SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2025, Vol. 68, Iss. 1, 112201:1–112201:19
https://doi.org/10.1007/s11432-023-4095-y

Inexact proximal gradient algorithm with random reshuffling for nonsmooth optimization

Xia JIANG[1], Yanyan FANG[1], Xianlin ZENG[1*], Jian SUN[1,2] & Jie CHEN[3,2]

[1]*National Key Lab of Autonomous Intelligent Unmanned Systems, School of Automation, Beijing Institute of Technology, Beijing 100081, China;*
[2]*Beijing Institute of Technology Chongqing Innovation Center, Chongqing 401120, China;*
[3]*School of Electronic and Information Engineering, Tongji University, Shanghai 200082, China*

Received 14 July 2023/Revised 18 December 2023/Accepted 11 March 2024/Published online 19 November 2024

Abstract Proximal gradient algorithms are popularly implemented to achieve convex optimization with nonsmooth regularization. Obtaining the exact solution of the proximal operator for nonsmooth regularization is challenging because errors exist in the computation of the gradient; consequently, the design and application of inexact proximal gradient algorithms have attracted considerable attention from researchers. This paper proposes computationally efficient basic and inexact proximal gradient descent algorithms with random reshuffling. The proposed stochastic algorithms take randomly reshuffled data to perform successive gradient descents and implement only one proximal operator after all data pass through. We prove the convergence results of the proposed proximal gradient algorithms under the sampling-without-replacement reshuffling scheme. When computational errors exist in gradients and proximal operations, the proposed inexact proximal gradient algorithms can converge to an optimal solution neighborhood. Finally, we apply the proposed algorithms to compressed sensing and compare their efficiency with some popular algorithms.

Keywords proximal operator, random reshuffling, inexact computation, compressed sensing, nonsmooth optimization

Citation Jiang X, Fang Y Y, Zeng X L, et al. Inexact proximal gradient algorithm with random reshuffling for nonsmooth optimization. Sci China Inf Sci, 2025, 68(1): 112201, https://doi.org/10.1007/s11432-023-4095-y

1 Introduction

Nonsmooth optimization is an important issue in signal processing [1] and cooperative control [2]. The traditional optimization theory and methods based on the differentiable concept are not applicable to noncontinuously differentiable objective functions or constraints. Moreover, many nonsmooth optimization problems require large-scale data, which makes the computation of the full gradient expensive; consequently, deterministic algorithms that necessitate computing full gradients are inefficient in solving these problems. Therefore, this paper proposes efficient stochastic gradient methods for the following nonsmooth optimization problems:

$$\min_{x\in\mathbb{R}^d}\left\{F(x)\triangleq\frac{1}{n}\sum_{i=1}^{n}f_i(x)+\phi(x)\right\},\tag{1}$$

where $f_i:\mathbb{R}^d\to\mathbb{R}$ is differentiable, and $\phi:\mathbb{R}^d\to\mathbb{R}$ is a nondifferentiable regularization.

One of the most recurring problems during the application of compressed sensing is the following l_1-regularized nonsmooth optimization problem:

$$\min_{x\in\mathbb{R}^d}\left\{F(x)\triangleq\frac{1}{n}\sum_{i=1}^{n}\|A_ix-y_i\|^2+\lambda\|x\|_1\right\},\tag{2}$$

where x is the sparse signal that needs to be recovered, $\{A_i\}_{i=1}^n$ are multiple observation matrices, and $\{y_i\}_{i=1}^n$ are corresponding compressed signals. Compressed sensing can reconstruct the compressed

* Corresponding author (email: xianlin.zeng@bit.edu.cn)

signals/images to the original sparse signal using only some known observation matrices. Furthermore, nonsmooth optimization is extensively applied in neural network training and classification problems. Contextually, nonsmooth regularization is utilized to prevent overfitting of the predictor x. Furthermore, nonsmooth optimization techniques are instrumental in various other reconstruction problems, such as tomographic reconstruction and image deconvolution [3], where the sparse prior knowledge of the unknown data is exploited. Designing efficient algorithms is the key (2) to recovering the original sparse signal with high probability.

For achieving nonsmooth convex optimization (1), one efficient method is the proximal gradient algorithm, where the proximal operator is used as an approximate gradient for gradient descent [4–7]. In recent years, deterministic proximal gradient algorithms with guaranteed convergence for nonsmooth optimization have emerged [8–11]. These proximal gradient algorithms implement proximal operators for the nonsmooth optimization and show more stability than subgradient-based methods [12], since subgradient methods may increase the objective function value for small step sizes. For nonsmooth convex optimization, proximal gradient algorithms have a convergence rate of $\mathcal{O}(1/T)$, whereas the traditional subgradient-based algorithms only achieve a convergence rate of $\mathcal{O}(1/\sqrt{T})$, in which T is the number of iterations [13].

Considering the burden of computing full gradients in deterministic algorithms, stochastic proximal gradient (SPG) algorithms using stochastic gradient descent (SGD) with the evaluation of the proximal operator have attracted substantial attention of researchers [14–16]. The results in [17] consider proximal methods with stochastic gradients using the online-to-batch approach and establish an $\mathcal{O}(1/\sqrt{T})$ convergence rate. Traditional SPG algorithms take the stochastic gradients uniformly sampled from the entire dataset and execute a proximal operation on the nonsmooth regularized function following each gradient descent. For large-scale datasets, these algorithms need to execute numerous proximal operations; this can be time-consuming and expensive, especially when the precise computation of the proximal operator for nondifferentiable functions is computationally costly.

Another popular stochastic method for convex optimization is the random reshuffling (RR) algorithm. In contrast to SGD, where data are uniformly sampled with replacement, the RR algorithm employs the sampling-without-replacement approach. In other words, the RR algorithm processes each data point within a given epoch and usually converges in fewer iterations than SGD [18–20]. However, the sampling-without-replacement approach in RR introduces biased gradients, which cannot approximate a full gradient descent. The presence of biased gradients complicates the convergence analysis of the RR scheme. For strongly convex nonsmooth finite-sum optimization, a recent work [21] has proposed a novel proximal gradient algorithm that integrates the RR scheme. Considering that a complete pass over the data results in an accurate approximation of a full gradient step, we aimed to design a proximal gradient algorithm with RR for general convex nonsmooth optimization.

For the nondifferentiable function ϕ in (1), the proximal operator in proximal gradient algorithms may not have an analytic solution; these algorithms are also computationally expensive. This scenario includes total variation regularization, nuclear norm regularization, and overlapping group l_1-regularization [22–24]. Considerably, inexact proximal gradient algorithms using approximate proximity operators [24–26] with the convergence analysis under weak assumptions have emerged. However, the effect of inexact calculation errors on the convergence rate is not shown in previous studies. Moreover, when it is expensive to precisely compute the gradients in the gradient descent steps or when these gradients are affected by disturbance [27, 28], some exact proximal algorithms have studied independent, zero-mean, or fixed deterministic errors in the gradient calculation [29, 30]. For nonsmooth convex optimization, [31] has considered the existence of errors in gradients and proximal operators.

We summarize some related studies in Table 1 briefly. The biased gradients inherent in the RR scheme, combined with the challenge of accounting for potential errors in proximal operators and gradients, make it challenging to realize nonsmooth convex optimization. Specifically, no study to date has explored the inexact SPG algorithm using the RR scheme for nonsmooth convex optimization (Table 1). Therefore, taking advantage of the excellent practical performance of RR and the low computation effort required for inexact proximal operators/gradients, this paper developed two efficient proximal gradient algorithms with RR for finite-sum optimization. The contributions of this paper are summarized as follows.

(1) This paper proposes computationally efficient basic and inexact SPG algorithms that utilize the sampling-without-replacement RR scheme. Unlike the proximal algorithm using SGD, which applies the proximal operator after each SGD step, the proposed algorithms implement the proximal operator only once after all data pass through. Thus, these proposed approaches save computational resources,

Table 1 Related exact and inexact proximal gradient algorithms.

Algorithm	Proximal operator	Gradient	Stochasticity	$f(x)$
[26]	Inexact	Exact	Deterministic	Convex
[14]	Inexact	Exact	SGD	Convex
[31]	Inexact	Inexact	Deterministic	Convex
[21]	Exact	Exact	RR	Strongly-convex
This paper	Inexact	Inexact	RR	Convex

particularly in scenarios where computing proximal operators is expensive.

(2) This paper proves that the proposed basic proximal gradient with RR (PG-RR) algorithm converges to a neighborhood of the optimal solution at an $\mathcal{O}(1/T)$ convergence rate. The convergence analysis leverages the Bregman divergence of objective functions and the forward per-epoch deviation to address the issue of biased gradients inherent in the RR scheme. Moreover, the errors in the vicinity of the optimal solution decrease as the step size is reduced.

(3) This paper proves that the inexact proximal gradient with the RR (IPG-RR) algorithm converges to a neighborhood of the optimal solution even when the proximal operator and gradients are computed imprecisely or subject to disturbance. The proposed IPG-RR algorithm extends the inexact deterministic algorithm in [31] to the RR setting. Moreover, this paper establishes the effect of inexact calculation errors on the convergence rate analysis. Compared with the recent studies on proximal RR in [21], this paper relaxes the assumption of strong convexity and considers calculation errors.

The remaining paper is organized as follows: the problem description and proposed algorithms are outlined in Section 2. The theoretical analysis of the proposed basic proximal and inexact proximal gradient algorithms with RR is provided in Sections 3 and 4. The numerical simulations of the proposed algorithms are provided in Section 5. Finally, the conclusion is presented in Section 6.

Notations and preliminaries. We define $\mathbb{R}$ as the set of real numbers, $\mathbb{R}^n$ as the set of n-dimensional real column vectors, $\mathbb{R}^{n\times m}$ as the set of n-by-m real matrices. All vectors in the paper are column vectors unless otherwise noted. For a vector $x\in\mathbb{R}^n$, the notation $\|x\|$ denotes the Euclidean norm, defined by $\|x\|=\sqrt{\sum_{i=1}^n x_i^2}$, and $\|x\|_1$ denotes the 1-norm, defined by $\|x\|_1=\sum_{i=1}^n|x_i|$. The notation $\langle\cdot,\cdot\rangle$ denotes the inner product, which is defined by $\langle a,b\rangle=a^{\mathrm{T}}b$. The notation $[m]$ denotes the set $\{1,\ldots,m\}$. For a differentiable function $f:\mathbb{R}^n\to\mathbb{R}$, $\nabla f(x)$ denotes the gradient of function f with respect to x. For a non-differentiable function $f:\mathbb{R}^n\to\mathbb{R}$, a vector $g\in\mathbb{R}^n$ is a subgradient of f at x if for all $z\in\mathbb{R}^n$, $f(z)\geqslant f(x)+g^{\mathrm{T}}(z-x)$. The set of subgradients of f at the point x is called the subdifferential of f at x, denoted by $\partial f(x)$. A function $f:\mathbb{R}^n\to\mathbb{R}$ is L-smooth if it is continously differentiable and its gradient satisfies $\|\nabla f(x)-\nabla f(y)\|\leqslant L\|x-y\|$ for all $x,y\in\mathbb{R}^n$. The proximal operator of a non-differentiable convex function $\phi:\mathbb{R}^n\to\mathbb{R}$ is defined as

$$\mathrm{prox}_{\gamma,\phi}(y)\triangleq\mathrm{argmin}_{x\in\mathbb{R}^n}\left\{\phi(x)+\frac{1}{2\gamma}\|x-y\|^2\right\},\tag{3}$$

where the step size $\gamma>0$. Let $z=\mathrm{prox}_{\gamma,\phi}(y)$. Then, it follows from the optimality condition for convex optimization problems that

$$\frac{1}{\gamma}(y-z)\in\partial\phi(z),\tag{4}$$

where $\partial\phi(z)$ is the subdifferential of ϕ at z. When there exist errors in the computation of the proximal operator of ϕ, we define the inexact proximal operator by $\mathrm{prox}^{\varepsilon}_{\gamma,\phi}(\cdot)$, where ε is the error in the proximal operator. Then, the inexact proximal operator is defined as

$$\mathrm{prox}^{\varepsilon}_{\gamma,\phi}(y)\triangleq\left\{\tilde{x}\Big|\frac{1}{2\gamma}\|\tilde{x}-y\|^2+\phi(\tilde{x})\leqslant\varepsilon+\min_{x\in\mathbb{R}^n}\left\{\frac{1}{2\gamma}\|x-y\|^2+\phi(x)\right\}\right\}.\tag{5}$$

The ε-subdifferential of a non-differentiable convex function $\phi:\mathbb{R}^n\to\mathbb{R}$ at x is

$$\partial_{\varepsilon}\phi(x)\triangleq\{y\in\mathbb{R}^n|\phi(q)-\phi(x)\geqslant y^{\mathrm{T}}(q-x)-\varepsilon,\ \forall q\in\mathbb{R}^n\}.\tag{6}$$

Algorithm 1 PG-RR algorithm.

(S.1) Require:
Step-size $\gamma > 0$; $x_0 \in \mathbb{R}^d$; number of epochs T.
(S.2) Iterations:
for $t = 0, 1, \ldots, T-1$ **do**
  Sample a permutation $\pi = (\pi_0, \pi_1, \ldots, \pi_{n-1})$ of $[n]$;
  $x_t^0 = x_t$;
  for $i = 0, 1, \ldots, n-1$ **do**
    $x_t^{i+1} = x_t^i - \gamma \nabla f_{\pi_i}(x_t^i)$;
  end for
  $x_{t+1} = \text{prox}_{n\gamma,\phi}(x_t^n)$;
end for

2 Problem description and algorithm design

This section formulates the problem and proposes basic and inexact proximal gradient algorithms with RR. Consider the finite-sum nonsmooth optimization problem,

$$\min_{x\in\mathbb{R}^d} F(x),\ F(x) = f(x) + \phi(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) + \phi(x), \tag{7}$$

where $x \in \mathbb{R}^d$ is the unknown decision variable, n is the number of samples, $f_i : \mathbb{R}^d \to \mathbb{R}$ is the cost function associated to sample i, and ϕ is a regularized function.

In this paper, we assume that the objective functions in (7) satisfy the following assumptions.

Assumption 1. Consider problem (7).

(1) Functions f_i $(i = 1, \ldots, n)$, and ϕ are convex, not necessarily strongly convex.

(2) Functions f_i $(i = 1, \ldots, n)$ are all L-smooth, while the regularized function ϕ is non-differentiable.

(3) There exists $G_\phi > 0$ such that $\|z\| < G_\phi$ for each $z \in \partial\phi(x)$ and $x \in \mathbb{R}^d$.

(4) There exists $G_f > 0$ such that $\|\nabla f_i(x)\| \leqslant G_f$ for each $x \in \mathbb{R}^d$.

(5) Function F is lower bounded by some $F_* \in \mathbb{R}$ and problem (7) owns at least one optimal solution $x_* \in \mathbb{R}^d$.

2.1 PG-RR algorithm

Using the RR sampling scheme, we propose an efficient proximal gradient algorithm (PG-RR) to solve (7) in Algorithm 1. The proposed Algorithm 1 includes three main steps.

(1) RR. At the start of each iteration t, we generate a permutation π through sampling without replacement.

(2) Incremental gradient (IG). Following the order π_i in the permutation, the variable x_t is updated using a gradient descent with respect to function f_{π_i}. After n such updates, we obtain the updated variable x_t^n.

(3) Proximal operator. We execute the proximal operator of the regularized function ϕ at the variable x_t^n, denoted as $\text{prox}_{n\gamma,\phi}(x_t^n)$[1)], only once after all data have been processed.

Different from the unbiased gradient in SGD, the gradient obtained from sampling-without-replacement is biased, so the intuition of applying the proximal operator in each inner iteration may not be superior due to the possible error accumulation.

Remark 1. The proposed algorithm executes the proximal operator only once after all data pass through, which is calculation-efficient when the proximal operator is computationally expensive. Note that the permutation π generated at the beginning of each epoch is different from the popular IG algorithm [32]. The IG generates a fixed permutation before all epochs and reuses the permutation in all subsequent epochs. In addition, the IG needs to carefully choose a suitable permutation and is susceptible to bad orderings compared with the RR algorithms [33]. Therefore, we take the RR scheme in the proposed algorithms.

Considering the biased gradients caused by the sampling-without-replacement RR scheme, define the variance at the optimal solution x_* as

$$\sigma_*^2 \triangleq \frac{1}{n}\sum_{i=1}^{n} \|\nabla f_i(x_*) - \nabla f(x_*)\|^2,$$

1) The definition of the proximal operator $\text{prox}_{\gamma n,\phi}(x_t^n)$ is given as $\text{prox}_{\gamma,\phi}(\cdot)$ in (3), using a step size of $n\gamma$.

Algorithm 2 IPG-RR algorithm.

(S.1) Require:
Step-size $\gamma > 0$; $x_0 \in \mathbb{R}^d$; number of epochs T.
(S.2) Iterations:
for $t = 0, 1, \ldots, T-1$ **do**
  Sample a permutation $\pi = (\pi_0, \pi_1, \ldots, \pi_{n-1})$ of $[n]$;
  $x_t^0 = x_t$;
  for $i = 0, 1, \ldots, n-1$ **do**
    $x_t^{i+1} = x_t^i - \gamma(\nabla f_{\pi_i}(x_t^i) + e_{\pi_i}(x_t^i))$;
  end for
  $x_{t+1} = \mathrm{prox}_{n\gamma,\phi}^{\varepsilon_{t+1}}(x_t^n)$;
end for

where the notation f_i represents the ith component function of f, which satisfies the relationship $f(x) = \frac{1}{n}\sum_{i=1}^n f_i(x)$.

Theorem 1. If Assumption 1 holds and step-size γ satisfies $3\gamma^2L^2n^2 \leqslant 1$, the average iterate $\tilde{x}_T \triangleq \frac{1}{T}\sum_{t=1}^T x_t$ satisfies

$$\mathbb{E}[F(\tilde{x}_T) - F_*] \leqslant \frac{\|x_0 - x_*\|^2}{2\gamma nT} + \frac{3}{8}\gamma^2 Ln\sigma_*^2 + L\gamma^2 n^2\left(G_f^2 + \frac{3}{2}G_\phi^2\right), \tag{8}$$

where σ_* is the variance at the optimal solution.

Remark 2. The convergence rate in (8) has a three-term structure. The first term corresponds to the deterministic gradient descent for general convex optimization. The second term reflects the stochastic property of the proposed algorithm. The gradient variance σ_* is caused by the random sampling-without-replacement reshuffling scheme. The third term is caused by the nonsmooth nature of the optimization problem. What is more, the second and third terms show that the neighbor errors at the optimal solution decrease with the step size γ. If the regularization ϕ in (7) is zero, the last two terms will be zeros and the convergence rate is consistent with the results in [34] for smooth non-strongly convex optimization. To be specific, when the regularization ϕ is zero, it is evident that $G_\phi = 0$ in the last term. The impact of the absence of regularization ϕ on the G_f term in the convergence result is primarily evident in the proof of Lemma 2. If the regularization ϕ is not zero, the equality $\nabla F(x^*) = \nabla f(x^*) + \nabla g(x^*)$ holds and $\nabla f(x^*)$ may not necessarily be zero. However, if the regularization ϕ is zero, we have $\nabla F(x^*) = \nabla f(x^*) = 0$. By following a similar convergence analysis as in Lemma 2, the G_f term in the convergence result will be zero.

Remark 3. Define the solution accuracy is ϵ, i.e., $\mathbb{E}[F(\tilde{x}_T) - F_*] \leqslant \epsilon$. According to (8) and given that $3\gamma^2L^2n^2 \leqslant 1$, we deduce that to achieve the ϵ-accuracy solution, the number of iterations T must satisfy $T \geqslant \frac{\|x_0 - x_*\|^2}{2\gamma n}\frac{1}{(\epsilon - \frac{1}{8Ln}\sigma_*^2 - \frac{1}{3L}(G_f^2 + \frac{3}{2}G_\phi^2))}$. In addition, the proposed PG-RR algorithm processes n samples in each iteration. Therefore, the lower bound of the sample complexity is $\mathcal{O}(\frac{1}{\gamma\epsilon})$.

2.2 IPG-RR algorithm

In order to tackle errors in the proximal operation and gradient calculation, which can be caused by external disturbances or attacks in the field of signal processing, we develop an IPG-RR algorithm for (7) in Algorithm 2. Similar to Algorithm 1, the proposed IPG-RR algorithm also includes three main steps.

(1) RR. At the start of each iteration t, we generate a permutation π through sampling without replacement.

(2) IG. Following the order π_i in the permutation, the variable x_t executes an inexact gradient descent step, where $e_{\pi_i}(x)$ denotes the calculation error of the gradient $\nabla f_{\pi_i}(x)$. After n cycles of inexact gradient descents, we obtain the variable x_t^n.

(3) Proximal operator. We perform the inexact proximal operator of the regularized function ϕ with a computing error ε_{t+1}, denoted as $\mathrm{prox}_{n\gamma,\phi}^{\varepsilon_{t+1}}(x_t^n)$[2)], only once after all data have been processed.

The first RR step is the same as that in the PG-RR algorithm. The second and third steps take into account the calculation errors in the gradient and proximal operations.

The convergence rate of the inexact Algorithm 2 is discussed in Theorem 2, whose proof is provided in Section 4. The error of gradient calculation through the inner iterations is denoted by $e_{t+1} \triangleq \sum_{i=0}^{n-1} e_{\pi_i}(x_t^i)$.

2) The definition of the inexact proximal operator $\mathrm{prox}_{n\gamma,\phi}^{\varepsilon}(x_t^n)$ is given as $\mathrm{prox}_{\gamma,\phi}^{\varepsilon}(\cdot)$ in (5), using a step size of $n\gamma$.

Theorem 2. Suppose error sequences $\{\|e_t\|\}$ and $\{\varepsilon_t\}$ are summable[3)]. If Assumption 1 holds and step-size γ satisfies $12\gamma^2L^2n^2 \leqslant 1$, the average iterate $\tilde{x}_T \triangleq \frac{1}{T}\sum_{t=1}^{T} x_t$ satisfies

$$\begin{aligned}\mathbb{E}[F(\tilde{x}_T) - F_*] \leqslant & \frac{\|x_0 - x_*\|^2}{2\gamma nT} + \frac{A_T\|x_0 - x_*\|}{\gamma nT} \\ & + \frac{A_T\sigma_*\sqrt{\gamma T/L} + \gamma T\sigma_*^2/4L}{2\gamma nT} + \frac{2A_T(2A_T + \sqrt{B_T}) + B_T}{2\gamma nT},\end{aligned} \tag{9}$$

where $\sigma_*^2 \triangleq \frac{1}{n}\sum_{i=1}^{n}\|\nabla f_i(x_*) - \nabla f(x_*)\|^2$, $A_T = \sum_{t=1}^{T}(\gamma\|e_t\| + \sqrt{2n\gamma\varepsilon_t})$ and $B_T = \frac{2\gamma nTG_f^2}{3L} + \frac{\gamma nTG_\phi^2}{2L} + 6\gamma^3Ln\sum_{t=1}^{\mathrm{T}}\|e_t\|^2 + 2\gamma n(1 + 2\gamma nL)\sum_{t=1}^{T}\varepsilon_t$.

Remark 4. The convergence rate in Theorem 2 is similar to that in Theorem 1. The main difference is that the three terms in (8) now include the effect of calculation errors. In addition, under the constant step-size setting, the error sequences need to be summable. If the step size is diminishing, the summable assumption can be relaxed, which is not discussed in this paper.

Remark 5. Let the solution accuracy be ϵ. By (9), we derive that the iteration complexity is $\mathcal{O}(\gamma^{-1}n^{-1}\epsilon^{-2})$. Since the proposed IPG-RR algorithm takes n samples at each iteration, it follows that the lower bound of sample complexity is $\mathcal{O}(\gamma^{-1}\epsilon^{-2})$.

Remark 6. The proposed algorithms can be easily extended to the parallel setting. In this setting, clients perform successive gradient descent steps on local optimization variables, following an RR permutation of their local datasets. Subsequently, all clients send their updated local optimization variables to a central server. The server then applies the proximal operator to the average of the received updates. However, the theoretical analysis for the parallel setting may differ from our current work.

Remark 7. Building on tensor algebraic frameworks, such as [35–37], several studies have developed optimization models with non-smooth regularized terms or non-smooth constraints for higher-order tensor factorization [38, 39] and tensor recovery [40, 41]. These studies proposed proximal stochastic gradient algorithms using the sampling-with-replacement scheme. The sampling-without-replacement RR scheme in this paper can be adapted to high-order tensor scenarios following a similar design idea. In addition, the approach of applying a single proximal operator in each iteration could be beneficial for higher-order tensor problems, especially when computing proximal operators is expensive. However, the theoretical analysis will be different from our work due to the more intricate properties of high-order tensors.

3 Theoretical analysis of PG-RR algorithm

In this section, we present theoretical proofs for the convergence properties of the proposed PG-RR algorithm. At first, we provide some necessary quantities for the RR algorithms.

Definition 1. For any i, the quantity $D_{f_i}(x, y) \triangleq f_i(x) - f_i(y) - \langle\nabla f_i(y), x - y\rangle$ is the Bregman divergence between x and y associated with function f_i.

If f_i is L-smooth, then for all $x, y \in \mathbb{R}^d$

$$D_{f_i}(x, y) \leqslant \frac{L}{2}\|x - y\|^2. \tag{10}$$

The difference between the gradients of a convex and L-smooth function f_i at x and y is related to its Bregman divergence by

$$\|\nabla f_i(x) - \nabla f_i(y)\|^2 \leqslant 2LD_{f_i}(x, y). \tag{11}$$

In addition, the forward per-epoch deviation over the t-th epoch is defined as follows.

Definition 2. Let $x_t^0, x_t^1, \ldots, x_t^n$ be iterates generated by the RR algorithms. We define the forward per-epoch deviation over the t-th epoch as

$$\mathcal{V}_t \triangleq \sum_{i=0}^{n-1}\|x_t^i - x_{t+1}\|^2. \tag{12}$$

3) Let $\{a_n\}$ be a sequence of real numbers. The series $\sum_{k=1}^{\infty} a_k$ is summable if and only if the sequence $x_n \triangleq \sum_{k=1}^{n} a_k,\ n \in \mathbb{N}$ converges.

The next lemma characterizes the variance of sampling without replacement, which is a key ingredient in the theoretical analysis.

Lemma 1 (Lemma 1 in [34]). Let $X_1, \ldots, X_n \in \mathbb{R}^d$ be fixed vectors, $\bar{X} \triangleq \frac{1}{n}\sum_{i=1}^n X_i$ be their average and $\sigma^2 \triangleq \frac{1}{n}\sum_{i=1}^n \|X_i - \bar{X}\|^2$ be the population variance. Fix any $k \in \{1, \ldots, n\}$, let $X_{\pi_1}, \ldots, X_{\pi_k}$ be sampled uniformly without replacement from $\{X_1, \ldots, X_n\}$ and $\bar{X}_\pi$ be their average. Then, the sample average and variance are given by

$$\mathbb{E}[\bar{X}_\pi] = \bar{X}, \quad \mathbb{E}[\|\bar{X}_\pi - \bar{X}\|^2] = \frac{n-k}{k(n-1)}\sigma^2.$$

Now, with the above auxiliary quantities and vital lemma, we show that the forward per-epoch deviation $\mathcal{V}_t$ of the PG-RR algorithm is upper bounded.

Lemma 2. Consider the iterates of Algorithm 1. If Assumption 1 holds, then

$$\mathbb{E}[\mathcal{V}_t] \leqslant 6\gamma^2 L n^2 \sum_{i=0}^{n-1} \mathbb{E}[D_{f_{\pi_i}}(x_*, x_t^i)] + \frac{3}{4}\gamma^2 n^2 \sigma_*^2 + 2\gamma^2 n^3 G_f^2 + 3\gamma^2 n^3 G_\phi^2. \tag{13}$$

Proof. It follows from lines 6–10 of Algorithm 1 and the optimality condition (4) that there exists $d_{t+1} \in \partial\phi(x_{t+1})$ such that

$$x_t - \gamma \sum_{i=0}^{n-1} \nabla f_{\pi_i}(x_t^i) - x_{t+1} = \gamma n d_{t+1}. \tag{14}$$

In addition, by iterates, $x_t^k = x_t - \gamma\sum_{i=0}^{k-1} \nabla f_{\pi_i}(x_t^i)$. Then,

$$\begin{aligned}
&\|x_t^k - x_{t+1}\|^2 \\
&= \left\| x_t - \gamma \sum_{i=0}^{k-1} \nabla f_{\pi_i}(x_t^i) - \left(x_t - \gamma \sum_{i=0}^{n-1} \nabla f_{\pi_i}(x_t^i) - \gamma n d_{t+1} \right) \right\|^2 \\
&= \left\| \gamma \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_t^i) + \gamma n d_{t+1} \right\|^2 \\
&= \gamma^2 \left\| \sum_{i=k}^{n-1} (\nabla f_{\pi_i}(x_t^i) - \nabla f_{\pi_i}(x_*)) + \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) + n d_{t+1} \right\|^2 \\
&\leqslant 3\gamma^2 (n-k) \sum_{i=k}^{n-1} \|\nabla f_{\pi_i}(x_t^i) - \nabla f_{\pi_i}(x_*)\|^2 + 3\gamma^2 \left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) \right\|^2 + 3\gamma^2 \|n d_{t+1}\|^2 \\
&\overset{(11)}{\leqslant} 6\gamma^2 L (n-k) \sum_{i=k}^{n-1} D_{f_{\pi_i}}(x_*, x_t^i) + 3\gamma^2 \left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) \right\|^2 + 3\gamma^2 \|n d_{t+1}\|^2 \\
&\leqslant 6\gamma^2 L n \sum_{i=0}^{n-1} D_{f_{\pi_i}}(x_*, x_t^i) + 3\gamma^2 \left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) \right\|^2 + 3\gamma^2 n^2 G_\phi^2.
\end{aligned}$$

Taking expectations of both sides and summing up from $k = 0$ to $n-1$ yields

$$\sum_{k=0}^{n-1} \mathbb{E}[\|x_t^k - x_{t+1}\|^2] \leqslant 6\gamma^2 L n^2 \sum_{i=0}^{n-1} \mathbb{E}[D_{f_{\pi_i}}(x_*, x_t^i)] + 3\gamma^2 \sum_{k=0}^{n-1} \mathbb{E}\left[\left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) \right\|^2 \right] + 3\gamma^2 n^3 G_\phi^2.$$

We now bound the second term on the right-hand side of the above inequality. First, by Lemma 1 with $\bar{X}_\pi = (1/(n-k))\sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*)$ and $\bar{X} = \nabla f(x_*)$, we have

$$\mathbb{E}\left[\left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) \right\|^2 \right]$$

$$
\begin{aligned}
&= (n-k)^2\mathbb{E}\left[\left\|\frac{1}{n-k}\sum_{i=k}^{n-1}\nabla f_{\pi_i}(x_*)\right\|^2\right]\\
&= (n-k)^2\mathbb{E}[\|\bar{X}_\pi\|^2]\\
&= (n-k)^2\big(\|\bar{X}\|^2+\mathbb{E}[\|\bar{X}_\pi-\bar{X}\|^2]\big)\\
&= (n-k)^2\left[\|\nabla f(x_*)\|^2+\mathbb{E}\left[\left\|\frac{1}{n-k}\sum_{i=k}^{n-1}\nabla f_{\pi_i}(x_*)-\nabla f(x_*)\right\|^2\right]\right]\\
&= (n-k)^2\|\nabla f(x_*)\|^2+(n-k)\frac{k}{(n-1)}\sigma_*^2,
\end{aligned}
$$

where the third equality holds due to the property $\mathbb{E}[\|x-\mathbb{E}[x]\|^2]=\mathbb{E}[\|x\|^2]-\|\mathbb{E}[x]\|^2$ and Lemma 1. Summing the above equality over k from 0 to $n-1$,

$$
\begin{aligned}
&\sum_{k=0}^{n-1}\mathbb{E}\left[\left\|\sum_{i=k}^{n-1}\nabla f_{\pi_i}(x_*)\right\|^2\right]\\
&=\sum_{k=0}^{n-1}\left[(n-k)^2\|\nabla f(x_*)\|^2+\frac{k(n-k)}{n-1}\sigma_*^2\right]\\
&=\frac{n(n+1)(2n+1)}{6}\|\nabla f(x_*)\|^2+\frac{1}{6}n(n+1)\sigma_*^2\\
&\leqslant\frac{2n^3}{3}G_f^2+\frac{n^2\sigma_*^2}{4},
\end{aligned}
$$

where the last inequality holds due to $n\geqslant 2$. Thus, we obtain (13).

Proof of Theorem 1. It follows from (14) that

$$
\begin{aligned}
&\|x_t-x_*\|^2\\
&=\|\gamma n d_{t+1}+x_{t+1}+\gamma g_t-x_*\|^2\\
&\geqslant\|x_{t+1}-x_*\|^2+2\langle x_{t+1}-x_*,\gamma n d_{t+1}\rangle+2\langle x_{t+1}-x_*,\gamma g_t\rangle\\
&=\|x_{t+1}-x_*\|^2+2\langle x_{t+1}-x_*,\gamma n d_{t+1}\rangle+2\gamma\sum_{i=0}^{n-1}\langle x_{t+1}-x_*,\nabla f_{\pi_i}(x_t^i)\rangle,
\end{aligned}
\tag{15}
$$

where $d_{t+1}\in\partial_{\varepsilon_{t+1}}\phi(x_{t+1})$ and $g_t\triangleq\sum_{i=0}^{n-1}\nabla f_{\pi_i}(x_t^i)$. For any i,

$$
\begin{aligned}
&\langle\nabla f_{\pi_i}(x_t^i),x_{t+1}-x_*\rangle\\
&=[f_{\pi_i}(x_{t+1})-f_{\pi_i}(x_*)]+[f_{\pi_i}(x_*)-f_{\pi_i}(x_t^i)-\langle\nabla f_{\pi_i}(x_t^i),x_*-x_t^i\rangle]\\
&\quad-[f_{\pi_i}(x_{t+1})-f_{\pi_i}(x_t^i)-\langle\nabla f_{\pi_i}(x_t^i),x_{t+1}-x_t^i\rangle]\\
&=[f_{\pi_i}(x_{t+1})-f_{\pi_i}(x_*)]+D_{f_{\pi_i}}(x_*,x_t^i)-D_{f_{\pi_i}}(x_{t+1},x_t^i).
\end{aligned}
\tag{16}
$$

Summing the first quantity in the right-hand side of (16) over i from 0 to $n-1$ gives

$$
\sum_{i=0}^{n-1}[f_{\pi_i}(x_{t+1})-f_{\pi_i}(x_*)]=n(f(x_{t+1})-f_*).
\tag{17}
$$

Next, because Assumption 1 (2) holds, we have

$$
D_{f_{\pi_i}}(x_{t+1},x_t^i)\leqslant\frac{L}{2}\|x_{t+1}-x_t^i\|^2.
$$

By taking the expectation of the above inequality and summing over i from 0 to $n-1$, we obtain an upper bound

$$
\sum_{i=0}^{n-1}\mathbb{E}[D_{f_{\pi_i}}(x_{t+1},x_t^i)]\leqslant\frac{L}{2}\mathbb{E}[\mathcal{V}_t]
$$

$$\leqslant 3\gamma^2L^2n^2\sum_{i=0}^{n-1}\mathbb{E}[D_{f_{\pi_i}}(x_*,x_t^i)]+L\gamma^2n^3G_f^2+\frac{3}{8}\gamma^2Ln^2\sigma_*^2+\frac{3L}{2}\gamma^2n^3G_\phi^2,$$

where the last inequality holds by Lemma 2. Then, with the above inequality, we can bound the sum of the second and third term in (16) as

$$\begin{aligned}&\sum_{i=0}^{n-1}\mathbb{E}[D_{f_{\pi_i}}(x_*,x_t^i)-D_{f_{\pi_i}}(x_{t+1},x_t^i)]\\&\geqslant(1-3\gamma^2L^2n^2)\sum_{i=0}^{n-1}\mathbb{E}[D_{f_{\pi_i}}(x_*,x_t^i)]-L\gamma^2n^3G_f^2-\frac{3}{8}\gamma^2Ln^2\sigma_*^2-\frac{3L}{2}\gamma^2n^3G_\phi^2\\&\geqslant-L\gamma^2n^3G_f^2-\frac{3}{8}\gamma^2Ln^2\sigma_*^2-\frac{3L}{2}\gamma^2n^3G_\phi^2,\end{aligned}\tag{18}$$

where the last inequality holds since $(1-3\gamma^2L^2n^2)\geqslant 0$ and $D_{f_{\pi_i}}(x_*,x_t^i)\geqslant 0$.

Taking expectation of (15) and using (16)–(18) gives

$$\begin{aligned}&\mathbb{E}[\|x_{t+1}-x_*\|^2]\\&\leqslant\mathbb{E}[\|x_t-x_*\|^2]-2\gamma n\mathbb{E}[f(x_{t+1})-f_*]-2\gamma n\mathbb{E}[\langle x_{t+1}-x_*,d_{t+1}\rangle]\\&\quad+2\gamma^3Ln^3G_f^2+\frac{3}{4}\gamma^3Ln^2\sigma_*^2+3L\gamma^3n^3G_\phi^2\\&\leqslant\mathbb{E}[\|x_t-x_*\|^2]-2\gamma n\mathbb{E}[f(x_{t+1})-f_*]-2\gamma n\mathbb{E}[\phi(x_{t+1})-\phi_*]\\&\quad+\gamma^3Ln^3(2G_f^2+3G_\phi^2)+\frac{3}{4}\gamma^3Ln^2\sigma_*^2,\end{aligned}$$

where the second inequality holds because $\langle x_{t+1}-x_*,d_{t+1}\rangle\geqslant\phi(x_{t+1})-\phi_*$. By rearrangement, we obtain

$$2\gamma n\mathbb{E}[F(x_{t+1})-F_*]\leqslant\mathbb{E}[\|x_t-x_*\|^2]-\mathbb{E}[\|x_{t+1}-x_*\|^2]+\frac{3}{4}\gamma^3Ln^2\sigma_*^2+\gamma^3Ln^3(2G_f^2+3G_\phi^2).$$

Summing over t from 0 to $T-1$ gives

$$\begin{aligned}&2\gamma n\sum_{t=0}^{T-1}\mathbb{E}[F(x_{t+1})-F_*]\\&\leqslant\sum_{t=0}^{T-1}\left(\mathbb{E}[\|x_t-x_*\|^2]-\mathbb{E}[\|x_{t+1}-x_*\|^2]\right)+\frac{3}{4}\gamma^3LTn^2\sigma_*^2+\gamma^3Ln^3T(2G_f^2+3G_\phi^2)\\&=\|x_0-x_*\|^2-\|x_T-x_*\|^2+\frac{3}{4}\gamma^3LTn^2\sigma_*^2+\gamma^3Ln^3T(2G_f^2+3G_\phi^2)\\&\leqslant\|x_0-x_*\|^2+\frac{3}{4}\gamma^3LTn^2\sigma_*^2+\gamma^3Ln^3T(2G_f^2+3G_\phi^2),\end{aligned}$$

and dividing both sides by $2\gamma nT$ gives

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[F(x_{t+1})-F_*]\leqslant\frac{\|x_0-x_*\|^2}{2\gamma nT}+\frac{3}{8}\gamma^2Ln\sigma_*^2+L\gamma^2n^2\left(G_f^2+\frac{3}{2}G_\phi^2\right).$$

Finally, using the convexity of F, the average iterate $\tilde{x}_T\triangleq\frac{1}{T}\sum_{t=1}^{T}x_t$ satisfies

$$\begin{aligned}&\mathbb{E}[F(\tilde{x}_T)-F_*]\\&\leqslant\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[F(x_t)-F_*]\\&\leqslant\frac{\|x_0-x_*\|^2}{2\gamma nT}+\frac{3}{8}\gamma^2Ln\sigma_*^2+L\gamma^2n^2\left(G_f^2+\frac{3}{2}G_\phi^2\right).\end{aligned}$$

4 Theoretical analysis of IPG-RR algorithm

This section provides the theoretical proofs for Algorithm 2. Since there exist inexact proximal operators of non-differentiable function ϕ in Algorithm 2, the following vital lemma firstly characterizes the ε_t-subdifferential of ϕ at x_t, $\partial_{\varepsilon_t}\phi(x_t)$, defined in (6).

Lemma 3 (Lemma 2 in [31])**.** If x_t is an ε_t-optimal solution to (3) in the sense of (5) with $y = x_{t-1} - \gamma(\nabla g(x_{t-1}) + e_t)$, then there exists $p_t \in \mathbb{R}^d$ such that $\|p_t\| \leqslant \sqrt{2n\gamma\varepsilon_t}$ and

$$\frac{1}{n\gamma}(x_{t-1} - x_t - \gamma\nabla g(x_{t-1}) - \gamma e_t - p_t) \in \partial_{\varepsilon_t}\phi(x_t).$$

With Lemma 3, we show that the forward per-epoch deviation $\mathcal{V}_t$ of the IPG-RR algorithm is upper bounded despite the existence of calculation errors in gradients and proximal operator.

Lemma 4. Consider the iterates of Algorithm 2. If Assumption 1 holds, then

$$\begin{aligned}\mathbb{E}[\mathcal{V}_t] \leqslant & 24\gamma^2 Ln^2 \sum_{i=0}^{n-1} \mathbb{E}[D_{f_{\pi_i}}(x_*, x_t^i)] + 8\gamma^2 n^3 G_f^2 + 3\gamma^2 n^2 \sigma_*^2 \\ & + 6\gamma^2 n^3 G_\phi^2 + 6\gamma^2 n\|e_{t+1}\|^2 + 2n\|p_{t+1}\|^2. \end{aligned} \tag{19}$$

Proof. It follows from lines 6–10 in Algorithm 2 and the optimality condition in (4) that

$$x_t - \gamma g_t - x_{t+1} - \gamma e_{t+1} - p_{t+1} = n\gamma d_{t+1}, \tag{20}$$

where $d_{t+1} \in \partial_{\varepsilon_{t+1}}\phi(x_{t+1})$, $g_t \triangleq \sum_{i=0}^{n-1} \nabla f_{\pi_i}(x_t^i)$ and $e_{t+1} \triangleq \sum_{i=0}^{n-1} e_{\pi_i}(x_t^i)$.

In addition, by iterates, $x_t^k = x_t - \gamma \sum_{i=0}^{k-1} \nabla f_{\pi_i}(x_t^i)$. Then,

$$\begin{aligned} & x_t^k - x_{t+1} \\ & = x_t - \gamma \sum_{i=0}^{k-1} \nabla f_{\pi_i}(x_t^i) - \left(x_t - \gamma \sum_{i=0}^{n-1} \nabla f_{\pi_i}(x_t^i) - \gamma e_{t+1} - p_{t+1} - \gamma n d_{t+1} \right) \\ & = \gamma \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_t^i) + \gamma e_{t+1} + p_{t+1} + \gamma n d_{t+1}. \end{aligned}$$

Taking the norm of $x_t^k - x_{t+1}$ yields

$$\begin{aligned} \|x_t^k - x_{t+1}\|^2 = & \left\| \gamma \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_t^i) + \gamma e_{t+1} + p_{t+1} + \gamma n d_{t+1} \right\|^2 \\ \leqslant & 2 \left\| \gamma \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_t^i) + \gamma e_{t+1} + \gamma n d_{t+1} \right\|^2 + 2\|p_{t+1}\|^2 \\ \leqslant & 6\gamma^2 \left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_t^i) \right\|^2 + 6\gamma^2 \|e_{t+1}\|^2 + 6\gamma^2 \|n d_{t+1}\|^2 + 2\|p_{t+1}\|^2 \\ \leqslant & 6\gamma^2 \left(2 \left\| \sum_{i=k}^{n-1} (\nabla f_{\pi_i}(x_t^i) - \nabla f_{\pi_i}(x_*)) \right\|^2 + 2 \left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) \right\|^2 \right) \\ & + 6\gamma^2 \|e_{t+1}\|^2 + 6\gamma^2 \|n d_{t+1}\|^2 + 2\|p_{t+1}\|^2 \\ \leqslant & 6\gamma^2 \left(2n \sum_{i=k}^{n-1} \|\nabla f_{\pi_i}(x_t^i) - \nabla f_{\pi_i}(x_*)\|^2 + 2 \left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) \right\|^2 \right) \\ & + 6\gamma^2 \|e_{t+1}\|^2 + 6\gamma^2 \|n d_{t+1}\|^2 + 2\|p_{t+1}\|^2 \\ \leqslant & 6\gamma^2 \left(4Ln \sum_{i=0}^{n-1} D_{f_{\pi_i}}(x_*, x_t^i) + 2 \left\| \sum_{i=k}^{n-1} \nabla f_{\pi_i}(x_*) \right\|^2 \right) \end{aligned}$$

$$+ 6\gamma^2\|e_{t+1}\|^2 + 6\gamma^2\|nd_{t+1}\|^2 + 2\|p_{t+1}\|^2,$$

where the fourth inequality holds due to $\|\sum_{i=k}^{n-1} a_i\|^2 \leqslant (n-k)\sum_{i=k}^{n-1}\|a_i\|^2$ and the last inequality is justified by (11). Next, we consider the term $\|\sum_{i=k}^{n-1}\nabla f_{\pi_i}(x_*)\|^2$. With Lemma 1,

$$\begin{aligned}
&\mathbb{E}\left[\left\|\sum_{i=k}^{n-1}\nabla f_{\pi_i}(x_*)\right\|^2\right]\\
&= (n-k)^2\mathbb{E}\left[\left\|\frac{1}{n-k}\sum_{i=k}^{n-1}\nabla f_{\pi_i}(x_*)\right\|^2\right]\\
&= (n-k)^2\left[\|\nabla f(x_*)\|^2+\mathbb{E}\left[\left\|\frac{1}{n-k}\sum_{i=k}^{n-1}\nabla f_{\pi_i}(x_*)-\nabla f(x_*)\right\|^2\right]\right]\\
&= (n-k)^2\|\nabla f(x_*)\|^2 + (n-k)\frac{k}{(n-1)}\sigma_*^2.
\end{aligned}$$

Summing this over k from 0 to $n-1$,

$$\begin{aligned}
&\sum_{k=0}^{n-1}\mathbb{E}\left[\left\|\sum_{i=k}^{n-1}\nabla f_{\pi_i}(x_*)\right\|^2\right]\\
&= \sum_{k=0}^{n-1}\left[(n-k)^2\|\nabla f(x_*)\|^2 + \frac{k(n-k)}{n-1}\sigma_*^2\right]\\
&= \frac{n(n+1)(2n+1)}{6}\|\nabla f(x_*)\|^2 + \frac{1}{6}n(n+1)\sigma_*^2\\
&\leqslant \frac{2n^3}{3}G_f^2 + \frac{n^2\sigma_*^2}{4},
\end{aligned}$$

where the last inequality holds due to $n \geqslant 2$. Thus, we obtain (19).

Before providing the proof of Theorem 2, we give a lemma about the bound of one special nonnegative sequence.

Lemma 5 (Lemma 1 in [31])**.** Assume that the nonnegative sequence $\{u_T\}$ satisfies the following recursion for all $T \geqslant 1$:

$$u_T^2 \leqslant S_T + \sum_{t=1}^{T}\lambda_t u_t \tag{21}$$

with $\{S_T\}$ an increasing sequence, $S_0 \geqslant u_0^2$ and $\lambda_t \geqslant 0$ for all t. Then, for all $T \geqslant 1$,

$$u_T \leqslant \frac{1}{2}\sum_{t=1}^{T}\lambda_t + \left(S_T + \left(\frac{1}{2}\sum_{t=1}^{T}\lambda_t\right)^2\right)^{\frac{1}{2}}. \tag{22}$$

Proof of Theorem 2. It follows from (20) that

$$\begin{aligned}
&\|x_t - x_*\|^2\\
&= \|\gamma g_t + \gamma e_{t+1} + p_{t+1} + \gamma n d_{t+1} + x_{t+1} - x_*\|^2\\
&\geqslant \|x_{t+1} - x_*\|^2 + 2\gamma\sum_{i=0}^{n-1}\langle x_{t+1} - x_*, \nabla f_{\pi_i}(x_t^i)\rangle\\
&\quad + 2\langle x_{t+1} - x_*, \gamma e_{t+1} + p_{t+1} + \gamma n d_{t+1}\rangle,
\end{aligned} \tag{23}$$

where $d_{t+1} \in \partial_{\varepsilon_{t+1}}\phi(x_{t+1})$, $g_t \triangleq \sum_{i=0}^{n-1}\nabla f_{\pi_i}(x_t^i)$ and $e_{t+1} \triangleq \sum_{i=0}^{n-1} e_{\pi_i}(x_t^i)$.

Take the second term in (23). For any i,

$$\begin{aligned}
&\langle \nabla f_{\pi_i}(x_t^i), x_{t+1} - x_* \rangle \\
&= [f_{\pi_i}(x_{t+1}) - f_{\pi_i}(x_*)] + [f_{\pi_i}(x_*) - f_{\pi_i}(x_t^i) - \langle \nabla f_{\pi_i}(x_t^i), x_* - x_t^i \rangle] \\
&\quad - [f_{\pi_i}(x_{t+1}) - f_{\pi_i}(x_t^i) - \langle \nabla f_{\pi_i}(x_t^i), x_{t+1} - x_t^i \rangle] \\
&= [f_{\pi_i}(x_{t+1}) - f_{\pi_i}(x_*)] + D_{f_{\pi_i}}(x_*, x_t^i) - D_{f_{\pi_i}}(x_{t+1}, x_t^i). \qquad (24)
\end{aligned}$$

Summing the first quantity in (24) over i from 0 to $n-1$ gives

$$\sum_{i=0}^{n-1} [f_{\pi_i}(x_{t+1}) - f_{\pi_i}(x_*)] = n(f(x_{t+1}) - f_*). \qquad (25)$$

Next, we bound the third term in (24) with L-smoothness of f_i as follows:

$$D_{f_{\pi_i}}(x_{t+1}, x_t^i) \leqslant \frac{L}{2} \|x_{t+1} - x_t^i\|^2.$$

By summing over i from 0 to $n-1$, we get the forward deviation over an epoch $\mathcal{V}_t$,

$$\begin{aligned}
&\sum_{i=0}^{n-1} \mathbb{E}[D_{f_{\pi_i}}(x_{t+1}, x_t^i)] \leqslant \frac{L}{2} \mathbb{E}[\mathcal{V}_t] \\
&\leqslant 12\gamma^2 L^2 n^2 \sum_{i=0}^{n-1} \mathbb{E}[D_{f_{\pi_i}}(x_*, x_t^i)] + 3\gamma^2 L \frac{n^2 \sigma_*^2}{2} + 4\gamma^2 n^3 L G_f^2 \\
&\quad + 3\gamma^2 n^3 L G_\phi^2 + 3\gamma^2 L n \|e_{t+1}\|^2 + L n \|p_{t+1}\|^2,
\end{aligned}$$

where we use the result in Lemma 4. Then, we bound the sum of the second and third term in (24) as

$$\begin{aligned}
&\sum_{i=0}^{n-1} \mathbb{E}[D_{f_{\pi_i}}(x_*, x_t^i) - D_{f_{\pi_i}}(x_{t+1}, x_t^i)] \\
&\geqslant (1 - 12\gamma^2 L^2 n^2) \sum_{i=0}^{n-1} \mathbb{E}[D_{f_{\pi_i}}(x_*, x_t^i)] - 4\gamma^2 n^3 L G_f^2 \\
&\quad - 3\gamma^2 L \frac{n^2 \sigma_*^2}{2} - 3\gamma^2 n^3 L G_\phi^2 - 3\gamma^2 L n \|e_{t+1}\|^2 - L n \|p_{t+1}\|^2 \\
&\geqslant -3\gamma^2 L \frac{n^2 \sigma_*^2}{2} - 4\gamma^2 n^3 L G_f^2 - 3\gamma^2 n^3 L G_\phi^2 - 3\gamma^2 L n \|e_{t+1}\|^2 - L n \|p_{t+1}\|^2, \qquad (26)
\end{aligned}$$

where the last inequality holds since $(1 - 12\gamma^2 L^2 n^2) \geqslant 0$ and $D_{f_{\pi_i}}(x_*, x_t^i)$ is non-negative.

Rearranging (23) gives

$$\begin{aligned}
&\|x_{t+1} - x_*\|^2 \\
&\overset{(24),(25)}{\leqslant} \|x_t - x_*\|^2 - 2\gamma n (f(x_{t+1}) - f_*) - 2\gamma \sum_{i=0}^{n-1} \big(D_{f_{\pi_i}}(x_*, x_t^i) - D_{f_{\pi_i}}(x_{t+1}, x_t^i) \big) \\
&\quad - 2\gamma \langle x_{t+1} - x_*, n d_{t+1} \rangle - 2 \langle x_{t+1} - x_*, \gamma e_{t+1} + p_{t+1} \rangle \\
&\leqslant \|x_t - x_*\|^2 - 2\gamma n (f(x_{t+1}) - f_*) - 2\gamma \sum_{i=0}^{n-1} \big(D_{f_{\pi_i}}(x_*, x_t^i) - D_{f_{\pi_i}}(x_{t+1}, x_t^i) \big) \\
&\quad - 2\gamma \langle x_{t+1} - x_*, n d_{t+1} \rangle + 2 \|x_{t+1} - x_*\| \big(\gamma \|e_{t+1}\| + \|p_{t+1}\| \big).
\end{aligned}$$

Taking expectation of the above inequality and using (26) yields

$$\begin{aligned}
&\mathbb{E}[\|x_{t+1} - x_*\|^2] \\
&\leqslant \mathbb{E}[\|x_t - x_*\|^2] - 2\gamma n \mathbb{E}[f(x_{t+1}) - f_*] + 3\gamma^3 L n^2 \sigma_*^2 + 8\gamma^3 n^3 L G_f^2
\end{aligned}$$

$$
\begin{aligned}
&+6\gamma^3n^3LG_\phi^2+6\gamma^3Ln\|e_{t+1}\|^2+2\gamma Ln\|p_{t+1}\|^2-2\gamma\mathbb{E}[\langle x_{t+1}-x_*,nd_{t+1}\rangle]\\
&+2\mathbb{E}[\|x_{t+1}-x_*\|](\gamma\|e_{t+1}\|+\|p_{t+1}\|)\\
\leqslant\ &\mathbb{E}[\|x_t-x_*\|^2]-2\gamma n\mathbb{E}[f(x_{t+1})-f_*]+3\gamma^3Ln^2\sigma_*^2+8\gamma^3n^3LG_f^2\\
&+6\gamma^3n^3LG_\phi^2+6\gamma^3Ln\|e_{t+1}\|^2+2\gamma Ln\|p_{t+1}\|^2-2\gamma n\mathbb{E}[\phi(x_{t+1})-\phi_*]\\
&+2\gamma n\varepsilon_{t+1}+2\|x_{t+1}-x_*\|(\gamma\|e_{t+1}\|+\|p_{t+1}\|),
\end{aligned}
$$

where the second inequality holds since $\langle x_{t+1}-x_*,d_{t+1}\rangle+\varepsilon_{t+1}\geqslant\phi(x_{t+1})-\phi_*$.

Then, by rearranging the above inequality,

$$
\begin{aligned}
&2\gamma n\mathbb{E}[F(x_{t+1})-F_*]\\
&\leqslant\mathbb{E}[\|x_t-x_*\|^2]-\mathbb{E}[\|x_{t+1}-x_*\|^2]+3\gamma^3Ln^2\sigma_*^2\\
&\quad+8\gamma^3n^3LG_f^2+6\gamma^3n^3LG_\phi^2+6\gamma^3Ln\|e_{t+1}\|^2+2\gamma Ln\|p_{t+1}\|^2\\
&\quad+2\gamma n\varepsilon_{t+1}+2\mathbb{E}[\|x_{t+1}-x_*\|](\gamma\|e_{t+1}\|+\|p_{t+1}\|).
\end{aligned}
$$

Summing over t from 0 to $T-1$,

$$
\begin{aligned}
&2\gamma n\sum_{t=0}^{T-1}\mathbb{E}[F(x_{t+1})-F_*]\\
&\leqslant\|x_0-x_*\|^2-\|x_T-x_*\|^2+3\gamma^3LTn^2\sigma_*^2+8\gamma^3n^3LTG_f^2\\
&\quad+6\gamma^3n^3LTG_\phi^2+6\gamma^3Ln\sum_{t=0}^{T-1}\|e_{t+1}\|^2+2\gamma n(1+2\gamma nL)\sum_{t=0}^{T-1}\varepsilon_{t+1}\\
&\quad+2\sum_{t=0}^{T-1}\mathbb{E}[\|x_{t+1}-x_*\|](\gamma\|e_{t+1}\|+\|p_{t+1}\|),
\end{aligned}
\tag{27}
$$

where we use the fact that $\|p_{t+1}\|^2\leqslant 2n\gamma\varepsilon_{t+1}$.

Next, we provide the upper bound of $\mathbb{E}[\|x_{t+1}-x_*\|]$. By simple transformation of (27) and the fact that $\mathbb{E}[\|x_T-x_*\|]^2\leqslant\mathbb{E}[\|x_T-x_*\|^2]$,

$$
\begin{aligned}
&\mathbb{E}[\|x_T-x_*\|]^2\\
&\leqslant\|x_0-x_*\|^2+\frac{\gamma T}{4L}\sigma_*^2+\frac{2\gamma nTG_f^2}{3L}+\frac{\gamma nTG_\phi^2}{2L}+6\gamma^3Ln\sum_{t=1}^{T}\|e_t\|^2\\
&\quad+2\gamma n(1+2\gamma nL)\sum_{t=1}^{T}\varepsilon_t+2\sum_{t=1}^{T}\mathbb{E}[\|x_t-x_*\|](\gamma\|e_t\|+\|p_t\|),
\end{aligned}
\tag{28}
$$

where we use $12\gamma^2L^2n^2\leqslant 1$. Let $S_T=\|x_0-x_*\|^2+\frac{\gamma T}{4L}\sigma_*^2+\frac{2\gamma nTG_f^2}{3L}+\frac{\gamma nTG_\phi^2}{2L}+6\gamma^3Ln\sum_{t=1}^{\mathrm{T}}\|e_t\|^2+2\gamma n(1+2\gamma nL)\sum_{t=1}^{T}\varepsilon_t$, $u_t=\mathbb{E}[\|x_t-x_*\|]$ and $\lambda_t=2(\gamma\|e_t\|+\|p_t\|)$. It follows from Lemma 5 and (28),

$$
\mathbb{E}[\|x_T-x_*\|]\leqslant\sum_{t=1}^{T}(\gamma\|e_t\|+p_t)+\left(S_T+\left(\sum_{t=1}^{T}(\gamma\|e_t\|+\|p_t\|)\right)^2\right)^{\frac{1}{2}}.
$$

Denoting $A_T=\sum_{t=1}^{T}(\gamma\|e_t\|+\|p_t\|)$ and $D_T=\frac{\gamma T}{4L}\sigma_*^2+\frac{2\gamma nTG_f^2}{3L}+\frac{\gamma nTG_\phi^2}{2L}+6\gamma^3Ln\sum_{t=1}^{T}\|e_t\|^2+2\gamma n(1+2\gamma nL)\sum_{t=1}^{T}\varepsilon_t$, we get

$$
\mathbb{E}[\|x_T-x_*\|]\leqslant A_T+(\|x_0-x_*\|^2+D_T+A_T^2)^{\frac{1}{2}}.
$$

Since A_t and D_t are increasing sequences, we have for $t\leqslant T$,

$$
\mathbb{E}[\|x_t-x_*\|]\leqslant A_T+(\|x_0-x_*\|^2+D_T+A_T^2)^{\frac{1}{2}}
$$

$$\leqslant 2A_T + \|x_0 - x_*\| + \sqrt{D_T}. \tag{29}$$

Now, we can bound the right-hand side of (28) with (29) as follows:

$$2\gamma n \sum_{t=0}^{T-1} \mathbb{E}[F(x_{t+1}) - F_*] \leqslant [\|x_0 - x_*\|^2] + D_T + 2A_T(2A_T + \|x_0 - x_*\| + \sqrt{D_T}).$$

Dividing both sides by $2\gamma nT$, we get

$$\frac{1}{T}\sum_{t=0}^{T-1} E(F(x_{t+1}) - F_*) \leqslant \frac{\|x_0 - x_*\|^2}{2\gamma nT} + \frac{D_T}{2\gamma nT} + \frac{2A_T(2A_T + \|x_0 - x_*\| + \sqrt{D_T})}{2\gamma nT}.$$

Finally, using the convexity of F, the average iterate $\tilde{x}_T \triangleq \frac{1}{T}\sum_{t=1}^{T} x_t$ satisfies

$$\begin{aligned} E(F(\tilde{x}_T) - F_*) &\leqslant \frac{1}{T}\sum_{t=1}^{T} E(F(x_t) - F_*) \\ &\leqslant \frac{\|x_0 - x_*\|^2}{2\gamma nT} + \frac{2A_T(2A_T + \|x_0 - x_*\| + \sqrt{D_T}) + D_T}{2\gamma nT}. \end{aligned}$$

By rearrangement, we obtain the desired result.

5 Numerical simulation

In this section, we apply the proposed algorithms to compressed sensing, whose measurement process can be modeled as one linear algebraic equation $y = Ax$, where $x \in \mathbb{R}^d$ is the sparse signal, $y \in \mathbb{R}^m$ is the compressed signal, $A \in \mathbb{R}^{m\times d}$ $(m < d)$ is the observation matrix. The sparse signal recovery problem under one single observation matrix A is the l_0-norm minimization model with an equality constraint,

$$\min_{x\in\mathbb{R}^d} \|x\|_0, \text{ s.t. } y = Ax, \tag{30}$$

where $\|x\|_0$ denotes the number of non-zero elements in x. Since the l_0-norm minimization is non-deterministic polynomial-time hard (NP-hard), in practice, the l_1-norm is usually used to replace l_0-norm to find the sparse solution [42]. The recovery problem (30) is rewritten into an unconstrained form as

$$\min_{x\in\mathbb{R}^d} F(x) = f(x) + \phi(x) = \|y - Ax\|^2 + \lambda \|x\|_1,$$

where λ is a regularized parameter. If there are multiple sensors measuring the same signal, the sparse signal recovery problem under multiple observation matrices can be expressed as

$$\min_{x\in\mathbb{R}^d} F(x) = f(x) + \phi(x) = \frac{1}{n}\sum_{i=1}^{n} \|y_i - A_i x\|^2 + \lambda \|x\|_1, \tag{31}$$

where the pair $\{A_i, y_i\}$ corresponds to the ith sensor. Multiple observed information pairs $\{A_i, y_i\}_{i=1}^n$ are used to recover the original sparse signal x.

In compressed sensing, the recovery algorithms are mainly divided into two categories: greedy algorithms and convex optimization algorithms. Compared with the intuitive greedy algorithms [43, 44], the convex optimization algorithms require a smaller number of observations and have higher recovery accuracy, such as projected gradient algorithms [45] and ADMM algorithms [46]. In this section, we compare the proposed algorithms with some existing popular first-order optimization algorithms to solve (31).

In the simulation, we conduct experiments using the real MNIST dataset, where the image matrix can be vectorized into a signal vector s with the dimension $28 \times 28 = 784$. At first, we sparse the selected handwriting image and obtain the sparse basis $\Phi \in \mathbb{R}^{784\times 100}$, so that $s = \Phi x$, where x is the sparse signal with a lower dimension. With a set of information pairs $\{O_i, y_i\}_{i=1}^n$, where $y_i \in \mathbb{R}^{100}$ is the known compressed signal and $O_i \in \mathbb{R}^{100\times 784}$ is the observation matrix, we aim to recover the sparse signal $x \in \mathbb{R}^{100}$. In this context, the matrix A_i in (31) is $A_i \triangleq O_i\Phi \in \mathbb{R}^{100\times 100}$, $n = 10$ and $\lambda = 10^{-5}$. To verify

the efficiency of the proposed stochastic algorithms, we compare PG-RR and IPG-RR with three existing popular optimization algorithms, including proximal gradient (PG) algorithm [47], SPG algorithm, block proximal gradient (B-PG) algorithm [48], and alternating direction method of multipliers (ADMM) [46]. The brief descriptions of different comparative algorithms are introduced as follows:

(a) PG. Stacking multiple observed information, we obtain the stacked observation matrix $A = [A_1^{\mathrm{T}}, \ldots, A_n^{\mathrm{T}}]^{\mathrm{T}}$ and the stacked compressed signal $y = [y_1^{\mathrm{T}}, \ldots, y_n^{\mathrm{T}}]^{\mathrm{T}}$. In each iteration of PG, one gradient descent step and one proximal operator are performed with the stacked data A and y.

(b) B-PG. In each iteration, for a fixed permutation of $\{(A_1, y_1), \ldots, (A_n, y_n)\}$, one gradient descent and one proximal operator are performed with each pair (A_i, y_i) successively. After all information pairs pass through, the next iteration is executed. Note that the proposed PG-RR algorithm uses the RR permutation at each iteration. B-PG is one special case of our proposed PG-RR algorithm. In addition, the work [48] does not provide a rigorous proof of convergence rate for B-PG.

(c) SPG. The algorithm design is similar to B-PG. The only difference is that the fixed permutation in B-PG is replaced by an unordered sample set of length n, which is uniformly sampled with replacement from the global dataset $\{A_i, y_i\}_{i=1}^n$. This sampling scheme means that there may exist repeated elements in the generated set.

(d) ADMM. Stacking multiple observed information A_i and y_i yields the stacked observation matrix A and the stacked compressed signal y, respectively. In each iteration, the primal and dual updating of traditional ADMM is performed with the stacked data A and y.

Then, we compare the performance of the above algorithms with the proposed PG-RR algorithm in three settings: exact algorithms (no error), inexact algorithms with diminishing errors, and inexact algorithms with constant errors.

(1) No error setting. In this setting, the error is defined as $\frac{1}{n}\left\|x - x^*\right\|^2$, where x^* is the true sparse signal, and x is the recovery sparse signal. Each algorithm takes a proper step size according to the performance. The convergence results of the comparative algorithms are shown in Figures 1(a)–(c). Figure 1(a) shows that the error trajectory of PG-RR diminishes more rapidly with the number of iterations than the other algorithms, achieving the highest final accuracy. Figures 1(b) and (c) illustrate that PG-RR owns a better computational performance than the others. Notably, the error trajectories in terms of the number of gradient computations align with those in terms of iterations shown in Figure 1(a), since all algorithms process n gradient computations per iteration. Figure 1(b) indicates that the proposed PG-RR algorithm has better computational complexity in terms of proximal operators. Furthermore, the faster convergence of the PG-RR algorithm in comparison to the B-PG algorithm verifies that RR permutations enhance convergence efficacy over fixed permutations.

(2) Diminishing errors setting. This setting considers that the compressed signal y is affected by a disturbance term $e = C \cdot r^t$, where t is the iteration number, and r is a Gaussian vector. This deliberate inclusion of a disturbance term introduces error terms in the gradient computations. Gaussian distributions are commonly used to model noise and disturbances in various systems and have well-defined statistical properties. All other parameters are the same as those in the no error setting. Due to these computational errors, the comparative algorithms are rendered inexact. The convergence results of the inexact comparative algorithms are shown in Figures 2(a)–(d). The convergence performance of the inexact algorithms varies with different amplitudes C. Figures 2(a) and (c) show that the proposed IPG-RR converges faster than the other comparative algorithms for $C = 20$ and $C = 50$. Additionally, Figures 2(b) and (d) demonstrate that the proposed IPG-RR algorithm exhibits superior computing efficiency in scenarios where there are diminishing errors in the computation.

(3) Constant error setting. When the compressed signal y is affected by disturbance with a constant upper bound, indicated as $e = C \cdot r$ with r being a Gaussian vector, we evaluate the convergence performance of different inexact algorithms. Other parameters remain unchanged from the no error setting. The convergence results are shown in Figures 3(a)–(d). Figures 3(a) and (c) show that the proposed IPG-RR converges faster than the other comparative algorithms when $C = 3$ and $C = 5$. Figures 3(b) and (d) illustrate that the proposed IPG-RR algorithm maintains superior computing efficiency even in the presence of constant computation errors. The faster convergence of IPG-RR than B-IPG also demonstrates the superiority of RR permutations over fixed permutations.

We compare the recovered signal obtained from the proposed PG-RR algorithm with the original true sparse signal in Figure 4. The recovered signal is very similar to the true sparse signal. In Figure 5(a), we show the original sparse image, which is obtained by the sparse operation to the selected handwriting image. The recovery images generated by different comparable algorithms are shown in Figures 5(b)–(f).

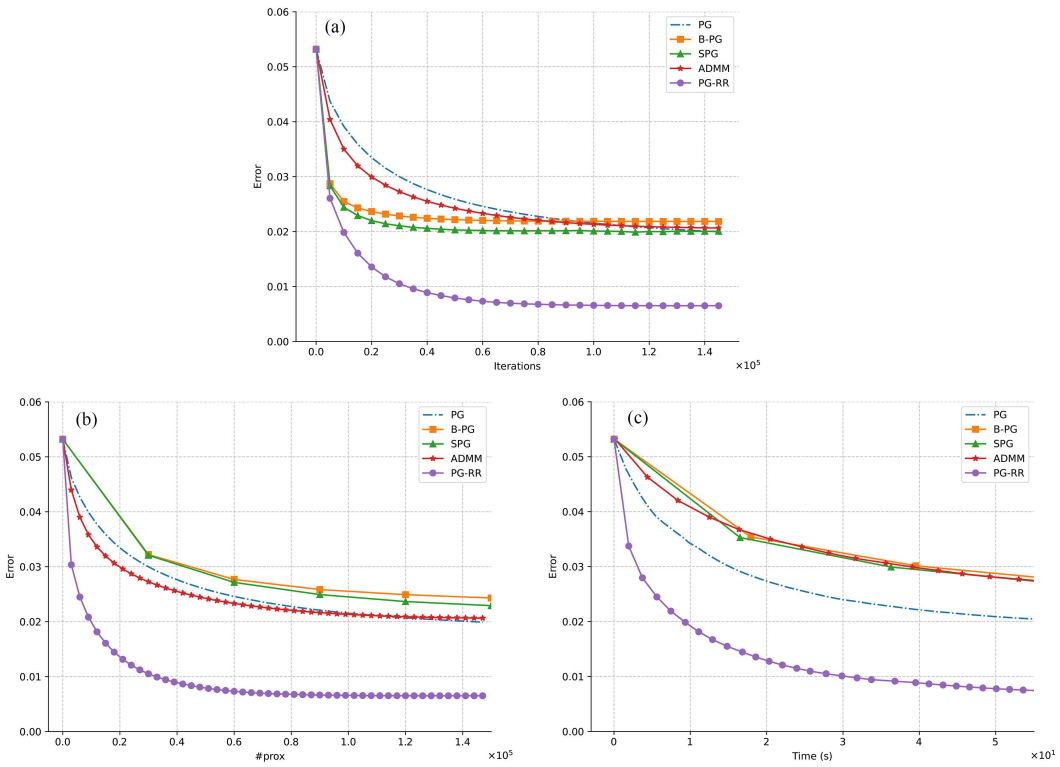

Figure 1 (Color online) (a) Convergence of exact comparative algorithms; (b) convergence of exact comparative algorithms in terms of proximal operators; (c) convergence of exact comparative algorithms in terms of time.

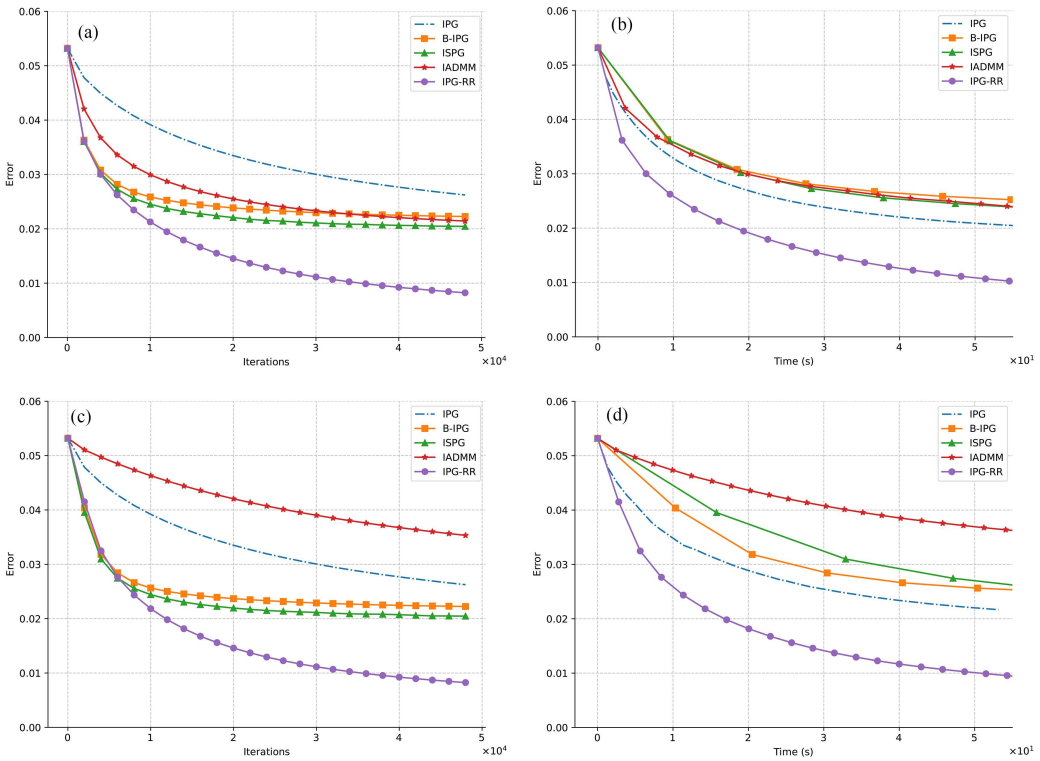

Figure 2 (Color online) Convergence of inexact algorithms with diminishing errors (a) $C = 20$, (b) $C = 20$ in terms of time, (c) $C = 50$, and (d) $C = 50$ in terms of time.

We observe that the proposed PG-RR algorithm yields a recovery image of superior quality compared to those generated by other algorithms, which demonstrates the efficiency of the proposed algorithm.

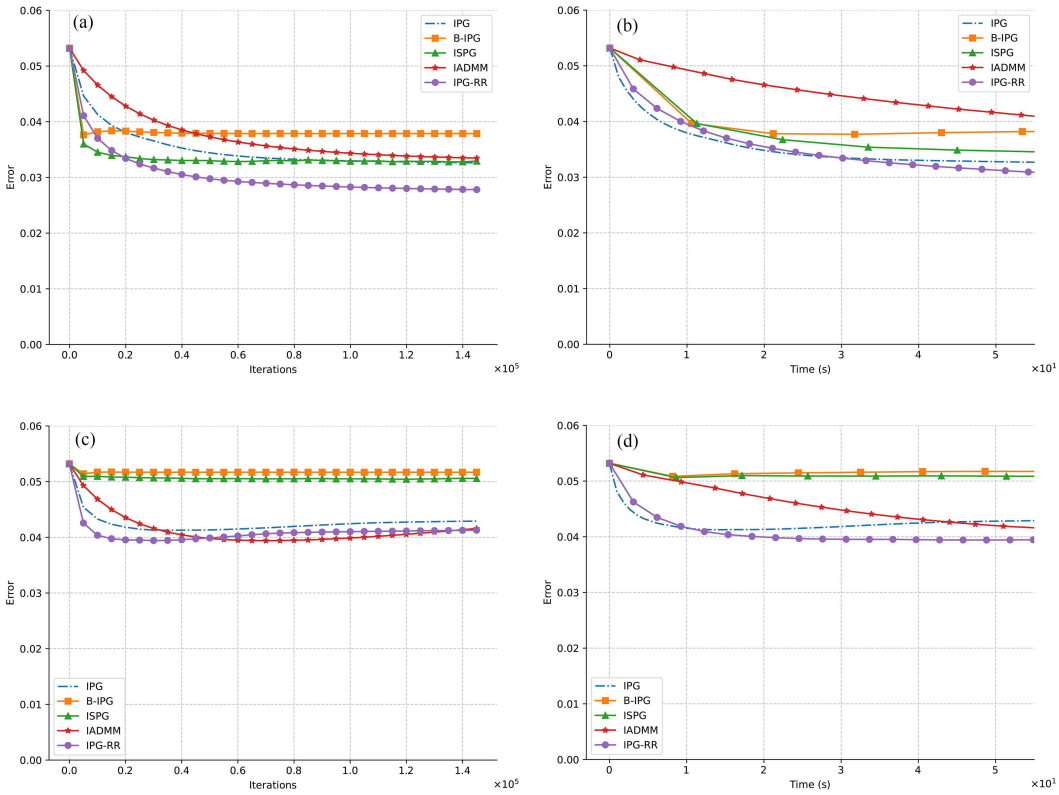

Figure 3 (Color online) Convergence of inexact algorithms with constant errors (a) $C = 3$, (b) $C = 3$ in terms of time, (c) $C = 5$, and (d) $C = 5$ in terms of time.

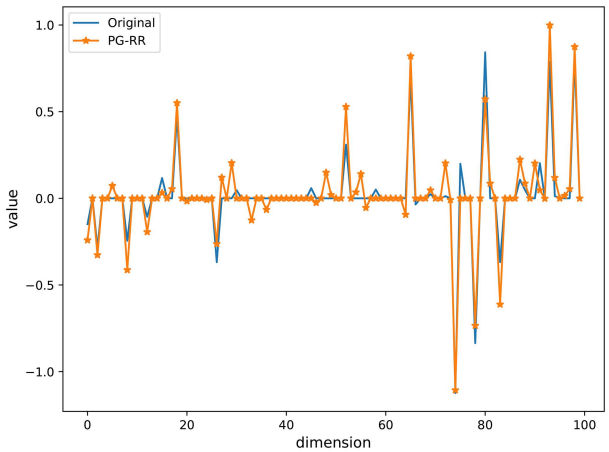

Figure 4 (Color online) Recovered signal versus the original sparse signal.

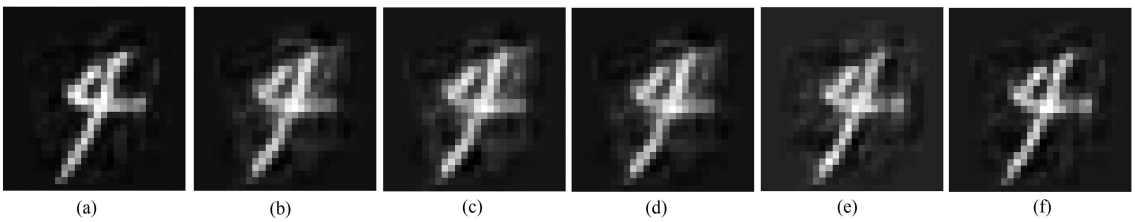

Figure 5 (a) Original; (b) PG; (c) B-PG; (d) SPG; (e) ADMM; (f) PG-RR.

6 Conclusion

This paper has proposed two proximal gradient algorithms that utilize a sampling-without-replacement scheme for nonsmooth optimization. The proposed exact stochastic proximal gradient algorithm has a sublinear convergence rate, in which the magnitude of errors near the solution decreases with the increase in the step size. Furthermore, when computing errors exist in the gradient and proximal operator, the proposed inexact proximal gradient algorithm converges to an optimal solution neighborhood. In the simulation, both proposed algorithms show superior convergence performance compared with that of several existing algorithms. One potential future research direction is the development of variance-reduced proximal RR algorithms to eliminate gradient variance and consequently advance convergence performance.

Acknowledgements This work was supported in part by National Key R&D Program of China (Grant No. 2021YFB1714800), National Natural Science Foundation of China (Grant Nos. 61925303, 62088101, 62073035, 62173034), and Natural Science Foundation of Chongqing (Grant No. 2021ZX4100027).

References

1 Berinde R, Gilbert A C, Indyk P, et al. Combining geometry and combinatorics: a unified approach to sparse signal recovery. In: Proceedings of the 46th Annual Allerton Conference on Communication, Control, and Computing, 2008. 798–805
2 Chen J, Kai S X. Cooperative transportation control of multiple mobile manipulators through distributed optimization. Sci China Inf Sci, 2018, 61: 120201
3 Banert S, Ringh A, Adler J, et al. Data-driven nonsmooth optimization. SIAM J Optim, 2020, 30: 102–131
4 Shi W, Ling Q, Wu G, et al. A proximal gradient algorithm for decentralized composite optimization. IEEE Trans Signal Process, 2015, 63: 6013–6023
5 Li G C, Song S J, Wu C. Generalized gradient projection neural networks for nonsmooth optimization problems. Sci China Inf Sci, 2010, 53: 990–1005
6 Li Z, Li Y, Tan B, et al. Structured sparse coding with the group log-regularizer for key frame extraction. IEEE CAA J Autom Sin, 2022, 9: 1818–1830
7 Wang J H, Meng F Y, Pang L P, et al. An adaptive fixed-point proximity algorithm for solving total variation denoising models. Inf Sci, 2017, 402: 69–81
8 Hassan-Moghaddam S, Jovanović M R. On the exponential convergence rate of proximal gradient flow algorithms. In: Proceedings of IEEE Conference on Decision and Control (CDC), 2018. 4246–4251
9 Huang Y, Meng Z, Sun J, et al. Distributed multiproximal algorithm for nonsmooth convex optimization with coupled inequality constraints. IEEE Trans Automat Contr, 2023, 68: 8126–8133
10 Niu L, Zhou R, Tian Y, et al. Nonsmooth penalized clustering via ℓ_p regularized sparse regression. IEEE Trans Cybern, 2017, 47: 1423–1433
11 Alghunaim S A, Ryu E K, Yuan K, et al. Decentralized proximal gradient algorithms with linear convergence rates. IEEE Trans Automat Contr, 2021, 66: 2787–2794
12 Vandenberghe L. ECE236C — optimization methods for large-scale systems. 2022. https://www.seas.ucla.edu/~vandenbe/ee236c.html
13 Beck A, Teboulle M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J Imag Sci, 2009, 2: 183–202
14 Wang X, Wang S, Zhang H. Inexact proximal stochastic gradient method for convex composite optimization. Comput Optim Appl, 2017, 68: 579–618
15 Rosasco L, Villa S, Vũ B C. Convergence of stochastic proximal gradient algorithm. Appl Math Optim, 2020, 82: 891–917
16 Nitanda A. Stochastic proximal gradient descent with acceleration techniques. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, 2014. 1574–1582
17 Duchi J, Singer Y. Efficient online and batch learning using forward backward splitting. J Mach Learn Res, 2009, 10: 2899–2934
18 Ahn K, Yun C, Sra S. SGD with shuffling: optimal rates without component convexity and large epoch requirements. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020
19 Gürbüzbalaban M, Ozdaglar A, Parrilo P A. Why random reshuffling beats stochastic gradient descent. Math Program, 2021, 186: 49–84
20 Huang X, Yuan K, Mao X, et al. Improved analysis and rates for variance reduction under without-replacement sampling orders. In: Proceedings of the 35th Conference on Neural Information Processing System, 2021. 3232–3243
21 Mishchenko K, Khaled A, Richtarik P. Proximal and federated random reshuffling. In: Proceedings of the 39th International Conference on Machine Learning, 2022. 15718–15749
22 Pan H, Jing Z L, Qiao L F, et al. Visible and infrared image fusion using $\ell 0$-generalized total variation model. Sci China Inf Sci, 2018, 61: 049103
23 Yang X H, Wang Z, Sun J, et al. Unlabeled data driven cost-sensitive inverse projection sparse representation-based classification with 1/2 regularization. Sci China Inf Sci, 2022, 65: 182102
24 Brbic M, Kopriva I. ℓ_0-motivated low-rank sparse subspace clustering. IEEE Trans Cybern, 2020, 50: 1711–1725
25 Gu B, Wang D, Huo Z, et al. Inexact proximal gradient methods for non-convex and non-smooth optimization. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018. 3093–3100
26 Jenatton R, Mairal J, Obozinski G, et al. Proximal methods for sparse hierarchical dictionary learning. In: Proceedings of the 27th International Conference on International Conference on Machine Learning, 2010. 487–494
27 Huo S C, Huang D L, Zhang Y. Secure output synchronization of heterogeneous multi-agent systems against false data injection attacks. Sci China Inf Sci, 2022, 65: 162204
28 Guo H, Sun J, Pang Z H. Residual-based false data injection attacks against multi-sensor estimation systems. IEEE CAA J Autom Sin, 2023, 10: 1181–1191
29 Devolder O, Glineur F, Nesterov Y. First-order methods of smooth convex optimization with inexact oracle. Math Program, 2014, 146: 37–75
30 Duchi J, Singer Y. Efficient online and batch learning using forward backward splitting. J Mach Learn Res, 2009, 10: 2899–2934
31 Schmidt M, Roux N, Bach F. Convergence rates of inexact proximal-gradient methods for convex optimization. In: Proceedings of the 24th International Conference on Neural Information Processing Systems, 2011. 1458–1466

32 Nedic A, Bertsekas D P. Incremental subgradient methods for nondifferentiable optimization. SIAM J Optim, 2001, 12: 109–138
33 Bertsekas D P. Optimization for Machine Learning. Cambridge: MIT Press, 2011
34 Mishchenko K, Khaled A, Richtarik P. Random reshuffling: simple analysis with vast improvements. In: Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), 2020. 17309–17320
35 Kilmer M E, Martin C D. Factorization strategies for third-order tensors. Linear Algebra its Appl, 2011, 435: 641–658
36 Kilmer M E, Braman K, Hao N, et al. Third-order tensors as operators on matrices: a theoretical and computational framework with applications in imaging. SIAM J Matrix Anal Appl, 2013, 34: 148–172
37 Qin W, Wang H, Zhang F, et al. Low-rank high-order tensor completion with applications in visual data. IEEE Trans Image Process, 2022, 31: 2433–2448
38 Fu X, Gao C, Wai H T, et al. Block-randomized stochastic proximal gradient for constrained low-rank tensor factorization. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019. 7485–7489
39 Xu Y. Alternating proximal gradient method for sparse nonnegative Tucker decomposition. Math Prog Comp, 2015, 7: 39–70
40 Bumin A, Huang K. Efficient implementation of stochastic proximal point algorithm for matrix and tensor completion. In: Proceedings of the 29th European Signal Processing Conference (EUSIPCO), 2021. 1050–1054
41 Yao Q, Kwok J T Y, Han B. Efficient nonconvex regularized tensor completion with structure-aware proximal iterations. In: Proceedings of the 36th International Conference on Machine Learning, 2019. 7035–7044
42 Donoho D L. For most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution. Comm Pure Appl Math, 2006, 59: 797–829
43 Tropp J A, Gilbert A C. Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans Inform Theor, 2007, 53: 4655–4666
44 Do T T, Gan L, Nguyen N, et al. Sparsity adaptive matching pursuit algorithm for practical compressed sensing. In: Proceedings of the 42nd Asilomar Conference on Signals, Systems and Computers, 2008. 581–587
45 Figueiredo M A T, Nowak R D, Wright S J. Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. IEEE J Sel Top Signal Process, 2007, 1: 586–597
46 Boyd S. Distributed optimization and statistical learning via the alternating direction method of multipliers. FNT Machine Learn, 2011, 3: 1–122
47 Daubechies I, Defrise M, De Mol C. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Comm Pure Appl Math, 2004, 57: 1413–1457
48 Peng Z, Yan M, Yin W. Parallel and distributed sparse optimization. In: Proceedings of Asilomar Conference on Signals, Systems and Computers, 2013. 659–646