- For Problem $\mathscr{P}_1$, we have $\mathcal{L}(\boldsymbol{X}) = \sum_i (X_{ii} - 1)^2 + \sum_{(i,j)\in\mathcal{V}} X_{ij}^2$ and $\lambda = 0$. This is solved by the Riemannian pursuit algorithm via alternatively performing fixed-rank optimization and rank updating [4].
- For Problem $\mathscr{P}_2$, we have $\mathcal{L}(\boldsymbol{X}) = \sum_i (X_{ii} - 1)^2$, $\mathcal{R}_\epsilon(\boldsymbol{X}) = \|\boldsymbol{X}\|_{1,\epsilon}$, and $\lambda \geq 0$. Here, $\|\boldsymbol{X}\|_{1,\epsilon} = \sum_{ij} (\boldsymbol{X}_i^2 + \epsilon^2)^{1/2}$ with $\epsilon \geq 0$ as a smoothing parameter [5]. Once the sparsity pattern is obtained, a subsequent matrix completion step refines the estimate.
- Problem $\mathscr{P}_3$, we have $\mathcal{L}(\boldsymbol{X}) = \sum_{(i,j)\in\mathcal{V}} X_{ij}^2$, $\mathcal{R}_\epsilon(\boldsymbol{X}) = \rho\|\mathrm{diag}(\boldsymbol{X})\|_2^2 - \|\mathrm{diag}(\boldsymbol{X})\|_{1,\epsilon}$ (which is *non-convex*), $\lambda \geq 0$, and $\rho \geq 0$. Here, ρ is a weighting parameter and the regularized term $\rho\|\mathrm{diag}(\boldsymbol{X})\|_2^2$ provides a novel way to bound the overall objective function [13], which allows to apply efficient manifold optimization algorithms. The dual way for bounding the non-convex objective is to add one additional constraint, e.g., the ℓ_1-norm constraint serves the purpose of bounding the non-convex objective in the high-dimensional regression problem [19]. Similar to the earlier case, once the sparsity pattern is obtained, we refine the estimate with a matrix completion step.

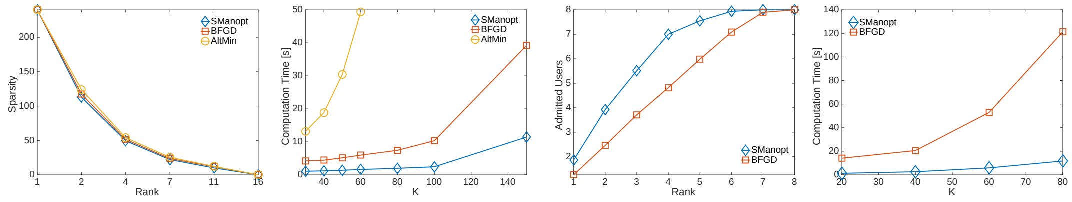
2.2 A matrix manifold optimization framework for $\mathscr{P}$

One popular approach to solve the smoothed version of Problem $\mathscr{P}$ is based on the following parameterization:

$$\underset{\boldsymbol{U}\in\mathbb{R}^{K\times r}, \boldsymbol{V}\in\mathbb{R}^{K\times r}}{\mathsf{minimize}} \quad f(\boldsymbol{U}\boldsymbol{V}^T) := \mathcal{L}(\boldsymbol{U}\boldsymbol{V}^T) + \lambda\mathcal{R}_\epsilon(\boldsymbol{U}\boldsymbol{V}^T), \tag{6}$$

where the rank-r matrix $\boldsymbol{X}$ is represented as $\boldsymbol{U}\boldsymbol{V}^T$ with $\boldsymbol{U} \in \mathbb{R}^{K\times r}$ and $\boldsymbol{V} \in \mathbb{R}^{K\times r}$. It should be noted that $\mathcal{R}_\epsilon$ is the smooth version of $\mathcal{R}$, which is discussed in Section 2.1 for different problems. With $r \ll K$, such a parameterization significantly reduces the number of optimization variables, thereby yielding low computational and memory costs. Although $f(\boldsymbol{U}\boldsymbol{V}^T)$ in (6) becomes a non-convex function w.r.t. both $\boldsymbol{U}$ and $\boldsymbol{V}$, there are exists recent non-convex algorithms operating on the $\boldsymbol{U}$ and $\boldsymbol{V}$ factors. For example, the Bi-Factored Gradient Descent (BFGD) algorithm in [20] can be adopted if f is smooth, while the alternating minimization (AltMin) algorithm [21] can be applied when f is convex in $\boldsymbol{U}$ and $\boldsymbol{V}$ individually. In this paper, we adopt a geometric approach that exploits non-uniqueness of the factorization $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{V}^T$ and allows joint update of $\boldsymbol{U}$ and $\boldsymbol{V}$.

It should be noted that $\boldsymbol{X}$ remains unchanged under the transformation of the factors $(\boldsymbol{U}, \boldsymbol{V}) \mapsto (\boldsymbol{U}\boldsymbol{M}^{-1}, \boldsymbol{V}\boldsymbol{M}^T)$ for all non-singular matrices $\boldsymbol{M} \in \mathrm{GL}(r)$, which is the set of $r \times r$ non-singular matrices. As a result, the critical points of an objective function parameterized with $\boldsymbol{U}$ and $\boldsymbol{V}$ are *not isolated* on $\mathbb{R}^{K\times r} \times \mathbb{R}^{K\times r}$. This issue is effectively resolved by considering the set of equivalence classes $[(\boldsymbol{U}, \boldsymbol{V})] := \{(\boldsymbol{U}\boldsymbol{M}^{-1}, \boldsymbol{V}\boldsymbol{M}^T) : \boldsymbol{M} \in \mathrm{GL}(r)\}$ as the search space instead of $\mathbb{R}^{K\times r} \times \mathbb{R}^{K\times r}$. In this search space, the critical points are isolated. Mathematically, our search



(a) Network control (lower is better). (b) Computation time. (c) User control (higher is better). (d) Computation time.

Figure 2: Performance of the proposed algorithms on $\mathscr{P}_2$ and $\mathscr{P}_3$.

space, i.e., the rank constraint $\mathrm{rank}(\boldsymbol{X})$, is the *quotient space* $\mathcal{M}/\sim$, where $\mathcal{M} := \mathbb{R}^{K\times r} \times \mathbb{R}^{K\times r}$ and $\sim$ is the equivalence relationship. The dimension of $\mathcal{M}/\sim$ is $nr + mr - r^2$.

The set of fixed-rank matrices has the structure of a Riemannian manifold [16, 17] that allows to propose a second order *trust-region* optimization algorithm in a systematic way. Particularly, we need the notion of "linearization" of the search space (tangent space), "search" direction computed from a local second order model (Riemannian gradient and Hessian), and a way "move" on a manifold (retraction). These notions are well defined on a Riemannian manifold. Figure 1 shows a schematic viewpoint of optimization. An additional requirement is the notion of an inner product well suited for regularized *least-squares* such as the ones in Section 2.1 [16]. Once the ingredients are in place, the toolbox Manopt allows a ready implementation of Riemannian trust-region algorithm for $\mathscr{P}$.

3 Experimental results

In this section, we compare our proposed smoothed Riemannian optimization (SManopt) algorithm with BFGD [20] and AltMin [21] on the regularized formulations (Section 2.1) of $\mathscr{P}_2$ and $\mathscr{P}_3$. All simulations are performed in Matlab on a 2.4 GHz octa-core Intel Xeon E5-2630 v3 machine (2 processors) with 64 GB RAM. The algorithms and problem instances are initialized randomly. The sets of $\mathcal{V}$ are generated uniformly at random. Each algorithm is stopped if the change in the objective value after every five iterations is less than δ_c. Numerical comparisons show that proposed SManopt outperforms state-of-art algorithms both in quality of solutions and numerical tractability.

Network topology control problem $\mathscr{P}_2$. We set ϵ to a high value of 0.01 and $\delta_c = 10^{-4}$. A good choice of λ is 0.01, which is obtained by cross validation. The maximum number of iterations of SManopt, AltMin, and BFGD are set to 500, 500, and 10^5, respectively. Figure 2 (a) demonstrates the sparsity and low-rankness tradeoff in $\boldsymbol{X} \in \mathbb{R}^{K\times K}$ with $K = 16$, followed by the time results illustrated in Figure 2 (b) with the fixed-rank $r = 10$. Each point in the simulations is averaged over 200 randomly generated initial points for each algorithm.

User admission control problem $\mathscr{P}_3$. We set $\rho = 0.5$, $\lambda = 0.02$, $\epsilon = 0.001$, and $\delta_c = 10^{-11}$ because of the extremely low convergence rate of BFGD in this scenario. Figure 2 (c) demonstrates the average number of admitted users with different rank allocations of matrix $\boldsymbol{X}$ with $K = 8$, followed by the time results shown in Figure 2 (d) with the fixed rank $r = 8$. Each point in the simulations is averaged over 100 randomly generated network topology realizations $\mathcal{V}$. It should be noted that we only compare with BFGD. As the objective function in $\mathscr{P}$ is non-convex for Problem $\mathscr{P}_3$, AltMin is not directly applicable to this problem.

4 Conclusions and future work

In this paper, we have presented a unified sparse and low-rank decomposition approach for problems arising in mobile big data systems, for which the conventional convex relaxation approaches (e.g., ℓ_1-norm and nuclear-norm relaxations) fail. We propose a regularization approach to deal with challenging constraints that boils down to a smooth fixed-rank optimization problem. This allows to exploit the Riemannian structure of fixed-rank matrices. Specifically, we exploit the Riemannian trust-region algorithm. Numerical comparisons demonstrate the significant performance improvement of the proposed algorithm. As a future research direction, we intend to establish optimality of the proposed algorithm for the problems on the lines of [22].

References

[1] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, pp. 717–772, Apr. 2009.

[2] Y. Chen, A. Jalali, S. Sanghavi, and H. Xu, "Clustering partially observed graphs via convex optimization.," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 2213–2238, 2014.

[3] D. Park, J. Neeman, J. Zhang, S. Sanghavi, and I. Dhillon, "Preference completion: Large-scale collaborative ranking from pairwise comparisons," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2015.

[4] Y. Shi, J. Zhang, and K. B. Letaief, "Low-rank matrix completion for topological interference management by Riemannian pursuit," *IEEE Trans. Wireless Commun.*, vol. 15, pp. 1–15, Jul. 2016.

[5] Y. Shi and B. Mishra, "A sparse and low-rank optimization framework for index coding via Riemannian optimization," tech. rep., arXiv preprint arXiv:1604.04325, 2016.

[6] S. Jafar, "Topological interference management through index coding," *IEEE Trans. Inf. Theory*, vol. 60, pp. 529–568, Jan. 2014.

[7] Y. Shi, J. Zhang, K. Letaief, B. Bai, and W. Chen, "Large-scale convex optimization for ultra-dense Cloud-RAN," *IEEE Wireless Commun. Mag.*, vol. 22, pp. 84–91, Jun. 2015.

[8] K. Yang, Y. Shi, and Z. Ding, "Low-rank matrix completion for mobile edge caching in fog-ran via riemannian optimization," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, (Washington, DC, USA), Dec. 2016.

[9] S. Li, Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "A scalable framework for wireless distributed computing," *arXiv:1608.05743*, 2016.

[10] S. Li, M. A. Maddah-Ali, Q. Yu, and A. S. Avestimehr, "A fundamental tradeoff between computation and communication in distributed computing," *arXiv preprint arXiv:1604.07086*, 2016.

[11] K. Lee, M. Lam, R. Pedarsani, D. Papailiopoulos, and K. Ramchandran, "Speeding up distributed machine learning using codes," *arXiv preprint arXiv:1512.02673*, 2015.

[12] M. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, pp. 2856–2867, May 2014.

[13] Y. Shi and B. Mishra, "Topological interference management with user admission control via Riemannian optimization," *arXiv preprint arXiv:1607.07252*, 2016.

[14] H. Maleki, V. Cadambe, and S. Jafar, "Index coding-An interference alignment perspective," *IEEE Trans. Inf. Theory*, vol. 60, pp. 5402–5432, Sep. 2014.

[15] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.

[16] B. Mishra, K. Adithya Apuroop, and R. Sepulchre, "A Riemannian geometry for low-rank matrix completion," tech. rep., arXiv preprint arXiv:1211.1550, 2012.

[17] B. Mishra, G. Meyer, S. Bonnabel, and R. Sepulchre, "Fixed-rank matrix factorizations and Riemannian low-rank optimization," *Comput. Statist.*, vol. 29, no. 3-4, pp. 591–621, 2014.

[18] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, "Manopt, a Matlab toolbox for optimization on manifolds," *J. Mach. Learn. Res.*, vol. 15, pp. 1455–1459, 2014.

[19] P.-L. Loh and M. J. Wainwright, "High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity," *Ann. Statist.*, vol. 40, pp. 1637–1664, Mar. 2012.

[20] D. Park, A. Kyrillidis, C. Caramanis, and S. Sanghavi, "Finding low-rank solutions to matrix problems, efficiently and provably," *arXiv preprint arXiv:1606.03168*, 2016.

[21] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *ACM Symp. Theory Comput.*, pp. 665–674, ACM, 2013.

[22] N. Boumal, V. Voroninski, and A. S. Bandeira, "The non-convex Burer-Monteiro approach works on smooth semidefinite programs," *arXiv preprint arXiv:1606.04970*, 2016.