RANDOM GRADIENT EXTRAPOLATION FOR DISTRIBUTED AND STOCHASTIC OPTIMIZATION

GUANGHUI LAN * AND YI ZHOU †

Abstract. In this paper, we consider a class of finite-sum convex optimization problems defined over a distributed multiagent network with m agents connected to a central server. In particular, the objective function consists of the average of m (≥ 1) smooth components associated with each network agent together with a strongly convex term. Our major contribution is to develop a new randomized incremental gradient algorithm, namely random gradient extrapolation method (RGEM), which does not require any exact gradient evaluation even for the initial point, but can achieve the optimal $\mathcal{O}(\log(1/\epsilon))$ complexity bound in terms of the total number of gradient evaluations of component functions to solve the finite-sum problems. Furthermore, we demonstrate that for stochastic finite-sum optimization problems, RGEM maintains the optimal $\mathcal{O}(1/\epsilon)$ complexity (up to a certain logarithmic factor) in terms of the number of stochastic gradient computations, but attains an $\mathcal{O}(\log(1/\epsilon))$ complexity in terms of communication rounds (each round involves only one agent). It is worth noting that the former bound is independent of the number of agents m, while the latter one only linearly depends on m or even $\sqrt{m}$ for ill-conditioned problems. To the best of our knowledge, this is the first time that these complexity bounds have been obtained for distributed and stochastic optimization problems. Moreover, our algorithms were developed based on a novel dual perspective of Nesterov's accelerated gradient method.

Keywords: finite-sum optimization, gradient extrapolation, randomized method, distributed machine learning, stochastic optimization.

1. Introduction. The main problem of interest in this paper is the finite-sum convex programming (CP) problem given in the form of

$$\psi^* := \min_{x\in X} \left\{\psi(x) := \tfrac{1}{m}\textstyle\sum_{i=1}^m f_i(x) + \mu w(x)\right\}. \tag{1.1}$$

Here, $X \subseteq \mathbb{R}^n$ is a closed convex set, $f_i : X \to \mathbb{R}$, $i = 1, \ldots, m$, are smooth convex functions with Lipschitz continuous gradients over X, i.e., $\exists L_i \geq 0$ such that

$$\|\nabla f_i(x_1) - \nabla f_i(x_2)\|_* \leq L_i\|x_1 - x_2\|, \quad \forall x_1, x_2 \in X, \tag{1.2}$$

$w : X \to \mathbb{R}$ is a strongly convex function with modulus 1 w.r.t. a norm $\|\cdot\|$, i.e.,

$$w(x_1) - w(x_2) - \langle w'(x_2), x_1 - x_2\rangle \geq \tfrac{1}{2}\|x_1 - x_2\|^2, \quad \forall x_1, x_2 \in X, \tag{1.3}$$

where $w'(\cdot)$ denotes any subgradient (or gradient) of $w(\cdot)$ and $\mu \geq 0$ is a given constant. Hence, the objective function ψ is strongly convex whenever $\mu > 0$. For notational convenience, we also denote $f(x) \equiv \frac{1}{m}\sum_{i=1}^m f_i(x)$, $L \equiv \frac{1}{m}\sum_{i=1}^m L_i$, and $\hat{L} = \max_{i=1,\ldots,m} L_i$. It is easy to see that for some $L_f \geq 0$,

$$\|\nabla f(x_1) - \nabla f(x_2)\|_* \leq L_f\|x_1 - x_2\| \leq L\|x_1 - x_2\|, \quad \forall x_1, x_2 \in X. \tag{1.4}$$

We also consider a class of stochastic finite-sum optimization problems given by

$$\psi^* := \min_{x\in X} \left\{\psi(x) := \tfrac{1}{m}\textstyle\sum_{i=1}^m \mathbb{E}_{\xi_i}[F_i(x, \xi_i)] + \mu w(x)\right\}, \tag{1.5}$$

where ξ_i's are random variables with support $\Xi_i \subseteq \mathbb{R}^d$. It can be easily seen that (1.5) is a special case of (1.1) with $f_i(x) = \mathbb{E}_{\xi_i}[F_i(x, \xi_i)]$, $i = 1, \ldots, m$. However, different from deterministic finite-sum optimization problems, only noisy gradient information of each component function f_i can be accessed for the stochastic finite-sum optimization problem in (1.5).

The deterministic finite-sum problem (1.1) can model the empirical risk minimization in machine learning and statistical inferences, and hence has become the subject of intensive studies during the past few years.

*H. Milton Stewart School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 30332 . (email: george.lan@isye.gatech.edu).

†H. Milton Stewart School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 30332 . (email: yizhou@gatech.edu).

Our study on finite-sum problems (1.1) and (1.5) has also been motivated by the emerging need for distributed optimization and machine learning. Under such settings, each component function f_i is associated with an agent i, $i = 1, \ldots, m$, which are connected through a distributed network. While different topologies can be considered for distributed optimization (see, e.g., Figure 1.1 and 1.2), in this paper, we focus on the star network where m agents are connected to one central server, and all agents only communicate with the server (see Figure 1.1). These types of distributed optimization problems have several unique features. Firstly, they allow for data privacy, since no local data is stored in the server. Secondly, network agents behave independently and they may not be responsive at the same time. Thirdly, the communication between the server and agent can be expensive and has high latency. Finally, by considering the stochastic finite-sum optimization problem, we are interested in not only the deterministic empirical risk minimization, but also the generalization risk for distributed machine learning. Moreover, we allow the private data for each agent to be collected in an online (steaming) fashion. One typical example of the aforementioned distributed problems is *Federated Learning* recently introduced by Google in [25]. As a particular example, in the ℓ_2-regularized logistic regression problem, we have

$$f_i(x) = l_i(x) := \tfrac{1}{N_i}\sum_{j=1}^{N_i} \log(1 + \exp(-b_j^i {a_j^i}^T x)),\ i = 1, \ldots, m, \quad w(x) = R(x) := \tfrac{1}{2}\|x\|_2^2,$$

provided that f_i is the loss function of agent i with training data $\{a_j^i, b_j^i\}_{j=1}^{N_i} \in \mathbb{R}^n \times \{-1, 1\}$, and $\mu := \lambda$ is the penalty parameter. For minimization of the generalized risk, f_i's are given in the form of expectation, i.e.,

$$f_i(x) = l_i(x) := \mathbb{E}_{\xi_i}[\log(1 + \exp(-\xi_i^T x))],\ i = 1, \ldots, m,$$

where the random variable ξ_i models the underlying distribution for training dataset of agent i. Note that

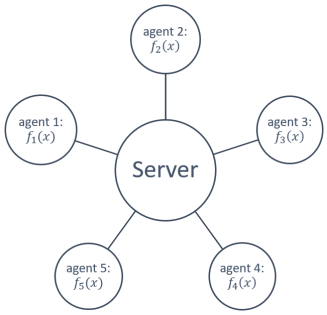

FIG. 1.1. *A distributed network with* 5 *agents and one server*

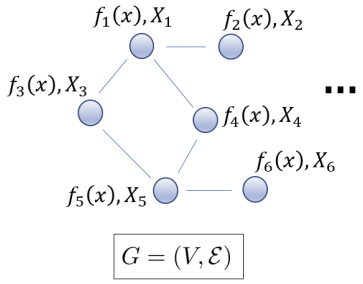

FIG. 1.2. *An example of the decentralized network*

another type of topology for distributed optimization is the multi-agent network without a central server, namely the decentralized setting, as shown in Figure 1.2, where the agents can only communicate with their neighbors to update information, please refer to [21, 32, 23] and reference therein for decentralized algorithms.

During the past few years, randomized incremental gradient (RIG) methods have emerged as an important class of first-order methods for finite-sum optimization (e.g.,[4, 16, 35, 8, 29, 22, 1, 14, 24]). For solving nonsmooth finite-sum problems, Nemirovski et al. [26, 27] showed that stochastic subgradient (mirror) descent methods can possibly save up to $\mathcal{O}(\sqrt{m})$ subgradient evaluations. By utilizing the smoothness properties of the objective, Lan [18] showed that one can separate the impact of variance from other deterministic components for stochastic gradient descent and presented a new class of accelerated stochastic gradient descent methods to further improve these complexity bounds. However, the overall rate of convergence of these stochastic methods is still sublinear even for smooth and strongly finite-sum problems (see [11, 12]). Inspired by these works and the success of the incremental aggregated gradient method by Blatt et al.[4], Schimidt et al. [29] presented a stochastic average gradient (SAG) method, which uses randomized sampling of f_i to update the gradients, and can achieve a linear rate of convergence, i.e., an $\mathcal{O}\{m + (mL/\mu)\log(1/\epsilon)\}$ complexity bound, to solve unconstrained finite-sum problems (1.1). Johnson and Zhang later in [16] presented a stochastic variance

reduced gradient (SVRG) method, which computes an estimator of ∇f by iteratively updating the gradient of one randomly selected f_i of the current exact gradient information and re-evaluating the exact gradient from time to time. Xiao and Zhang [35] later extended SVRG to solve proximal finite-sum problems (1.1). All these methods exhibit an improved $\mathcal{O}\{(m + L/\mu)\log(1/\epsilon)\}$ complexity bound, and Defazio et al. [8] also presented an improved SAG method, called SAGA, that can achieve such a complexity result. Comparing to the class of stochastic dual methods (e.g., [31, 30, 36]), each iteration of the RIG methods only involves the computation ∇f_i, rather than solving a more complicated subproblem

$$\operatorname{argmin}\{\langle g, y\rangle + f_i^*(y) + \|y\|_*^2\},$$

which may not have explicit solutions [31].

Noting that most of these RIG methods are not optimal even for $m = 1$, much recent research effort has been directed to the acceleration of RIG methods. In 2015, Lan and Zhou in [22] proposed a RIG method, namely randomized primal-dual gradient (RPDG) method, and show that its total number of gradient computations of f_i can be bounded by

$$\mathcal{O}\left\{\left(m + \sqrt{\tfrac{mL}{\mu}}\right)\log\tfrac{1}{\epsilon}\right\}. \tag{1.6}$$

The RPDG method utilizes a direct acceleration without even using the concept of variance reduction, evolving from the randomized primal-dual methods developed in [36, 7] for solving saddle-point problems. Lan and Zhou [22] also established a lower complexity bound for the RIG methods by showing that the number of gradient evaluations of f_i required by any RIG methods to find an ϵ-solution of (1.1), i.e., a point $\bar{x} \in X$ s.t. $\mathbb{E}[\|\bar{x} - x^*\|_2^2] \le \epsilon$, cannot be smaller than

$$\Omega\left(\left(m + \sqrt{\tfrac{mL}{\mu}}\right)\log\tfrac{1}{\epsilon}\right), \tag{1.7}$$

whenever the dimension

$$n \ge (k + m/2)/\log(1/q),$$

where k is the total number of iterations and $q = 1 + 2/(\sqrt{L/((m+1)\mu)} - 1)$. Simultaneously, Lin et al. [24] presented a catalyst scheme which utilizes a restarting technique to accelerate the SAG method in [29] (or other "non-accelerated" first-order methods) and thus can possibly improve the complexity bounds obtained by SVRG and SAGA to (1.6) (under the Euclidean setting). Allen-Zhu [1] later showed that one can also directly accelerate SVRG to achieve the optimal rate of convergence (1.6). All these accelerated RIG methods can save up to $\mathcal{O}(\sqrt{m})$ in the number of gradient evaluations of f_i comparing to optimal deterministic first-order methods when $L/\mu \ge m$.

It should be noted that most existing RIG methods were inspired by empirical risk minimization on a single server (or cluster) in machine learning rather than on a set of agents distributed over a network. Under the distributed setting, methods requiring full gradient computation and/or restarting from time to time may incur extra communication and synchronization costs. As a consequence, methods which require fewer full gradient computations (e.g. SAG, SAGA and RPDG) seem to be more advantageous in this regard. An interesting but yet unresolved question in stochastic optimization is whether there exists a method which does not require the computation of any full gradients (even at the initial point), but can still achieve the optimal rate of convergence in (1.6). Moreover, little attention in the study of RIG methods has been paid to the stochastic finite-sum problem in (1.5), which is important for generalization risk minimization in machine learning. Very recently, there are some progresses on stochastic primal-dual type methods for solving problem (1.5). For example, Lan, Lee and Zhou [21] proposed a stochastic decentralized communication sliding method that can achieve the optimal sampling complexity of $\mathcal{O}(1/\epsilon)$ and best-known $\mathcal{O}(1/\sqrt{\epsilon})$ complexity bounds for communication rounds for solving stochastic decentralized strongly convex problems. For the distributed setting with a central sever, by using mini-batch technique to collect gradient information and any stochastic gradient based algorithm as a black box to update iterates, Dekel et al. [9] presented a distributed mini-batch algorithm with a batch size of $o(m^{1/2})$ that can obtain $\mathcal{O}(1/\epsilon)$ sampling complexity (i.e., number of

stochastic gradients) for stochastic strongly convex problems, and hence implies at least $\mathcal{O}(1/\sqrt{\epsilon})$ bound for communication complexity. An asynchronous version was later proposed by Feyzmahdavian et al. in [10] that maintained the above convergence rate for regularized stochastic strongly convex problems. It should be pointed out that these mini-batch based distributed algorithms require sampling from all network agents iteratively and hence leads to at least $\mathcal{O}(m/\sqrt{\epsilon})$ rate of convergence in terms of communication costs among server and agents. It is unknown whether there exists an algorithm which only requires a significantly smaller communication rounds (e.g. $\mathcal{O}(\log 1/\epsilon)$), but can achieve the optimal $\mathcal{O}(1/\epsilon)$ sampling complexity for solving the stochastic finite-sum problem in (1.5).

The main contribution of this paper is to introduce a new randomized incremental gradient type method to solve (1.1) and (1.5). Firstly, we develop a random gradient extrapolation method (RGEM) for solving (1.1) that does not require any exact gradient evaluations of f. For strongly convex problems, we demonstrate that RGEM can still achieve the optimal rate of convergence (1.6) under the assumption that the average of gradients of f_i at the initial point x^0 is bounded by σ_0^2. To the best of our knowledge, this is the first time that such an optimal RIG methods without any exact gradient evaluations has been presented for solving (1.1) in the literature. In fact, without any full gradient computation, RGEM possesses iteration costs as low as pure stochastic gradient descent (SGD) methods, but achieves a much faster and optimal linear rate of convergence for solving deterministic finite-sum problems. In comparison with the well-known randomized Kaczmarz method [33], which can be viewed as an enhanced version of SGD, but can achieve a linear rate of convergence for solving linear systems, RGEM has a better convergence rate in terms of the dependence on the condition number L/μ. Secondly, we develop a stochastic version of RGEM and establish its optimal convergence properties for solving stochastic finite-sum problems (1.5). More specifically, we assume that only noisy first-order information of one randomly selected component function f_i can be accessed via a stochastic first-order (SFO) oracle iteratively. In other words, at each iteration only one randomly selected network agent needs to compute an estimator of its gradient by sampling from its local data using a SFO oracle instead of performing exact gradient evaluation of its component function f_i. Note that for these problems, it is difficult to compute the exact gradients even at the initial point. Under standard assumptions for centralized stochastic optimization, i.e., the gradient estimators computed by the SFO oracle are unbiased and have bounded variance σ^2, the number of stochastic gradient evaluations performed by RGEM to solve (1.5) can be bounded by[1]

$$\tilde{\mathcal{O}}\left\{\frac{\sigma_0^2/m+\sigma^2}{\mu^2\epsilon}+\frac{\mu\|x^0-x^*\|_2^2+\psi(x^0)-\psi^*}{\mu\epsilon}\right\}, \tag{1.8}$$

for finding a point $\bar{x}\in X$ s.t. $\mathbb{E}[\|\bar{x}-x^*\|_2^2]\leq\epsilon$. Moreover, by utilizing the mini-batch technique, RGEM can achieve an

$$\mathcal{O}\left\{\left(m+\sqrt{\frac{m\hat{L}}{\mu}}\right)\log\frac{1}{\epsilon}\right\}, \tag{1.9}$$

complexity bound in terms of the number of communication rounds, and each round only involves the communication between the server and a randomly selected agent. This bound seems to be optimal, since it matches the lower complexity bound for RIG methods to solve deterministic finite-sum problems. It is worth noting that the former bound (1.8) is independent of the number of agents m, while the latter one (1.9) only linearly depends on m or even $\sqrt{m}$ for ill-conditioned problems. To the best of our knowledge, this is the first time that such a RIG type method has been developed for solving stochastic finite-sum problems (1.5) that can achieve the optimal communication complexity and nearly optimal (up to a logarithmic factor) sampling complexity in the literature.

RGEM is developed based on a novel algorithmic framework, namely gradient extrapolation method (GEM), that we introduce in this paper for solving black-box convex optimization (i.e., $m=1$). The development of GEM was inspired by our recent studies on the relation between accelerated gradient methods and the primal-dual gradient methods. In particular, it is observed in [22] that Nesterov's accelerated gradient method is a special primal-dual gradient (PDG) method where the extrapolation step is performed in the primal space.

[1] $\tilde{\mathcal{O}}$ indicates the rate of convergence is up to a logarithmic factor - $\log(1/\epsilon)$.

Such a primal extrapolation step, however, might result in a search point outside the feasible region under the randomized setting in the RPDG method mentioned above. In view of this deficiency of PDG and RPDG methods, we propose to switch the primal and dual spaces for primal-dual gradient methods, and to perform the extrapolation step in the dual (gradient) space. The resulting new first-order method, i.e., GEM, can be viewed as a dual version of Nesterov's accelerated gradient method, and we show that it can also achieve the optimal rate of convergence for black-box convex optimization.

RGEM is a randomized version of GEM which only computes the gradient of a randomly selected component function f_i at each iteration. It utilizes the gradient extrapolation step also for estimating exact gradients in addition to predicting dual information as in GEM. As a result, it has several advantages over RPDG. Firstly, RPDG requires a restricted assumption that each f_i has to be differentiable and has Lipschitz continuous gradients over the whole $\mathbb{R}^n$ due to its primal extrapolation step. RGEM relaxes this assumption to having Lipschitz gradients over the feasible set X (see (1.2)), and hence can be applied to a much broader class of problems. Secondly, RGEM possesses simpler convergence analysis carried out in the primal space due to its simplified algorithmic scheme. However, RPDG has a complicated algorithmic scheme, which contains a primal extrapolation step and a gradient (dual) prediction step in addition to solving a primal proximal subproblem, and thus leads to an intricate primal-dual convergence analysis. Last but not least, it is unknown whether RPDG could maintain the optimal convergence rate (1.6) without the exact gradient evaluation of f during initialization.

This paper is organized as follows. In Section 2 we present the proposed random gradient extrapolation methods (RGEM), and their convergence properties for solving (1.1) and (1.5). In order to provide more insights into the design of the algorithmic scheme of RGEM, we provide an introduction to the gradient extrapolation method (GEM) and its relation to the primal-dual gradient method, as well as Nesterov's method in Section 3. Section 4 is devoted to the convergence analysis of RGEM. Some concluding remarks are made in Section 5.

1.1. Notation and terminology. We use $\|\cdot\|$ to denote a general norm in $\mathbb{R}^n$ without specific mention. We also use $\|\cdot\|_*$ to denote the conjugate norm of $\|\cdot\|$. For any $p \geq 1$, $\|\cdot\|_p$ denotes the standard p-norm in $\mathbb{R}^n$, i.e., $\|x\|_p^p = \sum_{i=1}^n |x_i|^p$, for any $x \in \mathbb{R}^n$. For any convex function h, $\partial h(x)$ is the set of subdifferential at x. For a given strongly convex function w with modulus 1 (see (1.1)), we define a *prox-function* associated with w as

$$P(x^0, x) \equiv P_w(x^0, x) := w(x) - \left[w(x^0) + \langle w'(x^0), x - x^0\rangle\right], \tag{1.10}$$

where $w'(x^0) \in \partial w(x^0)$ is an arbitrary subgradient of w at x^0. By the strong convexity of w, we have

$$P(x^0, x) \geq \frac{1}{2}\|x - x^0\|^2, \quad \forall x, x^0 \in X. \tag{1.11}$$

It should be pointed out that the prox-function $P(\cdot,\cdot)$ described above is a generalized Bregman distance in the sense that w is not necessarily differentiable. This is different from the standard definition for Bregman distance [5, 2, 3, 17, 6]. Throughout this paper, we assume that the prox-mapping associated with X and w, given by

$$\mathcal{M}_X(g, x^0, \eta) := \mathrm{argmin}_{x \in X} \left\{\langle g, x\rangle + \mu w(x) + \eta P(x^0, x)\right\}, \tag{1.12}$$

is easily computable for any $x^0 \in X, g \in \mathbb{R}^n, \mu \geq 0, \eta > 0$. For any real number r, $\lceil r \rceil$ and $\lfloor r \rfloor$ denote the nearest integer to r from above and below, respectively. $\mathbb{R}_+$ and $\mathbb{R}_{++}$, respectively, denote the set of nonnegative and positive real numbers.

2. Algorithms and main results. This section contains three subsections. We first present in Subsection 2.1 an optimal random gradient extrapolation method (RGEM) for solving the distributed finite-sum problem in (1.1), and then discuss in Subsection 2.2, a stochastic version of RGEM for solving the stochastic finite-sum problem in (1.5). Subsection 2.3 is devoted to the implementation of RGEM in a distributed setting and the discussion about its communication complexity.

2.1. RGEM for deterministic finite-sum optimization. The basic scheme of RGEM is formally stated in Algorithm 1. This algorithm simply initializes the gradient as $y^{-1} = y^0 = \mathbf{0}$. At each iteration, RGEM requires the new gradient information of only one randomly selected component function f_i, but maintains m pairs of search points and gradients $(\underline{x}_i^t, y_i^t)$, $i = 1, \ldots, m$, which are stored by their corresponding agents in the distributed network. More specifically, it first performs a gradient extrapolation step in (2.1) and the primal proximal mapping in (2.2). Then a randomly selected block $\underline{x}_{i_t}^t$ is updated in (2.3) and the corresponding component gradient ∇f_{i_t} is computed in (2.4). As can be seen from Algorithm 1, RGEM does not require any exact gradient evaluations.

Algorithm 1 A random gradient extrapolation method (RGEM)

Input: Let $x^0 \in X$, and the nonnegative parameters $\{\alpha_t\}$, $\{\eta_t\}$, and $\{\tau_t\}$ be given.

Initialization:

Set $\underline{x}_i^0 = x^0$, $i = 1, \ldots, m$, $y^{-1} = y^0 = \mathbf{0}$. ▷ No exact gradient evaluation for initialization

for $t = 1, \ldots, k$ **do**

Choose i_t according to $\mathrm{Prob}\{i_t = i\} = \frac{1}{m}, i = 1, \ldots, m$.

Update $z^t = (x^t, y^t)$ according to

$$\tilde{y}^t = y^{t-1} + \alpha_t(y^{t-1} - y^{t-2}), \tag{2.1}$$

$$x^t = \mathcal{M}_X(\tfrac{1}{m}\textstyle\sum_{i=1}^m \tilde{y}_i^t, x^{t-1}, \eta_t), \tag{2.2}$$

$$\underline{x}_i^t = \begin{cases} (1+\tau_t)^{-1}(x^t + \tau_t \underline{x}_i^{t-1}), & i = i_t, \\ \underline{x}_i^{t-1}, & i \neq i_t. \end{cases} \tag{2.3}$$

$$y_i^t = \begin{cases} \nabla f_i(\underline{x}_i^t), & i = i_t, \\ y_i^{t-1}, & i \neq i_t. \end{cases} \tag{2.4}$$

end for

Output: For some $\theta_t > 0$, $t = 1, \ldots, k$, set

$$\underline{x}^k := (\textstyle\sum_{t=1}^k \theta_t)^{-1} \sum_{t=1}^k \theta_t x^t. \tag{2.5}$$

Note that the computation of x^t in (2.2) requires an involved computation of $\frac{1}{m}\sum_{i=1}^m \tilde{y}_i^t$. In order to save computational time when implementing this algorithm, we suggest to compute this quantity in a recursive manner as follows. Let us denote $g^t \equiv \frac{1}{m}\sum_{i=1}^m y_i^t$, $t = 1, \ldots, k$. Clearly, in view of the fact that $y_i^t = y_i^{t-1}$, $\forall i \neq i_t$, we have

$$g^t = g^{t-1} + \tfrac{1}{m}(y_{i_t}^t - y_{i_t}^{t-1}). \tag{2.6}$$

Also, by the definition of g^t and (2.1), we have

$$\tfrac{1}{m}\textstyle\sum_{i=1}^m \tilde{y}_i^t = \tfrac{1}{m}\sum_{i=1}^m y_i^{t-1} + \tfrac{\alpha_t}{m}(y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}) = g^{t-1} + \tfrac{\alpha_t}{m}(y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}). \tag{2.7}$$

Using these two ideas mentioned above, we can compute $\frac{1}{m}\sum_{i=1}^m \tilde{y}_i^t$ in two steps: i) initialize $g^0 = \mathbf{0}$, and update g^t as in (2.6) after the gradient evaluation step (2.4); ii) replace (2.1) by (2.7) to compute $\frac{1}{m}\sum_{i=1}^m \tilde{y}_i^t$. Also note that the difference $y_{i_t}^t - y_{i_t}^{t-1}$ can be saved as it is used in both (2.6) and (2.7) for the next iteration. These enhancements will be incorporated into the distributed setting in Subsection 2.3 to possibly save communication costs.

It is also interesting to observe the differences between RGEM and RPDG [22]. RGEM has only one extrapolation step (2.1) which combines two types of predictions. One is to predict future gradients using historic data, and the other is to obtain an estimator of the current exact gradient of f from the randomly updated gradient information of f_i. However, RPDG method needs two extrapolation steps in both the

primal and dual spaces. Due to the existence of the primal extrapolation step, RPDG cannot guarantee the search points where it performs gradient evaluations to fall within the feasible set X. Hence, it requires the assumption that f_i's are differentiable with Lipschitz continuous gradients over $\mathbb{R}^n$. Such a strong assumption is not required by RGEM, since all the primal iterates generated by RGEM stay within the feasible region X. As a result, RGEM can deal with a much wider class of problems than RPDG. Moreover, RGEM allows no exact gradient computation for initialization, which provides a fully-distributed algorithmic framework under the assumption that there exists $\sigma_0 \geq 0$ such that

$$\tfrac{1}{m}\textstyle\sum_{i=1}^m \|\nabla f_i(x^0)\|_*^2 \leq \sigma_0^2, \tag{2.8}$$

where x^0 is the given initial point.

We now provide a constant step-size policy for RGEM to solve strongly convex problems given in the form of (1.1) and show that the resulting algorithm exhibits an optimal linear rate of convergence in Theorem 2.1. The proof of Theorem 2.1 can be found in Subsection 4.1.

THEOREM 2.1. *Let x^* be an optimal solution of (1.1), x^k and $\underline{x}^k$ be defined in* (2.2) *and (2.5), respectively, and $\hat{L} = \max_{i=1,\ldots,m} L_i$. Also let $\{\tau_t\}$, $\{\eta_t\}$ and $\{\alpha_t\}$ be set to*

$$\tau_t \equiv \tau = \tfrac{1}{m(1-\alpha)} - 1, \quad \eta_t \equiv \eta = \tfrac{\alpha}{1-\alpha}\mu, \quad \textit{and} \quad \alpha_t \equiv m\alpha. \tag{2.9}$$

If (2.8) *holds and α is set as*

$$\alpha = 1 - \tfrac{1}{m+\sqrt{m^2+16m\hat{L}/\mu}}, \tag{2.10}$$

then

$$\mathbb{E}[P(x^k, x^*)] \leq \tfrac{2\Delta_{0,\sigma_0}\alpha^k}{\mu}, \tag{2.11}$$

$$\mathbb{E}[\psi(\underline{x}^k) - \psi(x^*)] \leq 16\max\left\{m, \tfrac{\hat{L}}{\mu}\right\}\Delta_{0,\sigma_0}\alpha^{k/2}, \tag{2.12}$$

where

$$\Delta_{0,\sigma_0} := \mu P(x^0, x^*) + \psi(x^0) - \psi^* + \tfrac{\sigma_0^2}{m\mu}. \tag{2.13}$$

In view of Theorem 2.1, we can provide bounds on the total number of gradient evaluations performed by RGEM to find a stochastic ϵ-solution of problem (1.1), i.e., a point $\bar{x} \in X$ s.t. $\mathbb{E}[\psi(\bar{x}) - \psi^*] \leq \epsilon$. Theorem 2.1 implies the number of gradient evaluations of f_i performed by RGEM to find a stochastic ϵ-solution of (1.1) can be bounded by

$$K(\epsilon, C, \sigma_0^2) = 2\left(m + \sqrt{m^2+16mC}\right)\log\tfrac{16\max\{m,C\}\Delta_{0,\sigma_0}}{\epsilon} = \mathcal{O}\left\{\left(m + \sqrt{\tfrac{m\hat{L}}{\mu}}\right)\log\tfrac{1}{\epsilon}\right\}. \tag{2.14}$$

Here $C = \hat{L}/\mu$. Therefore, whenever $\sqrt{mC}\log(1/\epsilon)$ is dominating, and L_f and $\hat{L}$ are in the same order of magnitude, RGEM can save up to $\mathcal{O}(\sqrt{m})$ gradient evaluations of the component function f_i than the optimal deterministic first-order methods. More specifically, RGEM does not require any exact gradient computation and its communication cost is similar to pure stochastic gradient descent. To the best of our knowledge, it is the first time that such an optimal RIG method is presented for solving (1.1) in the literature. It should be pointed out that while the rates of convergence of RGEM obtained in Theorem 2.1 is stated in terms of expectation, we can develop large-deviation results for these rates of convergence using similar techniques in [22] for solving strongly convex problems.

Furthermore, if a one-time exact gradient evaluation is available at the initial point, i.e., $y^{-1} = y^0 = (\nabla f_1(x^0), \ldots, \nabla f_m(x^0))$, we can drop the assumption in (2.8) and employ a more aggressive stepsize policy with

$$\alpha = 1 - \tfrac{2}{m+\sqrt{m^2+8m\hat{L}/\mu}},$$

Similarly, we can demonstrate that the number of gradient evaluations of f_i performed by RGEM with this initialization method to find a stochastic ϵ-solution can be bounded by

$$\left(m + \sqrt{m^2 + 8mC}\right) \log\left(\tfrac{6\max\{m,C\}\Delta_{0,0}}{\epsilon}\right) + m = \mathcal{O}\left\{\left(m + \sqrt{\tfrac{m\hat{L}}{\mu}}\right)\log\tfrac{1}{\epsilon}\right\}.$$

2.2. RGEM for stochastic finite-sum optimization. We discuss in this subsection the stochastic finite-sum optimization and online learning problems, where only noisy gradient information of f_i can be accessed via a stochastic first-order (SFO) oracle. In particular, for any given point $\underline{x}_i^t \in X$, the SFO oracle outputs a vector $G_i(\underline{x}_i^t, \xi_i^t)$ s.t.

$$\mathbb{E}_\xi[G_i(\underline{x}_i^t, \xi_i^t)] = \nabla f_i(\underline{x}_i^t),\ i = 1, \ldots, m, \tag{2.15}$$

$$\mathbb{E}_\xi[\|G_i(\underline{x}_i^t, \xi_i^t) - \nabla f_i(\underline{x}_i^t)\|_*^2] \le \sigma^2,\ i = 1, \ldots, m. \tag{2.16}$$

We also assume that throughout this subsection that the $\|\cdot\|$ is associated with the inner product $\langle\cdot,\cdot\rangle$.

As shown in Algorithm 2, the RGEM for stochastic finite-sum optimization is naturally obtained by replacing the gradient evaluation of f_i in Algorithm 1 (see (2.4)) with a stochastic gradient estimator of f_i given in (2.17). In particular, at each iteration, we collect B_t number of stochastic gradients of only one randomly selected component f_i and take their average as the stochastic estimator of ∇f_i. Moreover, it needs to be mentioned that the way RGEM initializes gradients, i.e, $y^{-1} = y^0 = \mathbf{0}$, is very important for stochastic optimization, since it is usually impossible to compute exact gradient for expectation functions even at the initial point.

Algorithm 2 RGEM for stochastic finite-sum optimization

This algorithm is the same as Algorithm 1 except that (2.4) is replaced by

$$y_i^t = \begin{cases} \frac{1}{B_t}\sum_{j=1}^{B_t} G_i(\underline{x}_i^t, \xi_{i,j}^t), & i = i_t, \\ y_i^{t-1}, & i \neq i_t. \end{cases} \tag{2.17}$$

Here, $G_i(\underline{x}_i^t, \xi_{i,j}^t)$, $j = 1, \ldots, B_t$, are stochastic gradients of f_i computed by the SFO oracle at $\underline{x}_i^t$.

Under the standard assumptions in (2.15) and (2.16) for stochastic optimization, and with proper choices of algorithmic parameters, Theorem 2.2 shows that RGEM can achieve the optimal $\mathcal{O}\{\sigma^2/\mu^2\epsilon\}$ rate of convergence (up to a certain logarithmic factor) for solving strongly convex problems given in the form of (1.5) in terms of the number of stochastic gradients of f_i. The proof of the this result can be found in Subsection 4.2.

THEOREM 2.2. *Let x^* be an optimal solution of (1.5), x^k and $\underline{x}^k$ be generated by Algorithm 2, and $\hat{L} = \max_{i=1,\ldots,m} L_i$. Suppose that σ_0 and σ are defined in* (2.8) *and* (2.16)*, respectively. Given the iteration limit k, let $\{\tau_t\}$, $\{\eta_t\}$ and $\{\alpha_t\}$ be set to* (2.9) *with α being set as* (2.10)*, and we also set*

$$B_t = \lceil k(1-\alpha)^2\alpha^{-t}\rceil,\ t = 1, \ldots, k, \tag{2.18}$$

then

$$\mathbb{E}[P(x^k, x^*)] \le \tfrac{2\alpha^k\Delta_{0,\sigma_0,\sigma}}{\mu}, \tag{2.19}$$

$$\mathbb{E}[\psi(\underline{x}^k) - \psi(x^*)] \le 6\max\left\{m, \tfrac{\hat{L}}{\mu}\right\}\Delta_{0,\sigma_0,\sigma}\alpha^{k/2}, \tag{2.20}$$

where the expectation is taken w.r.t. $\{i_t\}$ and $\{\xi_i^t\}$ and

$$\Delta_{0,\sigma_0,\sigma} := \mu P(x^0, x^*) + \psi(x^0) - \psi(x^*) + \tfrac{\sigma_0^2/m + 5\sigma^2}{\mu}. \tag{2.21}$$

In view of (2.20), the number of iterations performed by RGEM to find a stochastic ϵ-solution of (1.5), can be bounded by

$$\hat{K}(\epsilon, C, \sigma_0^2, \sigma^2) := 2\left(m + \sqrt{m^2 + 16mC}\right) \log \frac{6\max\{m, C\}\Delta_{0,\sigma_0,\sigma}}{\epsilon}. \tag{2.22}$$

Furthermore, in view of (2.19) this iteration complexity bound can be improved to

$$\bar{K}(\epsilon, \alpha, \sigma_0^2, \sigma^2) := \log_{1/\alpha} \frac{2\tilde{\Delta}_{0,\sigma_0,\sigma}}{\mu\epsilon}, \tag{2.23}$$

in terms of finding a point $\bar{x} \in X$ s.t. $\mathbb{E}[P(\bar{x}, x^*)] \leq \epsilon$. Therefore, the corresponding number of stochastic gradient evaluations performed by RGEM for solving problem (1.5) can be bounded by

$$\textstyle\sum_{t=1}^k B_t \leq k\sum_{t=1}^k (1-\alpha)^2\alpha^{-t} + k = \mathcal{O}\left\{\left(\frac{\Delta_{0,\sigma_0,\sigma}}{\mu\epsilon} + m + \sqrt{mC}\right)\log\frac{\Delta_{0,\sigma_0,\sigma}}{\mu\epsilon}\right\}, \tag{2.24}$$

which together with (2.21) imply that the total number of required stochastic gradients or samples of the random variables $\xi_i,\ i = 1, \ldots, m$, can be bounded by

$$\tilde{\mathcal{O}}\left\{\frac{\sigma_0^2/m + \sigma^2}{\mu^2\epsilon} + \frac{\mu P(x^0, x^*) + \psi(x^0) - \psi^*}{\mu\epsilon} + m + \sqrt{\frac{m\hat{L}}{\mu}}\right\}.$$

Observe that this bound does not depend on the number of terms m for small enough ϵ. To the best of our knowledge, it is the first time that such a convergence result is established for RIG algorithms to solve distributed stochastic finite-sum problems. This complexity bound in fact is in the same order of magnitude (up to a logarithmic factor) as the complexity bound achieved by the optimal accelerated stochastic approximation methods [11, 12, 19], which uniformly sample all the random variables $\xi_i,\ i = 1, \ldots, m$. However, this latter approach will thus involve much higher communication costs in the distributed setting (see Subsection 2.3 for more discussions).

2.3. RGEM for distributed optimization and machine learning. This subsection is devoted to RGEMs (see Algorithm 1 and Algorithm 2) from two different perspectives, i.e., the server and the activated agent under a distributed setting. We also discuss the communication costs incurred by RGEM under this setting.

Both the server and agents in the distributed network start with the same global initial point x^0, i.e., $\underline{x}_i^0 = x^0,\ i = 1, \ldots, m$, and the server also sets $\Delta y = \mathbf{0}$ and $g^0 = \mathbf{0}$. During the process of RGEM, the server updates iterate x^t and calculates the output solution $\underline{x}^k$ (cf. (2.5)) which is given by sumx/sumθ. Each agent only stores its local variable $\underline{x}_i^t$ and updates it according to the information received from the server (i.e., x^t) when activated. The activated agent also needs to upload the changes of gradient Δy_i to the server. Observe that since Δy might be sparse, uploading it will incur smaller amount of communication costs than uploading the new gradient y_i^t. Note that line 5 of RGEM from the i_t-th agent's perspective is optional if the agent saves historic gradient information from the last update.

RGEM The server's perspective

1: **while** $t \leq k$ **do**
2: $x^t \leftarrow \mathcal{M}_X(g^{t-1} + \frac{\alpha_t}{m}\Delta y, x^{t-1}, \eta_t)$
3: $\mathrm{sumx} \leftarrow \mathrm{sumx} + \theta_t \mathrm{x}^t$
4: $\mathrm{sum}\theta \leftarrow \mathrm{sum}\theta + \theta_t$
5: Send signal to the i_t-th agent where i_t is selected uniformly from $\{1, \ldots, m\}$
6: **if** i_t-th agent is responsive **then**
7: Send current iterate x^t to i_t-th agent
8: **if** Receive feedback Δy **then**
9: $g^t \leftarrow g^{t-1} + \Delta y$
10: $t \leftarrow t + 1$
11: **else goto** *Line 5*
12: **end if**
13: **else goto** *Line 5*
14: **end if**
15: **end while**

RGEM The activated i_t-th agent's perspective

1: Download the current iterate x^t from the server
2: **if** $t = 1$ **then**
3: $y_i^{t-1} \leftarrow \mathbf{0}$
4: **else**
5: $y_i^{t-1} \leftarrow \nabla f_i(\underline{x}_i^{t-1})$ ▷ Optional
6: **end if**
7: $\underline{x}_i^t \leftarrow (1 + \tau_t)^{-1}(x^t + \tau_t \underline{x}_i^{t-1})$
8: $y_i^t \leftarrow \nabla f_i(\underline{x}_i^t)$
9: Upload the local changes to the server, i.e., $\Delta y_i = y_i^t - y_i^{t-1}$

We now add some remarks about the potential benefits of RGEM for distributed optimization and machine learning. Firstly, since RGEM does not require any exact gradient evaluation of f, it does not need to wait for the responses from all agents in order to compute an exact gradient. Each iteration of RGEM only involves communication between the server and the activated i_t-th agent. In fact, RGEM will move to the next iteration in case no response is received from the i_t-th agent. This scheme works under the assumption that the probability for any agent being responsive or available at a certain point of time is equal. However, all other optimal RIG algorithms, except RPDG [22], need the exact gradient information from all network agents once in a while, which incurs high communication costs and synchronous delays as long as one agent is not responsive. Even RPDG requires a full round of communications and synchronization at the initial point.

Secondly, since each iteration of RGEM involves only constant number of communication rounds between the server and one selected agent, the communication complexity for RGEM under distributed setting can be bounded by

$$\mathcal{O}\left\{\left(m + \sqrt{\tfrac{m\hat{L}}{\mu}}\right)\log\tfrac{1}{\epsilon}\right\}.$$

Therefore, it can save up to $\mathcal{O}\{\sqrt{m}\}$ rounds of communication than the optimal deterministic first-order methods.

For solving distributed stochastic finite-sum optimization problems (1.5), RGEM from the i_t-th agent's perspective will be slightly modified as follows.

RGEM The activated i_t-th agent's perspective for solving (1.5)

1: Download the current iterate x^t from the server
2: **if** $t = 1$ **then**
3: $y_i^{t-1} \leftarrow \mathbf{0}$ ▷ Assuming RGEM saves y_i^{t-1} for $t \geq 2$ at the latest update
4: **end if**
5: $\underline{x}_i^t \leftarrow (1 + \tau_t)^{-1}(x^t + \tau_t \underline{x}_i^{t-1})$
6: $y_i^t \leftarrow \frac{1}{B_t}\sum_{j=1}^{B_t} G_i(\underline{x}_i^t, \xi_{i,j}^t)$ ▷ B_t is the batch size, and G_i's are the stochastic gradients given by SFO
7: Upload the local changes to the server, i.e., $\Delta y_i = y_i^t - y_i^{t-1}$

Similar to the case for the deterministic finite-sum optimization, the total number of communication

rounds performed by the above RGEM can be bounded by

$$\mathcal{O}\left\{\left(m+\sqrt{\frac{m\hat{L}}{\mu}}\right)\log\tfrac{1}{\epsilon}\right\},$$

for solving (1.5). Each round of communication only involves the server and a randomly selected agent. This communication complexity seems to be optimal, since it matches the lower complexity bound (1.7) established in [22]. Moreover, the sampling complexity, i.e., the total number of samples to be collected by all the agents, is also nearly optimal and comparable to the case when all these samples are collected in a centralized location and processed by an optimal stochastic approximation method. On the other hand, if one applies an existing optimal stochastic approximation method to solve the distributed stochastic optimization problem, the communication complexity will be as high as $\mathcal{O}(1/\sqrt{\epsilon})$, which is much worse than RGEM.

3. Gradient extrapolation method: dual of Nesterov's acceleration. Our goal in this section is to introduce a new algorithmic framework, referred to as the gradient extrapolation method (GEM), for solving the convex optimization problem given by

$$\psi^* := \min_{x\in X}\{\psi(x) := f(x)+\mu w(x)\}. \tag{3.1}$$

We show that GEM can be viewed as a dual of Nesterov's accelerated gradient method although these two algorithms appear to be quite different. Moreover, GEM possess some nice properties which enable us to develop and analyze the random gradient extrapolation method for distributed and stochastic optimization.

3.1. Generalized Bregman distance. In this subsection, we provide a brief introduction to the generalized Bregman distance defined in (1.10) and some properties for its associated prox-mapping defined in(1.12).

Note that whenever w is non-differentiable, we need to specify a particular selection of the subgradient w' before performing the prox-mapping. We assume throughout this paper that such a selection of w' is defined recursively as follows. Denote $x^1 \equiv \mathcal{M}_X(g, x^0, \eta)$. By the optimality condition of (1.12), we have

$$g+(\mu+\eta)w'(x^1)-\eta w'(x^0)\in\mathcal{N}_X(x^1),$$

where $\mathcal{N}_X(x^1) := \{v\in\mathbb{R}^n : v^T(x-x^1)\le 0, \forall x\in X\}$ denotes the normal cone of X at x^1. Once such a $w'(x^1)$ satisfying the above relation is identified, we will use it as a subgradient when defining $P(x^1,x)$ in the next iteration. Note that such a subgradient can be identified as long as x^1 is obtained, since it satisfies the optimality condition of (1.12).

The following lemma, which generalizes Lemma 6 of [20] and Lemma 2 of [11], characterizes the solutions to (1.12). The proof of this result can be found in Lemma 5 of [22].

LEMMA 3.1. *Let U be a closed convex set and a point $\tilde{u}\in U$ be given. Also let $w: U\to\mathbb{R}$ be a convex function and*

$$W(\tilde{u},u) = w(u)-w(\tilde{u})-\langle w'(\tilde{u}), u-\tilde{u}\rangle$$

for some $w'(\tilde{u})\in\partial w(\tilde{u})$. Assume that the function $q: U\to\mathbb{R}$ satisfies

$$q(u_1)-q(u_2)-\langle q'(u_2), u_1-u_2\rangle \ge \mu_0 W(u_2,u_1), \quad \forall u_1,u_2\in U$$

for some $\mu_0\ge 0$. Also assume that the scalars μ_1 and μ_2 are chosen such that $\mu_0+\mu_1+\mu_2\ge 0$. If

$$u^*\in \mathrm{Argmin}\{q(u)+\mu_1 w(u)+\mu_2 W(\tilde{u},u) : u\in U\},$$

then for any $u\in U$, we have

$$q(u^*)+\mu_1 w(u^*)+\mu_2 W(\tilde{u},u^*)+(\mu_0+\mu_1+\mu_2)W(u^*,u)\le q(u)+\mu_1 w(u)+\mu_2 W(\tilde{u},u).$$

3.2. The algorithm. As shown in Algorithm 3, GEM starts with a gradient extrapolation step (3.2) to compute $\tilde{g}^t$ from the two previous gradients g^{t-1} and g^{t-2}. Based on $\tilde{g}^t$, it performs a proximal gradient descent step in (3.3) and updates the output solution $\underline{x}^t$. Finally, the gradient at $\underline{x}^t$ is computed for gradient extrapolation in the next iteration. This algorithm is a special case of RGEM in Algorithm 1 (with $m = 1$).

Algorithm 3 An optimal gradient extrapolation method (GEM)

Input: Let $x^0 \in X$, and the nonnegative parameters $\{\alpha_t\}$, $\{\eta_t\}$, and $\{\tau_t\}$ be given.
Set $\underline{x}^0 = x^0$ and $g^{-1} = g^0 = \nabla f(x^0)$.
for $t = 1, 2, \ldots, k$ **do**

$$\tilde{g}^t = \alpha_t(g^{t-1} - g^{t-2}) + g^{t-1}. \tag{3.2}$$
$$x^t = \mathcal{M}_X(\tilde{g}^t, x^{t-1}, \eta_t). \tag{3.3}$$
$$\underline{x}^t = \left(x^t + \tau_t \underline{x}^{t-1}\right)/(1+\tau_t). \tag{3.4}$$
$$g^t = \nabla f(\underline{x}^t). \tag{3.5}$$

end for
Output: $\underline{x}^k$.

We now show that GEM can be viewed as the dual of the well-known Nesterov's accelerated gradient (NAG) method as studied in [22]. To see such a relationship, we will first rewrite GEM in a primal-dual form. Let us consider the dual space $\mathcal{G}$, where the gradients of f reside, and equip it with the conjugate norm $\|\cdot\|_*$. Let $J_f : \mathcal{G} \to \mathbb{R}$ be the conjugate function of f such that $f(x) := \max_{g\in\mathcal{G}}\{\langle x, g\rangle - J_f(g)\}$. We can reformulate the original problem in (3.1) as the following saddle point problem:

$$\psi^* := \min_{x\in X}\left\{\max_{g\in\mathcal{G}}\{\langle x, g\rangle - J_f(g)\} + \mu\, w(x)\right\}. \tag{3.6}$$

It is clear that J_f is strongly convex with modulus $1/L_f$ w.r.t. $\|\cdot\|_*$ (See Chapter E in [15] for details). Therefore, we can define its associated dual generalized Bregman distance and dual prox-mappings as

$$D_f(g^0, g) := J_f(g) - [J_f(g^0) + \langle J_f'(g^0), g - g^0\rangle], \tag{3.7}$$
$$\mathcal{M}_{\mathcal{G}}(-\tilde{x}, g^0, \tau) := \arg\min_{g\in\mathcal{G}}\left\{\langle -\tilde{x}, g\rangle + J_f(g) + \tau D_f(g^0, g)\right\}, \tag{3.8}$$

for any $g^0, g \in \mathcal{G}$. The following result, whose proof is given in Lemma 1 of [22], shows that the computation of the dual prox-mapping associated with D_f is equivalent to the computation of ∇f.

LEMMA 3.2. *Let $\tilde{x} \in X$ and $g^0 \in \mathcal{G}$ be given and $D_f(g^0, g)$ be defined in (3.7). For any $\tau > 0$, let us denote $z = [\tilde{x} + \tau J_f'(g^0)]/(1+\tau)$. Then we have $\nabla f(z) = \mathcal{M}_{\mathcal{G}}(-\tilde{x}, g^0, \tau)$.*

Using this result, we can see that the GEM iteration can be written a primal-dual form. Given $(x^0, g^{-1}, g^0) \in X \times \mathcal{G} \times \mathcal{G}$, it updates (x^t, g^t) by

$$\tilde{g}^t = \alpha_t(g^{t-1} - g^{t-2}) + g^{t-1}, \tag{3.9}$$
$$x^t = \mathcal{M}_X(\tilde{g}^t, x^{t-1}, \eta_t), \tag{3.10}$$
$$g^t = \mathcal{M}_{\mathcal{G}}(-x^t, g^{t-1}, \tau_t), \tag{3.11}$$

with a specific selection of $J_f'(g^{t-1}) = \underline{x}^{t-1}$ in $D_f(g^{t-1}, g)$. Indeed, by denoting $\underline{x}^0 = x^0$, we can easily see from $g^0 = \nabla f(\underline{x}^0)$ that $\underline{x}^0 \in \partial J_f(g^0)$. Now assume that $g^{t-1} = \nabla f(\underline{x}^{t-1})$ and hence that $\underline{x}^{t-1} \in \partial J_f(g^{t-1})$. By the definition of g^t in (3.11) and Lemma 3.2, we conclude that $g^t = \nabla f(\underline{x}^t)$ with $\underline{x}^t = (x^t + \tau_t \underline{x}^{t-1})/(1+\tau_t)$, which are exactly the definitions given in (3.4) and (3.5).

Recall that in a simple version of the NAG method (e.g., [28, 34, 19, 11, 12, 13]), given $(x^{t-1}, \bar{x}^{t-1}) \in X \times X$, it updates $(x^t, \bar{x}^t)$ by

$$\underline{x}^t = (1-\lambda_t)\bar{x}^{t-1} + \lambda_t x^{t-1}, \tag{3.12}$$
$$g^t = \nabla f(\underline{x}^t), \tag{3.13}$$
$$x^t = \mathcal{M}_X(g^t, x^{t-1}, \eta_t), \tag{3.14}$$
$$\bar{x}^t = (1-\lambda_t)\bar{x}^{t-1} + \lambda_t x^t, \tag{3.15}$$

for some $\lambda_t \in [0,1]$. Moreover, we have shown in [22] that (3.12)-(3.15) can be viewed as a specific instantiation of the following primal-dual updates:

$$\tilde{x}^t = \alpha_t(x^{t-1} - x^{t-2}) + x^{t-1}, \tag{3.16}$$
$$g^t = \mathcal{M}_{\mathcal{G}}(-\tilde{x}^t, g^{t-1}, \tau_t), \tag{3.17}$$
$$x^t = \mathcal{M}_X(g^t, x^{t-1}, \eta_t). \tag{3.18}$$

Comparing (3.9)-(3.11) with (3.16)-(3.18), we can clearly see that GEM is a dual version of NAG, obtained by switching the primal and dual variables in each equation of (3.16)-(3.18). The major difference exists in that the extrapolation step in GEM is performed in the dual space while the one in NAG is performed in the primal space. In fact, extrapolation in the dual space will help us to greatly simplify and further enhance the randomized incremental gradient methods developed in [22] based on NAG. Another interesting fact is that in GEM, the gradients are computed for the output solutions $\{\underline{x}^t\}$. On the other hand, the output solutions in the NAG method are given by $\{\bar{x}^t\}$ while the gradients are computed for the extrapolation sequence $\{\underline{x}^t\}$.

3.3. Convergence of GEM. Our goal in this subsection is to establish the convergence properties of the GEM method for solving (3.1). Observe that our analysis is carried out completely in the primal space and does not rely on the primal-dual interpretation described in the previous section. This type of analysis technique appears to be new for solving problem (3.1) in the literature as it also differs significantly from that of NAG.

We first establish some general convergence properties for GEM for both smooth convex ($\mu = 0$) and strongly convex cases ($\mu > 0$).

THEOREM 3.3. *Suppose that $\{\eta_t\}$, $\{\tau_t\}$, and $\{\alpha_t\}$ in GEM satisfy*

$$\theta_{t-1} = \alpha_t\theta_t, \quad t = 2, \ldots, k, \tag{3.19}$$
$$\theta_t\eta_t \le \theta_{t-1}(\mu + \eta_{t-1}), \quad t = 2, \ldots, k, \tag{3.20}$$
$$\theta_t\tau_t = \theta_{t-1}(1 + \tau_{t-1}), \ t = 2, \ldots, k, \tag{3.21}$$
$$\alpha_t L_f \le \tau_{t-1}\eta_t, \quad t = 2, \ldots, k, \tag{3.22}$$
$$2L_f \le \tau_k(\mu + \eta_k), \tag{3.23}$$

for some $\theta_t \ge 0$, $t = 1, \ldots, k$. Then, for any $k \ge 1$ and any given $x \in X$, we have

$$\theta_k(1+\tau_k)[\psi(\underline{x}^k) - \psi(x)] + \tfrac{\theta_k(\mu+\eta_k)}{2}P(x^k, x) \le \theta_1\tau_1[\psi(x^0) - \psi(x)] + \theta_1\eta_1 P(x^0, x). \tag{3.24}$$

Proof. Applying Lemma 3.1 to (3.3), we obtain

$$\langle x^t - x, \alpha_t(g^{t-1} - g^{t-2}) + g^{t-1}\rangle + \mu w(x^t) - \mu w(x) \le \eta_t P(x^{t-1}, x) - (\mu + \eta_t)P(x^t, x) - \eta_t P(x^{t-1}, x^t). \tag{3.25}$$

Moreover, using the definition of ψ, the convexity of f, and the fact that $g^t = \nabla f(\underline{x}^t)$, we have

$$\begin{aligned}(1+\tau_t)f(\underline{x}^t) + \mu w(x^t) - \psi(x) &\le (1+\tau_t)f(\underline{x}^t) + \mu w(x^t) - \mu w(x) - [f(\underline{x}^t) + \langle g^t, x - \underline{x}^t\rangle] \\ &= \tau_t[f(\underline{x}^t) - \langle g^t, \underline{x}^t - \underline{x}^{t-1}\rangle] - \langle g^t, x - x^t\rangle + \mu w(x^t) - \mu w(x) \\ &\le -\tfrac{\tau_t}{2L_f}\|g^t - g^{t-1}\|_*^2 + \tau_t f(\underline{x}^{t-1}) - \langle g^t, x - x^t\rangle + \mu w(x^t) - \mu w(x) \\ &\le -\tfrac{\tau_t}{2L_f}\|g^t - g^{t-1}\|_*^2 + \tau_t f(\underline{x}^{t-1}) + \langle x^t - x, g^t - g^{t-1} - \alpha_t(g^{t-1} - g^{t-2})\rangle \\ &\quad + \eta_t P(x^{t-1}, x) - (\mu+\eta_t)P(x^t, x) - \eta_t P(x^{t-1}, x^t),\end{aligned}$$

where the first equality follows from the definition of $\underline{x}^t$ in (3.4), the second inequality follows from the smoothness of f (see Theorem 2.1.5 in [28]), and the last inequality follows from (3.25). Multiplying both sides of the above inequality by θ_t, and summing up the resulting inequalities from $t = 1$ to k, we obtain

$$\begin{aligned}&\textstyle\sum_{t=1}^k \theta_t(1+\tau_t)f(\underline{x}^t) + \sum_{t=1}^k \theta_t[\mu w(x^t) - \psi(x)] \le -\sum_{t=1}^k \tfrac{\theta_t\tau_t}{2L_f}\|g^t - g^{t-1}\|_*^2 + \sum_{t=1}^k \theta_t\tau_t f(\underline{x}^{t-1}) \\ &\quad + \textstyle\sum_{t=1}^k \theta_t\langle x^t - x, g^t - g^{t-1} - \alpha_t(g^{t-1} - g^{t-2})\rangle \\ &\quad + \textstyle\sum_{t=1}^k \theta_t[\eta_t P(x^{t-1}, x) - (\mu+\eta_t)P(x^t, x) - \eta_t P(x^{t-1}, x^t)].\end{aligned} \tag{3.26}$$

Now by (3.19) and the fact that $g^{-1} = g^0$, we have

$$\begin{aligned}&\textstyle\sum_{t=1}^k \theta_t\langle x^t - x, g^t - g^{t-1} - \alpha_t(g^{t-1} - g^{t-2})\rangle \\ &= \textstyle\sum_{t=1}^k \theta_t[\langle x^t - x, g^t - g^{t-1}\rangle - \alpha_t\langle x^{t-1} - x, g^{t-1} - g^{t-2}\rangle] - \sum_{t=2}^k \theta_t\alpha_t\langle x^t - x^{t-1}, g^{t-1} - g^{t-2}\rangle \\ &= \theta_k\langle x^k - x, g^k - g^{k-1}\rangle - \textstyle\sum_{t=2}^k \theta_t\alpha_t\langle x^t - x^{t-1}, g^{t-1} - g^{t-2}\rangle.\end{aligned}$$

Moreover, in view of (3.20), (3.21) and the definition of $\underline{x}^t$ (3.4), we obtain

$$\begin{aligned}\textstyle\sum_{t=1}^k \theta_t[\eta_t P(x^{t-1}, x) - (\mu+\eta_t)P(x^t, x)] &\overset{(3.20)}{\le} \theta_1\eta_1 P(x^0, x) - \theta_k(\mu+\eta_k)P(x^k, x), \\ \textstyle\sum_{t=1}^k \theta_t[(1+\tau_t)f(\underline{x}^t) - \tau_t f(\underline{x}^{t-1})] &\overset{(3.21)}{=} \theta_k(1+\tau_k)f(\underline{x}^k) - \theta_1\tau_1 f(\underline{x}^0), \\ \textstyle\sum_{t=1}^k \theta_t &\overset{(3.21)}{=} \textstyle\sum_{t=2}^k[\theta_t\tau_t - \theta_{t-1}\tau_{t-1}] + \theta_k = \theta_k(1+\tau_k) - \theta_1\tau_1, \\ \theta_k(1+\tau_k)\underline{x}^k &\overset{(3.4)}{=} \theta_k(x^k + \tfrac{\tau_k}{1+\tau_{k-1}}x^{k-1} + \cdots + \textstyle\prod_{t=2}^k \frac{\tau_t}{1+\tau_{t-1}}x^1 + \prod_{t=2}^k \frac{\tau_t}{1+\tau_{t-1}}\tau_1 x^0) \\ &\overset{(3.21)}{=} \textstyle\sum_{t=1}^k \theta_t x^t + \theta_1\tau_1 x^0.\end{aligned}$$

The last two relations, in view of the convexity of $w(\cdot)$, also imply that

$$\theta_k(1+\tau_k)\mu w(\underline{x}^k) \le \textstyle\sum_{t=1}^k \theta_t\mu w(x^t) + \theta_1\tau_1\mu w(x^0).$$

Therefore, by (3.26), the above relations, and the definition of ψ, we conclude that

$$\begin{aligned}\theta_k(1+\tau_k)[\psi(\underline{x}^k) - \psi(x)] \le \textstyle\sum_{t=2}^k &\left[-\tfrac{\theta_{t-1}\tau_{t-1}}{2L_f}\|g^{t-1} - g^{t-2}\|_*^2 - \theta_t\alpha_t\langle x^t - x^{t-1}, g^{t-1} - g^{t-2}\rangle - \theta_t\eta_t P(x^{t-1}, x^t)\right] \\ &- \theta_k\left[\tfrac{\tau_k}{2L_f}\|g^k - g^{k-1}\|_*^2 - \langle x^k - x, g^k - g^{k-1}\rangle + (\mu+\eta_k)P(x^k, x)\right] + \theta_1\eta_1 P(x^0, x) \\ &+ \theta_1\tau_1[\psi(x^0) - \psi(x)] - \theta_1\eta_1 P(x^0, x^1).\end{aligned} \tag{3.27}$$

By the strong convexity of $P(\cdot,\cdot)$ in (1.11), the simple relation that $b\langle u, v\rangle - a\|v\|^2/2 \le b^2\|u\|^2/(2a),\ \forall a > 0$,

and the conditions in (3.22) and (3.23), we have

$$
\begin{aligned}
&-\sum_{t=2}^{k}\left[\tfrac{\theta_{t-1}\tau_{t-1}}{2L_f}\|g^{t-1}-g^{t-2}\|_*^2+\theta_t\alpha_t\langle x^t-x^{t-1},g^{t-1}-g^{t-2}\rangle+\theta_t\eta_t P(x^{t-1},x^t)\right]\\
&\quad\le\sum_{t=2}^{k}\tfrac{\theta_t}{2}\left(\tfrac{\alpha_t L_f}{\tau_{t-1}}-\eta_t\right)\|x^{t-1}-x^t\|^2\le 0\\
&-\theta_k\left[\tfrac{\tau_k}{2L_f}\|g^k-g^{k-1}\|_*^2-\langle x^k-x,g^k-g^{k-1}\rangle+\tfrac{(\mu+\eta_k)}{2}P(x^k,x)\right]\\
&\quad\le\tfrac{\theta_k}{2}\left(\tfrac{L_f}{\tau_k}-\tfrac{\mu+\eta_k}{2}\right)\|x^k-x\|^2\le 0.
\end{aligned}
$$

Using the above relations in (3.27), we obtain (3.24). □

We are now ready to establish the optimal convergence behavior of GEM as a consequence of Theorem 3.3. We first provide a constant step-size policy which guarantees an optimal linear rate of convergence for the strongly convex case ($\mu > 0$).

COROLLARY 3.4. *Let x^* be an optimal solution of (1.1), x^k and $\underline{x}^k$ be defined in (3.3) and (3.4), respectively. Suppose that $\mu > 0$, and that $\{\tau_t\}$, $\{\eta_t\}$ and $\{\alpha_t\}$ are set to*

$$
\tau_t\equiv\tau=\sqrt{\tfrac{2L_f}{\mu}},\quad \eta_t\equiv\eta=\sqrt{2L_f\mu},\quad \textit{and}\quad \alpha_t\equiv\alpha=\tfrac{\sqrt{2L_f/\mu}}{1+\sqrt{2L_f/\mu}},\ \forall t=1,\dots,k. \tag{3.28}
$$

Then,

$$
P(x^k,x^*)\le 2\alpha^k[P(x^0,x^*)+\tfrac{1}{\mu}(\psi(x^0)-\psi^*)], \tag{3.29}
$$

$$
\psi(\underline{x}^k)-\psi^*\le\alpha^k\left[\mu P(x^0,x^*)+\psi(x^0)-\psi^*\right]. \tag{3.30}
$$

Proof. Let us set $\theta_t=\alpha^{-t},\ t=1,\dots,k$. It is easy to check that the selection of $\{\tau_t\},\{\eta_t\}$ and $\{\alpha_t\}$ in (3.28) satisfies conditions (3.19)-(3.23). In view of Theorem 3.3 and (3.28), we have

$$
\begin{aligned}
\psi(\underline{x}^k)-\psi(x^*)+\tfrac{\mu+\eta}{2(1+\tau)}P(x^k,x^*)&\le\tfrac{\theta_1\tau}{\theta_k(1+\tau)}[\psi(x^0)-\psi(x^*)]+\tfrac{\theta_1\eta}{\theta_k(1+\tau)}P(x^0,x^*)\\
&=\alpha^k[\psi(x^0)-\psi(x^*)+\mu P(x^0,x^*)].
\end{aligned}
$$

It also follows from the above relation, the fact $\psi(\underline{x}^k)-\psi(x^*)\ge 0$, and (3.28) that

$$
P(x^k,x^*)\le\tfrac{2(1+\tau)\alpha^k}{\mu+\eta}[\mu P(x^0,x^*)+\psi(x^0)-\psi(x^*)]=2\alpha^k[P(x^0,x^*)+\tfrac{1}{\mu}(\psi(x^0)-\psi(x^*))].
$$

□

We now provide a stepsize policy which guarantees the optimal rate of convergence for the smooth case ($\mu = 0$). Observe that in smooth case we can estimate the solution quality for the sequence $\{\underline{x}^k\}$ only.

COROLLARY 3.5. *Let x^* be an optimal solution of (1.1), and $\underline{x}^k$ be defined in (3.4). Suppose that $\mu = 0$, and that $\{\tau_t\}$, $\{\eta_t\}$ and $\{\alpha_t\}$ are set to*

$$
\tau_t=\tfrac{t}{2},\quad \eta_t=\tfrac{4L_f}{t},\quad \textit{and}\quad \alpha_t=\tfrac{t}{t+1},\ \forall t=1,\dots,k. \tag{3.31}
$$

Then,

$$
\psi(\underline{x}^k)-\psi(x^*)=f(\underline{x}^k)-f(x^*)\le\tfrac{2}{(k+1)(k+2)}[f(x^0)-f(x^*)+8L_f P(x^0,x^*)]. \tag{3.32}
$$

Proof. Let us set $\theta_t=t+1$, $t=1,\dots,k$. It is easy to check that the parameters in (3.31) satisfy conditions (3.22)-(3.23). In view of (3.24) and (3.31), we conclude that

$$
\psi(\underline{x}^k)-\psi(x^*)\le\tfrac{2}{(k+1)(k+2)}[\psi(x^0)-\psi(x^*)+8L_f P(x^0,x^*)].
$$

□

In Corollary 3.6, we improve the above complexity result in terms of the dependence on $f(x^0) - f(x^*)$ by using a different step-size policy and a slightly more involved analysis for the smooth case ($\mu = 0$).

COROLLARY 3.6. *Let x^* be an optimal solution of (1.1), x^k and $\underline{x}^k$ be defined in (3.3) and (3.4), respectively. Suppose that $\mu = 0$, and that $\{\tau_t\}$, $\{\eta_t\}$ and $\{\alpha_t\}$ are set to*

$$\tau_t = \tfrac{t-1}{2}, \quad \eta_t = \tfrac{6L_f}{t}, \quad \text{and} \quad \alpha_t = \tfrac{t-1}{t}, \ \forall t = 1, \ldots, k. \tag{3.33}$$

Then, for any $k \geq 1$,

$$\psi(\underline{x}^k) - \psi(x^*) = f(\underline{x}^k) - f(x^*) \leq \tfrac{12L_f}{k(k+1)} P(x^0, x^*). \tag{3.34}$$

Proof. If we set $\theta_t = t, \ t = 1, \ldots, k$. It is easy to check that the parameters in (3.33) satisfy conditions (3.19)-(3.21) and (3.23). However, condition (3.22) only holds for $t = 3, \ldots, k$, i.e.,

$$\alpha_t L_f \leq \tau_{t-1}\eta_t, \quad t = 3, \ldots, k. \tag{3.35}$$

In view of (3.27) and the fact that $\tau_1 = 0$, we have

$$\begin{aligned}
\theta_k(1+\tau_k)[\psi(\underline{x}^k) - \psi(x)] \leq & -\theta_2[\alpha_2\langle x^2 - x^1, g^1 - g^0\rangle + \eta_2 P(x^1, x^2)] - \theta_1\eta_1 P(x^0, x^1) \\
& - \textstyle\sum_{t=3}^{k}\left[\frac{\theta_{t-1}\tau_{t-1}}{2L_f}\|g^{t-1} - g^{t-2}\|_*^2 + \theta_t\alpha_t\langle x^t - x^{t-1}, g^{t-1} - g^{t-2}\rangle + \theta_t\eta_t P(x^{t-1}, x^t)\right] \\
& - \theta_k\left[\tfrac{\tau_k}{2L_f}\|g^k - g^{k-1}\|_*^2 - \langle x^k - x, g^k - g^{k-1}\rangle + (\mu + \eta_k)P(x^k, x)\right] + \theta_1\eta_1 P(x^0, x) \\
\leq & \tfrac{\theta_1\alpha_2}{2\eta_2}\|g^1 - g^0\|_*^2 - \tfrac{\theta_1\eta_1}{2}\|x^1 - x^0\|^2 + \textstyle\sum_{t=3}^{k}\tfrac{\theta_t}{2}\left(\tfrac{\alpha_t L_f}{\tau_{t-1}} - \eta_t\right)\|x^{t-1} - x^t\|^2 \\
& + \tfrac{\theta_k}{2}\left(\tfrac{L_f}{\tau_k} - \tfrac{\eta_k}{2}\right)\|x^k - x\|^2 + \theta_1\eta_1 P(x^0, x) - \tfrac{\theta_k\eta_k}{2}P(x^k, x) \\
\leq & \tfrac{\theta_1\alpha_2 L_f^2}{2\eta_2}\|\underline{x}^1 - \underline{x}^0\|^2 - \tfrac{\theta_1\eta_1}{2}\|x^1 - x^0\|^2 + \theta_1\eta_1 P(x^0, x) - \tfrac{\theta_k\eta_k}{2}P(x^k, x) \\
\leq & \theta_1\left(\tfrac{\alpha_2 L_f^2}{2\eta_2} - \eta_1\right)\|x^1 - x^0\|^2 + \theta_1\eta_1 P(x^0, x) - \tfrac{\theta_k\eta_k}{2}P(x^k, x),
\end{aligned}$$

where the second inequality follows from the simple relation that $b\langle u, v\rangle - a\|v\|^2/2 \leq b^2\|u\|^2/(2a), \ \forall a > 0$ and (1.11), the third inequality follows from (3.35), (3.23), the definition of g^t in (3.5) and (1.4), and the last inequality follows from the facts that $\underline{x}^0 = x^0$ and $\underline{x}^1 = x^1$ (due to $\tau_1 = 0$). Therefore, by plugging the parameter setting in (3.33) into the above inequality, we conclude that

$$\psi(\underline{x}^k) - \psi^* = f(\underline{x}^k) - f(x^*) \leq [\theta_k(1+\tau_k)]^{-1}[\theta_1\eta_1 P(x^0, x^*) - \tfrac{\theta_k\eta_k}{2}P(x^k, x)] \leq \tfrac{12L_f}{k(k+1)}P(x^0, x^*).$$

□

In view of the results obtained in the above three corollaries, GEM exhibits optimal rates of convergence for both strongly convex and smooth cases. Different from the classical NAG method, GEM performs extrapolation on the gradients, rather than the iterates. This fact will help us to develop an enhanced randomized incremental gradient method than RPDG in [22], i.e., the Random Gradient Extrapolation Method, with a much simpler analysis.

4. Convergence analysis of RGEM. Our main goal in this section is to establish the convergence properties of RGEM for solving (1.1) and (1.5), i.e., the main results stated in Theorem 2.1 and 2.2. In fact, comparing RGEM in Algorithm 1 with GEM in Algorithm 3, RGEM is a direct randomization of GEM. Therefore, inheriting from GEM, its convergence analysis is carried out completely in the primal space. However, the analysis for RGEM is more challenging especially because we need to 1) build up the relationship

between $\frac{1}{m}\sum_{i=1}^m f_i(\underline{x}_i^k)$ and $f(\underline{x}^k)$, for which we exploit the function Q defined in (4.3) as an intermediate tool; 2) bound the error caused by inexact gradients at the initial point and 3) analyze the accumulated error caused by randomization and noisy stochastic gradients.

Before proving Theorem 2.1 and 2.2, we first need to provide some important technical results. The following simple result demonstrates a few identities related to $\underline{x}_i^t$ (cf. (2.3)) and y^t (cf. (2.4) or (2.17)).

LEMMA 4.1. *Let x^t and y^t be defined in (2.2) and (2.4) (or (2.17)), respectively, and $\hat{\underline{x}}_i^t$ and $\hat{y}^t$ be defined as*

$$\hat{\underline{x}}_i^t = (1+\tau_t)^{-1}(x^t + \tau_t \underline{x}_i^{t-1}),\ i = 1,\dots,m,\ t \geq 1, \tag{4.1}$$

$$\hat{y}_i^t = \begin{cases} \nabla f_i(\hat{\underline{x}}_i^t), & \text{if } y^t \text{ is defined in (2.4)}, \\ \frac{1}{B_t}\sum_{j=1}^{B_t} G_i(\hat{\underline{x}}_i^t, \xi_{i,j}^t), & \text{if } y^t \text{ is defined in (2.17)}, \end{cases} \quad i = 1,\dots,m,\ t \geq 1, \tag{4.2}$$

respectively. Then we have, for any $i = 1,\dots,m$ and $t = 1,\dots,k$,

$$\begin{aligned} \mathbb{E}_t[y_i^t] &= \tfrac{1}{m}\hat{y}_i^t + (1-\tfrac{1}{m})y_i^{t-1}, \\ \mathbb{E}_t[\underline{x}_i^t] &= \tfrac{1}{m}\hat{\underline{x}}_i^t + (1-\tfrac{1}{m})\underline{x}_i^{t-1}, \\ \mathbb{E}_t[f_i(\underline{x}_i^t)] &= \tfrac{1}{m} f_i(\hat{\underline{x}}_i^t) + (1-\tfrac{1}{m}) f_i(\underline{x}_i^{t-1}), \\ \mathbb{E}_t[\|\nabla f_i(\underline{x}_i^t) - \nabla f_i(\underline{x}_i^{t-1})\|_*^2] &= \tfrac{1}{m}\|\nabla f_i(\hat{\underline{x}}_i^t) - \nabla f_i(\underline{x}_i^{t-1})\|_*^2, \end{aligned}$$

where $\mathbb{E}_t$ denotes the conditional expectation w.r.t. i_t given $i_1,\dots,i_{t-1}$ when y^t is defined in (2.4), and w.r.t. i_t given $i_1,\dots,i_{t-1},\xi_1^t,\dots,\xi_m^t$ when y^t is defined in (2.17), respectively.

Proof. This first equality follows immediately from the facts that $\mathrm{Prob}_t\{y_i^t = \hat{y}_i^t\} = \mathrm{Prob}_t\{i_t = i\} = \frac{1}{m}$ and $\mathrm{Prob}_t\{y_i^t = y_i^{t-1}\} = 1 - \frac{1}{m}$. Here Prob_t denotes the conditional probability w.r.t. i_t given $i_1,\dots,i_{t-1}$ when y^t is defined in (2.4) and w.r.t i_t given $i_1,\dots,i_{t-1},\xi_1^t,\dots,\xi_m^t$ when y^t is defined in (2.17), respectively. Similarly, we can prove the rest equalities. □

We define the following function Q to help us analyze the convergence properties of RGEM. Let $\underline{x}, x \in X$ be two feasible solutions of (1.1) (or (1.5)), we define the corresponding $Q(\underline{x}, x)$ by

$$Q(\underline{x}, x) := \langle \nabla f(x), \underline{x} - x\rangle + \mu w(\underline{x}) - \mu w(x). \tag{4.3}$$

It is obvious that if we fix $x = x^*$, an optimal solution of (1.1) (or (1.5)), by the convexity of w and the optimality condition of x^*, for any feasible solution $\underline{x}$, we can conclude that

$$Q(\underline{x}, x^*) \geq \langle \nabla f(x^*) + \mu w'(x^*), \underline{x} - x^*\rangle \geq 0.$$

Moreover, observing that f is smooth, we conclude that

$$Q(\underline{x}, x^*) = f(x^*) + \langle \nabla f(x^*), \underline{x} - x^*\rangle + \mu w(\underline{x}) - \psi(x^*) \geq -\tfrac{L_f}{2}\|\underline{x} - x^*\|^2 + \psi(\underline{x}) - \psi(x^*). \tag{4.4}$$

The following lemma establishes an important relationship regarding Q.

LEMMA 4.2. *Let x^t be defined in (2.2), and $x \in X$ be any feasible solution of (1.1) or (1.5). Suppose that τ_t in RGEM satisfy*

$$\theta_t(m(1+\tau_t) - 1) = \theta_{t-1} m(1+\tau_{t-1}), \quad t = 2,\dots,k, \tag{4.5}$$

for some $\theta_t \geq 0,\ t = 1,\dots,k$. Then, we have

$$\begin{aligned} \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[Q(x^t, x)] \leq \theta_k(1+\tau_k)\textstyle\sum_{i=1}^m \mathbb{E}[f_i(\underline{x}_i^k)] + \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[\mu w(x^t) - \psi(x)] \\ - \theta_1(m(1+\tau_1) - 1)[\langle x^0 - x, \nabla f(x)\rangle + f(x)]. \end{aligned} \tag{4.6}$$

Proof. In view of the definition of Q in (4.3), we have

$$\begin{aligned} Q(x^t,x) &= \tfrac{1}{m}\textstyle\sum_{i=1}^m \langle \nabla f_i(x), x^t - x\rangle + \mu w(x^t) - \mu w(x) \\ &\overset{(4.1)}{=} \tfrac{1}{m}\textstyle\sum_{i=1}^m [(1+\tau_t)\langle \hat{\underline{x}}_i^t - x, \nabla f_i(x)\rangle - \tau_t \langle \underline{x}_i^{t-1} - x, \nabla f_i(x)\rangle] + \mu w(x^t) - \mu w(x). \end{aligned}$$

Taking expectation on both sides of the above relation over $\{i_1,\ldots,i_k\}$, and using Lemma 4.1, we obtain

$$\mathbb{E}[Q(x^t,x)] = \textstyle\sum_{i=1}^m \mathbb{E}[(1+\tau_t)\langle \underline{x}_i^t - x, \nabla f_i(x)\rangle - ((1+\tau_t) - \tfrac{1}{m})\langle \underline{x}_i^{t-1} - x, \nabla f_i(x)\rangle] + \mathbb{E}[\mu w(x^t) - \mu w(x)].$$

Multiplying both sides of the above inequality by θ_t, and summing up the resulting inequalities from $t=1$ to k, we conclude that

$$\begin{aligned} \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[Q(x^t,x)] = \textstyle\sum_{i=1}^m \sum_{t=1}^k \mathbb{E}[\theta_t(1+\tau_t)\langle \underline{x}_i^t - x, \nabla f_i(x)\rangle - \theta_t((1+\tau_t) - \tfrac{1}{m})\langle \underline{x}_i^{t-1} - x, \nabla f_i(x)\rangle] \\ + \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[\mu w(x^t) - \mu w(x)]. \end{aligned}$$

Note that by (4.5) and the fact that $\underline{x}_i^0 = x^0,\ i=1,\ldots,m$, we have

$$\begin{aligned} \textstyle\sum_{t=1}^k \theta_t &= \textstyle\sum_{t=2}^k [\theta_t m(1+\tau_t) - \theta_{t-1} m(1+\tau_{t-1})] + \theta_1 = \theta_k m(1+\tau_k) - \theta_1(m(1+\tau_1) - 1), \qquad (4.7) \\ \textstyle\sum_{t=1}^k &[\theta_t(1+\tau_t)\langle \underline{x}_i^t - x, \nabla f_i(x)\rangle - \theta_t((1+\tau_t) - \tfrac{1}{m})\langle \underline{x}_i^{t-1} - x, \nabla f_i(x)\rangle] \\ &= \theta_k(1+\tau_k)\langle \underline{x}_i^k - x, \nabla f_i(x)\rangle - \theta_1((1+\tau_1) - \tfrac{1}{m})\langle x^0 - x, \nabla f_i(x)\rangle,\ i=1,\ldots,m. \end{aligned}$$

Combining the above three relations and using the convexity of f_i, we obtain

$$\begin{aligned} \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[Q(x^t,x)] \le \theta_k(1+\tau_k)\textstyle\sum_{i=1}^m \mathbb{E}[f_i(\underline{x}_i^k) - f_i(x)] - \theta_1(m(1+\tau_1)-1)\langle x^0 - x, \nabla f(x)\rangle \\ + \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[\mu w(x^t) - \mu w(x)], \end{aligned}$$

which in view of (4.7) implies (4.6). □

4.1. Convergence analysis of RGEM for deterministic finite-sum optimization. We now prove the main convergence properties for RGEM to solve (1.1). Observe that RGEM starts with $y^0 = \mathbf{0}$ and only updates the corresponding i_t-block of $(\underline{x}_i^t, y_i^t)$, $i=1,\ldots,m$, according to (2.3) and (2.4), respectively. Therefore, for y^t generated by RGEM, we have

$$y_i^t = \begin{cases} \mathbf{0}, & \text{if the } i\text{-th block has never been updated for the first } t \text{ iterations,} \\ \nabla f_i(\underline{x}_i^t), & \text{o.w.} \end{cases} \qquad (4.8)$$

Throughout this subsection, we assume that there exists $\sigma_0 \ge 0$ which is the upper bound of the initial gradients, i.e., (2.8) holds. Proposition 4.3 below establishes some general convergence properties of RGEM for solving strongly convex problems.

PROPOSITION 4.3. *Let x^t and $\underline{x}^k$ be defined as in* (2.2) *and* (2.5)*, respectively, and x^* be an optimal solution of* (1.1)*. Under the assumption that there exists σ_0 satisfying* (2.8)*, and suppose that $\{\eta_t\}$, $\{\tau_t\}$, and $\{\alpha_t\}$ in RGEM satisfy* (4.5) *and*

$$m\theta_{t-1} = \alpha_t\theta_t, \quad t \ge 2, \qquad (4.9)$$

$$\theta_t\eta_t \le \theta_{t-1}(\mu + \eta_{t-1}), \quad t \ge 2, \qquad (4.10)$$

$$2\alpha_t L_i \le m\tau_{t-1}\eta_t, \quad i=1,\ldots,m;\ t \ge 2, \qquad (4.11)$$

$$4L_i \le \tau_k(\mu + \eta_k), \quad i=1,\ldots,m, \qquad (4.12)$$

for some $\theta_t \geq 0$, $t = 1, \ldots, k$. Then, for any $k \geq 1$, we have

$$\begin{aligned} \mathbb{E}[Q(\underline{x}^k, x^*)] &\leq (\textstyle\sum_{t=1}^k \theta_t)^{-1} \tilde{\Delta}_{0,\sigma_0}, \\ \mathbb{E}[P(x^k, x^*)] &\leq \tfrac{2\tilde{\Delta}_{0,\sigma_0}}{\theta_k(\mu+\eta_k)}, \end{aligned} \tag{4.13}$$

where

$$\tilde{\Delta}_{0,\sigma_0} := \theta_1(m(1+\tau_1)-1)(\psi(x^0)-\psi^*) + \theta_1\eta_1 P(x^0,x^*) + \textstyle\sum_{t=1}^k (\frac{m-1}{m})^{t-1} \frac{2\theta_t\alpha_{t+1}}{m\eta_{t+1}}\sigma_0^2. \tag{4.14}$$

Proof. In view of the definition of x^t in (2.2) and Lemma 3.1, we have

$$\langle x^t - x, \tfrac{1}{m}\textstyle\sum_{i=1}^m \tilde{y}_i^t\rangle + \mu w(x^t) - \mu w(x) \leq \eta_t P(x^{t-1}, x) - (\mu+\eta_t)P(x^t, x) - \eta_t P(x^{t-1}, x^t). \tag{4.15}$$

Moreover, using the definition of ψ in (1.1), the convexity of f_i, and the fact that $\hat{y}_i^t = \nabla f_i(\hat{\underline{x}}_i^t)$ (see (4.2) with y^t defined in (2.4)), we obtain

$$\begin{aligned} \tfrac{1+\tau_t}{m}\textstyle\sum_{i=1}^m f_i(\hat{\underline{x}}_i^t) + \mu w(x^t) - \psi(x) &\leq \tfrac{1+\tau_t}{m}\textstyle\sum_{i=1}^m f_i(\hat{\underline{x}}_i^t) + \mu w(x^t) - \mu w(x) - \tfrac{1}{m}\textstyle\sum_{i=1}^m [f_i(\hat{\underline{x}}_i^t) + \langle \hat{y}_i^t, x - \hat{\underline{x}}_i^t\rangle] \\ &= \tfrac{\tau_t}{m}\textstyle\sum_{i=1}^m [f_i(\hat{\underline{x}}_i^t) + \langle \hat{y}_i^t, \underline{x}_i^{t-1} - \hat{\underline{x}}_i^t\rangle] + \mu w(x^t) - \mu w(x) - \tfrac{1}{m}\textstyle\sum_{i=1}^m \langle \hat{y}_i^t, x - x^t\rangle \\ &\leq -\tfrac{\tau_t}{2m}\textstyle\sum_{i=1}^m \tfrac{1}{L_i}\|\nabla f_i(\hat{\underline{x}}_i^t) - \nabla f_i(\underline{x}_i^{t-1})\|_*^2 + \tfrac{\tau_t}{m}\textstyle\sum_{i=1}^m f_i(\underline{x}_i^{t-1}) \\ &\quad + \mu w(x^t) - \mu w(x) - \tfrac{1}{m}\textstyle\sum_{i=1}^m \langle \hat{y}_i^t, x - x^t\rangle \\ &\leq -\tfrac{\tau_t}{2m}\textstyle\sum_{i=1}^m \tfrac{1}{L_i}\|\nabla f_i(\hat{\underline{x}}_i^t) - \nabla f_i(\underline{x}_i^{t-1})\|_*^2 + \tfrac{\tau_t}{m}\textstyle\sum_{i=1}^m f_i(\underline{x}_i^{t-1}) \\ &\quad + \langle x^t - x, \tfrac{1}{m}\textstyle\sum_{i=1}^m [\hat{y}_i^t - y_i^{t-1} - \alpha_t(y_i^{t-1} - y_i^{t-2})]\rangle \\ &\quad + \eta_t P(x^{t-1}, x) - (\mu+\eta_t)P(x^t, x) - \eta_t P(x^{t-1}, x^t), \end{aligned} \tag{4.16}$$

where the first equality follows from the definition of $\hat{\underline{x}}_i^t$ in (4.1), the second inequality follows from the smoothness of f_i (see Theorem 2.1.5 in [28]) and (4.2), and the last inequality follows from (4.15) and the definition of $\tilde{y}^t$ in (2.1). Therefore, taking expectation on both sides of the above relation over $\{i_1, \ldots, i_k\}$, and using Lemma 4.1, we have

$$\begin{aligned} \mathbb{E}[(1+\tau_t)\textstyle\sum_{i=1}^m f_i(\underline{x}_i^t) + \mu w(x^t) - \psi(x)] &\leq \mathbb{E}[-\tfrac{\tau_t}{2L_{i_t}}\|\nabla f_{i_t}(\underline{x}_{i_t}^t) - \nabla f_{i_t}(\underline{x}_{i_t}^{t-1})\|_*^2 + \tfrac{1}{m}\textstyle\sum_{i=1}^m (m(1+\tau_t)-1) f_i(\underline{x}_i^{t-1})] \\ &\quad + \mathbb{E}\{\langle x^t - x, \tfrac{1}{m}\textstyle\sum_{i=1}^m [m(y_i^t - y_i^{t-1}) - \alpha_t(y_i^{t-1} - y_i^{t-2})]\rangle\} \\ &\quad + \mathbb{E}[\eta_t P(x^{t-1}, x) - (\mu+\eta_t)P(x^t, x) - \eta_t P(x^{t-1}, x^t)]. \end{aligned}$$

Multiplying both sides of the above inequality by θ_t, and summing up the resulting inequalities from $t = 1$ to k, we obtain

$$\begin{aligned} &\textstyle\sum_{t=1}^k \sum_{i=1}^m \mathbb{E}[\theta_t(1+\tau_t) f_i(\underline{x}_i^t)] + \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[\mu w(x^t) - \psi(x)] \\ &\qquad \leq \textstyle\sum_{t=1}^k \theta_t \mathbb{E}\left[-\tfrac{\tau_t}{2L_{i_t}}\|\nabla f_{i_t}(\underline{x}_{i_t}^t) - \nabla f_{i_t}(\underline{x}_{i_t}^{t-1})\|_*^2 + \textstyle\sum_{i=1}^m ((1+\tau_t) - \tfrac{1}{m}) f_i(\underline{x}_i^{t-1})\right] \\ &\qquad\quad + \textstyle\sum_{t=1}^k \sum_{i=1}^m \theta_t \mathbb{E}[\langle x^t - x, y_i^t - y_i^{t-1} - \tfrac{\alpha_t}{m}(y_i^{t-1} - y_i^{t-2})\rangle] \\ &\qquad\quad + \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[\eta_t P(x^{t-1}, x) - (\mu+\eta_t)P(x^t, x) - \eta_t P(x^{t-1}, x^t)]. \end{aligned} \tag{4.17}$$

Now by (4.9), and the facts that $y^{-1} = y^0$ and that we only update one block of y^t (see (2.4)), we have

$$\begin{aligned} &\textstyle\sum_{t=1}^k \sum_{i=1}^m \theta_t \mathbb{E}[\langle x^t - x, y_i^t - y_i^{t-1} - \tfrac{\alpha_t}{m}(y_i^{t-1} - y_i^{t-2})\rangle] \\ &= \textstyle\sum_{t=1}^k \mathbb{E}[\theta_t \langle x^t - x, y_{i_t}^t - y_{i_t}^{t-1}\rangle - \tfrac{\theta_t\alpha_t}{m}\langle x^{t-1} - x, y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\rangle] - \textstyle\sum_{t=2}^k \tfrac{\theta_t\alpha_t}{m}\mathbb{E}[\langle x^t - x^{t-1}, y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\rangle] \\ &\overset{(4.9)}{=} \theta_k \mathbb{E}[\langle x^k - x, y_{i_k}^k - y_{i_k}^{k-1}\rangle - \textstyle\sum_{t=2}^k \tfrac{\theta_t\alpha_t}{m}\mathbb{E}[\langle x^t - x^{t-1}, y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\rangle]. \end{aligned}$$

Moreover, in view of (4.10), (4.5), and the fact that $\underline{x}_i^0 = x^0,\ i = 1, \ldots, m$, we obtain

$$\textstyle\sum_{t=1}^k \theta_t[\eta_t P(x^{t-1}, x) - (\mu + \eta_t)P(x^t, x)] \overset{(4.10)}{\leq} \theta_1\eta_1 P(x^0, x) - \theta_k(\mu + \eta_k)P(x^k, x),$$

$$\textstyle\sum_{t=1}^k \sum_{i=1}^m \mathbb{E}[\theta_t(1+\tau_t) f_i(\underline{x}_i^t) - \theta_t((1+\tau_t) - \frac{1}{m}) f_i(\underline{x}_i^{t-1})] \overset{(4.5)}{=} \sum_{i=1}^m \mathbb{E}[\theta_k(1+\tau_k) f_i(\underline{x}_i^k)] - \theta_1(m(1+\tau_1) - 1) f(x^0)$$

which together with (4.17) and (4.8) imply that

$$\begin{aligned}
&\theta_k(1+\tau_k)\textstyle\sum_{i=1}^m \mathbb{E}[f_i(\underline{x}_i^k)] + \sum_{t=1}^k \theta_t \mathbb{E}[\mu w(x^t) - \psi(x)] + \frac{\theta_k(\mu+\eta_k)}{2}\mathbb{E}[P(x^k, x)] \\
&\quad \leq \theta_1(m(1+\tau_1) - 1) f(x^0) + \theta_1\eta_1 P(x^0, x) \\
&\qquad + \textstyle\sum_{t=2}^k \mathbb{E}\left[-\frac{\theta_t\alpha_t}{m}\langle x^t - x^{t-1}, y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\rangle - \theta_t\eta_t P(x^{t-1}, x^t) - \frac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}}\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2 \right] \\
&\qquad + \theta_k \mathbb{E}\left[\langle x^k - x, y_{i_k}^k - y_{i_k}^{k-1}\rangle - \frac{(\mu+\eta_k)}{2} P(x^k, x) - \frac{\tau_k}{2L_{i_k}}\|y_{i_k}^k - \nabla f_{i_k}(\underline{x}_{i_k}^{k-1})\|_*^2 \right].
\end{aligned} \tag{4.18}$$

By the strong convexity of $P(\cdot,\cdot)$ in (1.11), the simple relations that $b\langle u, v\rangle - a\|v\|^2/2 \leq b^2\|u\|^2/(2a), \forall a > 0$ and $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$, and the conditions in (4.11) and (4.12), we have

$$\begin{aligned}
&\textstyle\sum_{t=2}^k \left[-\frac{\theta_t\alpha_t}{m}\langle x^t - x^{t-1}, y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\rangle - \theta_t\eta_t P(x^{t-1}, x^t) - \frac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}}\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2 \right] \\
&\quad \overset{(1.11)}{\leq} \textstyle\sum_{t=2}^k \left[-\frac{\theta_t\alpha_t}{m}\langle x^t - x^{t-1}, y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\rangle - \frac{\theta_t\eta_t}{2}\|x^{t-1} - x^t\|^2 - \frac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}}\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2 \right] \\
&\quad \leq \textstyle\sum_{t=2}^k \left[\frac{\theta_{t-1}\alpha_t}{2m\eta_t}\|y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\|_*^2 - \frac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}}\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2 \right] \\
&\quad \leq \textstyle\sum_{t=2}^k \left[\left(\frac{\theta_{t-1}\alpha_t}{m\eta_t} - \frac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}} \right)\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2 + \frac{\theta_{t-1}\alpha_t}{m\eta_t}\|\nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2}) - y_{i_{t-1}}^{t-2}\|_*^2 \right] \\
&\quad \overset{(4.11)}{\leq} \textstyle\sum_{t=2}^k \frac{\theta_{t-1}\alpha_t}{m\eta_t}\left[\|\nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2}) - y_{i_{t-1}}^{t-2}\|_*^2 \right],
\end{aligned}$$

and similarly,

$$\begin{aligned}
&\theta_k\left[\langle x^k - x, y_{i_k}^k - y_{i_k}^{k-1}\rangle - \frac{(\mu+\eta_k)}{2} P(x^k, x) - \frac{\tau_k}{2L_{i_k}}\|y_{i_k}^k - \nabla f_{i_k}(\underline{x}_{i_k}^{k-1})\|_*^2 \right] \\
&\quad \leq \frac{2\theta_k}{\mu+\eta_k}\left[\|\nabla f_{i_k}(\underline{x}_{i_k}^{k-1}) - y_{i_k}^{k-1}\|_*^2 \right] \leq \frac{2\theta_k\alpha_{k+1}}{m\eta_{k+1}}\left[\|\nabla f_{i_k}(\underline{x}_{i_k}^{k-1}) - y_{i_k}^{k-1}\|_*^2 \right],
\end{aligned}$$

where the last inequality follows from the fact that $m\eta_{k+1} \leq \alpha_{k+1}(\mu + \eta_k)$ (induced from (4.9) and (4.10)). Therefore, combing the above three relations, we conclude that

$$\begin{aligned}
&\theta_k(1+\tau_k)\textstyle\sum_{i=1}^m \mathbb{E}[f_i(\underline{x}_i^k)] + \sum_{t=1}^k \theta_t \mathbb{E}[\mu w(x^t) - \psi(x)] + \frac{\theta_k(\mu+\eta_k)}{2}\mathbb{E}[P(x^k, x)] \\
&\quad \leq \theta_1(m(1+\tau_1) - 1) f(x^0) + \theta_1\eta_1 P(x^0, x) + \textstyle\sum_{t=1}^k \frac{2\theta_t\alpha_{t+1}}{m\eta_{t+1}}\mathbb{E}[\|\nabla f_{i_t}(\underline{x}_{i_t}^{t-1}) - y_{i_t}^{t-1}\|_*^2].
\end{aligned} \tag{4.19}$$

We now provide a bound on $\mathbb{E}[\|\nabla f_{i_t}(\underline{x}_{i_t}^{t-1}) - y_{i_t}^{t-1}\|_*^2]$. In view of (4.8), we have

$$\|\nabla f_{i_t}(\underline{x}_{i_t}^{t-1}) - y_{i_t}^{t-1}\|_*^2 = \begin{cases} \|\nabla f_{i_t}(\underline{x}_{i_t}^{t-1})\|_*^2, & \text{if the } i_t\text{-th block has never been updated until iteration } t; \\ 0, & \text{o.w.} \end{cases}$$

Let us denote event $\mathcal{B}_{i_t} := \{\text{the } i_t\text{-th block has never been updated until iteration } t\}$, for all $t = 1, \ldots, k$, we have

$$\mathbb{E}[\|\nabla f_{i_t}(\underline{x}_{i_t}^{t-1}) - y_{i_t}^{t-1}\|_*^2] = \mathbb{E}[\|\nabla f_{i_t}(\underline{x}_{i_t}^{t-1})\|_*^2 | \mathcal{B}_{i_t}] \text{Prob}\{\mathcal{B}_{i_t}\} \leq \left(\tfrac{m-1}{m}\right)^{t-1}\sigma_0^2,$$

where the last inequality follows from the definitions of $\mathcal{B}_{i_t}$, $\underline{x}_i^t$ in (2.3) and σ_0^2 in (2.8). Fixing $x = x^*$, and using the above result in (4.19), we then conclude from (4.19) and Lemma 4.2 that

$$\begin{aligned}0 \le \textstyle\sum_{t=1}^k \theta_t \mathbb{E}[Q(x^t, x^*)] \;&\le \theta_1(m(1+\tau_1)-1)[f(x^0) - \langle x^0 - x^*, \nabla f(x^*)\rangle - f(x^*)] \\ &\quad + \theta_1\eta_1 P(x^0, x^*) + \textstyle\sum_{t=1}^k (\frac{m-1}{m})^{t-1} \frac{2\theta_t\alpha_{t+1}}{m\eta_{t+1}}\sigma_0^2 - \frac{\theta_k(\mu+\eta_k)}{2}\mathbb{E}[P(x^k, x^*)],\end{aligned}$$

which, in view of the relation $-\langle x^0 - x^*, \nabla f(x^*)\rangle \le \langle x^0 - x^*, \mu w'(x^*)\rangle \le \mu w(x^0) - \mu w(x^*)$ and the convexity of $Q(\cdot, x^*)$, implies the first result in (4.13). Moreover, we can also conclude from the above inequality that

$$\tfrac{\theta_k(\mu+\eta_k)}{2}\mathbb{E}[P(x^k, x^*)] \;\le \theta_1(m(1+\tau_1)-1)[\psi(x^0) - \psi(x^*)] + \theta_1\eta_1 P(x^0, x^*) + \textstyle\sum_{t=1}^k (\frac{m-1}{m})^{t-1}\frac{2\theta_t\alpha_{t+1}}{m\eta_{t+1}}\sigma_0^2,$$

from which the second result in (4.13) follows. □

With the help of Proposition 4.3, we are now ready to prove Theorem 2.1, which establishes the convergence properties of RGEM. In particular, Theorem 2.1 shows that RGEM can achieve the optimal convergence rate as $\mathcal{O}\left\{\left(m + \sqrt{m\hat{L}/\mu}\right)\log 1/\epsilon\right\}$ for strongly convex problems.

Proof of Theorem 2.1. Letting $\theta_t = \alpha^{-t},\ t = 1, \ldots, k$, we can easily check that parameter setting in (2.9) with α defined in (2.10) satisfies conditions (4.5) and (4.9)-(4.12) stated in Proposition 4.3. It then follows from (2.9) and (4.13) that

$$\begin{aligned}\mathbb{E}[Q(\underline{x}^k, x^*)] &\le \tfrac{\alpha^k}{1-\alpha^k}\left[\mu P(x^0, x^*) + \psi(x^0) - \psi^* + \tfrac{2m(1-\alpha)^2\sigma_0^2}{(m-1)\mu}\textstyle\sum_{t=1}^k \left(\frac{m-1}{m\alpha}\right)^t\right], \\ \mathbb{E}[P(x^k, x^*)] &\le 2\alpha^k\left[P(x^0, x^*) + \tfrac{\psi(x^0)-\psi^*}{\mu} + \tfrac{2m(1-\alpha)^2\sigma_0^2}{(m-1)\mu^2}\textstyle\sum_{t=1}^k \left(\frac{m-1}{m\alpha}\right)^t\right],\ \forall k \ge 1.\end{aligned}$$

Also observe that $\alpha \ge \frac{2m-1}{2m}$, we then have

$$\textstyle\sum_{t=1}^k \left(\frac{m-1}{m\alpha}\right)^t \le \sum_{t=1}^k \left(\frac{2(m-1)}{2m-1}\right)^t \le 2(m-1).$$

Combining the above three relations and the fact that $m(1-\alpha) \le 1/2$, we have

$$\begin{aligned}&\mathbb{E}[Q(\underline{x}^k, x^*)] \le \tfrac{\alpha^k}{1-\alpha^k}\Delta_{0,\sigma_0}, \\ &\mathbb{E}[P(x^k, x^*)] \le 2\alpha^k\Delta_{0,\sigma_0}/\mu,\ \forall k \ge 1,\end{aligned} \tag{4.20}$$

where Δ_{0,σ_0} is defined in (2.13). The second relation immediately implies our bound in (2.11). Moreover, by the strong convexity of $P(\cdot,\cdot)$ in (1.11) and (2.11), we have

$$\begin{aligned}\tfrac{L_f}{2}\mathbb{E}[\|\underline{x}^k - x^*\|^2] \;&\le \tfrac{L_f}{2}(\textstyle\sum_{t=1}^k\theta_t)^{-1}\sum_{t=1}^k\theta_t\mathbb{E}[\|x^t - x^*\|^2] \overset{(1.11)}{\le} L_f\tfrac{(1-\alpha)\alpha^k}{1-\alpha^k}\textstyle\sum_{t=1}^k\alpha^{-t}\mathbb{E}[P(x^t, x^*)] \\ &\overset{(2.11)}{\le} \tfrac{L_f(1-\alpha)\alpha^k}{1-\alpha^k}\textstyle\sum_{t=1}^k\frac{2\Delta_{0,\sigma_0}}{\mu} = \frac{2L_f(1-\alpha)\Delta_{0,\sigma_0}k\alpha^k}{\mu(1-\alpha^k)}.\end{aligned}$$

Combining the above relation with the first inequality in (4.20) and (4.4), we obtain

$$\mathbb{E}[\psi(\underline{x}^k) - \psi(x^*)] \overset{(4.4)}{\le} \mathbb{E}[Q(\underline{x}^k, x^*)] + \tfrac{L_f}{2}\mathbb{E}[\|\underline{x}^k - x^*\|^2] \le \left(1 + \tfrac{2L_f(1-\alpha)}{\mu}k\right)\tfrac{\Delta_{0,\sigma_0}\alpha^k}{1-\alpha^k}.$$

Observing that

$$\begin{aligned}\tfrac{1}{1-\alpha} &\le \tfrac{16}{3}\max\{m, \hat{L}/\mu\}, \\ (k+1)\tfrac{\alpha^k(1-\alpha)}{1-\alpha^k} &= \left(\textstyle\sum_{t=1}^k\frac{\alpha^t}{\alpha^t} + 1\right)\tfrac{\alpha^k(1-\alpha)}{1-\alpha^k} \le \left(\textstyle\sum_{t=1}^k\frac{\alpha^t}{\alpha^{3t/2}} + 1\right)\tfrac{\alpha^k(1-\alpha)}{1-\alpha^k} \\ &\le \tfrac{1-\alpha^{k/2}}{\alpha^{k/2}(1-\alpha^{1/2})}\tfrac{\alpha^k(1-\alpha)}{1-\alpha^k} + \alpha^k \le 2\alpha^{k/2} + \alpha^k \le 3\alpha^{k/2},\end{aligned}$$

we have

$$\mathbb{E}[\psi(\underline{x}^k) - \psi(x^*)] \le \tfrac{16}{3}\max\left\{m, \tfrac{\hat{L}}{\mu}\right\}\tfrac{\Delta_{0,\sigma_0}(k+1)\alpha^k(1-\alpha)}{1-\alpha^k} \le 16\max\left\{m, \tfrac{\hat{L}}{\mu}\right\}\Delta_{0,\sigma_0}\alpha^{k/2}.$$

4.2. Convergence analysis of RGEM for stochastic finite-sum optimization. Our goal in this section is to establish the convergence properties of RGEM for solving stochastic finite-sum optimization problems in (1.5). For notation convenience, we use $\mathbb{E}_{[i_k]}$ for taking expectation over $\{i_1, \ldots, i_k\}$, $\mathbb{E}_\xi$ for expectations over $\{\xi^1, \ldots, \xi^k\}$, respectively, we use $\mathbb{E}$ to denote the expectations over all random variables.

Note that the parameter $\{B_t\}$ in Algorithm 2 denotes the batch size used to compute $y_{i_t}^t$ in (2.17). Since we now assume that $\|\cdot\|$ is associated with a certain inner product, it can be easily seen from (2.17), and the two assumptions we have for the stochastic gradients computed by SFO oracle, i.e., (2.15) and (2.16), that

$$\mathbb{E}_\xi[y_{i_t}^t] = \nabla f_{i_t}(\underline{x}_{i_t}^t) \text{ and } \mathbb{E}_\xi[\|y_{i_t}^t - \nabla f_{i_t}(\underline{x}_{i_t}^t)\|_*^2] \le \tfrac{\sigma^2}{B_t},\ \forall i_t, t = 1, \ldots, k, \tag{4.21}$$

and hence $y_{i_t}^t$ is an unbiased estimator of $\nabla f_{i_t}(\underline{x}_{i_t}^t)$. Moreover, for y^t generated by Algorithm 2, we can see that

$$y_i^t = \begin{cases} \mathbf{0}, & \text{if the } i\text{-th block has never been updated for the first } t \text{ iterations;} \\ \frac{1}{B_l}\sum_{j=1}^{B_l} G_i(\underline{x}_i^l, \xi_{i,j}^l), & \text{if the latest update happened at } l\text{-th iteration, for } 1 \le l \le t. \end{cases} \tag{4.22}$$

We first establish some general convergence properties for Algorithm 2.

PROPOSITION 4.4. *Let x^t and $\underline{x}^k$ be defined as in (2.2) and (2.5), respectively, and x^* be an optimal solution of (1.5). Suppose that σ_0 and σ are defined in (2.8) and (2.16), respectively, and $\{\eta_t\}$, $\{\tau_t\}$, and $\{\alpha_t\}$ in Algorithm 2 satisfy (4.5), (4.9) (4.10), and (4.12) for some $\theta_t \ge 0$, $t = 1, \ldots, k$. Moreover, if*

$$3\alpha_t L_i \le m\tau_{t-1}\eta_t,\ i = 1, \ldots, m; t \ge 2, \tag{4.23}$$

then for any $k \ge 1$, we have

$$\begin{aligned} \mathbb{E}[Q(\underline{x}^k, x^*)] &\le (\textstyle\sum_{t=1}^k \theta_t)^{-1}\tilde{\Delta}_{0,\sigma_0,\sigma}, \\ \mathbb{E}[P(x^k, x^*)] &\le \tfrac{2\tilde{\Delta}_{0,\sigma_0,\sigma}}{\theta_k(\mu+\eta_k)}, \end{aligned} \tag{4.24}$$

where

$$\tilde{\Delta}_{0,\sigma_0,\sigma} := \tilde{\Delta}_{0,\sigma_0} + \textstyle\sum_{t=2}^k \frac{3\theta_{t-1}\alpha_t\sigma^2}{2m\eta_t B_{t-1}} + \sum_{t=1}^k \frac{2\theta_t\alpha_{t+1}}{m^2\eta_{t+1}}\sum_{l=1}^{t-1}(\frac{m-1}{m})^{t-1-l}\frac{\sigma^2}{B_l}, \tag{4.25}$$

with $\tilde{\Delta}_{0,\sigma_0}$ defined in (4.14).

Proof. Observe that in Algorithm 2 y^t is updated as in (2.17). Therefore, according to (4.2), we have

$$\hat{y}_i^t = \tfrac{1}{B_t}\textstyle\sum_{j=1}^{B_t} G_i(\hat{\underline{x}}_i^t, \xi_{i,j}^t),\ i = 1, \ldots, m,\ t \ge 1,$$

which together with the first relation in (4.21) imply that $\mathbb{E}_\xi[\langle \hat{y}_i^t, x - \hat{\underline{x}}_i^t\rangle] = \mathbb{E}_\xi[\langle \nabla f_i(\hat{\underline{x}}_i^t), x - \hat{\underline{x}}_i^t\rangle]$. Hence, we can rewrite (4.16) as

$$\begin{aligned} \mathbb{E}_\xi[\tfrac{1+\tau_t}{m}\textstyle\sum_{i=1}^m f_i(\hat{\underline{x}}_i^t) + \mu w(x^t) - \psi(x)] &\le \mathbb{E}_\xi\left[\tfrac{1+\tau_t}{m}\textstyle\sum_{i=1}^m f_i(\hat{\underline{x}}_i^t) + \mu w(x^t) - \mu w(x) - \tfrac{1}{m}\sum_{i=1}^m [f_i(\hat{\underline{x}}_i^t) + \langle \nabla f_i(\hat{\underline{x}}_i^t), x - \hat{\underline{x}}_i^t\rangle]\right] \\ &= \mathbb{E}_\xi\left[\tfrac{1+\tau_t}{m}\textstyle\sum_{i=1}^m f_i(\hat{\underline{x}}_i^t) + \mu w(x^t) - \mu w(x) - \tfrac{1}{m}\sum_{i=1}^m [f_i(\hat{\underline{x}}_i^t) + \langle \hat{y}_i^t, x - \hat{\underline{x}}_i^t\rangle]\right] \\ &\le \mathbb{E}_\xi\big[-\tfrac{\tau_t}{2m}\textstyle\sum_{i=1}^m \frac{1}{L_i}\|\nabla f_i(\hat{\underline{x}}_i^t) - \nabla f_i(\underline{x}_i^{t-1})\|_*^2 + \frac{\tau_t}{m}\sum_{i=1}^m f_i(\underline{x}_i^{t-1}) \\ &\quad + \langle x^t - x, \tfrac{1}{m}\textstyle\sum_{i=1}^m [\hat{y}_i^t - y_i^{t-1} - \alpha_t(y_i^{t-1} - y_i^{t-2})]\rangle \\ &\quad + \eta_t P(x^{t-1}, x) - (\mu+\eta_t)P(x^t, x) - \eta_t P(x^{t-1}, x^t)\big], \end{aligned}$$

Following the same procedure as in the proof of Proposition 4.3, we obtain the following similar relation (cf. (4.18))

$$\begin{aligned}
&\theta_k(1+\tau_k)\textstyle\sum_{i=1}^m \mathbb{E}[f_i(\underline{x}_i^k)] + \sum_{t=1}^k \theta_t \mathbb{E}[\mu w(x^t) - \psi(x)] + \frac{\theta_k(\mu+\eta_k)}{2}\mathbb{E}[P(x^k,x)] \\
&\quad \le \theta_1(m(1+\tau_1)-1)f(x^0) + \theta_1\eta_1 P(x^0,x) \\
&\quad + \textstyle\sum_{t=2}^k \mathbb{E}\left[-\frac{\theta_t\alpha_t}{m}\langle x^t - x^{t-1}, y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\rangle - \theta_t\eta_t P(x^{t-1},x^t) - \frac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}}\|\nabla f_{i_{t-1}}(x_{i_{t-1}}^{t-1}) - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2\right] \\
&\quad + \theta_k\mathbb{E}\left[\langle x^k - x, y_{i_k}^k - y_{i_k}^{k-1}\rangle - \frac{(\mu+\eta_k)}{2}P(x^k,x) - \frac{\tau_k}{2L_{i_k}}\|\nabla f_{i_k}(x_{i_k}^k) - \nabla f_{i_k}(\underline{x}_{i_k}^{k-1})\|_*^2\right].
\end{aligned}$$

By the strong convexity of $P(\cdot,\cdot)$ in (1.11), the fact that $b\langle u,v\rangle - a\|v\|^2/2 \le b^2\|u\|^2/(2a), \forall a>0$, and the Cauchy-Schwartz inequality, we have, for $t=2,\ldots,k$,

$$\begin{aligned}
&\mathbb{E}[-\tfrac{\theta_t\alpha_t}{m}\langle x^t - x^{t-1}, y_{i_{t-1}}^{t-1} - y_{i_{t-1}}^{t-2}\rangle - \theta_t\eta_t P(x^{t-1},x^t) - \tfrac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}}\|\nabla f_{i_{t-1}}(x_{i_{t-1}}^{t-1}) - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2] \\
&\overset{(1.11)}{\le} \mathbb{E}[-\tfrac{\theta_t\alpha_t}{m}\langle x^t - x^{t-1}, y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-1}) + \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-1}) - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2}) + \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2}) - y_{i_{t-1}}^{t-2}\rangle] \\
&\quad - \mathbb{E}\left[\tfrac{\theta_t\eta_t}{2}\|x^{t-1}-x^t\|^2 + \tfrac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}}\|\nabla f_{i_{t-1}}(x_{i_{t-1}}^{t-1}) - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2\right] \\
&\le \mathbb{E}\left[\left(\tfrac{3\theta_{t-1}\alpha_t}{2m\eta_t} - \tfrac{\theta_{t-1}\tau_{t-1}}{2L_{i_{t-1}}}\right)\|\nabla f_{i_{t-1}}(x_{i_{t-1}}^{t-1}) - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2})\|_*^2\right] \\
&\quad + \tfrac{3\theta_{t-1}\alpha_t}{2m\eta_t}\mathbb{E}\left[\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-1})\|_*^2 + \|\nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2}) - y_{i_{t-1}}^{t-2}\|_*^2\right] \\
&\overset{(4.23)}{\le} \tfrac{3\theta_{t-1}\alpha_t}{2m\eta_t}\mathbb{E}\left[\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-1})\|_*^2 + \|\nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-2}) - y_{i_{t-1}}^{t-2}\|_*^2\right].
\end{aligned}$$

Similarly, we can also obtain

$$\begin{aligned}
&\mathbb{E}\left[\langle x^k - x, y_{i_k}^k - y_{i_k}^{k-1}\rangle - \tfrac{(\mu+\eta_k)}{2}P(x^k,x) - \tfrac{\tau_k}{2L_{i_k}}\|f_{i_k}(x_{i_k}^k) - \nabla f_{i_k}(\underline{x}_{i_k}^{k-1})\|_*^2\right] \\
&\overset{(4.21),(1.11)}{\le} \mathbb{E}\left[\langle x^k - x, \nabla f_{i_k}(\underline{x}_{i_k}^k) - \nabla f_{i_k}(\underline{x}_{i_k}^{k-1}) + \nabla f_{i_k}(\underline{x}_{i_k}^{k-1}) - y_{i_k}^{k-1}\rangle\right] \\
&\quad - \mathbb{E}\left[\tfrac{(\mu+\eta_k)}{4}\|x^k - x\|^2 + \tfrac{\tau_k}{2L_{i_k}}\|f_{i_k}(x_{i_k}^k) - \nabla f_{i_k}(\underline{x}_{i_k}^{k-1})\|_*^2\right] \\
&\le \mathbb{E}\left[\left(\tfrac{2}{\mu+\eta_k} - \tfrac{\tau_k}{2L_{i_k}}\right)\|\nabla f_{i_k}(x_{i_k}^k) - \nabla f_{i_k}(\underline{x}_{i_k}^{k-1})\|_*^2 + \tfrac{2}{\mu+\eta_k}\|\nabla f_{i_k}(\underline{x}_{i_k}^{k-1}) - y_{i_k}^{k-1}\|_*^2\right] \\
&\overset{(4.12)}{\le} \mathbb{E}\left[\tfrac{2}{\mu+\eta_k}\|\nabla f_{i_k}(\underline{x}_{i_k}^{k-1}) - y_{i_k}^{k-1}\|_*^2\right].
\end{aligned}$$

Combining the above three relations, and using the fact that $m\eta_{k+1} \le \alpha_{k+1}(\mu+\eta_k)$ (induced from (4.9) and (4.10)), we have

$$\begin{aligned}
&\theta_k(1+\tau_k)\textstyle\sum_{i=1}^m \mathbb{E}[f_i(\underline{x}_i^k)] + \sum_{t=1}^k \theta_t \mathbb{E}[\mu w(x^t) - \psi(x)] + \frac{\theta_k(\mu+\eta_k)}{2}\mathbb{E}[P(x^k,x)] \\
&\quad \le \theta_1(m(1+\tau_1)-1)f(x^0) + \theta_1\eta_1 P(x^0,x) \\
&\quad\quad + \textstyle\sum_{t=2}^k \frac{3\theta_{t-1}\alpha_t}{2m\eta_t}\mathbb{E}[\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-1})\|_*^2] + \sum_{t=1}^k \frac{2\theta_t\alpha_{t+1}}{m\eta_{t+1}}\mathbb{E}[\|\nabla f_{i_t}(\underline{x}_{i_t}^{t-1}) - y_{i_t}^{t-1}\|_*^2].
\end{aligned}$$

Moreover, in view of the second relation in (4.21), we have

$$\mathbb{E}[\|y_{i_{t-1}}^{t-1} - \nabla f_{i_{t-1}}(\underline{x}_{i_{t-1}}^{t-1})\|_*^2] \le \tfrac{\sigma^2}{B_{t-1}},\ \forall t \ge 2.$$

Let us denote $\mathcal{E}_{i_t,t} := \max\{l : i_l = i_t, l < t\}$ with $\mathcal{E}_{i_t,t} = 0$ denoting the event that the i_t-th block has never been updated until iteration t, we can also conclude that for any $t \ge 1$

$$\begin{aligned}
\mathbb{E}[\|\nabla f_{i_t}(\underline{x}_{i_t}^{t-1}) - y_{i_t}^{t-1}\|_*^2] &= \textstyle\sum_{l=0}^{t-1}\mathbb{E}\left[\|\nabla f_{i_l}(\underline{x}_{i_l}^l) - y_{i_l}^l\|_*^2 | \{\mathcal{E}_{i_t,t} = l\}\right]\mathrm{Prob}\{\mathcal{E}_{i_t,t} = l\} \\
&\le (\tfrac{m-1}{m})^{t-1}\sigma_0^2 + \textstyle\sum_{l=1}^{t-1}\frac{1}{m}(\frac{m-1}{m})^{t-1-l}\frac{\sigma^2}{B_l},
\end{aligned}$$

where the first term in the inequality corresponds to the case when the i_t-block has never been updated for the first $t-1$ iterations, and the second term represents that its latest update for the first $t-1$ iterations happened at the l-th iteration. Hence, using Lemma 4.2 and following the same argument as in the proof of Proposition 4.3, we obtain our results in (4.24). □

We are now ready to prove Theorem 2.2, which establishes an optimal complexity bound (up to a logarithmic factor) on the number of calls to the SFO oracle and a linear rate of convergence in terms of the communication complexity for solving problem (1.5).

Proof of Theorem 2.2 Let us set $\theta_t = \alpha^{-t},\ t = 1, \ldots, k$. It is easy to check that the parameter setting in (2.9) with α defined in (2.10) satisfies conditions (4.5), (4.9), (4.10), (4.12), and (4.23) as required by Proposition 4.4. By (2.9), the definition of B_t in (2.18), and the fact that $\alpha \geq \frac{2m-1}{2m} > (m-1)/m$, we have

$$\begin{aligned}
\textstyle\sum_{t=2}^{k} \frac{3\theta_{t-1}\alpha_t\sigma^2}{2m\eta_t B_{t-1}} &\leq \textstyle\sum_{t=2}^{k} \frac{3\sigma^2}{2\mu(1-\alpha)k} \leq \frac{3\sigma^2}{2\mu(1-\alpha)}, \\
\textstyle\sum_{t=1}^{k} \frac{2\theta_t\alpha_{t+1}}{m^2\eta_{t+1}} \textstyle\sum_{l=1}^{t-1} (\frac{m-1}{m})^{t-1-l} \frac{\sigma^2}{B_l} &\leq \frac{2\sigma^2}{\alpha\mu m(1-\alpha)k} \textstyle\sum_{t=1}^{k} (\frac{m-1}{m\alpha})^{t-1} \textstyle\sum_{l=1}^{t-1} (\frac{m\alpha}{m-1})^{l} \\
&\leq \frac{2\sigma^2}{\mu(1-\alpha)m\alpha k} \textstyle\sum_{t=1}^{k} (\frac{m-1}{m\alpha})^{t-1} (\frac{m\alpha}{m-1})^{t-1} \frac{1}{1-(m-1)/(m\alpha)} \\
&\leq \frac{2\sigma^2}{\mu(1-\alpha)} \frac{1}{m\alpha-(m-1)} \leq \frac{4\sigma^2}{\mu(1-\alpha)}.
\end{aligned}$$

Hence, similar to the proof of Theorem 2.1, using the above relations and (2.9) in (4.24), we obtain

$$\begin{aligned}
\mathbb{E}[Q(\underline{x}^k, x^*)] &\leq \tfrac{\alpha^k}{1-\alpha^k} \left[\Delta_{0,\sigma_0} + \tfrac{5\sigma^2}{\mu}\right], \\
\mathbb{E}[P(x^k, x^*)] &\leq 2\alpha^k \left[\Delta_{0,\sigma_0} + \tfrac{5\sigma^2}{\mu^2}\right],
\end{aligned}$$

where Δ_{0,σ_0} is defined in (2.13). The second relation implies our results in (2.19). Moreover, (2.20) follows from the same argument as we used in proving Theorem 2.1.

5. Concluding remarks. In this paper, we propose a new randomized incremental gradient method, referred to as random gradient extrapolation method, for solving the classes of deterministic finite-sum optimization problems in (1.1) and stochastic finite-sum optimization problems in (1.5), respectively. We demonstrate that without any exact gradient evaluation even at the initial point, this algorithm achieves optimal linear rate of convergence for deterministic strongly convex problems, as well as exhibiting optimal sublinear rate of convergence (up to a logarithmic factor) for stochastic strongly convex problems. All these complexity bounds have been established in terms of the total number of gradient computations of component function f_i and the latter complexity bound on the computation of stochastic gradients is in fact asymptotically independent of the number of components m. Moreover, we consider solving finite-sum problems in (1.1) and (1.5) in a distributed network setting with m agents connected to a central server. Since each iteration of our proposed algorithm only involves constant number of communication rounds between the server and one randomly selected agent, it achieves linear communication complexity and avoids synchronous delays among agents. It is worth pointing out that by exploiting the mini-batch technique, the algorithm can also achieve linear communication complexity for solving stochastic finite-sum problems, which is the best-known communication complexity for distributed stochastic optimization problems in the literature.

REFERENCES

[1] Zeyuan Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. *ArXiv e-prints, abs/1603.05953*, 2016.

[2] A. Auslender and M. Teboulle. Interior gradient and proximal methods for convex and conic optimization. *SIAM Journal on Optimization*, 16:697–725, 2006.

[3] H.H. Bauschke, J.M. Borwein, and P.L. Combettes. Bregman monotone optimization algorithms. *SIAM Journal on Controal and Optimization*, 42:596–636, 2003.

[4] D. Blatt, A. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2007.

[5] L.M. Bregman. The relaxation method of finding the common point convex sets and its application to the solution of problems in convex programming. *USSR Comput. Math. Phys.*, 7:200–217, 1967.

[6] Yair Censor and Arnold Lent. An iterative row-action method for interval convex programming. *Journal of Optimization theory and Applications*, 34(3):321–353, 1981.

[7] C. Dang and G. Lan. Randomized first-order methods for saddle point optimization. Manuscript, Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611, USA, September 2014.

[8] A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. *Advances of Neural Information Processing Systems (NIPS)*, 27, 2014.

[9] Ofer Dekel, Ran Gilad-Bachrach, Ohad Shamir, and Lin Xiao. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(Jan):165–202, 2012.

[10] Hamid Reza Feyzmahdavian, Arda Aytekin, and Mikael Johansson. An asynchronous mini-batch algorithm for regularized stochastic optimization. *IEEE Transactions on Automatic Control*, 61(12):3740–3754, 2016.

[11] S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, I: a generic algorithmic framework. *SIAM Journal on Optimization*, 22:1469–1492, 2012.

[12] S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, II: shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23:2061–2089, 2013.

[13] S. Ghadimi and G. Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic optimization. Technical report, Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611, USA, June 2013.

[14] Elad Hazan and Haipeng Luo. Variance-reduced and projection-free stochastic optimization. *CoRR, abs/1602.02101*, 2, 2016.

[15] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Fundamentals of convex analysis*. Springer Science & Business Media, 2012.

[16] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances of Neural Information Processing Systems (NIPS)*, 26:315–323, 2013.

[17] K.C. Kiwiel. Proximal minimization methods with generalized bregman functions. *SIAM Journal on Controal and Optimization*, 35:1142–1168, 1997.

[18] G. Lan. Efficient methods for stochastic composite optimization. Manuscript, Georgia Institute of Technology, 2008.

[19] G. Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1):365–397, 2012.

[20] G. Lan, Z. Lu, and R. D. C. Monteiro. Primal-dual first-order methods with $\mathcal{O}(1/\epsilon)$ iteration-complexity for cone programming. *Mathematical Programming*, 126:1–29, 2011.

[21] Guanghui Lan, Soomin Lee, and Yi Zhou. Communication-efficient algorithms for decentralized and stochastic optimization. *arXiv preprint arXiv:1701.03961*, 2017.

[22] Guanghui Lan and Yi Zhou. An optimal randomized incremental gradient method. *arXiv preprint arXiv:1507.02000*, 2015.

[23] Soomin Lee, Angelia Nedich, and Maxim Raginsky. Stochastic dual averaging for decentralized online optimization on time-varying communication graphs. *IEEE Transactions on Automatic Control*, 2017.

[24] H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. Technical report, 2015. hal-01160728.

[25] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, et al. Communication-efficient learning of deep networks from decentralized data. *arXiv preprint arXiv:1602.05629*, 2016.

[26] A. S. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19:1574–1609, 2009.

[27] A. S. Nemirovskii and D. Yudin. On cezari's convergence of the steepest descent method for approximating saddle point of convex-concave functions. *Doklady Akademiia Nauk SSSR*, 239:No. 5, 1978. English translation: *Soviet Mathematics Loklady* 19, No. 2.

[28] Y. E. Nesterov. *Introductory Lectures on Convex Optimization: a basic course*. Kluwer Academic Publishers, Massachusetts, 2004.

[29] M. Schmidt, N. L. Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. Technical report, September 2013.

[30] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss. *Journal of Machine Learning Research*, 14(1):567599, 2013.

[31] S. Shalev-Shwartz and T. Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, 2015. to appear.

[32] Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.

[33] Thomas Strohmer and Roman Vershynin. A randomized kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, 2009.

[34] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. Manuscript, University of Washington, Seattle, May 2008.

[35] Lin Xiao and Tong Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

[36] Yuchen Zhang and Lin Xiao. Stochastic primal-dual coordinate method for regularized empirical risk minimization. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 353–361, 2015.