Riemannian preconditioning for tensor completion

Hiroyuki Kasai
The University of Electro-Communications
Tokyo, Japan
kasai@is.uec.ac.jp

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

Abstract

We propose a novel Riemannian preconditioning approach for the tensor completion problem with rank constraint. A Riemannian metric or inner product is proposed that exploits the least-squares structure of the cost function and takes into account the structured symmetry in Tucker decomposition. The specific metric allows to use the versatile framework of Riemannian optimization on quotient manifolds to develop a preconditioned nonlinear conjugate gradient algorithm for the problem. Numerical comparisons suggest that our proposed algorithm robustly outperforms state-of-the-art algorithms across different problem instances encompassing various synthetic and real-world datasets.

1 Introduction

This paper addresses the problem of low-rank tensor completion when the rank is a priori known or estimated. Without loss of generality, we focus on 3-order tensors. Given a tensor $\boldsymbol{\mathcal{X}}^{n_1 \times n_2 \times n_3}$, whose entries $\boldsymbol{\mathcal{X}}^{\star}_{i_1,i_2,i_3}$ are only known for some indices $(i_1, i_2, i_3) \in \Omega$, where Ω is a subset of the complete set of indices $\{(i_1, i_2, i_3) : i_d \in \{1, \dots, n_d\}, d \in \{1, 2, 3\}\}$, the *fixed-rank tensor completion problem* is formulated as

$$\min_{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}} \frac{1}{|\Omega|} \|\boldsymbol{\mathcal{P}}_{\Omega}(\boldsymbol{\mathcal{X}}) - \boldsymbol{\mathcal{P}}_{\Omega}(\boldsymbol{\mathcal{X}}^{\star})\|_F^2 \quad \text{subject to } \operatorname{rank}(\boldsymbol{\mathcal{X}}) = \mathbf{r}, \tag{1}$$

where the operator $\boldsymbol{\mathcal{P}}_{\Omega}(\boldsymbol{\mathcal{X}})_{i_1 i_2 i_3} = \boldsymbol{\mathcal{X}}_{i_1 i_2 i_3}$ if $(i_1, i_2, i_3) \in \Omega$ and $\boldsymbol{\mathcal{P}}_{\Omega}(\boldsymbol{\mathcal{X}})_{i_1 i_2 i_3} = 0$ otherwise and (with a slight abuse of notation) $\|\cdot\|_F$ is the Frobenius norm. $\operatorname{rank}(\boldsymbol{\mathcal{X}})$ $(= \mathbf{r} = (r_1, r_2, r_3))$, called the *multilinear rank* of $\boldsymbol{\mathcal{X}}$, is the set of the ranks of for each of mode-d unfolding matrices. $r_d \ll n_d$ enforces a low-rank structure. The *mode* is a matrix obtained by concatenating the mode-d fibers along column and mode-d *unfolding* of $\boldsymbol{\mathcal{X}}$ is $\mathbf{X}_d \in \mathbb{R}^{n_d \times n_{d+1} \cdots n_D n_1 \cdots n_{d-1}}$ for $d = \{1, \dots, D\}$.

The optimization problem (1) has many variants, and one of those is extending the nuclear norm regularization approach from the matrix case [1] to the tensor case. While this generalization leads to good results [2, 3, 4], its scalabilityto large-scale instances is not trivial, especially due to the necessity of high-dimensional singular value decomposition computations. A different approach exploits *Tucker decomposition* [5, Section 4] of a low-rank tensor $\boldsymbol{\mathcal{X}}$ to develop large-scale algorithms for (1), e.g., in [6, 7]. The present paper exploits both the *symmetry* present in Tucker decomposition and the *least-squares* structure of the cost function of (1) by using the concept of *preconditioning*. While preconditioning in unconstrained optimization is well studied [8, Chapter 5], preconditioning on constraints with *symmetries*, owing to non-uniqueness of Tucker decomposition [5, Section 4.3], is not straightforward. We build upon the recent work [9] that suggests to use *Riemannian preconditioning* with a *tailored metric* (inner product) in the Riemannian optimization framework on quotient manifolds [10, 11, 12]. Our proposed preconditioned nonlinear conjugate gradient algorithm is implemented in the Matlab toolbox Manopt [13] and it outperforms state-of-the-art methods. The codes are available at `http://bamdevmishra.com/codes/tensorcompletion/`.

2 Exploiting the problem structure

We focus on the two fundamental structures present in (1): *symmetry* in the constraints, and the *least-squares structure* of the cost function. Finally, a novel metric is proposed.

The quotient and least-squares structures. The Tucker decomposition of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ of rank $\mathbf{r}$ (=(r_1, r_2, r_3)) is [5, Section 4.1] $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{G}} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3$, where $\mathbf{U}_d \in \mathrm{St}(r_d, n_d)$ for $d \in \{1, 2, 3\}$ belongs to the *Stiefel manifold* of matrices of size $n_d \times r_d$ with orthogonal columns and $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$. Here, $\boldsymbol{\mathcal{W}} \times_d \mathbf{V} \in \mathbb{R}^{n_1 \times \cdots n_{d-1} \times m \times n_{d+1} \times \cdots n_N}$ computes the *d-mode product* of a tensor $\boldsymbol{\mathcal{W}} \in \mathbb{R}^{n_1 \times \cdots \times n_N}$ and a matrix $\mathbf{V} \in \mathbb{R}^{m \times n_d}$. Tucker decomposition is *not unique* as $\boldsymbol{\mathcal{X}}$ remains unchanged under the transformation $(\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3, \boldsymbol{\mathcal{G}}) \mapsto (\mathbf{U}_1\mathbf{O}_1, \mathbf{U}_2\mathbf{O}_2, \mathbf{U}_3\mathbf{O}_3, \boldsymbol{\mathcal{G}} \times_1 \mathbf{O}_1^T \times_2 \mathbf{O}_2^T \times_3 \mathbf{O}_3^T)$ for all $\mathbf{O}_d \in \mathcal{O}(r_d)$, which is the set of orthogonal matrices of size of $r_d \times r_d$. The classical remedy to remove this indeterminacy is to have additional structures on $\boldsymbol{\mathcal{G}}$ like sparsity or restricted orthogonal rotations [5, Section 4.3]. In contrast, we encode the transformation in an abstract search space of *equivalence classes*, defined as, $[(\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3, \boldsymbol{\mathcal{G}})] := \{(\mathbf{U}_1\mathbf{O}_1, \mathbf{U}_2\mathbf{O}_2, \mathbf{U}_3\mathbf{O}_3, \boldsymbol{\mathcal{G}} \times_1 \mathbf{O}_1^T \times_2 \mathbf{O}_2^T \times_3 \mathbf{O}_3^T) : \mathbf{O}_d \in \mathcal{O}(r_d)\}$. The set of equivalence classes is the quotient manifold [14, Theorem 9.16]

$$\mathcal{M}/\sim \quad := \quad \mathcal{M}/(\mathcal{O}(r_1) \times \mathcal{O}(r_2) \times \mathcal{O}(r_3)), \tag{2}$$

where $\mathcal{M}$ is called the *total space* (computational space) that is the product space $\mathcal{M} := \mathrm{St}(r_1, n_1) \times \mathrm{St}(r_2, n_2) \times \mathrm{St}(r_3, n_3) \times \mathbb{R}^{r_1 \times r_2 \times r_3}$. Due to the invariance of the Tucker decomposition, the local minima of (1) in $\mathcal{M}$ are not isolated, but they become isolated on $\mathcal{M}/\sim$. Consequently, the problem (1) is an optimization problem on a quotient manifold for which systematic procedures are proposed in [10, 11, 12] by endowing $\mathcal{M}/\sim$ with a Riemannian structure. We call $\mathcal{M}/\sim$ the *Tucker manifold*.

Another structure that is present in (1) is the least-squares structure of the cost function. A way to exploit it is to endow the search space with a metric (inner product) induced by the Hessian of the cost function [8]. This induced metric (or its approximation) resolves convergence issues of first-order optimization algorithms. Specifically for the case of quadratic optimization with rank constraint (matrix case), Mishra and Sepulchre [9, Section 5] propose a family of Riemannian metrics from the Hessian of the cost function. Since applying this approach directly for (1) is computationally costly, we consider a simplified cost function by assuming that Ω contains the full set of indices, i.e., we focus on $\|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{X}}^\star\|_F^2$ to propose a metric candidate. A good candidate is by considering only the *block diagonal* elements of the Hessian of $\|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{X}}^\star\|_F^2$. It should emphasized that the cost function $\|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{X}}^\star\|_F^2$ is *convex and quadratic* in $\boldsymbol{\mathcal{X}}$. Consequently, it is also convex and quadratic in the arguments $(\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3, \boldsymbol{\mathcal{G}})$ individually. The block diagonal approximation of the Hessian of $\|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{X}}^\star\|_F^2$ in $(\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3, \boldsymbol{\mathcal{G}})$ is $((\mathbf{G}_1\mathbf{G}_1^T) \otimes \mathbf{I}_{n_1}, (\mathbf{G}_2\mathbf{G}_2^T) \otimes \mathbf{I}_{n_2}, (\mathbf{G}_3\mathbf{G}_3^T) \otimes \mathbf{I}_{n_3}, \mathbf{I}_{r_1 r_2 r_3})$, where $\mathbf{G}_d$ is the mode-d unfolding of $\boldsymbol{\mathcal{G}}$ and is assumed to be full rank. The terms $\mathbf{G}_d\mathbf{G}_d^T$ for $d \in \{1, 2, 3\}$ are *positive definite* when $r_1 \leq r_2 r_3$, $r_2 \leq r_1 r_3$, and $r_3 \leq r_1 r_2$.

A novel Riemannian metric and its motivation. An element x in the total space $\mathcal{M}$ has the matrix representation $(\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3, \boldsymbol{\mathcal{G}})$. Consequently, the tangent space $T_x\mathcal{M}$ is the Cartesian product of the tangent spaces of the individual manifolds, i.e., $T_x\mathcal{M}$ has the matrix characterization [12] $T_x\mathcal{M} = \{(\mathbf{Z}_{\mathbf{U}_1}, \mathbf{Z}_{\mathbf{U}_2}, \mathbf{Z}_{\mathbf{U}_3}, \mathbf{Z}_{\boldsymbol{\mathcal{G}}}) \in \mathbb{R}^{n_1 \times r_1} \times \mathbb{R}^{n_2 \times r_2} \times \mathbb{R}^{n_3 \times r_3} \times \mathbb{R}^{r_1 \times r_2 \times r_3} : \mathbf{U}_d^T\mathbf{Z}_{\mathbf{U}_d} + \mathbf{Z}_{\mathbf{U}_d}^T\mathbf{U}_d = 0, \text{ for } d \in \{1, 2, 3\}\}$. The earlier discussion on symmetry and least-squares structure leads to the novel metric $g_x : T_x\mathcal{M} \times T_x\mathcal{M} \to \mathbb{R}$

$$g_x(\xi_x, \eta_x) = \langle \xi_{\mathbf{U}_1}, \eta_{\mathbf{U}_1}(\mathbf{G}_1\mathbf{G}_1^T) \rangle + \langle \xi_{\mathbf{U}_2}, \eta_{\mathbf{U}_2}(\mathbf{G}_2\mathbf{G}_2^T) \rangle + \langle \xi_{\mathbf{U}_3}, \eta_{\mathbf{U}_3}(\mathbf{G}_3\mathbf{G}_3^T) \rangle + \langle \xi_{\boldsymbol{\mathcal{G}}}, \eta_{\boldsymbol{\mathcal{G}}} \rangle, \tag{3}$$

where $\xi_x, \eta_x \in T_x\mathcal{M}$ are tangent vectors with matrix characterizations, $(\xi_{\mathbf{U}_1}, \xi_{\mathbf{U}_2}, \xi_{\mathbf{U}_3}, \xi_{\boldsymbol{\mathcal{G}}})$ and $(\eta_{\mathbf{U}_1}, \eta_{\mathbf{U}_2}, \eta_{\mathbf{U}_3}, \eta_{\boldsymbol{\mathcal{G}}})$, respectively and $\langle \cdot, \cdot \rangle$ is the Euclidean inner product. As contrasts to the classical Euclidean metric, the metric (3) *scales* the level sets of the cost function on the search space that leads a preconditioning effect on the algorithms developed on the Tucker manifold.

3 Notions of optimization on quotient manifolds

Each point on a quotient manifold represents an entire equivalence class of matrices in the total space. Abstract geometric objects on a quotient manifold call for matrix representatives in the total space. Similarly, algorithms are run in the total space $\mathcal{M}$, but under appropriate compatibility

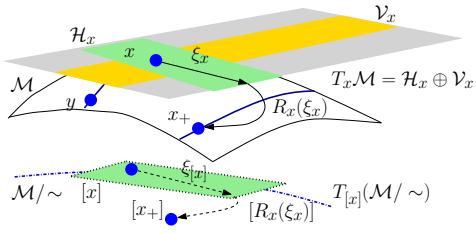


Figure 1: Riemannian optimization framework.

between the Riemannian structure of $\mathcal{M}$ and the Riemannian structure of the quotient manifold $\mathcal{M}/\sim$, they define algorithms on the quotient manifold. Once we endow $\mathcal{M}/\sim$ with a Riemannian structure, the constraint optimization problem (1) is conceptually transformed into an unconstrained optimization over the Riemannian quotient manifold (2). Figure 1 illustrates a schematic view of optimization with equivalence classes, where the points x and y in $\mathcal{M}$ belong to the same equivalence class (shown in solid blue color) and they represent a single point $[x] := \{y \in \mathcal{M} : y \sim x\}$ on the quotient manifold $\mathcal{M}/\sim$. The abstract tangent space $T_{[x]}(\mathcal{M}/\sim)$ at $[x] \in \mathcal{M}/\sim$ has the matrix representation in $T_x\mathcal{M}$, but restricted to the directions that do not induce a displacement along the equivalence class $[x]$. This is realized by decomposing $T_x\mathcal{M}$ into two complementary subspaces. The vertical space $\mathcal{V}_x$ is the tangent space of the equivalence class $[x]$. On the other hand, the horizontal space $\mathcal{H}_x$ is the *orthogonal subspace* to $\mathcal{V}_x$, i.e., $T_x\mathcal{M} = \mathcal{V}_x \oplus \mathcal{H}_x$. The horizontal subspace provides a valid matrix representation to the abstract tangent space $T_{[x]}(\mathcal{M}/\sim)$ [10, Section 3.5.8]. An abstract tangent vector $\xi_{[x]} \in T_{[x]}(\mathcal{M}/\sim)$ at $[x]$ has a unique element $\xi_x \in \mathcal{H}_x$ that is called its *horizontal lift*. Endowed with the Riemannian metric (3), the quotient manifold $\mathcal{M}/\sim$ is a *Riemannian submersion* of $\mathcal{M}$. The submersion principle then allows to work out concrete matrix representations of abstract object on $\mathcal{M}/\sim$. Particularly, starting from an arbitrary matrix (with appropriate dimensions), two linear projections are needed: the first projection Ψ_x is onto the tangent space $T_x\mathcal{M}$, while the second projection Π_x is onto the horizontal subspace $\mathcal{H}_x$. Their formulas are shown in Table 1. The computation cost of these projections is $O(n_1r_1^2 + n_2r_2^2 + n_3r_3^2)$.

Finally, we propose a Riemannian nonlinear conjugate gradient algorithm for (1) that scales well to large-scale instances. Specifically, we use the conjugate gradient implementation of Manopt with the ingredients described in Table 1. The convergence analysis of this method follows from [15, 16, 10]. If $f(\boldsymbol{\mathcal{X}}) = \|\boldsymbol{\mathcal{P}}_\Omega(\boldsymbol{\mathcal{X}}) - \boldsymbol{\mathcal{P}}_\Omega(\boldsymbol{\mathcal{X}}^\star)\|_F^2/|\Omega|$, then the Riemannian gradient $\mathrm{grad}_x f$, which has the matrix characterization $\Psi(\mathrm{egrad}_x f)$, where $\mathrm{egrad}_x f$ is the Euclidean gradient of f. We also show a way to compute a step-size guess effectively. The total computational cost per iteration of our proposed algorithm is $O(|\Omega|r_1r_2r_3)$, where $|\Omega|$ is the number of known entries.

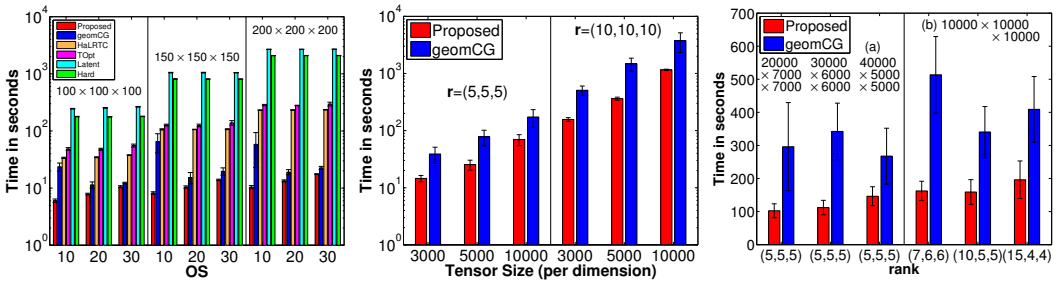
Table 1: Ingredients to implement an off-the-shelf conjugate gradient algorithm for (1).

Vertical tangent vectors in $\mathcal{V}_x$	$\{(\mathbf{U}_1\boldsymbol{\Omega}_1, \mathbf{U}_2\boldsymbol{\Omega}_2, \mathbf{U}_3\boldsymbol{\Omega}_3, -(\boldsymbol{\mathcal{G}}\times_1\boldsymbol{\Omega}_1 + \boldsymbol{\mathcal{G}}\times_2\boldsymbol{\Omega}_2 + \boldsymbol{\mathcal{G}}\times_3\boldsymbol{\Omega}_3)) :$ $\boldsymbol{\Omega}_d \in \mathbb{R}^{r_d \times r_d}, \boldsymbol{\Omega}_d^T = -\boldsymbol{\Omega}_d$, for $d \in \{1,2,3\}\}$
Horizontal tangent vectors in $\mathcal{H}_x$	$\{(\zeta_{\mathbf{U}_1}, \zeta_{\mathbf{U}_2}, \zeta_{\mathbf{U}_3}, \zeta_{\boldsymbol{\mathcal{G}}}) \in T_x\mathcal{M} :$ $(\mathbf{G}_d\mathbf{G}_d^T)\zeta_{\mathbf{U}_d}^T\mathbf{U}_d + \zeta_{\mathbf{G}_d}\mathbf{G}_d^T$ is symmetric, for $d \in \{1,2,3\}\}$
$\Psi(\cdot)$ projects an ambient vector $(\mathbf{Y}_{\mathbf{U}_1}, \mathbf{Y}_{\mathbf{U}_2}, \mathbf{Y}_{\mathbf{U}_3}, \mathbf{Y}_{\boldsymbol{\mathcal{G}}})$ onto $T_x\mathcal{M}$	$(\mathbf{Y}_{\mathbf{U}_1} - \mathbf{U}_1\mathbf{S}_{\mathbf{U}_1}(\mathbf{G}_1\mathbf{G}_1^T)^{-1}, \mathbf{Y}_{\mathbf{U}_2} - \mathbf{U}_2\mathbf{S}_{\mathbf{U}_2}(\mathbf{G}_2\mathbf{G}_2^T)^{-1},$ $\mathbf{Y}_{\mathbf{U}_3} - \mathbf{U}_3\mathbf{S}_{\mathbf{U}_3}(\mathbf{G}_3\mathbf{G}_3^T)^{-1}, \mathbf{Y}_{\boldsymbol{\mathcal{G}}})$, where $\mathbf{S}_{\mathbf{U}_d}$ for $d \in \{1,2,3\}$ are solutions to $\mathbf{S}_{\mathbf{U}_d}\mathbf{G}_d\mathbf{G}_d^T + \mathbf{G}_d\mathbf{G}_d^T\mathbf{S}_{\mathbf{U}_d} = \mathbf{G}_d\mathbf{G}_d^T(\mathbf{Y}_{\mathbf{U}_d}^T\mathbf{U}_d + \mathbf{U}_d^T\mathbf{Y}_{\mathbf{U}_d})\mathbf{G}_d\mathbf{G}_d^T$
$\Pi(\cdot)$ projects a tangent vector ξ onto $\mathcal{H}_x$	$(\xi_{\mathbf{U}_1} - \mathbf{U}_1\boldsymbol{\Omega}_1, \xi_{\mathbf{U}_2} - \mathbf{U}_2\boldsymbol{\Omega}_2, \xi_{\mathbf{U}_3} - \mathbf{U}_3\boldsymbol{\Omega}_3,$ $\xi_{\boldsymbol{\mathcal{G}}} - (-(\boldsymbol{\mathcal{G}}\times_1\boldsymbol{\Omega}_1 + \boldsymbol{\mathcal{G}}\times_2\boldsymbol{\Omega}_2 + \boldsymbol{\mathcal{G}}\times_3\boldsymbol{\Omega}_3)))$, where $\boldsymbol{\Omega}_d$ are solutions to particular *coupled* Lyapunov equations.
$\mathrm{egrad}_x f$	$(\mathbf{S}_1(\mathbf{U}_3 \otimes \mathbf{U}_2)\mathbf{G}_1^T(\mathbf{G}_1\mathbf{G}_1^T)^{-1}, \mathbf{S}_2(\mathbf{U}_3 \otimes \mathbf{U}_1)\mathbf{G}_2^T(\mathbf{G}_2\mathbf{G}_2^T)^{-1},$ $\mathbf{S}_3(\mathbf{U}_2 \otimes \mathbf{U}_1)\mathbf{G}_3^T(\mathbf{G}_3\mathbf{G}_3^T)^{-1}, \boldsymbol{\mathcal{S}} \times_1 \mathbf{U}_1^T \times_2 \mathbf{U}_2^T \times_3 \mathbf{U}_3^T) \times_3 \mathbf{U}_3^T)$, where $\boldsymbol{\mathcal{S}} = \frac{2}{\lvert\Omega\rvert}(\boldsymbol{\mathcal{P}}_\Omega(\boldsymbol{\mathcal{G}}\times_1\mathbf{U}_1\times_2\mathbf{U}_2\times_3\mathbf{U}_3) - \boldsymbol{\mathcal{P}}_\Omega(\boldsymbol{\mathcal{X}}^\star))$.

4 Numerical comparisons

We show numerical comparisons of our proposed algorithm with state-of-the-art algorithms that include TOpt [6] and geomCG [7], for comparisons with Tucker decomposition based algorithms, and

HaLRTC [2], Latent [3], and Hard [4] as nuclear norm minimization algorithms. All simulations are performed in Matlab on a 2.6 GHz Intel Core i7 machine with 16 GB RAM. For specific operations with unfoldings of $\boldsymbol{\mathcal{S}}$, we use the `mex` interfaces that are provided in geomCG. For large-scale instances, our algorithm is only compared with geomCG as other algorithms cannot handle these instances. We randomly and uniformly select known entries based on a multiple of the dimension, called the *over-sampling* (OS) ratio, to create the training set Ω. Algorithms (and problem instances) are initialized randomly, as in [7], and are stopped when either the mean square error (MSE) on the training set Ω is below 10^{-12} or the number of iterations exceeds 250. We also evaluate the mean square error on a test set Γ, which is different from Ω. Five runs are performed in each scenario.



(a) **Case 1:** small-scale tensors. (b) **Case 2:** large-scale tensors. (c) **Case 3:** asymmetric tensors.

Figure 2: Experiments on synthetic datasets.

Case 1 considers synthetic small-scale tensors of size $100 \times 100 \times 100$, $150 \times 150 \times 150$, and $200 \times 200 \times 200$ and rank $\mathbf{r} = (10, 10, 10)$ are considered. OS is $\{10, 20, 30\}$. Figure 2(a) shows that the convergence behavior of our proposed algorithm is either competitive or faster than the others. Next, **Case 2** considers large-scale tensors of size $3000 \times 3000 \times 3000$, $5000 \times 5000 \times 5000$, and $10000 \times 10000 \times 10000$ and ranks $\mathbf{r} = (5, 5, 5)$ and $(10, 10, 10)$. OS is 10. Our proposed algorithm outperforms geomCG in Figure 2(b). **Case 3** considers instances where the dimensions and ranks along certain modes are different than others. Two cases are considered. Case (3.a) considers tensors size $20000 \times 7000 \times 7000$, $30000 \times 6000 \times 6000$, and $40000 \times 5000 \times 5000$ with rank $\mathbf{r} = (5, 5, 5)$. Case (3.b) considers a tensor of size $10000 \times 10000 \times 10000$ with ranks $(7, 6, 6)$, $(10, 5, 5)$, and $(15, 4, 4)$. In all the cases, the proposed algorithm converges faster than geomCG as shown in Figure 2(c). Finally, **Case 4** considers MovieLens-10M dataset that contains 10000054 ratings corresponding to 71567 users and 10681 movies. We split the time into 7-days wide bins results, and finally, get a tensor of size $71567 \times 10681 \times 731$. The fraction of known entries is less than 0.002%. We perform five random 80/10/10–train/validation/test partitions. The maximum iteration is set to 500. As shown in Table 2, our proposed algorithm consistently gives lower test errors than geomCG across different ranks.

Table 2: **Case 4:** test MSE on Γ and time in seconds.

MovieLens-10M	Proposed		geomCG	
r	Time	MSE on Γ	Time	MSE on Γ
$(4, 4, 4)$	$\mathbf{1748 \pm 441}$	$\mathbf{0.6762 \pm 1.5 \cdot 10^{-3}}$	2981 ± 40	$0.6956 \pm 2.8 \cdot 10^{-3}$
$(6, 6, 6)$	$\mathbf{6058 \pm 47}$	$\mathbf{0.6913 \pm 3.3 \cdot 10^{-3}}$	6554 ± 655	$0.7398 \pm 7.1 \cdot 10^{-3}$
$(8, 8, 8)$	$\mathbf{11370 \pm 103}$	$\mathbf{0.7589 \pm 7.1 \cdot 10^{-3}}$	13853 ± 118	$0.8955 \pm 3.3 \cdot 10^{-2}$
$(10, 10, 10)$	$\mathbf{32802 \pm 52}$	$\mathbf{1.0107 \pm 2.7 \cdot 10^{-2}}$	38145 ± 36	$1.6550 \pm 8.7 \cdot 10^{-2}$

5 Conclusion and future work

We have proposed a preconditioned nonlinear conjugate gradient algorithm for the tensor completion problem by exploiting the fundamental structures of symmetry, due to non-uniqueness of Tucker decomposition, and least-squares of the cost function. A novel Riemannian metric is proposed that enables to use the versatile Riemannian optimization framework. Numerical comparisons suggest that our proposed algorithm has a superior performance on different benchmarks.

Acknowledgments

We thank Rodolphe Sepulchre, Paul Van Dooren, and Nicolas Boumal for useful discussions on the paper. Hiroyuki Kasai is (partly) supported by the Ministry of Internal Affairs and Communications, Japan, as the SCOPE Project (150201002). This work was initiated while Bamdev Mishra was with the Department of Electrical Engineering and Computer Science, University of Liège, 4000 Liège, Belgium and was visiting the Department of Engineering (Control Group), University of Cambridge, Cambridge, UK. He was supported as a research fellow (aspirant) of the Belgian National Fund for Scientific Research (FNRS).

References

[1] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Found. Comput. Math.*, 9(6):717–772, 2009.

[2] J. Liu, P. Musialski, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(1):208–220, 2013.

[3] R. Tomioka, K. Hayashi, and H. Kashima. Estimation of low-rank tensors via convex optimization. Technical report, arXiv:1010.0789, 2011.

[4] M. Signoretto, Q. T. Dinh, L. D. Lathauwer, and J. A. K. Suykens. Learning with tensors: a framework based on convex optimization and spectral regularization. *Mach. Learn.*, 94(3):303–351, 2014.

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

[6] M. Filipović and A. Jukić. Tucker factorization with missing data with application to low-n-rank tensor completion. *Multidim. Syst. Sign. P.*, 2013. Doi: 10.1007/s11045-013-0269-9.

[7] D. Kressner, M. Steinlechner, and B. Vandereycken. Low-rank tensor completion by Riemannian optimization. *BIT Numer. Math.*, 54(2):447–468, 2014.

[8] J. Nocedal and S. J. Wright. *Numerical Optimization*, volume Second Edition. Springer, 2006.

[9] B. Mishra and R. Sepulchre. Riemannian preconditioning. Technical report, arXiv:1405.6055, 2014.

[10] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008.

[11] S. T. Smith. Optimization techniques on Riemannian manifold. In A. Bloch, editor, *Hamiltonian and Gradient Flows, Algorithms and Control*, volume 3, pages 113–136. Amer. Math. Soc., Providence, RI, 1994.

[12] A. Edelman, T.A. Arias, and S.T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20(2):303–353, 1998.

[13] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre. Manopt: a Matlab toolbox for optimization on manifolds. *JMLR*, 15(1):1455–1459, 2014.

[14] J. M. Lee. *Introduction to smooth manifolds*, volume 218 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, second edition, 2003.

[15] H. Sato and T. Iwai. A new, globally convergent Riemannian conjugate gradient method. *Optimization*, 64(4):1011–1031, 2015.

[16] W. Ring and B. Wirth. Optimization methods on riemannian manifolds and their application to shape space. *SIAM J. Optim.*, 22(2):596–627, 2012.