Riemannian Stochastic Gradient Method for Nested Composition Optimization

Dewei Zhang\(^1\) and Sam Davanloo Tajbakhsh\(^2\)

**Abstract**—This work considers the optimization of the composition of functions in a nested form over Riemannian manifolds where each function contains an expectation. This problem type is gaining popularity in applications such as policy evaluation in reinforcement learning or model customization in meta-learning. The standard Riemannian stochastic gradient methods for non-compositional optimization cannot be directly applied as the stochastic approximation of the inner functions creates biases in the gradients of the outer functions. For two-level composition optimization, we present a Riemannian Stochastic Composition Gradient Descent (R-SCGD) method that finds an approximate stationary point, with expected squared Riemannian gradient smaller than \(c\), in \(O(c^{-2})\) calls to the stochastic gradient oracle of the outer function and stochastic function and gradient oracles of the inner function.

**I. INTRODUCTION**

We consider optimizing the nested composition of functions over Riemannian manifolds. The two-level composition optimization problem has the form

\[
\min_{x \in \mathcal{M}} F(x) = \mathbb{E}_\xi[f_\xi(\mathbb{E}_\phi[g_\phi(x)])] \quad (1)
\]

where \(g_\phi : \mathcal{M} \to \mathcal{N} \) is a smooth map, \(\mathcal{M}\) and \(\mathcal{N} \subseteq \mathcal{E}\) are Riemannian manifolds, \(\mathcal{E}\) is a Euclidean space, \(f_\xi : \mathcal{E} \to \mathbb{R}\) is a continuously differentiable function, and \(\xi\) and \(\phi\) are independent random variables [2]. With a slight abuse of notation, we denote \(\mathbb{E}_\xi[f_\xi(\cdot)], \mathbb{E}_\phi[g_\phi(\cdot)]\) by \(f(\cdot)\) and \(g(\cdot)\), respectively. We assume throughout the paper that there exists at least one global optimal solution \(x^* \in \mathcal{M}\) to problem (1). We do not require either the outer function \(f_\xi\) or the inner function \(g_\phi\) to be (geodesically) convex or monotone. As a result, the composition problem cannot be reformulated into a saddle point problem in general [74]. The Riemannian stochastic gradient method [16] cannot be applied to solve (1) since the stochastic gradient \(\nabla f_\xi\) evaluated at the stochastic function value \(g_\phi\) does not result in an unbiased estimate of \(\nabla F(x)\) [52]. Hence, a special algorithmic setup is required to tackle this type of problem.

The stochastic compositional optimization finds applications in reinforcement learning for policy evaluation, meta-learning, stochastic minimax problems, dynamic programming and risk-averse problems [11], [12], [28], [26], [59]. The Riemannian manifold constraint arises due to 1) recent reformulation of some nonconvex problems as geodesically convex problems over manifolds [76], [63], 2) significant computational or performance gains in introducing manifold constraints in some applications, e.g., training deep neural networks by enforcing hidden-layer weight matrices to belong to Stiefel manifolds [9], [68], [10], [70], [3] new optimization problems that intrinsically involve manifold constraints [3], [32], [30], [62].

Below, we present a policy evaluation problem in reinforcement learning which has two-level nested compositional form over Riemannian manifolds as (1).

**Motivating example.** In reinforcement learning, finding the value function of a given policy is often referred to as policy evaluation problem [21], [64], [59]. Suppose there are \(S\) states with the state variable \(s \in \mathbb{R}^d\) and denote the policy of interest by \(\pi\). Let \(V^\pi \in \mathbb{R}^S\) denote the value function, where \(V^\pi(s)\) represents the value of state \(s\) under policy \(\pi\). The Bellman equation that should be satisfied by the optimal policy is

\[
V^\pi(s) = \mathbb{E}[r_{s,s'} + \rho V^\pi(s')] \forall s,s' \in \{1, \ldots, S\}, \quad (2)
\]

where \(\rho < 1\) is a discount factor, \(r_{s,s'}\) is the reward of transition from state \(s\) to state \(s'\), and the expectation is taken over all possible future states \(s'\) conditioned on current state \(s\) and the policy \(\pi\). In the blackbox simulation environment, the transition and reward matrices are unknown but can be sampled. Furthermore, a large number of states \(S\) makes solving the Bellman equation directly impractical. Therefore, we model the value function with Gaussian basis functions (see [14]) and consider an iterative procedure to estimate the parameters. In particular, we assume that \(V^\pi(s) \approx \sum_{i=1}^{D} w_i N(s; \mu_i, \Sigma_i), \quad D \ll S\). We denote \(\sum_{i=1}^{D} w_i N(s; \mu_i, \Sigma_i)\) by \(\phi(s; \Lambda)\), where \(\Lambda \triangleq \{\{w_i\}, \{\mu_i\}, \{\Sigma_i\}\}_{i=1}^{D}\) is the set of all parameters.

The policy evaluation problem is formulated as

\[
\min_{\Lambda} \sum_{s} \left[\phi(s; \Lambda) - \sum_{s'} \mathbb{E}[\hat{P}_{s,s'}] \left(\mathbb{E}[\hat{r}_{s,s'}] + \rho \phi(s'; \Lambda)\right)\right]^2 \quad (3)
\]

where the optimization variables are on the product of the Euclidean spaces and the manifolds of positive definite matrices. Problem (3) is an instance of (1) with \(f(\cdot) = \| \cdot \|^2\) and function \(g\) defined as \(g : \mathbb{R}^D \times \mathbb{R}^{D \times d} \times \mathcal{M}^D \to \mathbb{R}^S, \{\{w_i\}, \{\mu_i\}, \{\Sigma_i\}\}_{i=1}^{D} \mapsto \{g_1, \ldots, g_S\}\).

\[
g_s \triangleq \phi(s; \Lambda) - \sum_{s'} \mathbb{E}[\hat{P}_{s,s'}] \left(\mathbb{E}[\hat{r}_{s,s'}] + \rho \phi(s'; \Lambda)\right). \quad (4)
\]

Note that the outer function \(f\) in (1) is the deterministic function 2-norm square in this example.
A. Related work

This work stems from two different lines of research: 1. Manifold optimization, 2. Stochastic compositional optimization in Euclidean setting which are briefly reviewed below.

a) Manifold optimization.

The stochastic gradient descent (SGD) method over manifolds was first studied in [16] which proved Riemannian SGD converges to a critical point of the problem. Under geodesic convexity, [76] developed the first global complexity of first-order methods (in general) and established $O(1/k^2)$ complexity to attain an $\epsilon$-optimal solution, i.e., $f(x^k) - f^* \leq \epsilon$, for Riemannian SGD. In the Euclidean setting, many variance reduction techniques have been proposed to improve the sample complexity of SGD [53], [36], [69], [22], [50], [8]. As a generalization of [36], the Riemannian stochastic variance-reduced gradient descent (R-SVRG) method was developed in [75], establishing the linear rate for geodesically strongly convex functions. An extension of Riemannian SVRG with computationally more efficient retraction and vector transport was developed in [56]. In [60], authors adapted the Polyak-Ruppert iterate averaging over Riemannian manifolds [54], [48] reaching $O(1/k)$ rate. The Riemannian version of the stochastic recursive gradient method [46] was proposed in [38]. [37] proposed the adaptive gradient method with the convergence rate of $O(\log(k)/\sqrt{k})$.

Besides the stochastic gradient methods, numerous deterministic algorithms for Euclidean unconstrained optimization [47], [55] have also been generalized to Riemannian settings [2], [61], [17] - see, for instance, gradient descent [76], the Nesterov’s accelerated method [45], [77], [6], [7], proximal gradient method [34], Frank-Wolfe method [67], non-linear conjugate gradient method [58], [57], BFGS [51], Newton’s method [4], [38], trust-region method [1] and cubic regularized method [79], [5], [80].

b) Stochastic compositional optimization.

The stochastic compositional optimization is closely related to the classical SGD and stochastic approximation (SA) methods [52], [39], [40]. When the outer function $f$ is linear, the problem reduces to the standard stochastic non-compositional setting which has been extensively studied. While the problem is indeed compositional with two expected-value functions, [25] applied a simple two-timescale SA scheme, showing its convergence under basic assumptions.

The first nonasymptotic analysis for the stochastic compositional optimization appeared in [64], which uses two sequences of stepsizes in two different timescales: a slower one $\alpha_k$ for updating the optimization variable $x$ and a faster one $\beta_k$ for tracking the value of the inner function. Their analysis requires $\alpha_k/\beta_k \to 0$ as $k \to \infty$. For problems with smooth and convex composition objectives, their algorithm converges at a rate of $O(k^{-2/7})$, and with the rate of $O(k^{-4/5})$ in the strongly convex case, where $k$ represents the number of queries to the stochastic first-order oracles. The first finite-sample error bound is improved in [65] to $O(k^{-4/9})$ for convex and nonconvex settings. While most methods rely on the two-timescale stepsizes, the single timescale algorithm was recently developed in [27], achieving the sample complexity of $O(1/\epsilon^2)$ to find an $\epsilon$-approximate stationary point. Furthermore, [20] proposed a single-loop loop algorithm, without any need for accuracy-dependent stepsizes or increasing batch size, that can achieve the sample complexity of $O(1/\epsilon^2)$ as in [20] (or classic SGD for non-compositional problems).

In addition to the general stochastic optimization, the special setting with finite-sum structure recently gained popularity. The variants of the algorithm in [64] for finite-sum setting have been proposed in [42], [15], [23], [44], [71]. Furthermore, the stochastic compositional problem with certain nonsmooth components was investigated in [35], [81], [78]. A key feature of these works involves variance reduction techniques, which help to achieve better performance for finite-sum stochastic compositional problems. As these methods usually require increasing batch size, it is not possible to directly use them for general stochastic compositional problems. Stochastic composition optimization over Riemannian manifolds is also considered in [33]. While their RCG algorithm has the same complexity as our R-SCGD, RCG requires minibatch and, hence, does not apply to online setups. In recent work, [29] considers a bilevel optimization over manifolds both at upper and lower levels and shows how composition optimization can be formulated into their framework. While their paper allows the inner function to be a general manifold-valued map, their formulation for composition optimization results in another composition with the Riemannian distance square function.

B. Contributions

The main contributions of this paper are as follows: 1) The paper proposes algorithms to optimize the composition of two functions in the expectation form over Riemannian manifolds. We do not require the outer or inner function to be (geodesically) convex, thereby, broadening the scope of applicable problems. To our knowledge, this is the first work opening the discussion for geodesically nonconvex compositional problems. 2) We provide the sample complexity of the proposed algorithms obtaining $O(1/\epsilon^2)$ for an $\epsilon$-approximate stationary point, i.e., $\|\text{grad}(f(x))\| \leq \epsilon$, which is the same rate as Riemannian SGD for stochastic non-composition problems [30] or the algorithms in [20] and [27] for Euclidean composition problems. This result underscores the efficiency and practicality of our approach, especially in the context of Riemannian manifolds. 3) The proposed algorithm is inspired by the Riemannian extension of the ODE gradient flow presented in [20] for the unconstrained Euclidean setting. Our analysis reveals how manifold geometry affects the standard gradient flow and provides insights for future works. 4) We empirically verify the effectiveness of the proposed algorithm for two-level composition problems in the policy evaluation problem discussed in section I. Finally, we note that the extension of the proposed R-SCGD algorithm for multilevel stochastic compositional problem over Riemannian manifolds is available at [72] and eliminated here due to the page limit.
C. Preliminaries

A Riemannian manifold \((M, g)\) is a real smooth manifold \(M\) equipped with a Riemannian metric \(g\). The metric \(g\) induces an inner product structure in each tangent space \(T_xM\) associated with point \(x \in M\). We denote the inner product of \(u, v \in T_xM\) as \(\langle u \rangle_{x}\), and the norm of \(u\) is defined as \(\|u\| = \sqrt{\langle u \rangle_{x}}\). Given a smooth real-valued function \(f\) on a Riemannian manifold \(M\), the Riemannian gradient of \(f\) at \(x\) is denoted by \(\nabla f(x)\). We use \(\nabla f\) to denote the gradient (or Jacobian) of a scalar (or vector) valued function \(f\) in the Euclidean sense.

Definition 1 (Differential): Given manifolds \(M\) and \(M'\), the differential of a smooth map \(F : M \to M'\) at \(x\) is a linear operator \(DF(x) : T_xM \to T_{F(x)}M'\) defined by:

\[
DF(x)[v] = [t \mapsto F(c(t))],
\]

where \(F \circ c\) is seen as a map into \(M'\).

Definition 2 (Riemannian gradient): Let \(F : M \to \mathbb{R}\) be a smooth function on a Riemannian manifold \(M\). The Riemannian gradient of \(f\) is the vector field \(\nabla f\) on \(M\) such that

\[
\nabla f = \frac{\partial F}{\partial x} = \frac{\partial F}{\partial y},
\]

where \(DF(x)[v] = (F \circ c)'(0)\),

(5)

where \(F \circ c\) is seen as a map into \(M'\).

Definition 3 (Adjoint of an operator): Let \(E\) and \(E'\) be two Euclidean spaces, with inner products \(\langle \cdot \rangle_a\) and \(\langle \cdot \rangle_b\) respectively. Let \(A : E \to E'\) be a linear operator. The adjoint of \(A\) is a linear operator \(A^* : E' \to E\) and we have

\[
\forall u \in E, v \in E', \quad \langle A(u), v \rangle_b = \langle u, A^*(v) \rangle_a.
\]

II. TWO-LEVEL RIEMANNIAN COMPOSITION

We first characterize the Riemannian gradient of the composite function \(F(x)\)

\[
DF(x)[v] = (\langle \cdot \rangle)_{x} - (\langle \cdot \rangle)_{y}.
\]

where \(\nabla f = \partial F/\partial x\), \(\nabla g = \partial F/\partial y\), \(\nabla h = \partial F/\partial z\), and \(\nabla k = \partial F/\partial w\).

Following the standard SGD methodology applied on manifold optimization [52, 16, 75, 76, 30], a promising update is

\[
x^{k+1} = \text{Exp}_x^k(-\alpha(Dg_{g^k}(x^k))^*\nabla f_{k}(g(x^k))),
\]

where \(g^k\) and \(x^k\) are samples drawn at iteration \(k\). Note that \(Dg_{g^k}(x^k)^*\nabla f_{k}(g(x^k))\) is an unbiased estimator of grad \(F(x)\), see Assumption 2.4, but the exact evaluation of \(g(x^k)\), i.e., \(E_{\theta}[g_{\theta}(x^k)]\), is generally not attainable. Furthermore, the stochastic gradient is not unbiased if we replace \(g(x^k)\) by its stochastic estimate \(g_{\theta}(x^k)\). Therefore, the stochastic gradient method cannot be directly applied.

A. Algorithm development motivated by ODE analysis

Below, we provide the intuition behind our algorithm design based on the ODE gradient flow which carefully extends the analysis in [20] to the Riemannian setting.

Let \(t\) denote the time in this subsection. Consider the following ODE

\[
\dot{x}(t) = -\alpha(Dg(x))^*\text{Proj}_{g(x)}\nabla f(y(t)),
\]

for some \(\alpha > 0\). If we set \(y(t) = g(x(t))\), then we have

\[
\frac{d}{dt}f(g(x(t))) = \langle (Dg(x))^*\text{Proj}_{g(x)}\nabla f(g(x(t))), \dot{x}(t) \rangle_x(t) \leq -\frac{1}{\alpha}||\dot{x}(t)||^2_x(t) \leq 0.
\]

In this case, (6) describes a gradient flow that monotonically decreases \(f(g(x(t)))\). However, we can not evaluate \(g(x(t))\) exactly. Instead, we can evaluate \(\nabla f\) at \(y(t) \approx g(x(t))\), and the introduced inexactness results in \(f(g(x(t)))\) loosing its monotonicity:

\[
\frac{d}{dt}f(g(x(t))) = -\frac{1}{\alpha}||\dot{x}(t)||^2_x(t) + \langle (Dg(x))^*\text{Proj}_{g(x)}\nabla f(g(x(t))), \dot{x}(t) \rangle_x(t) \leq -\frac{1}{\alpha}||\dot{x}(t)||^2_x(t) + L_f C_g \|g(x(t)) - y(t)\| \cdot ||\dot{x}(t)||_{x(t)} \leq -\frac{1}{\alpha}||\dot{x}(t)||^2_x(t) + \frac{\alpha C_g L_g^2}{2} \|g(x(t)) - y(t)\|_x^2(t),
\]

where \(L_f\) and \(C_g\) are defined in Assumptions 2.2 and 2.5, respectively. This motivates an energy function,

\[
V(t) = f(g(x(t))) + \|g(x(t)) - y(t)\|^2.
\]

We want \(V(t)\) to monotonically decrease. By substitution, we have

\[
\ddot{V}(t) \leq -\frac{1}{2\alpha}||\dot{x}(t)||^2_x(t) + \frac{\alpha C_g L_g^2}{2} \|g(x(t)) - y(t)\|^2\]

\[
+ 2 \langle y(t) - g(x(t)), \dot{y}(t) - Dg(x(t))\dot{x}(t) \rangle_x(t) \leq -\frac{1}{2\alpha}||\dot{x}(t)||^2_x(t) + \frac{\alpha C_g L_g^2}{2} \|g(x(t)) - y(t)\|^2 + 2 \langle y(t) - g(x(t)), \dot{y}(t) - \beta(y(t) - g(x(t))) \rangle_x(t) - Dg(x(t))\dot{x}(t),
\]

2559

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
where $\beta > 0$ is a fixed constant. Following the maximum descent principle of $V(t)$, we are motivated to use the following dynamics

$$
\dot{y}(t) = -\beta(y(t) - g(x(t))) + Dg(x(t))[\dot{x}(t)].
$$

(7)

We approximate $Dg(x(t))[\dot{x}(t)]$ by the first-order Taylor expansion [17], i.e.,

$$
Dg(x(t))[\dot{x}(t)] \approx \gamma_k(g(x^k) - g(x^{k-1})),
$$

where $k$ is the discrete iteration index, and $\gamma_k$ is the weight controlling the approximation.

With the insights gained from (6) and (7), we propose the following stochastic update, which serves as the main components in Algorithm 1.

$$
y^{k+1} = y^k - \beta_k(y^k - g_{\phi_k}(x^k)) + \gamma_k(g_{\phi_k}(x^k) - g_{\phi_k}(x^{k-1})),
$$

$$
x^{k+1} = \exp_{x^k}(-\alpha(Dg_{\phi_k}(x^k))^*\text{Proj}_{g_{\phi_k}(x^k)}\nabla f_{\xi}(y^{k+1})).
$$

Algorithm 1 R-SCGD for two-level problem

**Require:** $x^0$, $y^0$, constant sequences $\{\alpha_k\}$, $\{\beta_k\}$, $\{\gamma_k\}$

1. for $k = 0, \ldots, K - 1$ do
2. Randomly sample $\phi^k$ and $\xi^k$
3. Update inner function estimate $y^{k+1}$ using
   $$
y^{k+1} = y^k - \beta_k(y^k - g_{\phi_k}(x^k)) + \gamma_k(g_{\phi_k}(x^k) - g_{\phi_k}(x^{k-1}))
   $$
4. Update Riemannian gradient of the composition function as
   $$
   \eta^{k+1} = (Dg_{\phi_k}(x^k))^*\text{Proj}_{g_{\phi_k}(x^k)}\nabla f_{\xi}(y^{k+1})
   $$
5. Update $x^{k+1} = \exp_{x^k}(-\alpha_k\eta^{k+1})$
6. end for

**B. Iteration complexity of the two-level R-SCGD method**

With the insights gained from the continuous-time Lyapunov function, our analysis in this section essentially builds on the following discrete-time Lyapunov function

$$
Y^k := F(x^k) - F(x^*) + \|g(x^k) - y^k\|^2,
$$

where $x^*$ is (one of) the optimal solution(s) of the problem (1).

**Assumption 2.1:** Function $F(x) : \mathcal{M} \rightarrow \mathbb{R}$ is geodesically $L_g$-smooth, i.e., $\nabla(x, s) \in \mathcal{T},$

$$
\|P_{\mathcal{T}}\gamma^{-1}\text{grad} F(\exp(s)) - \text{grad} F(x)\| \leq L_g\|s\|,
$$

where $\mathcal{T} \subseteq TM$ is the domain of $\text{Exp}$ and $P_{\mathcal{T}}$ denotes parallel transport along $\gamma(t) = \text{Exp}(t)s$ from $t = 0$ to $t = 1$.

**Assumption 2.2:** Function $f_\xi : E \rightarrow \mathbb{R}$ is $L_f$-smooth, i.e., for all $y, y' \in E$,

$$
\|\nabla f_\xi(y) - \nabla f_\xi(y')\| \leq L_f\|y - y'\|.
$$

(9)

**Assumption 2.3:** Random sample oracle of function value $g_{\phi}(x)$ is an unbiased estimator of $g(x)$ and has bounded variance, i.e.,

$$
\mathbb{E}[g_{\phi}(x)] = g(x),
$$

$$
\mathbb{E}[\|g_{\phi}(x) - g(x)\|^2] \leq V^2.
$$

**Assumption 2.4:** The chain rule holds in expectation, i.e.,

$$
\mathbb{E}[(Dg_{\phi}(x))^*\text{Proj}_{g_{\phi}(x)}\nabla f_{\xi}(g(x)) = (Dg(x))^*\text{Proj}_{g(x)}\nabla f_{\xi}(g(x)).
$$

**Remark 2.1:** If the random sample oracle of derivatives (gradients) satisfies

$$
\mathbb{E}[\|Dg_{\phi}(x)[\eta]\| = Dg(x)[\eta],
$$

$$
\mathbb{E}[(\nabla f_{\xi}(x)] = \nabla f(x),
$$

and with the independence between $\phi$ and $\xi$, Assumption 2.4 holds. See lemma below.

**Lemma 2.1:** If the random sample oracle for derivatives (gradients) satisfies

$$
\mathbb{E}[\|Dg_{\phi}(x)[\eta]\| = Dg(x)[\eta],
$$

and random variable $\phi$ is independent of $\xi$, then

$$
\mathbb{E}[\|Dg_{\phi}(x)[\eta]\| = Dg(x)[\eta],
$$

Assumption 2.5: The stochastic gradients of $f_\xi$ and $g_{\phi}$ are bounded in expectation, i.e.,

$$
\mathbb{E}[\|Dg_{\phi}(x)[\eta]\| \leq C_\phi^2,
$$

$$
\mathbb{E}[\|\nabla f_{\xi}(x)\|] \leq C_f^2.
$$

The Assumptions 2.1 and 2.2 require that the objective function and the outer function have Lipschitz continuous gradients. The Assumptions 2.3-2.5 require the stochastic oracles to satisfy certain unbiasedness and second-moment boundedness, which are typical assumptions for stochastic methods. When the manifold $\mathcal{M}$ in problem (1) degenerates to a linear space, then the presented assumptions are equivalent to the assumptions in stochastic compositional optimization in the Euclidean setting ([64], [65], [81], [20]).

**Lemma 2.2:** Consider $\mathcal{F}^k$ as the collection of random variables, i.e., $\mathcal{F}^k := \{\phi^0, \ldots, \phi^{k-1}, \xi^0, \ldots, \xi^{k-1}\}$. Suppose that the Assumptions 2.3 and 2.5 hold, and $y^{k+1}$ is generated by running R-SCGD in Algorithm 1 conditioned on $\mathcal{F}^k$, then the mean square error of $y^{k+1}$ satisfies

$$
\mathbb{E}[\|y^{k+1} - g(x)^2\|^{2}\mathcal{F}^k] \leq (1 - \beta_k)||y^k - g(x^{k-1})||^2 + 2\beta_k^2V_\phi^2 + B_0\alpha_k^2C_g^4C_f^2,
$$

where the constant is defined as $B_0 := 2(1 - \beta_k)^2 + \gamma_k^2 + \frac{1 - \gamma_k - \gamma_k^2}{\beta_k^2}$.

**Theorem 2.1 (Two-level R-SCGD):** Under the Assumption 2.1 to 2.5, if we choose the stepsizes as $\alpha_k = \frac{2h_k}{L_g^2}$ and $\gamma_k = 1 - h_k\beta_k$, such that $t := \sup\{|t_k|\}$ is finite, the iterates $\{x^k\}$ of SCSC in Algorithm 1 satisfy

$$
\sum_{k=0}^{K-1}\mathbb{E}[\|\text{grad} F(x^k)\|^2] \leq \frac{2\nu_0 + 2B_1}{\sqrt{K}}
$$

where the constant is defined as

$$
B_1 := \frac{L_f^2}{2}C_g^2C_f^2 + C_g^2L_g^4V_\phi^2 + [2 + 6(t + 1)^2]C_g^4C_f^2.
$$
III. NUMERICAL STUDIES

This section conducts numerical experiments over problem (3) to compare the proposed algorithm with the straightforward implementation of the Riemannian SGD method. The code is written in MATLAB and uses the MANOPT package [19]. All the studies are run on a laptop with a 1.4 GHz Quad-Core Intel Core i5 CPU and 8 GB memory.

A $9 \times 9$ grid over the state space, i.e., $d = 2$, is generated first. Next, we fix the number of basis functions to five, randomly generate the true parameters $\{w_i\}, \{\mu_i\}, \{\Sigma_i\}$ and generate the true value function, represented by a column vector $v$. We assume $\Sigma_i$ is unique across the basis functions and we fix $\{w_i\}$ and $\{\mu_i\}$ to their true value and optimize (3) over $\Sigma$ belonging to the symmetric positive definite manifold. Next, we find the “true” transition matrix $P$ and reward matrix $r$ based on the true parameters such that the Bellman equation holds. The true transition and reward matrices are added with zero mean Gaussian noises, and the transition matrix is further normalized to be doubly stochastic. These two matrices are used as the outputs of the stochastic oracle at each iteration of the algorithm. We also generate an $11 \times 11$ grid over the state space and follow a similar simulation procedure. We note that while some of the assumptions of our analysis are not easily verifiable for this example (similar to many other studies in Riemannian optimization), we still believe in the significant merits of this study to numerically evaluate the performance of the proposed algorithm.

Figure 1 and 2 compare the proposed method with the Riemannian SGD [16], [76] over 10 replicates for each scenario. The shades provide the percentile information based on the replicates. More specifically, Figure 1 presents the decreasing norm of the Riemannian gradients. The upper two plots select the last iterates as the output while the bottom two plots show the ergodic average of the iterates, on which the analysis is based. On those plots, the proposed R-SCGD method shows at least a linear convergence rate.

Though the test problem is not geodesically convex [17], the decreasing approximate bias and objective function value are illustrated in Figure 2. The results show better performance of the proposed R-SCGD compared to the biased Riemannian SGD for the composition problem.

IV. CONCLUSION

We present the Riemannian stochastic compositional gradient method (R-SCGD) to solve the composition of two functions in the expectation forms over Riemannian manifolds. The proposed algorithm which is motivated by the Riemannian gradient flow approximates the inner function value using a moving average corrected by the first-order information and its parameter is in the same timescale as the stepsize. We established the sample complexity of $O(1/\epsilon^2)$ for the proposed algorithms to obtain $\epsilon$-approximate stationary solution, i.e., $\|\text{grad}f(x)\| \leq \epsilon$. Finally, we empirically studied the performance of the proposed algorithm over a policy approximation example in reinforcement learning.

Fig. 1: Performance of the proposed R-SCGD algorithm compared to the Riemannian SGD, for the norm of the Riemannian gradient. (Left) The first setting with 81 states. (Right) The second setting with 121 states.

Fig. 2: Performance of the proposed R-SCGD algorithm compared to the Riemannian SGD. Left and right plots illustrate the 81- and 121-state settings, respectively. (Top) Top plots show the inner function approximation bias, (Bottom) Bottom plots show the function value gap.
APPENDIX

We provide the proofs related to the complexity analysis of the two-level and multi-level R-SCGD algorithms in this appendix.

A. Complexity analysis of the two-level R-SCGD

1) Proof of Lemma 2.1

Proof: It suffices to show that

$$\left\langle \mathbb{E}[(Dg_\phi(x))\ast Proj_{g_\phi(x)}\nabla f_\phi(g(x))], b_i \right\rangle_x$$

$$= \left\langle (Dg_\phi(x))\ast Proj_{g_\phi(x)}\nabla f(g(x)), b_i \right\rangle_x,$$

where \(\langle \cdot, \cdot \rangle_x\) is the inner product defined on the tangent plane of the manifold \(M\) at \(x, i.e., T_x M\), and \(\{b_i\}\) is the basis of \(T_x M\). We have

$$\langle (Dg_\phi(x))\ast Proj_{g_\phi(x)}\nabla f_\phi(g(x)), b_i \rangle_x$$

(a) $$= \mathbb{E} \left\langle Proj_{g_\phi(x)}\nabla f_\phi(g(x)), Dg_\phi(x)[b_i] \right\rangle_{g_\phi(x)}$$

(b) $$= \mathbb{E} \left\langle Proj_{g_\phi(x)}\nabla f_\phi(g(x)), Dg_\phi(x)[b_i] \right\rangle$$

(c) $$= \mathbb{E} \left\langle \nabla f_\phi(g(x)), Dg_\phi(x)[b_i] \right\rangle$$

(d) $$= \mathbb{E} \left\langle \nabla f(g(x)), Dg(x)[b_i] \right\rangle$$

(e) $$= \left\langle (Dg(x))\ast Proj_{g_\phi(x)}\nabla f(g(x)), b_i \right\rangle_x.$$ 

The equality (a) comes from the linearity of the inner product and the definition of the adjoint operator (see Definition 3). For the equality (b), since we assume the manifold \(N\) is embedded in the Euclidean space \(E\), the inner product is the induced metric inherited from the Euclidean metric [17]. The equality (c) follows from the orthogonality of \(Dg_\phi(x)[b_i]\) and \(\nabla f_\phi(g(x))\). The equality (d) follows from the independence between random variables \(\xi\) and \(\phi\). Finally, the last equality (e) follows from the definition of the adjoint operator.

2) Proof of Lemma 2.2

Proof: Under the update rule

$$y^{k+1} = y^k - \beta_k (y^k - g_{\phi^k}(x^k)) + \gamma_k (g_{\phi^k}(x^k) - g_{\phi^{k-1}}(x^k)),$$

we have

$$y^{k+1} - g(x^k) = (1 - \beta_k)(y^k - g(x^{k-1})) + (1 - \beta_k)T_{1,k} + \beta_k T_{2,k} + \gamma_k T_{3,k},$$

(10)

where \(T_{1,k} := g(x^{k-1}) - g(x^k),\ T_{2,k} := g_{\phi^k}(x^k) - g_{\phi^k}(x^{k-1}),\ T_{3,k} := g_{\phi^k}(x^k) - g_{\phi^k}(x^{k-1}).\) Conditioned on \(F^k\), taking expectation over \(\phi^k\), we have

$$\mathbb{E}[(1 - \beta_k)T_{1,k} + \beta_k T_{2,k} + \gamma_k T_{3,k}|F^k] = (1 - \beta_k - \gamma_k)T_{1,k},$$

where we have used the condition \(\mathbb{E}[g_{\phi^k}(x)] = g(x)\) in the Assumption 2.3.

Therefore, conditioned on \(F^k\), taking expectation on the both sides of (10), we have

$$\mathbb{E}[y^{k+1} - g(x^k)]^2|F^k|$$

$$= \mathbb{E}[(1 - \beta_k)(y^k - g(x^{k-1}))^2|F^k]$$

$$+ \mathbb{E}[(1 - \beta_k)T_{1,k} + \beta_k T_{2,k} + \gamma_k T_{3,k}|F^k]$$

$$\leq (1 - \beta_k)^2||y^k - g(x^{k-1})||^2$$

$$+ 2\beta_k^2||T_{1,k}||^2 + 2\gamma_k^2||T_{3,k}||^2$$

$$\leq (1 - \beta_k)^2||y^k - g(x^{k-1})||^2$$

$$+ 2\beta_k^2||T_{1,k}||^2 + 2\gamma_k^2||T_{3,k}||^2$$

$$\leq (1 - \beta_k)^2||y^k - g(x^{k-1})||^2 + 2\beta_k^2\beta_k + 2\gamma_k^2\gamma_k$$

$$+ (1 - \beta_k - \gamma_k)^2||T_{1,k}||^2,$$

where (a) is based on the Young’s inequality; (b) uses the Cauchy-Schwartz inequality and nonexpansiveness property of the projection operator; The last inequality follows from the Proposition 10.47 in [17].

Conditioned on \(F^k\), taking expectation over \(\phi^k\) and \(\xi^k\) on both sides, we have

$$\mathbb{E}[F(x^{k+1})|F^k]$$

(a) $$\leq \mathbb{E}[F(x^k)] + \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$= \mathbb{E}[F(x^k)] + \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$+ \alpha_k \mathbb{E}[\langle \nabla f_\phi(x^k), (Dg_\phi(x^k))\ast Proj_{g_\phi(x^k)}\nabla f(g(x^k)) \rangle |F^k]$$

(b) $$\leq \mathbb{E}[F(x^k)] + \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$+ \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$||F^{k+1}||^2$$

$$\leq \mathbb{E}[F(x^k)] + \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$+ \alpha_k \mathbb{E}[\langle \nabla f_\phi(x^k), (Dg_\phi(x^k))\ast Proj_{g_\phi(x^k)}\nabla f(g(x^k)) \rangle |F^k]$$

(c) $$\leq \mathbb{E}[F(x^k)] + \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$+ \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$||F^{k+1}||^2$$

$$\leq \mathbb{E}[F(x^k)] + \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$+ \alpha_k \mathbb{E}[\langle \nabla f_\phi(x^k), (Dg_\phi(x^k))\ast Proj_{g_\phi(x^k)}\nabla f(g(x^k)) \rangle |F^k]$$

(d) $$\leq \mathbb{E}[F(x^k)] + \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$+ \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$||F^{k+1}||^2$$

$$\leq \mathbb{E}[F(x^k)] + \frac{\alpha_k^2 L F}{2}||F^{k+1}||^2$$

$$+ \alpha_k \mathbb{E}[\langle \nabla f_\phi(x^k), (Dg_\phi(x^k))\ast Proj_{g_\phi(x^k)}\nabla f(g(x^k)) \rangle |F^k]$$

The inequality (a) follows from Assumption 2.4; The inequality (b) uses the Cauchy-Schwartz inequality and nonexpansiveness property of the projection operator; The
inequality (c) is based on Assumptions 2.2, 2.5 and the Jensen inequality; Finally, the inequality (d) uses the Young’s inequality. Based on the definition of the Lyapunov function in (8), it follows that
\[
E[\|y^{k+1}\| - y^k] - y^k
\]
\[
\leq -\alpha_k (1 - \alpha_k C_g^2 C_f^2) \|\nabla F(x^k)\|^2 + \frac{L_F}{2} \alpha_k^2 C_g^2 C_f^2
\]
\[
+ (1 + \beta_k) E[\|g(x^k)\| - y^{k+1} + \|\nabla F(x^k)\| - \|\nabla F(x^k) - y^k\|].
\]

In the following, we enforce \(\beta_k = \frac{\alpha_k C_g^2 C_f^2}{4}\), and \(\alpha_k\) sufficiently small such that \(\beta_k \in (0, 1)\). Combining the result in the Lemma 2.2, we have
\[
E[\|y^{k+1}\| - y^k] - y^k
\]
\[
\leq -\frac{\alpha_k}{2} \|\nabla F(x^k)\|^2 + \frac{L_F}{2} \alpha_k^2 C_g^2 C_f^2 - \beta_k \|g(x^k) - y^k\|^2
\]
\[
+ \frac{1}{2} \alpha_k^2 C_g^4 C_f^4 V_g^2 + [2 + 6(t + 1)^2] \alpha_k^2 C_g^4 C_f^4
\]
\[
\leq -\frac{\alpha_k}{2} \|\nabla F(x^k)\|^2 + \frac{L_F}{2} \alpha_k^2 C_g^2 C_f^2
\]
\[
+ \alpha_k^2 C_g^4 L_f^4 V_g^2 + [2 + 6(t + 1)^2] \alpha_k^2 C_g^4 C_f^2
\]

Defining \(B := \frac{L_F}{2} \alpha_k^2 C_g^2 C_f^2 + C_g^4 L_f^4 V_g^2 + [2 + 6(t + 1)^2] C_g^2 C_f^2\), and taking expectation over \(\mathcal{F}_k\) on both sides of the above inequality, it follows that
\[
E[\|y^{k+1}\| - y^k] \leq -\frac{\alpha_k}{2} E[\|\nabla F(x^k)\|^2] + B \alpha_k^2.
\]

By telescoping, we have
\[
E[\|y^K\| - y^0] \leq -\frac{1}{2} \sum_{k=0}^{K-1} \alpha_k E[\|\nabla F(x^k)\|^2] + B \sum_{k=0}^{K-1} \alpha_k^2.
\]

Using the fact that \(E[\|y^K\|] \geq 0\) and rearranging the terms, we have
\[
\sum_{k=0}^{K-1} \alpha_k E[\|\nabla F(x^k)\|^2] \leq 2y^0 + 2B \sum_{k=0}^{K-1} \alpha_k^2.
\]

Choosing the stepsize as \(\alpha_k = \frac{1}{\sqrt{K}}\) leads
\[
\sum_{k=0}^{K-1} \frac{1}{K} E[\|\nabla F(x^k)\|^2] \leq \frac{2y^0 + 2B}{\sqrt{K}},
\]

from which the proof is complete.

References

[1] Absil, P., Baker, C. & Gallivan, K. Trust-region methods on Riemannian manifolds. Foundations Of Computational Mathematics. 7, 303-330 (2007)

[2] Absil, P., Mahony, R. & Sepulchre, R. Optimization algorithms on matrix manifolds. (Princeton University Press, 2009)

[3] Absil, P. & Hosseini, S. A collection of nonsmooth Riemannian optimization problems. Nonsmooth Optimization And Its Applications. pp. 1-15 (2019)

[4] Adler, R., Dedieu, J., Margulies, J., Martins, M. & Shub, M. Newton’s method on Riemannian manifolds and a geometric model for the human spine. IMA Journal Of Numerical Analysis. 22, 389-390 (2002)

[5] Agarwal, N., Boumal, N., Bullins, B., and Cartis, C. Adaptive regularization with cubics on manifolds, Mathematical Programming, vol. 188, pp. 85–134, (2021)

[6] Ahn, K. & Sra, S. From Nesterov’s estimate sequence to Riemannian acceleration. Conference On Learning Theory. pp. 84-118 (2020)

[7] Alimisis, F., Orvieto, A., Bécigneul, G. & Lucchi, A. A continuous-time perspective for modeling acceleration in Riemannian optimization. International Conference On Artificial Intelligence And Statistics. pp. 1297-1307 (2020)

[8] Allen-Zhu, Z. & Hazan, E. Variance reduction for faster non-convex optimization. International Conference On Machine Learning. pp. 699-707 (2016)

[9] Arzovsky, M., Shah, A. & Bengio, Y. Unitary evolution recurrent neural networks. International Conference On Machine Learning. pp. 1120-1128 (2016)

[10] Bansal, N., Chen, X. & Wang, Z. Can we gain more from orthogonality regularizations in training deep networks? Advances in Neural Information Processing Systems, vol. 31, 2018

[11] Ben-Tal, A. & Nemirovski, A. Robust optimization—methodology and applications. Mathematical Programming. 92, 453-480 (2002)

[12] Bertsekas, D. Dynamic programming and optimal control: Volume I. (Athena scientific, 2012)

[13] Billingsley, P. Probability and measure. (John Wiley & Sons, 2008)

[14] Bishop, C. Pattern recognition and machine learning. (Springer, 2006)

[15] Blanchet, J., Goldfarb, D., Iyengar, G., Li, F. & Zhou, C. Unbiased simulation for optimizing stochastic function compositions. ArXiv Preprint ArXiv:1711.07564. (2017)

[16] Bonnabel, S. Stochastic gradient descent on Riemannian manifolds. IEEE Transactions On Automatic Control. 58, 2217-2229 (2013)

[17] Boumal, N. An introduction to optimization on smooth manifolds. Available Online, Aug. (2020)

[18] Boumal, N., Absil, P. & Cartis, C. Global rates of convergence for nonconvex optimization on manifolds. IMA Journal Of Numerical Analysis. 39, 1-33 (2019)

[19] Boumal, N., Mishra, B., Absil, P. & Sepulchre, R. Manopt, a Matlab toolbox for optimization on manifolds. The Journal Of Machine Learning Research. 15, 1455-1459 (2014)

[20] Chen, T., Sun, Y. & Yin, W. Solving stochastic compositional optimization is nearly as easy as solving stochastic optimization. IEEE Transactions On Signal Processing. (2021)

[21] Dann, C., Neumann, G., Peters, J. & Others Policy evaluation with temporal differences: A survey and comparison. Journal Of Machine Learning Research. 15 pp. 809-883 (2014)

[22] Defazio, A., Bach, F. & Lacoste-Julien, S. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. Advances In Neural Information Processing Systems. pp. 1646-1654 (2014)

[23] Devraj, A. M., and Chen, J. Stochastic variance reduced primal dual algorithms for empirical composition optimization. Advances In Neural Information Processing Systems, vol. 32, 2019.

[24] Dreisigmeyer, D. Direct search algorithms over Riemannian manifolds. Submitted To SIOP. (2007)

[25] Ernoliev, Y. Methods of stochastic programming. (Nauka, Moscow,1976)

[26] Finn, C., Abbeel, P. & Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. International Conference On Machine Learning. pp. 1126-1135 (2017)

[27] Ghadimi, S., Ruszczyński, A. & Wang, M. A single timescale stochastic approximation method for nested stochastic optimization. SIAM Journal On Optimization. 30, 960-979 (2020)

[28] Goodfellow, I., Bengio, Y. & Courville, A. Deep learning. (MIT Press, 2016)

[29] Han, A., Mishra, B., Javanpuria, P. & Takeda, A. A framework for bilevel optimization on Riemannian manifolds. (arXiv preprint arXiv:2002.03883, 2024)

[30] Hosseini, R. & Sra, S. An alternative to EM for Gaussian mixture models: batch and stochastic Riemannian optimization. Mathematical Programming. 181, 187-223 (2020)

[31] Hosseini, R. & Sra, S. Matrix manifold optimization for Gaussian mixtures. Advances In Neural Information Processing Systems. 28 pp. 910-918 (2015)

[32] Hu, J., Liu, X., Wen, Z. & Yuan, Y. A brief introduction to manifold optimization. Journal Of The Operations Research Society Of China. 8, 199-248 (2020)

[33] Huang, F. & Gao, S. Riemannian gradient methods for stochastic composition problems. (Neural Networks, vol. 153, pp. 224–234, 2019)

[34] Huang, W. & Wei, K. Riemannian proximal gradient methods. Mathematical Programming. pp. 1-43 (2021)
Huo, Z., Gu, B., Liu, J. & Huang, H. Accelerated method for stochastic composition optimization with nonsmooth regularization. *Advances In Neural Information Processing Systems*. 26 pp. 315-323 (2013)

Kasai, H., Jawanpuria, P. & Mishra, B. Riemannian adaptive stochastic gradient algorithms on matrix manifolds. *International Conference On Machine Learning*. pp. 3262-3271 (2019)

Kasai, H., Sato, H. & Mishra, B. Riemannian stochastic quasi-Newton algorithm with variance reduction and its convergence analysis. *International Conference On Artificial Intelligence And Statistics*. pp. 269-278 (2018)

Kiefer, J. & Wolfowitz, J. Stochastic estimation of the maximum of a regression function. *The Annals Of Mathematical Statistics*. pp. 462-466 (1952)

Kushner, H. & Yin, G. Stochastic approximation and recursive algorithms and applications. (Springer Science & Business Media,2003)

Lee, J. Introduction to Riemannian manifolds. (Springer,2018)

Lian, X., Wang, M. & Liu, J. Finite-sum composition optimization via variance reduced gradient descent. *Artificial Intelligence And Statistics*. pp. 1159-1167 (2017)

Lim, L., Sze-Wai Wong, K. & Ye, K. Numerical algorithms on the affine Grassmannian. *SIAM Journal On Matrix Analysis And Applications*. 40, 371-393 (2019)

Lin, T., Fan, C., Wang, M., and Jordan, M. I., Improved sample complexity for stochastic compositional variance reduced gradient, in Proceedings of the 2020 American Control Conference (ACC), pp. 126–131, IEEE, July 2020.

Liu, Y., Shang, F., Cheng, J., Cheng, H. & Jiao, L. Accelerated First-order Methods for Geodesically Convex Optimization on Riemannian Manifolds.. *NearIPS*. pp. 4868-4877 (2017)

Nguyen, L., Liu, J., Scheinberg, K. & Takáč, M. Stochastic recursive gradient algorithm for nonconvex optimization. *ArXiv Preprint ArXiv:1705.07261* (2017)

Nocedal, J. & Wright, S. Numerical optimization. (Springer Science & Business Media,2006)

Polyak, B. New method of stochastic approximation type. *Automatic And Remote Control*. 51, 937-946 (1990)

Rajeswaran, A., Finn, C., Kakade, S. & Levine, S. Meta-learning with implicit gradients. *Advances In Neural Information Processing Systems*. pp. 113-124 (2019)

Reddi, S., Hefny, A., Sra, S., Poczos, B. & Smola, A. Stochastic variance reduction for nonconvex optimization. *International Conference On Machine Learning*. pp. 314-323 (2016)

Ring, W. & Wirth, B. Optimization methods on Riemannian manifolds and their application to shape space. *SIAM Journal On Optimization*. 22, 596-627 (2012)

Robbins, H. & Monro, S. A stochastic approximation method. *The Annals Of Mathematical Statistics*. pp. 400-407 (1951)

Roux, N. Schmidt, M. and Bach, F. A stochastic gradient method with an exponential convergence rate for finite training sets. *Advances In Neural Information Processing Systems*, vol. 25, 2012.

Ruppert, D. Efficient estimations from a slowly convergent Robbins-Monro process. (Cornell University Operations Research, 1988)

Ruszczynski, A. Nonlinear optimization. (Princeton University Press, 2011)

Sato, H., Kasai, H. & Mishra, B. Riemannian stochastic variance reduced gradient algorithm with retraction and vector transport. *SIAM Journal On Optimization*. 29, 1444-1472 (2019)

Sato, H. & Iwai, T. A new, globally convergent Riemannian conjugate gradient method. *Optimization*. 64, 1011-1031 (2015)

Smith, S. Optimization techniques on Riemannian manifolds. *Fields Institute Communications*. 3, 113-135 (1994)

Sutton, R. & Barto, A. Reinforcement learning: An introduction. (MIT Press,2018)

Tripuraneni, N., Flammarion, N., Bach, F. & Jordan, M. Averaging stochastic gradient descent on Riemannian manifolds. *Conference On Learning Theory*. pp. 650-687 (2018)

Udriste, C. Convex functions and optimization methods on Riemannian manifolds. (Springer Science & Business Media,2013)

Vandereycken, B. Low-rank matrix completion by Riemannian optimization. *SIAM Journal On Optimization*. 23, 1214-1236 (2013)

Vishnoi, N. Geodesic convex optimization: Differentiation on manifolds, geodesics, and convexity. *ArXiv Preprint ArXiv:1806.06373* (2018)

Wang, M., Fang, E. & Liu, H. Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions. *Mathematical Programming*. 161, 419-449 (2017)

Wang, M., Liu, J. & Fang, E. Accelerating stochastic composition optimization. *Advances In Neural Information Processing Systems*. 29 pp. 1714-1722 (2016)

Wang, S., Zhu, B., Li, C., Wu, M., Zhang, J., Chu, W. & Qi, Y. Riemannian Proximal Policy Optimization. *ArXiv Preprint ArXiv:2005.09195* (2020)

Weber, M. and Sra, S. Projection-free nonconvex stochastic optimization on Riemannian manifolds, *IMA Journal of Numerical Analysis*, vol. 42, no. 4, pp. 3241–3271, 2022.

Wisdom, S., Powers, T., Hershey, J., Le Roux, J. & Atlas, L. Full-capacity unitary recurrent neural networks. *Advances In Neural Information Processing Systems*. pp. 4880-4888 (2016)

Xiao, L. & Zhang, T. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal On Optimization*. 24, 2057-2075 (2014)

Xie, D., Xiong, J. & Pu, S. All you need is beyond a good init: Exploring better solution for training extremely deep convolutional neural networks with orthonormality and modulation. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*, pp. 6176-6185 (2017)

Xu, Y. & Xu, Y. Katushya acceleration for convex finite-sum compositional optimization. *Informjs Journal On Optimization*. (2021)

Zhang, D. and Davanloo Tajbaksh, S. Riemannian Stochastic Gradient Method for Nested Composition Optimization, arXiv preprint arXiv:2207.09350, 2022.

Zhang, S. & He, N. On the convergence rate of stochastic mirror descent for nonsmooth nonconvex optimization. *ArXiv Preprint ArXiv:1806.04781* (2018)

Zhang, Z. & Lan, G. Optimal Algorithms for Convex Nesterov Stochastic Composite Optimization. *ArXiv Preprint ArXiv:2011.10076* (2020)

Zhang, H., J Reddi, S. and Sra, S. Riemannian SVRG: Fast stochastic optimization on Riemannian manifolds. *Advances In Neural Information Processing Systems*. 29 pp. 4592-4600 (2016)

Zhang, H. & Sra, S. First-order methods for geodesically convex optimization. *Conference On Learning Theory*, pp. 1617-1638 (2016)

Zhang, H. & Sra, S. Towards Riemannian accelerated gradient methods. *ArXiv Preprint ArXiv:1806.02812* (2018)

Zhang, J. & Xiao, L. A composite randomized incremental gradient method. *International Conference On Machine Learning*, pp. 7454-7462 (2019)

Zhang, J. & Zhang, S. A cubic regularized Newton’s method over Riemannian manifolds. *ArXiv Preprint ArXiv:1805.05565*, (2018)

Zhang, D. & Davanloo Tajbaksh, S. (2023). Riemannian Stochastic Gradient Method for Nested Composition Optimization, arXiv preprint arXiv:2207.09350, 2022.

Zhang, J. & Xiao, L. A stochastic composite gradient method with incremental variance reduction. *Advances In Neural Information Processing Systems*. 32 pp. 9078-9088 (2019)