Riemannian Stochastic Recursive Gradient Algorithm

Hiroyuki Kasai [1] **Hiroyuki Sato** [2] **Bamdev Mishra** [3]

Abstract

Stochastic variance reduction algorithms have recently become popular for minimizing the average of a large, but finite number of loss functions on a Riemannian manifold. The present paper proposes a Riemannian stochastic recursive gradient algorithm (R-SRG), which does not require the inverse of retraction between two distant iterates on the manifold. Convergence analyses of R-SRG are performed on both retraction-convex and non-convex functions under computationally efficient retraction and vector transport operations. The key challenge is analysis of the influence of vector transport along the retraction curve. Numerical evaluations reveal that R-SRG competes well with state-of-the-art Riemannian batch and stochastic gradient algorithms.

1 Introduction

Let $f\colon \mathcal{M} \to \mathbb{R}$ be a smooth real-valued function on a Riemannian manifold $\mathcal{M}$ (Absil et al., 2008). The target problem concerns a given model variable $w \in \mathcal{M}$, and is expressed as

$$\min_{w \in \mathcal{M}} \left\{ f(w) := \frac{1}{n} \sum_{i=1}^{n} f_i(w) \right\}, \tag{1}$$

where n is the total number of the elements. This problem has many applications; for example, in principal component analysis (PCA) and the subspace tracking problem (Balzano et al., 2010) on the Grassmann manifold. The low-rank matrix/tensor completion problem is a promising application concerning the manifold of fixed-rank matrices/tensors (Mishra & Sepulchre, 2014; Kasai & Mishra, 2016). The linear regression problem is also defined on the manifold of fixed-rank matrices (Meyer et al., 2011).

A popular choice of algorithms for solving (1) is the *Riemannian gradient* descent method, which calculates the *Riemannian full gradient* estimation, i.e., $\mathrm{grad} f(w) = \frac{1}{n} \sum_{i=1}^{n} \mathrm{grad} f_i(w)$, for every iteration, where $\mathrm{grad} f_i(w)$ is the Riemannian gradient on the Riemannian manifold $\mathcal{M}$ for the i-th sample. However, this estimation is computationally costly when n is extremely large. A popular alternative is the *Riemannian stochastic gradient descent* algorithm (R-SGD), which extends the *stochastic gradient descent* algorithm (SGD) in the Euclidean space (Bonnabel, 2013) to the Riemannian manifold. As R-SGD calculates only $\mathrm{grad} f_i(w)$ for the i-th sample, the complexity per iteration is independent of the sample size n. Although R-SGD requires retraction and vector transport operations in every iteration, those calculation costs can be ignored when they are lower than those of $\mathrm{grad} f_i(w)$; this applies to many important Riemannian optimization problems, including the low-rank tensor completion problem and the Riemannian centroid problem as seen in Section 5.

Similar to SGD (Robbins & Monro, 1951), R-SGD is hindered by a slow convergence rate due to a *decaying step size* sequence. To accelerate the rate of R-SGD, the Riemannian stochastic variance reduced gradient algorithm (R-SVRG) (Sato et al., 2017; Zhang et al., 2016) has recently been proposed; this technique reduces the variance of the stochastic gradient exploiting the finite-sum form of (1) based on recent progress in *variance reduction* methods in the Euclidean space (Johnson & Zhang, 2013; Roux et al., 2012; Shalev-Shwartz & Zhang, 2013; Defazio et al., 2014; Reddi et al., 2016). One distinguished feature is reduction of the variance of *noisy* stochastic gradients by periodical full gradient estimations, which yields a linear convergence rate. R-SQN-VR has also recently been proposed, where a stochastic quasi-Newton algorithm and the variance reduced methods are mutually combined (Kasai et al., 2018). Although it achieves practical improvements for ill-conditioned problems, its convergence rate is worse than that of R-SVRG. Both R-SVRG and R-SQN-VR transport vectors between two *distant* iterates on the manifold $\mathcal{M}$; thus, they must calculate a tangent vector to connect them at every iteration, and are hindered by additional larger errors caused by vector transport compared with the parallel translation approach.

Here, we propose a *Riemannian stochastic recursive gradient* algorithm (R-SRG) that does not rely on two distant

[1]The University of Electro-Communications, Japan. [2]Kyoto University, Japan. [3]Microsoft, India. Correspondence to: Hiroyuki Kasai <kasai@is.uec.ac.jp>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

iterates. This feature of R-SRG is relevant for practical implementations and is also interesting from a theoretical analysis perspective. The R-SRG counterpart in the Euclidean space is proposed in (Nguyen et al., 2017a;b). The advantage of R-SRG over R-SVRG is more notable in the Riemannian than Euclidean case.

Contributions. Our contributions are summarized below.

- Our convergence analysis of R-SRG deals with both (*strongly*) *retraction-convex* (Definition 3.3 and Lemma 3.6) and *non-convex* functions.
- Our analysis considers computationally efficient *retraction* and *vector transport* instead of the more restrictive *exponential mapping* and *parallel translation*. This is more challenging than R-SVRG (Zhang et al., 2016), which involves exponential mapping and parallel translation.
- The obtained *total complexity* is the first result with respect to retraction and vector transport. Sato et al. (2017) have analyzed R-SVRG under retraction and vector transport, but have not provided the total complexity. Zhang et al. (2016) have provided the total complexity with only exponential mapping and parallel translation. Here, we derive a key fact (Lemma 3.8): the constants of L-smooth and L_l-Lipschitz are *not identical with respect to retraction*. Addressing the L_l/L ratio and the deviation parameter θ between vector transport and parallel translation (Lemma 3.7), we provide completely new complexities.
- The proposed algorithm has a linear convergence rate for (strongly) retraction-convex functions; converges at the sublinear rate in a single outer loop for general non-convex functions; and provides a linear rate in the case of gradient-dominated functions (Definition 3.4) (Polyak, 1963; Reddi et al., 2016; Zhang et al., 2016).

The advantages of R-SRG are summarized below.

- In R-SRG, computationally efficient retraction and vector transport are employed. Whereas R-SVRG transports vectors between two distant iterates, R-SRG transports vectors from the previous iterate. Thus, calculation of the inverse of retraction is avoided and R-SRG is computationally more efficient.
- R-SRG alleviates the additional errors caused by vector transport between two distant points encountered by R-SVRG.
- A practical variant of R-SRG accelerates the convergence speed, exploiting the linear convergence of the modified stochastic gradient in the inner loop.
- Use of retraction and vector transport enables application of R-SRG to a wider range of manifolds. For example, unlike our analysis and algorithm, the algorithm proposed by Zhang et al. (2016) cannot be applied to the Stiefel and fixed-rank manifolds, because they do not have closed-form expressions for parallel translation.

This paper is organized as follows. Section 2 presents details of the proposed R-SRG. Sections 3 and 4 summarize the preliminaries and present the convergence analysis, respectively. In Section 5, numerical comparisons with R-SGD and R-SVRG on two manifolds are given. The results suggest superior performance of R-SRG. The codes of R-SRG are implemented in the Matlab toolbox Manopt (Boumal et al., 2014) and are available at `https://github.com/hiroyuki-kasai/RSOpt`. Concrete proofs of theorems and details of additional experiments are provided as supplementary material.

2 Riemannian stochastic recursive gradient algorithm (R-SRG)

We assume that the manifold $\mathcal{M}$ is endowed with a Riemannian metric structure; i.e., a smooth inner product $\langle \cdot, \cdot \rangle_w$ is associated with tangent space $T_w\mathcal{M}$ for each $w \in \mathcal{M}$ (Absil et al., 2008). The *norm* $\| \cdot \|_w$ of a tangent vector in $T_w\mathcal{M}$ is that associated with the Riemannian metric. The metric structure allows a systematic framework for optimization over manifolds. Conceptually, the constrained optimization problem (1) is translated into an *unconstrained* problem over $\mathcal{M}$.

2.1 R-SGD and R-SVRG

R-SGD: Given a starting point $w_0 \in \mathcal{M}$, R-SGD produces a sequence $\{w_t\}$ in $\mathcal{M}$ that converges to a first-order critical point of (1). Specifically, it updates w as $w_{t+1} = R_{w_t}(-\alpha_t \mathrm{grad} f_{i_t}(w_t))$, where α_t is the step size and $\mathrm{grad} f_{i_t}(w_t)$ is a Riemannian stochastic gradient for the i_t-th sample, which is a tangent vector at $w_t \in \mathcal{M}$. $\mathrm{grad} f_{i_t}(w_t)$ represents an *unbiased* estimator of the Riemannian full gradient $\mathrm{grad} f(w_t)$, and the expectation of $\mathrm{grad} f_{i_t}(w_t)$ is $\mathrm{grad} f(w_t)$, i.e., $\mathbb{E}[\mathrm{grad} f_{i_t}(w_t) | \mathcal{F}_t] = \mathrm{grad} f(w_t)$. $\mathbb{E}[\cdot|\mathcal{F}_t]$ denotes an expected value taken with respect to the distribution of the random variable i_t. $\mathcal{F}_t = \sigma(w_0, i_1, \ldots, i_{t-1})$ is the σ-algebra that depends on $(w_0, i_1, \ldots, i_{t-1})$. The update moves from w_t in the direction $-\mathrm{grad} f_{i_t}(w_t)$ with step size α_t, remaining on $\mathcal{M}$. This mapping, denoted $R_w : T_w\mathcal{M} \to \mathcal{M} : \zeta \mapsto R_w(\zeta)$, is called a *retraction* at w, and maps tangent space $T_w\mathcal{M}$ onto $\mathcal{M}$ with a local rigidity condition that preserves the gradients at w. *Exponential mapping* Exp is an instance of retraction. Here, a curve defined by retraction R is called a *retraction curve* in this paper, being a *geodesic* when R is the exponential mapping.

R-SVRG: R-SVRG has a double loop structure where the s-th outer loop, called the *epoch*, has m_{s-1} inner iterations. Retaining a selected $w \in \mathcal{M}$ at the end of the $(s-1)$-th epoch as $\tilde{w}$, R-SVRG computes and stores the full Riemannian gradient $\mathrm{grad} f(\tilde{w})$ for this stored $\tilde{w}$. At the t-th inner iteration of the s-th epoch at w_t, picking the i_t-th sample, it computes the Riemannian stochastic gradient $\mathrm{grad} f_{i_t}(w_t)$ and $\mathrm{grad} f_{i_t}(\tilde{w})$ for this sample. Then, it calculates a *modified stochastic gradient* ξ_t by modifying $\mathrm{grad} f_{i_t}(w_t)$ using both $\mathrm{grad} f(\tilde{w})$ and $\mathrm{grad} f_{i_t}(\tilde{w})$. Because they belong to different tangent spaces, their simple addition is not well-defined, as Riemannian manifolds are not vector spaces. Consequently, after $\mathrm{grad} f_{i_t}(\tilde{w})$ and $\mathrm{grad} f(\tilde{w})$ are transported to $T_{w_t}\mathcal{M}$ by $\mathcal{T}_{\tilde{w}}^{w_t}$, the resultant update is performed as $w_{t+1} = R_{w_t}(-\alpha_t \xi_t)$, where ξ_t is set as

$$\xi_t = \mathrm{grad} f_{i_t}(w_t) - \mathcal{T}_{\tilde{w}}^{w_t}(\mathrm{grad} f_{i_t}(\tilde{w}) - \mathrm{grad} f(\tilde{w})).$$

Here, $\mathcal{T}_w^z$ or $\mathcal{T}_\eta$ represents *vector transport* from w to z satisfying $R_w(\eta) = z$. Vector transport $\mathcal{T} : T\mathcal{M} \oplus T\mathcal{M} \to T\mathcal{M}, (\eta, \xi) \mapsto \mathcal{T}_\eta \xi$ is associated with R, where $\xi, \zeta \in \mathcal{T}_w\mathcal{M}$ and $w \in \mathcal{M}$. It holds that (i) $\mathcal{T}_\eta \xi \in \mathcal{T}_{R(\eta)}\mathcal{M}$, (ii) $\mathcal{T}_0 \xi = \xi$, and (iii) $\mathcal{T}_\eta$ is a linear map. *Parallel translation* is a special instance of vector transport, which transports a vector along a curve γ from w to z. It is represented by $P(\gamma)_w^z$, or simply P_w^z, when γ is clear. Additionally, $P(\gamma)_\eta$ or P_η are also used.

2.2 Proposed R-SRG

Similar to R-SVRG, R-SRG has double loops. However, differently from R-SVRG, the inner loop of R-SRG generates the modified stochastic gradient v_t by adding and subtracting gradients to and from the previous v_{t-1}. More specifically, the recursive update of the stochastic gradient is calculated as $v_0 = \mathrm{grad} f(w_0)$ and, for $t \geq 1$, as

$$\begin{aligned} v_t &= \mathrm{grad} f_{i_t}(w_t) \\ &\quad - \mathcal{T}_{w_{t-1}}^{w_t} \mathrm{grad} f_{i_t}(w_{t-1}) + \mathcal{T}_{w_{t-1}}^{w_t} v_{t-1}. \end{aligned} \quad (2)$$

Then, the iterate update is calculated as $w_{t+1} = R_{w_t}(-\alpha_t v_t)$ with step size $\alpha_t > 0$. Note that, while the (modified) stochastic gradient of both R-SGD and R-SVRG is an unbiased estimator of the full gradient, that of R-SRG is not, i.e., $\mathbb{E}[v_t | \mathcal{F}_t] = \mathrm{grad} f(w_t) - \mathcal{T}_{w_{t-1}}^{w_t} \mathrm{grad} f(w_{t-1}) + \mathcal{T}_{w_{t-1}}^{w_t} v_{t-1} \neq \mathrm{grad} f(w_t)$. However, the total expectation $\mathbb{E}[v_t] = \mathbb{E}[\mathrm{grad} f(w_t)]$ holds. The algorithm is summarized in Algorithm 1.

Inspired by (Nguyen et al., 2017a), we propose a practical variant of R-SRG called R-SRG+. R-SRG has a linearly convergent v_t in retraction-convex functions (Proposition 4.4); thus, we propose an adaptive length for the inner loop size m. In detail, we stop the inner loop at $t = t_{\text{last}} < m$ when the norm of v_t decreases below the threshold of that of v_0, and proceed to the next outer loop. The threshold control parameter is denoted $\vartheta (0 \leq \vartheta \leq 1)$; the $\vartheta = 0$ case is identical to R-SRG. We also adopt a practical selection of $\tilde{w}^s$ as $w_{t_{\text{last}}+1}$. The proposed variant eliminates the need for careful selection of m in R-SRG. Note that this approach is inapplicable to R-SVRG because such a linearly convergent decrease of the modified stochastic gradient ξ_t is absent.

Algorithm 1 R-SRG algorithm

Require: Update frequency m and sequence $\{\alpha_t\}$ with $\alpha_t > 0$.

Initialize $\tilde{w}^0$.
for $s = 1, 2, \ldots$ **do**
  Store $w_0 = \tilde{w}^{s-1}$.
  Calculate Riemannian full gradient $\mathrm{grad} f(w_0)$.
  Store $v_0 = \mathrm{grad} f(w_0)$.
  Update $w_1 = R_{w_0}(-\alpha_0 v_0)$.
  for $t = 1, 2, \ldots, m-1$ **do**
    Choose $i_t \in \{1, 2, \ldots, n\}$ uniformly at random.
    Calculate v_t by (2):
    Update $w_{t+1} = R_{w_t}(-\alpha_t v_t)$.
  end for
  Set $\tilde{w}^s = w_{t'}$ for randomly chosen $t' \in \{0, 1, \ldots, m\}$.
end for

3 Preliminaries

For the convergence analysis, we derive the *retraction L_l-Lipschitz* lemma (Lemma 3.8) assuming the bound of the Hessian of f along a retraction curve, and exploiting the *retraction L-smooth* lemma (Lemma 3.5). First, we briefly present definitions and assumptions, followed by the essential lemmas.

3.1 Definitions and assumptions

We first summarize some definitions. Let R be a retraction. For linear transformations in tangent spaces, we use the operator norm with respect to the inner product from the Riemannian metric.

Definition 3.1 (Upper-Hessian bounded)**.** *f is said to be upper-Hessian bounded in $\mathcal{U} \subset \mathcal{M}$ with respect to R if there exists a constant $L > 0$ such that $\frac{d^2 f(R_w(t\eta))}{dt^2} \leq L$, for all $w \in \mathcal{U}$ and $\eta \in T_w\mathcal{M}$ with $\|\eta\|_w = 1$, and all t such that $R_w(\tau\eta) \in \mathcal{U}$ for all $\tau \in [0, t]$.*

Definition 3.2 (Lower-Hessian bounded)**.** *f is said to be lower-Hessian bounded in $\mathcal{U} \subset \mathcal{M}$ with respect to R if there exists a constant $\mu > 0$ such that $\mu \leq \frac{d^2 f(R_w(t\eta))}{dt^2}$, for all $w \in \mathcal{U}$ and $\eta \in T_w\mathcal{M}$ with $\|\eta\|_w = 1$, and all t such that $R_w(\tau\eta) \in \mathcal{U}$ for all $\tau \in [0, t]$.*

Definition 3.3 (Retraction convex (Huang et al., 2015b))**.** *f is retraction convex in $\mathcal{S} \subset \mathcal{M}$ with respect to R if, for*

all $w \in \mathcal{S}$ and $\eta \in T_w\mathcal{M}$ with $\|\eta\|_w = 1$, $f(R_w(\tau\eta))$ is convex for all t which satisfies $f(R_w(\tau\eta)) \in \mathcal{S}$ for all $\tau \in [0, t]$.

Definition 3.4 (τ-gradient dominated (Polyak, 1963))**.** *f is τ-gradient dominated in $\mathcal{U} \subset \mathcal{M}$ if there exists a constant $\tau > 0$ such that for $w \in \mathcal{U}$, $f(w) - f(w^*) \leq \tau\|\mathrm{grad} f(w)\|_w^2$, where w^* is a global minimizer of f.*

We make the following assumptions about (1).

Assumption 1. *For problem (1), we assume the following:*

(1.1) f and its components $f_1, f_2, \ldots, f_n$ are twice continuously differentiable.

(1.2) $\mathcal{T}$ is isometric on $\mathcal{M}$, i.e., $\langle \mathcal{T}_\xi\eta, \mathcal{T}_\xi\zeta\rangle_{R_w(\xi)} = \langle\eta, \zeta\rangle_w$ holds for any $w \in \mathcal{M}$ and $\xi, \eta, \zeta \in T_w\mathcal{M}$.

(1.3) The sequence generated by Algorithm 1 is continuously contained in a sufficiently small neighborhood $\mathcal{U} \subset \mathcal{M}$ of an optimal solution w^*. There exists a constant $c_0 > 0$ such that $\mathcal{T}$ satisfies $\|\mathcal{T}_\eta - \mathcal{T}_{R_\eta}\| \leq c_0\|\eta\|_w$, $\|\mathcal{T}_\eta^{-1} - \mathcal{T}_{R_\eta}^{-1}\| \leq c_0\|\eta\|_w$ for all $w, z \in \mathcal{U}$ with $R_w(\eta) = z$, where $\mathcal{T}_R$ denotes the differentiated retraction, i.e., $\mathcal{T}_{R_\zeta}\xi = \mathrm{D}R_w(\zeta)[\xi]$ with $\xi \in T_w\mathcal{M}$.

(1.4) The norms of the Riemannian gradient and Hessian are bounded, i.e., there exist constants $C_g > 0$ and $C_h > 0$ such that $\|\mathrm{grad} f_i(w_t)\|_w \leq C_g$ and $\|\mathrm{Hess} f_i(w)\| \leq C_h$ for $w \in \mathcal{U}$.

(1.5) The neighborhood $\mathcal{U}$ is a totally retractive neighborhood and totally normal neighborhood of w^* (see (Huang et al., 2015b)).

(1.6) There exists a constant $c_R > 0$ such that $\|\mathrm{Exp}_w^{-1}(z) - R_w^{-1}(z)\|_w \leq c_R\|R_w^{-1}(z)\|^2$ for all $w, z \in \mathcal{U}$.

Note that Assumption (1.1) is standard and (1.2) is guaranteed by the specific vector transport in (Huang et al., 2015b). Further, (1.3) is guaranteed when the vector transport is C^0, as derived from the Taylor expansion. Also, (1.4) holds when the manifold is compact like the Grassmann manifold, or through slight modification of the objective function and the algorithm. For (1.5), see (Huang et al., 2015b) for detail. Since R is a first-order approximation of Exp, (1.6) can be also considered natural.

3.2 Essential lemmas

Here, we present the lemmas essential for the convergence analysis under Assumption 1. The complete proofs are in the supplementary material.

Lemma 3.5 (Retraction L-smooth)**.** *Suppose that Assumptions (1.1) and (1.5) hold and that f is upper-Hessian bounded in $\mathcal{U}$. Then, for all $w, z \in \mathcal{U}$ and the constant $L > 0$ in Definition 3.1, we have*

$$f(z) \leq f(w) + \langle \mathrm{grad} f(w), \xi\rangle_w + \frac{1}{2}L\|\xi\|_w^2,$$

where $\xi \in T_w\mathcal{M}$ and $R_w(\xi) = z$. Here, such an f is called retraction L-smooth with respect to R.

Lemma 3.6 (Retraction μ-strongly convex (Huang et al., 2015b))**.** *Suppose that Assumptions (1.1) and (1.5) hold and that f is lower-Hessian bounded in $\mathcal{U}$. Then, for $\mu > 0$ from Definition 3.2 and for all $w, z \in \mathcal{U}$,*

$$f(z) \geq f(w) + \langle \mathrm{grad} f(w), \xi\rangle_w + \frac{1}{2}\mu\|\xi\|_w^2,$$

where $\xi \in T_w\mathcal{M}$ and $R_w(\xi) = z$. Here, such an f is called retraction μ-strongly convex with respect to R.

We also introduce the following lemma to quantify the difference between parallel translation and vector transport.

Lemma 3.7 (Difference between parallel translation and vector transport (Huang et al., 2015b, Lemma 3.5), (Huang et al., 2015a, Lemma 6))**.** *Let $\mathcal{T} \in C^0$ be a vector transport associated with the same retraction R as that of the parallel translation $P \in C^\infty$. Under Assumption (1.3), there exists a constant $\theta > 0$ such that for all $w, z \in \mathcal{U}$,*

$$\begin{aligned}\|\mathcal{T}_\eta\xi - P_\eta\xi\|_z &\leq \theta\|\xi\|_w\|\eta\|_w,\\ \|\mathcal{T}_\eta^{-1}\chi - P_\eta^{-1}\chi\|_w &\leq \theta\|\chi\|_z\|\eta\|_w,\end{aligned}$$

where $\xi, \eta \in T_w\mathcal{M}$, $\chi \in T_z\mathcal{M}$, and $R_w(\eta) = z$.

Finally, we derive the following key lemma:

Lemma 3.8 (Retraction L_l-Lipschitz)**.** *Let R be a retraction on $\mathcal{M}$. Suppose that Assumptions (1.1) and (1.3)–(1.5) hold. Then, there exists a constant $L_l > 0$ such that*

$$\|P(\gamma)_z^w \mathrm{grad} f(z) - \mathrm{grad} f(w)\|_w \leq L_l\|\eta\|_w,$$

for all $w, z \in \mathcal{U}$, where $L_l = C_h(1 + C_\eta\theta)$, with C_η being the upper bound of the norm of η for $\eta \in T_w\mathcal{M}$. Note that θ is in Lemma 3.7; γ is a curve $\gamma(t) := R_w(t\eta)$ defined by R on $\mathcal{M}$ with $\gamma(0) = w$ and $\gamma(1) = z$; and $P(\gamma)_z^w(\cdot)$ is a parallel translation operator along γ from z to w.

L and L_l are counterparts to those of L-smooth and L-Lipschitz for the geodesically L-smooth case and the L-smooth Euclidean case. However, it should be specifically emphasized that L and L_l are not identical in the retraction curve case.

Lemma 3.9. *Suppose that Assumptions (1.3) and (1.6) hold. Then, there exists a constant $\nu > 0$ such that $\langle\xi, \mathrm{Exp}_w^{-1}(z)\rangle_w \leq \langle\xi, R_w^{-1}(z)\rangle_w + \nu\|R_w^{-1}(z)\|_w^2$, for all $w, z \in \mathcal{U}$, where $\xi \in T_w\mathcal{M}$ and $\|\xi\|_w \leq 2C_g$, where C_g is a constant in Assumption (1.4).*

4 Convergence analysis

This section presents convergence analyses on both retraction-convex and non-convex functions under retrac-

tion and vector transport operations. To this end, we derive the bound on the number of iterations T to achieve an ϵ-accurate solution in terms of calls to the incremental first-order oracle (Agarwal & Bottou, 2015). In this particular case, we use the bound to guarantee the *expected* squared norm of a stochastic gradient $\mathbb{E}[\|\mathrm{grad} f(w_T)\|^2] \leq \epsilon$. Note that we derive the *total complexity* with regard to standard parameters such as n, ϵ, and L, and also $\rho_l = L_l/L$ in Lemma 3.8 and θ in Lemma 3.7 for this particular purpose.

4.1 Retraction-convex functions

We further suppose Assumption B.3 holds, which is equivalent to that f is L-smooth and convex in the Euclidean case (see the supplementary material).

Theorem 4.1 (Convergence analysis within a single outer loop on retraction convex functions)**.** *Let $\mathcal{M}$ be a Riemannian manifold and $w^* \in \mathcal{M}$ be a minimum of f. Suppose that Assumptions 1 and B.3 hold and f is upper-Hessian bounded. Consider Algorithm 1 with α that satisfies $\alpha < 2/L$ and $(2((2L_l + 2\theta C_g + L)\theta C_g + \nu L)m - L^2)\alpha^2 + 3L\alpha - 2 \leq 0$. Then, for any $s \geq 1$,*

$$\mathbb{E}[\|\mathrm{grad} f(\tilde{w}^s)\|^2_{\tilde{w}^s}] \leq \frac{2}{\alpha(m+1)}\mathbb{E}[f(\tilde{w}^{s-1}) - f(w^*)] + \frac{\alpha L}{2-\alpha L}\mathbb{E}[\|\mathrm{grad} f(\tilde{w}^{s-1})\|^2_{\tilde{w}^{s-1}}].$$

Proof. The complete proof is in the supplementary material. The proof proceeds as follows: Using Lemmas 3.5 and 3.8, and exploiting Lemma 3.7, the bound of $\sum_{t=0}^{m}\mathbb{E}[\|\mathrm{grad} f(w_t) - v_t\|^2_{w_t}]$ is derived. Then, conditioning α to eliminate additional terms caused by vector transport, $\mathbb{E}[\|\mathrm{grad} f(\tilde{w}^s)\|^2_{\tilde{w}^s}]$ is derived from Lemma 3.5. □

Suppose the θ in Lemma 3.7 and ν in Lemma 3.9 are sufficiently close to zero, i.e., $\mathcal{T}$ and R are close to P and Exp, respectively. This *reasonable* assumption yields $\beta < L^2$; thus, $(L^2 - \beta)\alpha^2 - 3L\alpha + 2 \geq 0$, where $\beta = 2((2L_l + 2\theta C_g + L)\theta C_g + \nu L)m$. The smaller root of $(L^2 - \beta)\alpha^2 - 3L\alpha + 2 = 0$, which is $\frac{3L - \sqrt{L^2+8\beta}}{2(L^2-\beta)} (= \alpha_l)$, is smaller than $1/L$, and the larger root exceeds $2/L$. Theorem 4.1 with $\alpha \leq 1/L$ implies $\mathbb{E}[\|\mathrm{grad} f(\tilde{w}^s)\|^2_{\tilde{w}^s}] \leq \frac{2}{\alpha(m+1)}\mathbb{E}[f(\tilde{w}^{s-1}) - f(w^*)] + \mathbb{E}[\|\mathrm{grad} f(\tilde{w}^{s-1})\|^2_{\tilde{w}^{s-1}}]$. Consequently, selecting $\alpha = \alpha^* := \sqrt{2\alpha_l/(m+1)}$ such that m satisfies $\alpha^* \leq 1/L$, we have

$$\begin{aligned}&\mathbb{E}[\|\mathrm{grad} f(\tilde{w}^s)\|^2_{\tilde{w}^s}] \\ &\leq 2\sqrt{\frac{L^2-\beta}{(3L-\sqrt{L^2+8\beta})(m+1)}}\mathbb{E}[f(\tilde{w}^{s-1}) - f(w^*)] \\ &\quad + \mathbb{E}[\|\mathrm{grad} f(\tilde{w}^{s-1})\|^2_{\tilde{w}^{s-1}}].\end{aligned}$$

This result means that the convergence rate within a single outer loop for retraction-convex functions is *sublinear*. Next, the convergence rates with multiple outer steps are derived based on this bound in Theorem 4.1.

Theorem 4.2 (Convergence analysis on retraction-convex functions)**.** *Suppose that all the conditions in Theorem 4.1 hold and define $\delta_k = \frac{2}{\alpha(m+1)}\mathbb{E}[f(\tilde{w}^k) - f(w^*)]$ for $k = 0, 1, \ldots, s-1$, and $\delta = \max_{0\leq k\leq s-1}\delta_k$. We also define $\Delta = \delta(1 + \frac{\alpha L}{2(1-\alpha L)})$, and $\varphi = \frac{\alpha L}{2-\alpha L}$. Then, we have*

$$\mathbb{E}[\|\mathrm{grad} f(\tilde{w}^s)\|^2_{\tilde{w}^s}] - \Delta \leq \varphi^s(\|\mathrm{grad} f(\tilde{w}^0)\|^2_{\tilde{w}^0} - \Delta).$$

The proof of Theorem 4.2 follows a similar approach to that given by (Nguyen et al., 2017a).

Suppose $\beta \leq L^2/3$ and that we select $\Delta = \epsilon/4$, $\varphi \leq \frac{2L^2}{3(L^2-\beta)}$ (with $\alpha = \alpha^* := 2\alpha_l/3 = \frac{3L-\sqrt{L^2+8\beta}}{3(L^2-\beta)}$) and $m = \mathcal{O}(1/\epsilon)$ in Theorem 4.2. As each inner loop evaluates $n + 2m$ gradients, the total complexity with respect to ϵ-accuracy is $\mathcal{O}((n + (1/\epsilon))\log(1/\epsilon)/\log(c(1 - \beta/L^2)))$ where $c > 1$ is a constant and $\mathcal{O}(\beta/L^2) = \mathcal{O}(\rho_l\theta/L)$.

Theorem 4.3 (Convergence analysis on retraction μ-strongly convex functions)**.** *Suppose that all the conditions in Theorem 4.1 hold and further assume that f is lower-Hessian bounded, where $\mu > 0$ satisfies the conditions in Definition 3.2 for both retraction R and exponential mapping* Exp*. Define $\sigma_m := \frac{1}{\mu\alpha(m+1)} + \frac{\alpha L}{2-\alpha L}$. Then, choosing α and m to satisfy $\sigma_m < 1$, we have*

$$\mathbb{E}[\|\mathrm{grad} f(\tilde{w}^s)\|^2_{\tilde{w}^s}] \leq (\sigma_m)^s\|\mathrm{grad} f(\tilde{w}^0)\|^2_{\tilde{w}^0}.$$

Let $\kappa := L/\mu$ be the condition number of f. Suppose $\beta \leq L^2/5$ and choose $\alpha = \alpha^* := \alpha_l/2$ and $m = 6.5\kappa$ for $T := \lceil\log(\|\mathrm{grad} f(\tilde{w}^0)\|^2_{\tilde{w}^0}/\epsilon)/\log(\frac{5(L^2-\beta)}{4L^2})\rceil$ outer iterations in Theorem 4.3. Then, the total complexity with respect to ϵ-accuracy is $\mathcal{O}((n + \kappa)\log(1/\epsilon)/\log(c(1 - \beta/L^2)))$, where $c > 1$ and $\mathcal{O}(\beta/L^2) = \mathcal{O}(\rho_l\theta/L)$.

The proof of Theorem 4.3 is also similar to that given by (Nguyen et al., 2017a).

Finally, we show the linear convergence of v_t in Algorithm 1 when f is retraction μ-strongly convex.

Proposition 4.4 (Linear convergence of v_t in inner loop)**.** *Suppose that Assumption B.4 and all the conditions in Theorem 4.3 except for α hold. Consider v_t in Algorithm 1 with a constant step size $\alpha < 2/L$. Then, there exist a function $\phi(\alpha)$ and constant $a_0 > 0$ such that, for $t \geq 1$, we have*

$$\mathbb{E}[\|v_t\|^2_{w_t}] \leq \left[1 - \left(\tfrac{2}{\alpha L} - 1\right)(a_0\mu - a_1C_g)^2\alpha^2 + \phi(\alpha)\right]^t \times\mathbb{E}[\|\mathrm{grad} f(w_0)\|^2_{w_0}].$$

When θ and ν are sufficiently close to zero, $\phi(\alpha)$ is close to zero regardless of α. Then, this result indicates that we obtain the linear convergence rate of $\|v_t\|^2_{w_t}$ for expectation rate $(1 - 1/\kappa^2)$ selecting $\alpha = \mathcal{O}(1/L)$.

4.2 Non-convex functions

We next find the convergence rate for non-convex functions.

Theorem 4.5 (Convergence analysis within a single outer loop on non-convex functions)**.** *Let $w^* \in \mathcal{M}$ be a minimizer of f and suppose Assumption 1 and that f is upper-Hessian bounded. Consider Algorithm 1 with a constant step size $\alpha \leq \frac{2}{L+\sqrt{L^2+8m(L_l^2+C_g^2\theta^2)}}$. Then, for $\tilde{w} = w_{t'}$, we have*

$$\mathbb{E}[\|\mathrm{grad} f(\tilde{w})\|_{\tilde{w}}^2] \leq \frac{2}{\alpha(m+1)}[f(w_0) - f(w^*)],$$

where t' is randomly chosen from $\{0, 1, \ldots, m\}$.

Proof. The complete proof is in the supplementary file, whose strategy is similar to those of Theorem 4.1 and (Nguyen et al., 2017b). Conditioning α to eliminate additional terms caused by retraction and vector transport, $\mathbb{E}[\|\mathrm{grad} f(\tilde{w})\|_{\tilde{w}}^2]$ is upper bounded by Lemma 3.5. □

We suppose that θ is sufficiently close to zero. Then, selecting the upper bound of α in Theorem 4.5 as α^* yields

$$\mathbb{E}[\|\mathrm{grad} f(\tilde{w})\|_{\tilde{w}}^2] \leq \frac{L+\sqrt{L^2+8m(L_l^2+C_g^2\theta^2)}}{m+1}\mathbb{E}[f(w_0)-f(w^*)].$$

Hence, when selecting $m = \mathcal{O}((L^2\rho_l^2+\theta^2)/\epsilon^2)$ and $\alpha^* = \mathcal{O}(1/\sqrt{m(L^2\rho_l^2+\theta^2)})$, we have $\mathbb{E}[\|\mathrm{grad} f(w_t)\|_{w_t}^2] \leq \epsilon$ to achieve a sublinear convergence rate with m for expectation rate $\mathcal{O}(\sqrt{(L^2\rho_l^2+\theta^2)/m})$. Hence, the total complexity for ϵ-accuracy is $\mathcal{O}(n + (L^2\rho_l^2+\theta^2)/\epsilon^2)$. Finally, the convergence rate with multiple outer steps is derived.

Theorem 4.6 (Convergence analysis on non-convex functions)**.** *Assume that all the conditions in Theorem 4.5 hold and that f is τ-gradient dominated. Consider Algorithm 1 with α as in Theorem 4.5 and assume $\bar{\sigma}_m := \frac{2\tau}{\alpha(m+1)} < 1$, i.e., $\alpha(m+1)/2 > \tau$. Then, we have*

$$\mathbb{E}[\|\mathrm{grad} f(\tilde{w}^s)\|_{\tilde{w}^s}^2] \leq (\bar{\sigma}_m)^s \|\mathrm{grad} f(\tilde{w}^0)\|_{\tilde{w}^0}^2.$$

Here, we choose α as the upper bound in Theorem 4.5. We need $m = \mathcal{O}(\tau^2(L^2\rho_l^2+\theta^2))$ to achieve $\frac{\alpha(m+1)}{2} > \tau$, and $s = \mathcal{O}(\log(1/\epsilon))$ to achieve ϵ-accuracy in the outer loop. Consequently, the total complexity with respect to ϵ-accuracy is $\mathcal{O}((n+\tau^2(L^2\rho_l^2+\theta^2))\log(1/\epsilon))$.

4.3 Discussions

We here discuss the total complexity of R-SRG as summarized in Table 1, addressing the special terms strongly related to retraction and vector transport, i.e., ρ_l $(= L_l/L)$, θ and β/L^2. It should be noted that, as previously, we restrict the discussion to the case where L_l and θ are sufficiently close to L and zero, respectively. Note that c in Table 1 is a constant satisfying $\beta \leq (1-\frac{1}{c})L^2$.

Table 1. Comparison of total complexity.

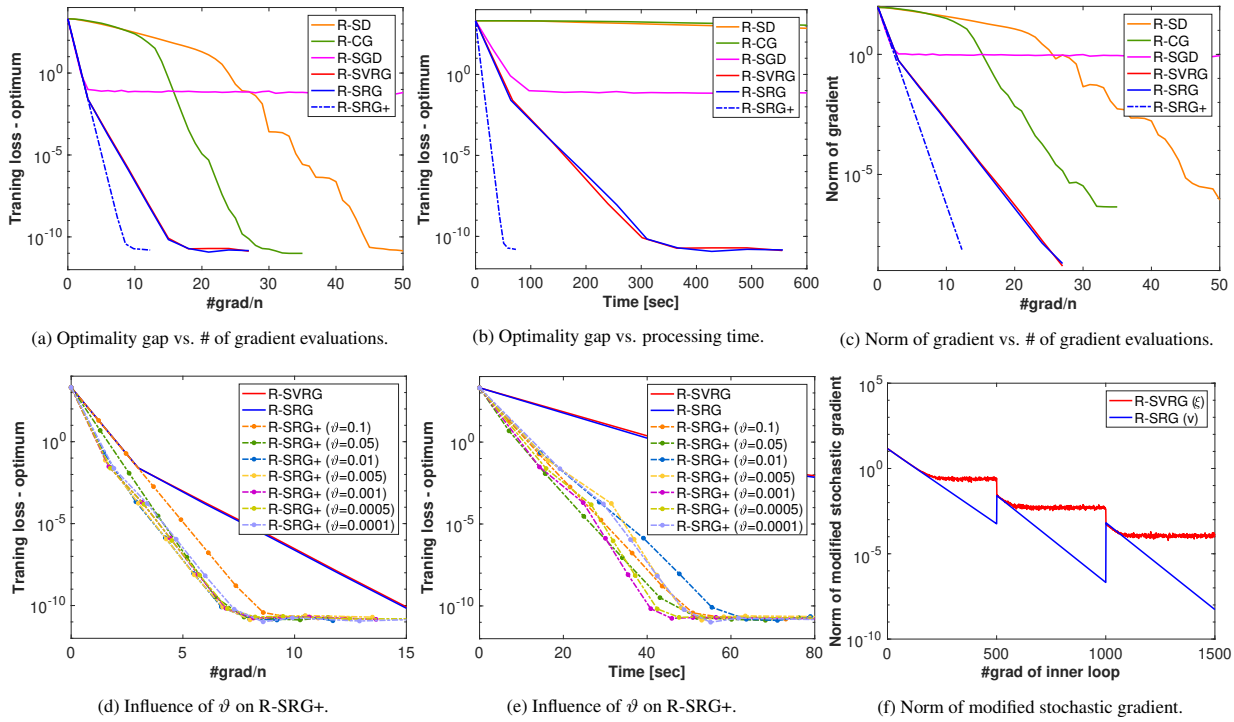
Function type	R-SRG (Proposed)	
Retraction convex (vector transport)	$\mathcal{O}((n+\frac{1}{\epsilon})\log(\frac{1}{\epsilon})/\log(c(1-\beta/L^2)))$	
Retraction μ-strongly convex (vector transport)	$\mathcal{O}((n+\kappa)\log(\frac{1}{\epsilon})/\log(c(1-\beta/L^2)))$	
Non-convex (vector transport)	$\mathcal{O}(n+\frac{L^2\rho_l^2+\theta^2}{\epsilon^2})$	
τ-gradient dominated (vector transport)	$\mathcal{O}((n+\tau^2(L^2\rho_l^2+\theta^2))\log(\frac{1}{\epsilon}))$	
Function type	R-SRG (Proposed)	R-SVRG (Zhang et al., 2016)
Geodesically convex (parallel translation)	$\mathcal{O}((n+\frac{1}{\epsilon})\log(\frac{1}{\epsilon}))$	–
Geodesically μ-strongly convex (parallel translation)	$\mathcal{O}((n+\kappa)\log(\frac{1}{\epsilon}))$	$\mathcal{O}((n+\zeta\kappa^2)\log(\frac{1}{\epsilon}))$
Non-convex (parallel translation)	$\mathcal{O}(n+\frac{L^2}{\epsilon^2})$	$\mathcal{O}(n+\zeta^{\frac{1}{2}}n^{\frac{2}{3}}/\epsilon)$
τ-gradient dominated (parallel translation)	$\mathcal{O}((n+\tau^2L^2)\log(\frac{1}{\epsilon}))$	$\mathcal{O}((n+L\tau\zeta^{\frac{1}{2}}n^{\frac{2}{3}})\log(\frac{1}{\epsilon}))$

Impact on complexity due to retraction and vector transport: Clearly, when a retraction curve deviates from the geodesic, the value of ρ_l increases, deviating from 1. Furthermore, θ deviates from 0 when the vectors from vector transport deviate from those of parallel translation. In those cases, the total complexity increases drastically. Those deviations more strongly influence the non-convex cases than the convex case. However, in the opposite case, the complexities retain values similar to the case of exponential mapping and parallel translation. Finally, the derived complexity is the *worst case* estimate for use of retraction and vector transport compared with the case of exponential mapping and parallel translation. Therefore, we leave investigation of more efficient retractions and vector transports than exponential mapping and parallel translation for future research.

Comparison with R-SVRG (Zhang et al., 2016): Additionally, we compare R-SRG with R-SVRG when exponential mapping and vector transport are used. In this case, we have $\rho_l = 1$, $\theta = 0$ and $\beta/L^2 = 0$; the results are also summarized as special cases in Table 1. Although R-SVRG has a curvature parameter $\zeta(\geq 1)$, R-SRG is superior to R-SVRG in the geodesically μ-strongly convex case. The convergence of R-SRG in a single outer loop for non-convex functions is inferior to that of R-SVRG in terms of ϵ, but superior with regard to n. R-SRG is superior to R-SVRG for τ-gradient dominated functions.

5 Numerical comparisons

In this section, we compare R-SRG(+) with R-SGD with a decaying step size sequence and R-SVRG with a fixed step size. The decaying step size sequence is $\alpha_k = \alpha(1+\alpha\lambda_\alpha\lfloor k/m\rfloor)^{-1}$, where k is the number of inner iterations, and where $\lfloor\cdot\rfloor$ denotes the floor function. As references, we also perform comparisons with two Riemannian batch methods with backtracking line search, R-SD and R-CG,



(a) Optimality gap vs. # of gradient evaluations.

(b) Optimality gap vs. processing time.

(c) Norm of gradient vs. # of gradient evaluations.

(d) Influence of ϑ on R-SRG+.

(e) Influence of ϑ on R-SRG+.

(f) Norm of modified stochastic gradient.

Figure 1. Riemannian centroid problem on SPD manifold.

which are the steepest descent and conjugate gradient algorithms on Riemannian manifolds, respectively (Absil et al., 2008). All experiments are executed in Matlab on a 4.0 GHz Intel Core i7 PC with 32 GB RAM, and are stopped when the gradient norm passes below 10^{-8} or a predefined maximum iteration is reached. All hyper parameters are selected by cross-validation. The supplementary material presents additional results.

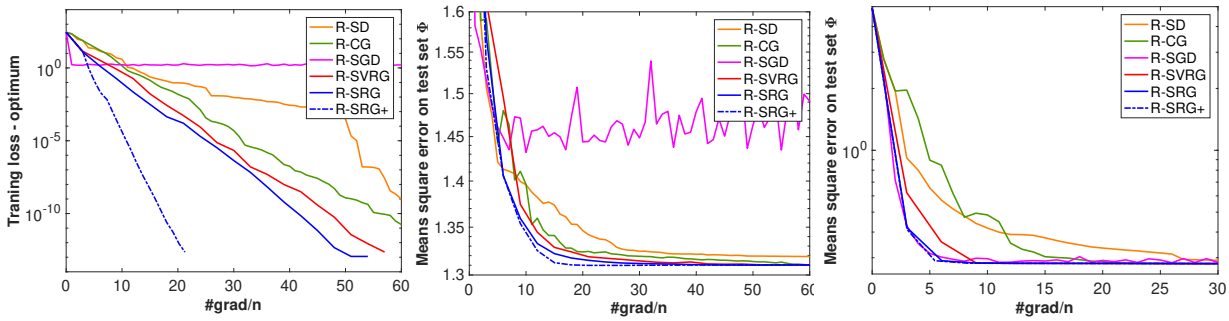
5.1 Riemannian centroid computation on symmetric positive-definite (SPD) manifold

We consider the problem of computing the Riemannian centroid on the $d \times d$ symmetric positive-definite (SPD) manifold $\mathcal{S}_{++}^d$, which frequently appears in computer vision problems such as visual object categorization and pose categorization (Jayasumana et al., 2015). Details of the SPD manifold are in the supplementary material.

Given n points $\{\mathbf{X}_1, \dots, \mathbf{X}_n\} \in \mathcal{S}_{++}^d$, the Riemannian centroid is derived from the solution to the problem $\min_{\mathbf{C} \in \mathcal{S}_{++}^d} \frac{1}{2n} \sum_{i=1}^n (\mathrm{dist}(\mathbf{C}, \mathbf{X}_i))^2$, where $\mathrm{dist}(a, b) = \| \log(a^{-1/2} b a^{-1/2}) \|_F$ represents the distance along the corresponding geodesic between the two points $a, b \in \mathcal{S}_{++}^d$ with respect to the affine-invariant Riemannian metric (AIRM). The gradient of the loss function is computed as $\frac{1}{n} \sum_{i=1}^n -\log(\mathbf{X}_i \mathbf{C}^{-1}) \mathbf{C}$.

We generate synthetic datasets and randomly initialize after setting the maximum iteration number as 20 for R-SVRG and R-SRG(+), and 60 for all others. α is tuned from $\{10^{-5}, \dots, 10^{-1}\}$. m and the batch size are n and 10, respectively. $\vartheta = 0.05$ is selected for R-SRG+. Our algorithm implementation for this particular problem uses the retraction and vector transport of (Huang et al., 2015b), which satisfy the requirements detailed in Section 4.

Figures 1(a)–(c), respectively, show two *optimality gap* results in terms of the number of gradient evaluations and processing time, and the norm of the gradient when $n = 10000$ with $d = 30$. The optimality gap indicates the performance against the minimum loss, which is calculated by R-CG with higher precision in advance. Hence, R-SRG and R-SVRG outperform the batch methods, R-SD and R-CG, even in terms of processing time. Further, R-SRG is competitive with R-SVRG, and R-SRG+ outperforms the others, especially in terms of processing time. Next, to investigate the influence of ϑ on R-SRG+, we consider the optimality gap when ϑ is changed to $\{0.1, 0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001\}$ (Figures 1 (d), (e)). Clearly, all results in this ϑ range indicate superior performance over the original R-SRG and R-SVRG. More importantly, R-SRG+ is insensitive to ϑ. Finally, the norms of the modified stochastic gradients ξ_t of R-SVRG and v_t of R-SRG are compared in Figure 1(f); the results for the first three outer loops are shown. While ξ_t of R-



(a) Optimality gap for PCA problem. (b) Test MSE for MC problem (Jester). (c) Test MSE for MC problem (MovieLens).

Figure 2. PCA problem and MC problem on Grassmann manifold.

SVRG fluctuates through each outer loop, v_t of R-SRG decreases monotonically, supporting Proposition 4.4.

5.2 PCA and matrix completion problems on Grassmann manifold

We also consider the PCA and matrix completion (MC) problems on the Grassmann manifold $\mathrm{Gr}(r, d)$, where a point is an equivalence class represented by a $d \times r$ orthogonal matrix $\mathbf{U}$ with orthonormal columns: $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. Note that projection-based vector transport and QR-decomposition-based retraction, which do not satisfy Assumption 1, are used in the following experiments for computational efficiency. The motivation is to show that our algorithm also empirically performs well without use of the specific vector transport. Details are given in the supplementary material.

PCA problem. Given an orthonormal matrix projector $\mathbf{U} \in \mathrm{St}(r, d)$, the PCA problem involves minimization of the sum of squared residual errors between projected data points and the original data, which is expressed as $\min_{\mathbf{U}\in\mathrm{St}(r,d)} \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{x}_i - \mathbf{U}\mathbf{U}^T\boldsymbol{x}_i\|_2^2$, where $\boldsymbol{x}_i$ is a data vector of size $d \times 1$. This problem is equivalent to maximizing $\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i^T\mathbf{U}\mathbf{U}^T\boldsymbol{x}_i$. Here, the critical points in the space $\mathrm{St}(r, d)$ are not isolated, because the cost function remains unchanged under the group action $\mathbf{U} \mapsto \mathbf{U}\mathbf{O}$ for all orthogonal matrices $\mathbf{O}$ of size $r \times r$. Subsequently, this is an optimization problem on the Grassmann manifold $\mathrm{Gr}(r, d)$. Figure 2(a) shows the results of the optimality gap when $n = 50000$, $d = 200$, and $r = 10$. α is from $\{10^{-3}, 2 \times 10^{-3}, \ldots, 10^{-2}\}$. The minimum loss for the optimality gap is obtained via the Matlab function `pca`. Figure 2(a) reveals that R-SRG(+) exhibits superior convergence performance to the alternatives.

MC problem. The MC problem involves completion of an incomplete matrix $\mathbf{X}$, say, of size $d \times n$, from a small number of entries by assuming that the latent structure of the matrix is low-rank. If Ω is the set of known indices in $\mathbf{X}$, the rank-r MC problem amounts to solving $\min_{\mathbf{U},\mathbf{A}} \|\mathcal{P}_\Omega(\mathbf{U}\mathbf{A}) - \mathcal{P}_\Omega(\mathbf{X})\|_F^2$, where $\mathbf{U} \in \mathbb{R}^{d\times r}, \mathbf{A} \in \mathbb{R}^{r\times n}$. The operator $\mathcal{P}_\Omega$ acts as $\mathcal{P}_\Omega(\mathbf{X}_{pq}) = \mathbf{X}_{pq}$ if $(p, q) \in \Omega$ and $\mathcal{P}_\Omega(\mathbf{X}_{pq}) = 0$ otherwise. Partitioning $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]$, the previous problem is equivalent to $\min_{\mathbf{U}\in\mathbb{R}^{d\times r}, \boldsymbol{a}_i\in\mathbb{R}^r} \frac{1}{n}\sum_{i=1}^{n} \|\mathcal{P}_{\Omega_i}(\mathbf{U}\boldsymbol{a}_i) - \mathcal{P}_{\Omega_i}(\boldsymbol{x}_i)\|_2^2$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ and $\mathcal{P}_{\Omega_i}$ is the sampling operator for the i-th column. Given $\mathbf{U}$, $\boldsymbol{a}_i$ admits a closed-form solution. Consequently, the problem depends on the column space of $\mathbf{U}$ only and is on $\mathrm{Gr}(r, d)$ (Boumal & Absil, 2015). Here, we use the Jester dataset (Goldberg et al., 2001) consisting of 24983 user ratings of 100 jokes. Each rating is a real number between -10 and 10. We randomly extract two ratings per user as the training set Ω and test set Φ. α is chosen from $\{10^{-7}, \ldots, 10^{-2}\}$ for R-SGD, R-SVRG, and R-SRG(+), and the batch size is 1, $r = 5$, and $\vartheta = 0.1$. The maximum number of outer iterations is 30 for R-SVRG and R-SRG(+), and 60 for the others. The algorithms are initialized randomly. We also use the MovieLens-1M dataset (Mov) containing one million ratings for 3952 movies (N) from 6040 users (d). We further randomly split this set into 80/10/10 percent datasets of the entire dataset as train/validation/test partitions. α is chosen from $\{10^{-5}, 5\times10^{-5}, \ldots, 10^{-2}, 5\times10^{-2}\}$, the batch size is 50, $r = 5$, and $\vartheta = 0.5$. The maximum number of outer iterations to stop is 20 for R-SVRG and R-SRG(+), and 60 for the others. Figures 2(b) and (c) show the superior performance of SRG(+) in both datasets.

6 Conclusions

We have proposed a Riemannian stochastic recursive gradient algorithm (R-SRG) on manifolds that is well suited for finite-sum minimization problems, and presented convergence analyses. R-SRG makes explicit use of retraction and vector transport, making it appealing for a wide variety of manifolds. Numerical comparisons have shown the benefits of R-SRG for a number of applications, with notable advantages over R-SVRG in both theory and practice.

Acknowledgements

H. Kasai was partially supported by JSPS KAKENHI Grant Numbers JP16K00031 and JP17H01732. H. Sato was partially supported by JSPS KAKENHI Grant Number JP16K17647.

References

MovieLens-10m dataset. URL http://grouplens.org/datasets/movielens/.

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

Agarwal, A. and Bottou, L. A lower bound for the optimization of finite sums. In *ICML*, 2015.

Balzano, L., Nowak, R., and Recht, B. Online identification and tracking of subspaces from highly incomplete information. In *Allerton*, pp. 704–711, 2010.

Bonnabel, S. Stochastic gradient descent on Riemannian manifolds. *IEEE Trans. on Automatic Control*, 58(9): 2217–2229, 2013.

Boumal, N. and Absil, P.-A. Low-rank matrix completion via preconditioned optimization on the Grassmann manifold. *Linear Algebra Appl.*, 475:200–239, 2015.

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

Defazio, A., Bach, F., and Lacoste-Julien, S. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *NIPS*, 2014.

Goldberg, K., Roeder, T., Gupta, D., and Perkins, C. Eigentaste: A constant time collaborative filtering algorithm. *Inform. Retrieval*, 4(2):133–151, 2001.

Huang, W., Absil, P.-A., and Gallivan, K. A. A Riemannian symmetric rank-one trust-region method. *Math. Program., Ser. A*, 150:179–216, 2015a.

Huang, W., Gallivan, K. A., and Absil, P.-A. A Broyden class of quasi-Newton methods for Riemannian optimization. *SIAM J. Optim.*, 25(3):1660–1685, 2015b.

Jayasumana, S., Hartley, R., Salzmann, M., Li, H., and Harandi, M. Kernel methods on Riemannian manifolds with Gaussian RBF kernels. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(12), 2015.

Johnson, R. and Zhang, T. Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS*, 2013.

Kasai, H. and Mishra, B. Low-rank tensor completion: a Riemannian manifold preconditioning approach. In *ICML*, 2016.

Kasai, H., Sato, H., and Mishra, B. Riemannian stochastic quasi-Newton algorithm with variance reduction and its convergence analysis. In *AISTATS*, 2018.

Meyer, G., Bonnabel, S., and Sepulchre, R. Linear regression under fixed-rank constraints: A Riemannian approach. In *ICML*, 2011.

Mishra, B. and Sepulchre, R. R3MC: A Riemannian three-factor algorithm for low-rank matrix completion. In *IEEE CDC*, 2014.

Nguyen, L. M., Liu, J., Scheinberg, K., and Takac, M. SARAH: A novel method for machine learning problems using stochastic recursive gradient. In *ICML*, 2017a.

Nguyen, L. M., Liu, J., Scheinberg, K., and Takac, M. Stochastic recursive gradient algorithm for nonconvex optimization. *arXiv preprint arXiv:1705.07261*, 2017b.

Polyak, B. Gradient methods for the minimisation of functionals. *USSR Computational Mathematics and Mathematical Physics*, 3(4):864–878, 1963.

Reddi, S. J., Hefny, A., Sra, S., Poczos, B., and Smola, A. Stochastic variance reduction for nonconvex optimization. In *ICML*, 2016.

Robbins, H. and Monro, S. A stochastic approximation method. *Ann. Math. Statistics*, pp. 400–407, 1951.

Roux, N. L., Schmidt, M., and Bach, F. R. A stochastic gradient method with an exponential convergence rate for finite training sets. In *NIPS*, 2012.

Sato, H., Kasai, H., and Mishra, B. Riemannian stochastic variance reduced gradient. *arXiv preprint: arXiv:1702.05594*, 2017.

Shalev-Shwartz, S. and Zhang, T. Stochastic dual coordinate ascent methods for regularized loss minimization. *JMLR*, 14:567–599, 2013.

Zhang, H., Reddi, S. J., and Sra, S. Fast stochastic optimization on Riemannian manifolds. In *NIPS*, 2016.