Global convergence rate of incremental aggregated gradient methods for nonsmooth problems

GLOBAL CONVERGENCE RATE OF PROXIMAL INCREMENTAL AGGREGATED GRADIENT METHODS

N. D. VANLI*, M. GÜRBÜZBALABAN*, AND A. OZDAGLAR*

Abstract. We focus on the problem of minimizing the sum of smooth component functions (where the sum is strongly convex) and a non-smooth convex function, which arises in regularized empirical risk minimization in machine learning and distributed constrained optimization in wireless sensor networks and smart grids. We consider solving this problem using the proximal incremental aggregated gradient (PIAG) method, which at each iteration moves along an aggregated gradient (formed by incrementally updating gradients of component functions according to a deterministic order) and taking a proximal step with respect to the non-smooth function. While the convergence properties of this method with randomized orders (in updating gradients of component functions) have been investigated, this paper, to the best of our knowledge, is the first study that establishes the convergence rate properties of the PIAG method for any deterministic order. In particular, we show that the PIAG algorithm is globally convergent with a linear rate provided that the step size is sufficiently small. We explicitly identify the rate of convergence and the corresponding step size to achieve this convergence rate. Our results improve upon the best known condition number dependence of the convergence rate of the incremental aggregated gradient methods used for minimizing a sum of smooth functions.

1. Introduction. We focus on *composite additive cost optimization problems*, where the objective function is given by the sum of m component functions $f_i(x)$ and a possibly non-smooth regularization function $r(x)$:

$$\min_{x\in\mathbb{R}^n} F(x) \triangleq f(x) + r(x), \tag{1.1}$$

and $f(x) = \frac{1}{m}\sum_{i=1}^m f_i(x)$. We assume each component function $f_i : \mathbb{R}^n \to (-\infty, \infty)$ is convex and continuously differentiable while the regularization function $r : \mathbb{R}^n \to (-\infty, \infty]$ is proper, closed, and convex but not necessarily differentiable. This formulation arises in many problems in machine learning, distributed optimization, and signal processing. Notable examples include constrained and regularized least squares problems that arise in various machine learning applications [7,21], distributed optimization problems that arise in wireless sensor network as well as smart grid applications [11, 19] and constrained optimization of separable problems [1]. An important feature of this formulation is that the number of component functions m is large, hence solving this problem using a standard gradient method that involves evaluating the full gradient of $f(x)$, i.e., $\nabla f(x) = \sum_{i=1}^m \nabla f_i(x)$, is costly. This motivates using *incremental methods* that exploit the additive structure of the problem and update the decision vector using one component function at a time.

When r is continuously differentiable, one widely studied approach is the incremental gradient (IG) method [1, 18, 24]. The IG method processes the component functions one at a time by taking steps

*Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139, USA. email: {denizcan, mertg, asuman}@mit.edu.

along the gradient of each individual function in a sequential manner, following a cyclic order [26,27] or a randomized order [13,22,27]. A particular randomized order, which at each iteration independently picks a component function uniformly at random from all component functions leads to the popular stochastic gradient descent (SGD) method. While SGD is the method of choice in practice for many machine learning applications due to its superior empirical performance and convergence rate estimates that does not depend on the number of component functions m, its convergence rate is sublinear, i.e., an ϵ-optimal solution can be computed within $O(1/\epsilon)$ iterations.[1] In a seminal paper, Blatt *et al.* [5] proposed the *incremental aggregated gradient (IAG) method*, which maintains the savings associated with incrementally accessing the component functions, but keeps the most recent component gradients in memory to approximate the full gradient $\nabla f(x)$ and updates the iterate using this aggregated gradient. Blatt *et al.* showed that under some assumptions, for a sufficiently small constant step size, the IAG method is globally convergent and when the component functions are quadratics, it achieves a linear rate. Two recent papers, [23] and [12], investigated the convergence rate of this method for general component functions that are convex and smooth (i.e., with Lipschitz gradients), where the sum of the component functions is strongly convex: In [23], the authors focused on a randomized version, called stochastic average gradient (SAG) method (which samples the component functions independently similar to SGD), and showed that it achieves a linear rate using a proof that relies on the stochastic nature of the algorithm. In a more recent work [12], the authors focused on deterministic IAG (i.e., component functions processed using an arbitrary deterministic order) and provided a simple analysis that uses a delayed dynamical system approach to study the evolution of the iterates generated by this algorithm.

While these recent advances suggest IAG as a promising approach with fast convergence rate guarantees for solving additive cost problems, in many applications listed above, the objective function takes a composite form and includes a non-smooth regularization function $r(x)$ (to avoid overfitting or to induce a sparse representation). Another important case of interest is smooth constrained optimization problems which can be represented in the composite form (1.1) where the function $r(x)$ is the indicator function of a nonempty closed convex set.

In this paper, we study the *proximal incremental aggregated gradient (PIAG) method* for solving composite additive cost optimization problems. Our method computes an aggregated gradient for the function $f(x)$ (with component gradients evaluated in a *deterministic manner* at outdated iterates over a finite window K, similar to IAG) and uses a proximal operator with respect to the regularization function $r(x)$ at the intermediate iterate obtained by moving along the aggregated gradient. Under the assumptions that $f(x)$ is strongly convex and each $f_i(x)$ is smooth with Lipschitz gradients, we show

[1] Let x^* be an optimal solution of the problem (1.1). A vector $x \in \mathbb{R}^n$ is an *ϵ-optimal solution* if $F(x) - F(x^*) \leq \epsilon$.

the first *linear convergence rate* result for the deterministic PIAG and provide explicit convergence rate estimates that highlight the dependence on the condition number of the problem (which we denote by Q) and the size of the window K over which outdated component gradients are evaluated. In particular, we show that in order to achieve an ϵ-optimal solution, the PIAG algorithm requires $\mathcal{O}(QK^2 \log^2(QK) \log(1/\epsilon))$ iterations, or equivalently $\widetilde{\mathcal{O}}(QK^2 \log(1/\epsilon))$ iterations, where the tilde is used to hide the logarithmic terms in Q and K. This result improves upon the condition number dependence of the deterministic IAG for smooth problems [12], where the authors proved that to achieve an ϵ-optimal solution, the IAG algorithm requires $\mathcal{O}(Q^2K^2 \log(1/\epsilon))$ iterations. We also note that two recent independent papers [9, 15] have analyzed the convergence rate of the prox-gradient algorithm (which is a special case of our algorithm with $K = 0$, i.e., where we have access to a full gradient at each iteration instead of an aggregated gradient) under strong convexity type assumptions and provided linear rate estimates. Our rate estimates for the PIAG algorithm with $K > 0$ matches the condition number dependence of the prox-gradient algorithm provided in these papers [9, 15] up to logarithmic factors. Furthermore, for the case $K = 0$ (i.e., for the prox-gradient algorithm), the rate estimates obtained using our analysis technique can be shown to have the same condition number dependence as the ones presented in [9, 15].

Our analysis uses function values to track the evolution of the iterates generated by the PIAG algorithm. This is in contrast with the recent analysis of the IAG algorithm provided in [12], which used distances of the iterates to the optimal solution as a *Lyapunov function* and relied on the smoothness of the problem to bound the gradient errors with distances. This approach does not extend to the non-smooth composite case, which motivates a new analysis using function values and the properties of the proximal operator. Since we work directly with function values, this approach also allows us to obtain iteration complexity results to achieve an ϵ-optimal solution.

In terms of the algorithmic structure, our paper is related to [7], where the authors introduce the SAGA method, which extends the SAG method to the composite case and provides a linear convergence rate result with an analysis that relies on the stochastic nature of the algorithm and does not extend to the deterministic case. Particularly, the SAGA method samples the component functions randomly and independently at each iteration without replacement (in contrast with the PIAG method, where the component functions are processed deterministically). However, such random sampling may not be possible for applications such as decentralized information processing in wireless sensor networks (where agents are subject to communication constraints imposed by the network topology and all agents are not necessarily connected to every other agent via a low-cost link [19]), motivating the study of the deterministic PIAG method. In [7], the authors prove that to achieve a point in the

ϵ-neighborhood of the optimal solution, SAGA requires $\mathcal{O}(\max(Q,K)\log(1/\epsilon))$ iterations.[2] However, note that this result does not translate into a guarantee in the function suboptimality of the resulting point because of lack of smoothness. Furthermore, the choice of Lyapunov function in [7] requires each $f_i(x)$ to be convex (to satisfy the non-negativity condition), whereas we do not need this assumption in our analysis.

Our work is also related to [26], where the authors propose a related linearly convergent incrementally updated gradient method for solving the composite additive cost problem in (1.1) under a local Lipschitzian error condition (a condition satisfied by locally strongly convex functions around an optimal solution). The PIAG algorithm is different from the algorithm proposed in [26]. Specifically, for constrained optimization problems (i.e., when the regularization function is the indicator function of a nonempty closed convex set), the iterates generated by the algorithm in [26] stay in the interior of the set since the algorithm in [26] searches for a feasible update direction. On the other hand, the PIAG algorithm uses the proximal map on the intermediate iterate obtained by moving in the opposite direction of the aggregated gradient, which operates as a projected gradient method and allows the iterates to be on the boundary of the set. Aside from algorithmic differences, [26] does not provide explicit rate estimates (even though the exact rate can be calculated after an elaborate analysis, the dependence on the condition number and the window length of the outdated gradients is significantly worse than the one presented in this paper). Furthermore, the results in [26] provides a K-step linear convergence, whereas the linear convergence results in our paper hold uniformly for each step.

Other than the papers mentioned above, our paper is also related to [4], which studies an alternative incremental aggregated proximal method and shows linear convergence when each $f_i(x)$ and $r(x)$ are continuously differentiable. This method forms a linear approximation to $f(x)$ and processes the component functions $f_i(x)$ with a proximal iteration, whereas our method processes $f_i(x)$ based on a gradient step. Furthermore, our linear convergence results do not require the differentiability of the objective function $r(x)$ in contrast to the analysis in [4].

Several recent papers in the machine learning literature (e.g., [7,8,14,16,17] and references therein) are also weakly related to our paper. In all these papers, the authors propose randomized order algorithms similar to the SAG algorithm [23] and analyze their convergence rates in expectation. In particular, in [8], the authors propose an algorithm, called Finito, which is closely related to the SAG algorithm but achieves a faster convergence rate than the SAG algorithm. These ideas are then extended to composite optimization problems with non-smooth objective functions (as in (1.1)) in [7, 17]. In particular, in [17], a majorization-minimization algorithm, called MISO, is proposed to solve smooth optimization problems and its global linear convergence is shown in expectation. In [16],

[2] Let x^* be an optimal solution of the problem (1.1). A vector $x \in \mathbb{R}^n$ is in the *ϵ-neighborhood of an optimal solution* if $||x - x^*|| \leq \epsilon$.

the ideas in [17] are then extended for non-smooth optimization problems using proximal operator. Similarly, in [14], a variance reduction technique is applied to the SGD algorithm for smooth problems and its global linear convergence in expectation is proven.

The rest of the paper is organized as follows. In Section 2, we introduce the PIAG algorithm. In Section 3, we first provide the assumptions on the objective functions and then prove the global linear convergence of the proposed algorithm under these assumptions. We conclude the paper in Section 4 with a summary of our results.

2. The PIAG Algorithm. Similar to the IAG method, at each iteration k, we form an aggregated gradient, which we denote as follows

$$g_k \triangleq \frac{1}{m}\sum_{i=1}^{m} \nabla f_i(x_{\tau_{i,k}}),$$

where $\nabla f_i(x_{\tau_{i,k}})$ represents the gradient of the ith component function sampled at time $\tau_{i,k}$. We assume that each component function is sampled at least once in the past $K \geq 0$ iterations, i.e., we have

$$k - K \leq \tau_{i,k} \leq k, \;\; \forall i \in \{1, \ldots, m\}.$$

This condition is typically satisfied in practical implementations of the deterministic incremental methods. For instance, if the functions are processed in a cyclic order, we have $K = m - 1$ [13, 26]. On the other hand, $K = 0$ corresponds to the case where we have the full gradient of the function $f(x)$ at each iteration (i.e., $g_k = \nabla f(x_k)$) and small K may represent a setting in which the gradients of the component functions are sent to a processor with some delay upper bounded by K.

Since the regularization function r is not necessarily differentiable, we propose to solve (1.1) with the proximal incremental aggregated gradient (PIAG) method, which uses the proximal operator with respect to the regularization function at the intermediate iterate obtained using the aggregated gradient. In particular, the PIAG algorithm, at each iteration $k \geq 0$, updates x_k as

$$x_{k+1} = \text{prox}_r^{\eta}(x_k - \eta g_k), \tag{2.1}$$

where η is a constant step size and the proximal mapping is defined as follows

$$\text{prox}_r^{\eta}(y) = \underset{x \in \mathbb{R}^n}{\arg\min} \left\{ \frac{1}{2} ||x - y||^2 + \eta r(x) \right\}. \tag{2.2}$$

Here, we define $\phi(x) \triangleq \frac{1}{2} ||x - y||^2 + \eta r(x)$ and let $\partial\phi(x)$ denote the set of subgradients of the function ϕ at x. Then, it follows from the optimality conditions [3] of the problem in (2.2) that $0 \in \partial\phi(x_{k+1})$. This yields $x_{k+1} - (x_k - \eta g_k) + \eta h_{k+1} = 0$, for some $h_{k+1} \in \partial r(x_{k+1})$. Hence, we can compactly represent our update rule as

$$x_{k+1} = x_k + \eta d_k, \tag{2.3}$$

where $d_k \triangleq -g_k - h_{k+1}$ is the direction of the update at time k.

3. Convergence Analysis.

3.1. Assumptions. Throughout the paper, we make the following standard assumptions.

ASSUMPTION 3.1. ***(Lipschitz gradients)*** *Each f_i has Lipschitz continuous gradients on $\mathbb{R}^n$ with some constant $L_i \geq 0$, i.e.,*

$$||\nabla f_i(x) - \nabla f_i(y)|| \leq L_i \, ||x - y|| \,,$$

for any $x, y \in \mathbb{R}^n$.[3]

Defining $L \triangleq \frac{1}{m} \sum_{i=1}^m L_i$, we observe that Assumption 3.1 and the triangle inequality yield

$$||\nabla f(x) - \nabla f(y)|| \leq L \, ||x - y|| \,,$$

for any $x, y \in \mathbb{R}^n$, i.e., the function f is L-smooth.

ASSUMPTION 3.2. ***(Strong Convexity)*** *The sum function f is μ-strongly convex on $\mathbb{R}^n$ for some $\mu > 0$, i.e., the function $x \mapsto f(x) - \frac{\mu}{2} ||x||^2$ is convex, and the regularization function r is convex on $(-\infty, \infty]$.*

A consequence of Assumption 3.2 is that F is strongly convex, hence there exists a unique optimal solution of problem (1.1) [20, Lemma 6], which we denote by x^*.

We emphasize that these assumptions hold for a variety of cost functions including regularized squared error loss, hinge loss, and logistic loss [6] and similar assumptions are widely used to analyze the convergence properties of incremental gradient methods in the literature [2, 4, 7, 12, 23]. Note that in contrast with many of these analyses, we do not assume that the component functions f_i are convex.

3.2. Rate of Convergence. In this section, we show that the PIAG algorithm attains a global linear convergence rate with a constant step size provided that the step size is sufficiently small. We define

$$F_k \triangleq F(x_k) - F(x^*), \tag{3.1}$$

which is the suboptimality in the objective value at iteration k. In our analysis, we will use F_k as a Lyapunov function to prove global linear convergence. Before providing the main theorems of the paper, we first introduce three lemmas that contain key relations in proving these theorems.

The first lemma investigates how the suboptimality in the objective value evolves over the iterations. In particular, it shows that the change in suboptimality $F_{k+1} - F_k$ can be bounded as a sum of two terms: The first term is negative and has a linear dependence in the step size η, whereas the

[3] If a function f has Lipschitz continuous gradients with some constant L, then f is called L-smooth. We use these terms interchangeably.

second term is positive and has a quadratic dependence in η. This suggests that if the step size η is small enough, the linear term in η will be dominant guaranteeing a descent in suboptimality.

LEMMA 3.3. *Suppose that Assumptions 1 and 2 hold. Then, the PIAG algorithm in* (2.1) *yields the following guarantee*

$$F_{k+1} \leq F_k - \frac{1}{2}\eta \left|\left|d_k\right|\right|^2 + \eta^2 \frac{L}{2} \sum_{j=(k-K)_+}^{k-1} \left|\left|d_j\right|\right|^2,$$

for any step size $0 < \eta \leq \frac{1}{L(K+1)}$.

Proof. We first consider the difference of the errors in consecutive time instances and write

$$\begin{aligned} F(x_{k+1}) - F(x_k) &= f(x_{k+1}) - f(x_k) + r(x_{k+1}) - r(x_k) \\ &\leq \langle \nabla f(x_k), x_{k+1} - x_k \rangle + \frac{L}{2} \left|\left|x_{k+1} - x_k\right|\right|^2 + r(x_{k+1}) - r(x_k), \end{aligned}$$

where the inequality follows from the Taylor series expansion of f around x_k and since the Hessian of f at any point is upper bounded by L by Assumption 3.1. Using the update rule $x_{k+1} = x_k + \eta d_k$ in this inequality, we obtain

$$\begin{aligned} F(x_{k+1}) - F(x_k) &= \eta \langle \nabla f(x_k), d_k \rangle + \eta^2 \frac{L}{2} \left|\left|d_k\right|\right|^2 + r(x_{k+1}) - r(x_k) \\ &= \eta \langle \nabla f(x_k) - g_k, d_k \rangle + \eta^2 \frac{L}{2} \left|\left|d_k\right|\right|^2 + \eta \langle g_k, d_k \rangle + r(x_{k+1}) - r(x_k) \\ &\leq \eta \left|\left|\nabla f(x_k) - g_k\right|\right| \left|\left|d_k\right|\right| + \eta^2 \frac{L}{2} \left|\left|d_k\right|\right|^2 - \eta \left|\left|d_k\right|\right|^2 - \eta \langle h_{k+1}, d_k \rangle + r(x_{k+1}) - r(x_k) \\ &= \eta \left|\left|\nabla f(x_k) - g_k\right|\right| \left|\left|d_k\right|\right| + \eta \left(\eta \frac{L}{2} - 1\right) \left|\left|d_k\right|\right|^2 + \langle h_{k+1}, x_k - x_{k+1} \rangle + r(x_{k+1}) - r(x_k) \\ &\leq \eta \left|\left|\nabla f(x_k) - g_k\right|\right| \left|\left|d_k\right|\right| + \eta \left(\eta \frac{L}{2} - 1\right) \left|\left|d_k\right|\right|^2, \end{aligned} \tag{3.2}$$

where the first inequality follows by the triangle inequality and the last inequality follows from the convexity of r.

The gradient error term in (3.2), i.e., $\left|\left|\nabla f(x_k) - g_k\right|\right|$, can be upper bounded as follows

$$\begin{aligned} \left|\left|\nabla f(x_k) - g_k\right|\right| &\leq \frac{1}{m} \sum_{i=1}^{m} \left|\left|\nabla f_i(x_k) - \nabla f_i(x_{\tau_{i,k}})\right|\right| \\ &\leq \frac{1}{m} \sum_{i=1}^{m} L_i \left|\left|x_k - x_{\tau_{i,k}}\right|\right| \\ &\leq \frac{1}{m} \sum_{i=1}^{m} L_i \sum_{j=\tau_{i,k}}^{k-1} \eta \left|\left|d_j\right|\right| \\ &\leq \eta L \sum_{j=(k-K)_+}^{k-1} \left|\left|d_j\right|\right|, \end{aligned} \tag{3.3}$$

where the first and third inequalities follow by the triangle inequality, the second inequality follows since each f_i is L_i-smooth, and the last inequality follows since $\tau_{i,k} \geq k - K$. Using (3.3) we can upper

bound (3.2) as follows

$$\begin{aligned} F(x_{k+1}) - F(x_k) &\leq \eta\left(\eta\frac{L}{2} - 1\right)||d_k||^2 + \eta^2 L \sum_{j=(k-K)_+}^{k-1} ||d_j||\,||d_k|| \\ &\leq \eta\left(\eta\frac{L(K+1)}{2} - 1\right)||d_k||^2 + \eta^2\frac{L}{2}\sum_{j=(k-K)_+}^{k-1} ||d_j||^2 \\ &\leq -\frac{\eta}{2}||d_k||^2 + \eta^2\frac{L}{2}\sum_{j=(k-K)_+}^{k-1} ||d_j||^2, \end{aligned} \tag{3.4}$$

where the second inequality follows from the arithmetic-geometric mean inequality, i.e., $||d_j||\,||d_k|| \leq \frac{1}{2}(||d_j||^2 + ||d_k||^2)$ and the last inequality follows since $0 < \eta \leq \frac{1}{L(K+1)}$. This concludes the proof of Lemma 3.3. □

We next introduce the following lemma, which can be viewed as an extension of [25, Theorem 4] into our framework with aggregated gradients. We provide a simplified proof compared to [25] with a tighter upper bound. This lemma can be interpreted as follows. When the regularization function is zero (i.e., $r(x) = 0$ for all $x \in \mathbb{R}^n$) and we have access to full gradients (i.e., $K = 0$), this lemma simply follows from the strong convexity of the sum function f since $||x_k - x^*|| \leq \frac{1}{\mu}||\nabla f(x_k) - \nabla f(x^*)||$ and $\nabla f(x^*) = 0$ due to the optimality condition of the problem. The following lemma indicates that even though we do not have such control over the subgradients of the regularization function (as the regularization function is neither strongly convex nor smooth), the properties of the proximal step yields a similar relation at the expense of a constant of 2 (instead of 1 compared to the $r(x) = 0$ case) and certain history dependent terms (that arise due to the incremental nature of the PIAG algorithm) that has a linear dependence in step size η. This lemma will be a key step in the proof of Lemma 3.5, where we illustrate how the descent term in Lemma 3.3 relates to our Lyapunov function.

LEMMA 3.4. *Suppose that Assumptions 1 and 2 hold and let $Q = L/\mu$ denote the condition number of the problem. Then, the distance of the iterates from the optimal solution is upper bounded as*

$$||x_k - x^*|| \leq \frac{2}{\mu}||d_k|| + 2\eta Q \sum_{j=(k-K)_+}^{k-1} ||d_j||,$$

for any $k \geq 0$ and $0 < \eta \leq \frac{1}{L}$.

Proof. Define

$$d'_k \triangleq \arg\min_{d\in\mathbb{R}^n}\left\{\frac{\eta}{2}||\nabla f(x_k) + d||^2 + r(x_k + \eta d)\right\},$$

as the direction of update with the full gradient. Non-expansiveness property of the proximal map implies

$$||\text{prox}_r^\eta(x) - \text{prox}_r^\eta(y)||^2 \leq \langle \text{prox}_r^\eta(x) - \text{prox}_r^\eta(y), x - y\rangle.$$

Putting $x = x_k - \eta \nabla f(x_k)$ and $y = x^* - \eta \nabla f(x^*)$ in the above inequality, we obtain

$$\begin{aligned} ||x_k + \eta d'_k - x^*||^2 &\leq \langle x_k + \eta d'_k - x^*,\, x_k - \eta \nabla f(x_k) - x^* + \eta \nabla f(x^*) \rangle \\ &\leq \langle x_k + \eta d'_k - x^*,\, x_k + \eta d'_k - x^* \rangle + \langle x_k + \eta d'_k - x^*,\, -\eta d'_k + \eta \nabla f(x^*) - \eta \nabla f(x_k) \rangle, \end{aligned}$$

which implies

$$0 \leq \langle x_k + \eta d'_k - x^*,\, -d'_k + \nabla f(x^*) - \nabla f(x_k) \rangle.$$

This inequality can be rewritten as follows

$$\begin{aligned} \langle x_k - x^*,\, \nabla f(x_k) - \nabla f(x^*) \rangle &\leq \langle x_k - x^*,\, -d'_k \rangle - \eta \, ||d'_k||^2 + \eta \langle d'_k,\, \nabla f(x^*) - \nabla f(x_k) \rangle \\ &\leq \langle x_k - x^*,\, -d'_k \rangle + \eta \langle d'_k,\, \nabla f(x^*) - \nabla f(x_k) \rangle \\ &\leq ||d'_k|| \, (||x_k - x^*|| + \eta \, ||\nabla f(x^*) - \nabla f(x_k)||) \\ &\leq ||d'_k|| \, (||x_k - x^*|| + \eta L \, ||x_k - x^*||) \\ &\leq 2 \, ||d'_k|| \, ||x_k - x^*|| \,, \end{aligned} \tag{3.5}$$

where the second inequality follows since $-\,||d'_k||^2 \leq 0$, the third inequality follows by the Cauchy-Schwarz inequality, the fourth inequality follows from the L-smoothness of f, and the last inequality follows since $\eta \leq \frac{1}{L}$. Since μ-strong convexity of f implies

$$\mu \, ||x_k - x^*||^2 \leq \langle x_k - x^*,\, \nabla f(x_k) - \nabla f(x^*) \rangle, \tag{3.6}$$

then combining (3.5) and (3.6), we obtain

$$\mu \, ||x_k - x^*|| \leq 2 \, ||d'_k|| \,. \tag{3.7}$$

In order to relate d'_k to the original direction of update d_k, we use the triangle inequality and write

$$\begin{aligned} ||d'_k|| &\leq ||d_k|| + ||d'_k - d_k|| \\ &= ||d_k|| + \frac{1}{\eta} \, ||x_k + \eta d'_k - x_k - \eta d_k|| \\ &= ||d_k|| + \frac{1}{\eta} \, ||\mathrm{prox}_r^\eta(x_k - \eta \nabla f(x_k)) - \mathrm{prox}_r^\eta(x_k - \eta g_k)|| \\ &\leq ||d_k|| + ||g_k - \nabla f(x_k)|| \,, \\ &\leq ||d_k|| + \eta L \sum_{j=(k-K)_+}^{k-1} ||d_j|| \,, \end{aligned} \tag{3.8}$$

where the last line follows by equation (3.3). Putting (3.8) back into (3.7) concludes the proof of Lemma 3.4. □

In the following lemma, we relate the direction of update to the suboptimality in the objective value at a given iteration k. In particular, we show that the descent term presented in Lemma 3.3

(i.e., $-\left\|d_k\right\|^2$) can be upper bounded by the negative of the suboptimality in the objective value of the next iteration (i.e., $-F_{k+1}$) and additional history dependent terms that arise due to the incremental nature of the PIAG algorithm.

LEMMA 3.5. *Suppose that Assumptions 1 and 2 hold. Then, for any* $0 < \eta \leq \frac{1}{L(K+1)}$, *the PIAG algorithm in* (2.1) *yields the following guarantee*

$$-\left\|d_k\right\|^2 \leq -\frac{\mu}{4} F_{k+1} + \eta L \sum_{j=(k-K)_+}^{k-1} \left\|d_j\right\|^2 .$$

Proof. In order to prove this lemma, we use Lemma 3.4, which can be rewritten as follows

$$-\left\|d_k\right\| \leq -\frac{\mu}{2} \left\|x_k - x^*\right\| + \eta L \sum_{j=(k-K)_+}^{k-1} \left\|d_j\right\| .$$

Then, we can upper bound $-\left\|d_k\right\|^2$ as

$$\begin{aligned} -\left\|d_k\right\|^2 &\leq -\frac{\mu}{2} \left\|d_k\right\| \left\|x_k - x^*\right\| + \eta L \sum_{j=(k-K)_+}^{k-1} \left\|d_k\right\| \left\|d_j\right\| \\ &\leq -\frac{\mu}{2} \langle d_k, x^* - x_k \rangle + \eta \frac{KL}{2} \left\|d_k\right\|^2 + \eta \frac{L}{2} \sum_{j=(k-K)_+}^{k-1} \left\|d_j\right\|^2 , \end{aligned} \tag{3.9}$$

where the last line follows by the Cauchy-Schwarz inequality and the arithmetic-geometric mean inequality. We can upper bound the inner product term in (3.9) as

$$\begin{aligned} -\langle d_k, x^* - x_k \rangle &= \langle g_k + h_{k+1}, x^* - x_k \rangle \\ &= \langle \nabla f(x_k), x^* - x_k \rangle + \langle h_{k+1}, x^* - x_k \rangle + \langle g_k - \nabla f(x_k), x^* - x_k \rangle \\ &\leq f(x^*) - f(x_k) + \langle h_{k+1}, x^* - x_{k+1} \rangle + \eta \langle h_{k+1}, d_k \rangle + \langle g_k - \nabla f(x_k), x^* - x_k \rangle \\ &\leq f(x^*) - f(x_k) + r(x^*) - r(x_{k+1}) + \eta \langle h_{k+1}, d_k \rangle + \left\|g_k - \nabla f(x_k)\right\| \left\|x^* - x_k\right\| , \end{aligned} \tag{3.10}$$

where the first inequality follows from the convexity of f and the second inequality follows from the convexity of r and the triangle inequality. The inner product term in (3.10) can be upper bounded as

$$\begin{aligned} \eta \langle h_{k+1}, d_k \rangle &= -\eta \left\|d_k\right\|^2 - \langle g_k, \eta d_k \rangle \\ &= -\eta \left\|d_k\right\|^2 + \langle \nabla f(x_k), -\eta d_k \rangle + \langle g_k - \nabla f(x_k), -\eta d_k \rangle \\ &\leq -\eta \left\|d_k\right\|^2 + \langle \nabla f(x_k), x_k - x_{k+1} \rangle + \eta \left\|d_k\right\| \left\|g_k - \nabla f(x_k)\right\| \\ &\leq -\eta \left\|d_k\right\|^2 + f(x_k) - f(x_{k+1}) + \eta^2 \frac{L}{2} \left\|d_k\right\|^2 + \eta \left\|d_k\right\| \left\|g_k - \nabla f(x_k)\right\| , \end{aligned} \tag{3.11}$$

where the first inequality follows by the triangle inequality and the second inequality follows from the L-smoothness of f. Putting (3.11) back in (3.10), we obtain

$$-\langle d_k, x^* - x_k \rangle \leq -F_{k+1} + \eta \left(\eta \frac{L}{2} - 1 \right) \left\|d_k\right\|^2 + \left\|g_k - \nabla f(x_k)\right\| \left(\left\|x^* - x_k\right\| + \eta \left\|d_k\right\| \right) . \tag{3.12}$$

The final term in (3.12) can be upper bounded as follows

$$||g_k - \nabla f(x_k)|| \left(||x^* - x_k|| + \eta \, ||d_k||\right) \leq \eta L \left(\sum_{j=(k-K)_+}^{k-1} ||d_j|| \right) \left(||x^* - x_k|| + \eta \, ||d_k||\right)$$
$$\leq \eta L \left(\sum_{j=(k-K)_+}^{k-1} ||d_j|| \right) \left[\left(\eta + \frac{2}{\mu} \right) ||d_k|| + 2\eta Q \sum_{j=(k-K)_+}^{k-1} ||d_j|| \right],$$

where the first line follows by equation (3.3) and the last line follows by Lemma 3.4. Using arithmetic-geometric mean inequality in the above inequality, we obtain

$$||g_k - \nabla f(x_k)|| \left(||x^* - x_k|| + \eta \, ||d_k||\right) \leq \eta \frac{KL}{2} \left(\eta + \frac{2}{\mu} \right) ||d_k|| + \eta \left[\eta \frac{L}{2} + Q + 2\eta KQL \right] \sum_{j=(k-K)_+}^{k-1} ||d_j|| \,. \tag{3.13}$$

Putting (3.13) back in (3.12) yields

$$-\langle d_k, x^* - x_k \rangle \leq -F_{k+1} + \eta \left(\eta \frac{L}{2} - 1 + \frac{KL}{2} \left(\eta + \frac{2}{\mu} \right) \right) ||d_k||^2 + \eta \left[\eta \frac{L}{2} + Q + 2\eta KQL \right] \sum_{j=(k-K)_+}^{k-1} ||d_j||$$
$$= -F_{k+1} + \eta \left(\eta \frac{(K+1)L}{2} - 1 + KQ \right) ||d_k||^2 + \eta \left[\eta \frac{L}{2} + Q + 2\eta KQL \right] \sum_{j=(k-K)_+}^{k-1} ||d_j||$$
$$\leq -F_{k+1} + \eta \left(KQ - \frac{1}{2} \right) ||d_k||^2 + \eta \left(\eta \frac{L}{2} + Q + 2\eta KQL \right) \sum_{j=(k-K)_+}^{k-1} ||d_j|| \,, \tag{3.14}$$

where the last line follows since $\eta \leq \frac{1}{L(K+1)}$. Finally, using (3.14) in our original inequality in (3.9), we obtain

$$-||d_k||^2 \leq -\frac{\mu}{2} F_{k+1} + \eta \left(\frac{KL}{2} - \frac{\mu}{4} + \frac{KL}{2} \right) ||d_k||^2 + \eta \left(\eta \frac{\mu L}{4} + \frac{L}{2} + \eta KL^2 + \frac{L}{2} \right) \sum_{j=(k-K)_+}^{k-1} ||d_j||^2 \,,$$
$$\leq -\frac{\mu}{2} F_{k+1} + \eta KL \, ||d_k||^2 + \eta L \left(\frac{\mu}{4} + \eta KL + 1 \right) \sum_{j=(k-K)_+}^{k-1} ||d_j||^2 \,,$$
$$\leq -\frac{\mu}{2} F_{k+1} + \eta KL \, ||d_k||^2 + \eta L \left(\eta (K+1) L + 1 \right) \sum_{j=(k-K)_+}^{k-1} ||d_j||^2 \,,$$
$$\leq -\frac{\mu}{2} F_{k+1} + ||d_k||^2 + 2\eta L \sum_{j=(k-K)_+}^{k-1} ||d_j||^2 \,, \tag{3.15}$$

where the second inequality follows since $\mu \geq 0$, the third inequality follows since $\frac{\mu}{4} \leq L$, and the last inequality follows since $\eta \leq \frac{1}{L(K+1)}$. Rearranging the terms in (3.15), we obtain

$$-||d_k||^2 \leq -\frac{\mu}{4} F_{k+1} + \eta L \sum_{j=(k-K)_+}^{k-1} ||d_j||^2 \,, \tag{3.16}$$

which completes the proof of Lemma 3.5. □

In the following theorem, we derive a recursive inequality to upper bound the suboptimality at iteration $k+1$ in terms of the suboptimality in the previous $aK+1$ iterations (where a is a positive integer that measures the length of history considered) and an additive remainder term. We will later show in Corollary 3.7 that this remainder term can also be upper bounded in terms of the suboptimality observed in previous iterations.

THEOREM 3.6. *Suppose that Assumptions 1 and 2 hold. Then, the PIAG algorithm with step size* $0 < \eta \leq \frac{1}{L(K+1)}$ *yields the following recursion*

$$\left(1+\eta\frac{\mu}{8}\right) F_{k+1} \leq \left(1-\sum_{i=1}^{a-1}\epsilon_i\right) F_k + \sum_{i=1}^{a-1} \epsilon_i \, F_{k-iK} + \eta^2 \frac{KL}{2} \, \epsilon_{a-1} \sum_{j=k-aK}^{k-1} ||d_j||^2 \,, \tag{3.17}$$

for any $k \geq aK+1$, *where* $a \geq 2$ *is an arbitrary constant and* $\epsilon_i \triangleq 2\eta L \, (\eta K L)^{i-1}$.

Proof. We use induction on the constant a to prove this theorem. For $a=2$, the recursion can be obtained as follows. Using Lemma 3.5 in Lemma 3.3 and rearranging terms, we get

$$\left(1+\eta\frac{\mu}{8}\right) F_{k+1} \leq F_k + \eta^2 L \sum_{j=k-K}^{k-1} ||d_j||^2 \,. \tag{3.18}$$

Rearranging terms in Lemma 3.3, we obtain

$$||d_j||^2 \leq \frac{2}{\eta} \left(F_j - F_{j+1}\right) + \eta L \sum_{i=j-K}^{j-1} ||d_i||^2 \,. \tag{3.19}$$

Putting (3.19) back in (3.18), we get

$$\begin{aligned}
\left(1+\eta\frac{\mu}{8}\right) F_{k+1} &\leq F_k + \eta^2 L \sum_{j=k-K}^{k-1} \left(\frac{2}{\eta} \left(F_j - F_{j+1}\right) + \eta L \sum_{i=j-K}^{j-1} ||d_i||^2 \right) \\
&\leq F_k + 2\eta L \left(F_{k-K} - F_k\right) + \eta^3 L^2 \sum_{j=k-K}^{k-1} \sum_{i=j-K}^{j-1} ||d_i||^2 \\
&\leq F_k + 2\eta L \left(F_{k-K} - F_k\right) + \eta^3 K L^2 \sum_{j=k-2K}^{k-1} ||d_i||^2
\end{aligned} \tag{3.20}$$

Since $\epsilon_1 = 2\eta L$, then (3.20) can be rewritten as follows

$$\left(1+\eta\frac{\mu}{8}\right) F_{k+1} \leq (1-\epsilon_1) F_k + \epsilon_1 F_{(k-K)_+} + \eta^2 \frac{KL}{2} \, \epsilon_1 \sum_{j=k-2K}^{k-1} ||d_j||^2 \,, \tag{3.21}$$

showing (3.17) for $a=2$. As a part of the induction procedure, we then assume that (3.17) holds for some arbitrary $a \geq 2$, which amounts to

$$\left(1+\eta\frac{\mu}{8}\right) F_{k+1} \leq \left(1-\sum_{i=1}^{a-1}\epsilon_i\right) F_k + \sum_{i=1}^{a-1} \epsilon_i \, F_{k-iK} + \eta^2 \frac{KL}{2} \, \epsilon_{a-1} \sum_{j=k-aK}^{k-1} ||d_j||^2 \,.$$

Using (3.19) in the above inequality, we obtain

$$\begin{aligned}\left(1+\eta\frac{\mu}{8}\right)F_{k+1} &\leq \left(1-\sum_{i=1}^{a-1}\epsilon_i\right)F_k+\sum_{i=1}^{a-1}\epsilon_i\,F_{k-iK}+\eta^2\frac{KL}{2}\,\epsilon_{a-1}\\ &\qquad\times\sum_{j=k-aK}^{k-1}\left(\frac{2}{\eta}\left(F_j-F_{j+1}\right)+\eta L\sum_{i=j-K}^{j-1}||d_i||^2\right)\\ &=\left(1-\sum_{i=1}^{a-1}\epsilon_i\right)F_k+\sum_{i=1}^{a-1}\epsilon_i\,F_{k-iK}+\epsilon_a\left(F_{(k-aK)_+}-F_k\right)+\eta^2\frac{L}{2}\,\epsilon_a\sum_{j=k-aK}^{k-1}\sum_{i=j-K}^{j-1}||d_j||^2\\ &\leq\left(1-\sum_{i=1}^{a}\epsilon_i\right)F_k+\sum_{i=1}^{a}\epsilon_i\,F_{k-iK}+\eta^2\frac{KL}{2}\,\epsilon_a\sum_{j=k-(a+1)K}^{k-1}||d_j||^2.\end{aligned}$$

Therefore, (3.17) holds for $a+1$ as well, which concludes the proof of Theorem 3.6. □

COROLLARY 3.7. *Suppose that Assumptions 1 and 2 hold. Then, the PIAG algorithm with step size* $0<\eta\leq\frac{1}{L(K+1)}$ *yields the following recursion*

$$\left(1+\eta\frac{\mu}{8}\right)F_{k+1}\leq\left(1-\sum_{i=1}^{a-1}\epsilon_i\right)F_k+\sum_{i=1}^{a-1}\epsilon_i\,F_{k-iK}+4KQ\,\epsilon_{a-1}\sum_{j=k-aK}^{k}F_j,\tag{3.22}$$

for any integer $a\geq 2$ *and* $k\geq aK+1$, *where* $\epsilon_i\triangleq 2\eta L\left(\eta KL\right)^{i-1}$.

Proof. Using Assumption 3.2, we obtain

$$\begin{aligned}\eta^2\,||d_j||^2&=||x_{j+1}-x_j||^2\\ &\leq\left(||x_{j+1}-x^*||+||x_j-x^*||\right)^2\\ &\leq 2\left(||x_{j+1}-x^*||^2+||x_j-x^*||^2\right)\\ &\leq\frac{4}{\mu}\left(F_{j+1}+F_j\right),\end{aligned}\tag{3.23}$$

where the first inequality follows from the triangle inequality, the second inequality follows from the arithmetic-geometric mean inequality, and the last inequality follows from the μ-strong convexity of the cost function. Using (3.23) in (3.17) leads to

$$\begin{aligned}\left(1+\eta\frac{8}{81Q^2L}\right)F_{k+1}&\leq\left(1-\sum_{i=1}^{a-1}\epsilon_i\right)F_k+\sum_{i=1}^{a-1}\epsilon_i\,F_{k-iK}+2K\frac{L}{\mu}\,\epsilon_{a-1}\sum_{j=k-aK}^{k-1}\left(F_{j+1}+F_j\right)\\ &=\left(1-\sum_{i=1}^{a-1}\epsilon_i\right)F_k+\sum_{i=1}^{a-1}\epsilon_i\,F_{k-iK}+4KQ\,\epsilon_{a-1}\sum_{j=k-aK}^{k}F_j,\end{aligned}$$

which completes the proof. □

Before presenting the main result of the paper, we first introduce the following lemma, which was presented in [10, Lemma 3] in a slightly different form.

LEMMA 3.8. *Let* $\{Z_k\}$ *be a sequence of non-negative real numbers satisfying*

$$Z_{k+1}\leq p\,Z_k+\sum_{j=k-A}^{k}q_j Z_j,$$

for any $k \geq 0$ for some non-negative constants p, q_j and A. If $r \triangleq p + \sum_{j=k-A}^{k} q_j < 1$ holds, then

$$Z_k \leq r^{\frac{k+1}{A+1}-1} \left(\max_{0 \leq j \leq A} Z_j \right), \tag{3.24}$$

for any $k \geq A+1$.

We next present the main theorem of the paper, which characterizes the linear convergence rate of the PIAG algorithm.

THEOREM 3.9. *Suppose that Assumptions 1 and 2 hold. Then, for any integer $a \geq 3$, the PIAG algorithm with step size $0 < \eta \leq \frac{1}{3L(K+1)} \overline{\eta}_a$ where*

$$\overline{\eta}_a \triangleq \left(\frac{1}{144(aK+1)KQ^2} \right)^{\frac{1}{a-2}},$$

is linearly convergent satisfying

$$F_k \leq \left(1 - \eta \frac{\mu}{18}\right)^{\frac{k+1}{aK+1}-1} \left(\max_{0 \leq j \leq aK} F_j \right), \tag{3.25}$$

for any $k \geq aK+1$.

This theorem implies that the exact linear convergence rate depends on the parameter a, which measures the length of the history considered in Theorem 3.6. We will show that increasing a will allow us to establish a linear convergence rate with better condition number Q and aggregation history K dependence. In particular, as $a \to \infty$, we have $\overline{\eta}_a \to 1$ and we can pick a step size of $\eta = \frac{1}{3L(K+1)}$, which yields a linear convergence rate of $1 - \mathcal{O}(1/(QK^2))$. However, this rate in (3.25) is only achievable after the first $aK+1$ iterations. Therefore, picking larger step sizes yields a faster convergence rate but at the expense of a larger number of iterations to achieve that rate. We address this tradeoff in Corollary 3.10 by showing the iteration complexity of the PIAG algorithm to achieve an ϵ-optimal solution.

Proof. We will apply Lemma 3.8 to Corollary 3.7 with $Z_k = F_k$, for which we require

(i) $0 < \eta \leq \frac{1}{L(K+1)}$,

(ii) $\epsilon_i \geq 0$ for all $i = 1, \ldots, a-1$,

(iii) $1 - \sum_{i=1}^{a-1} \epsilon_i \geq 0$, and

(iv) $1 + \eta \frac{\mu}{8} > 1 + 4KQ(aK+1)\epsilon_{a-1}$.

The first and second conditions $(i) - (ii)$ are trivially satisfied as $0 < \eta \leq \frac{1}{3L(K+1)} \overline{\eta}_a$ with $0 < \overline{\eta}_a < 1$, for any $a \in \mathbb{R}$. In order to show that the third condition (iii) holds, we use the definition of ϵ_i, which implies

$$\sum_{i=1}^{a-1} \epsilon_i = 2\eta L \sum_{i=1}^{a-1} (\eta KL)^{i-1} < 2\eta L \sum_{i=0}^{\infty} (\eta KL)^i = 2\eta L \left(\frac{1}{1 - \eta KL} \right), \tag{3.26}$$

where the last equality follows since $\eta KL \leq \frac{K}{3(K+1)}\overline{\eta}_a < \frac{1}{3}$ as $\eta \leq \frac{1}{3L(K+1)}\overline{\eta}_a$ and $\overline{\eta}_a < 1$. Then, we can upper bound the right hand side of (3.26) as follows

$$\sum_{i=1}^{a-1} \epsilon_i < 2\eta L \left(\frac{1}{1-\frac{1}{3}}\right) = 3\eta L \leq 1, \tag{3.27}$$

where the inequalities follow since $\eta \leq \frac{1}{3L(K+1)}\overline{\eta}_a$ with $K \geq 0$ and $\overline{\eta}_a < 1$. The inequality (3.27) implies $1 - \sum_{i=1}^{a-1} \epsilon_i > 0$, therefore the third condition (iii) holds as well. Using the definition of ϵ_{a-1}, the fourth condition (iv) can be written as follows

$$1 + \eta\frac{\mu}{8} > 1 + 8(aK+1)Q\left(\eta KL\right)^{a-1}, \tag{3.28}$$

which can be equivalently expressed as

$$\eta < \frac{1}{KL}\left(\frac{1}{64(aK+1)KQ^2}\right)^{\frac{1}{a-2}}, \tag{3.29}$$

for any $a \geq 3$ and $K > 0$. We can trivially observe that for any $0 < \eta \leq \frac{1}{3L(K+1)}\overline{\eta}_a$, the above inequality is satisfied, hence the fourth condition (iv) holds as well.

Therefore, we can apply Lemma 3.8 to Corollary 3.7. This yields that for any step size $0 < \eta \leq \frac{1}{3L(K+1)}\overline{\eta}_a$ with $a \geq 3$, we obtain a global linear convergence

$$F_k \leq \kappa^{\frac{k+1}{aK+1}-1} \max_{0 \leq j \leq aK} F_j, \tag{3.30}$$

where

$$\begin{aligned}\kappa &= \frac{1 + 8(aK+1)Q\left(\eta KL\right)^{a-1}}{1+\eta\frac{\mu}{8}} \\ &= 1 - \eta\frac{\frac{\mu}{8} - 8(aK+1)KQL\left(\eta KL\right)^{a-2}}{1+\eta\frac{\mu}{8}}.\end{aligned}$$

As $1 < 1 + \eta\frac{\mu}{8} \leq \frac{9}{8}$, we then have

$$\kappa \leq 1 - \eta\frac{\mu}{9} + 8\eta(aK+1)KQL\left(\eta KL\right)^{a-2}. \tag{3.31}$$

Noting that $\eta \leq \frac{1}{3L(K+1)}\overline{\eta}_a < \frac{1}{LK}\overline{\eta}_a$, we obtain $\left(\eta KL\right)^{a-2} < \left(\overline{\eta}_a\right)^{a-2} = \frac{1}{144(aK+1)KQ^2}$. Using this inequality in (3.31), we obtain

$$\begin{aligned}\kappa &< 1 - \eta\frac{\mu}{9} + 8\eta(aK+1)KQL\frac{1}{144(aK+1)KQ^2} \\ &\leq 1 - \eta\frac{\mu}{9} + \eta\frac{\mu}{18} \\ &\leq 1 - \eta\frac{\mu}{18},\end{aligned} \tag{3.32}$$

which concludes the proof of Theorem 3.9. □

We next introduce the following corollary, which highlights the main result of the paper. This corollary indicates that using a step size of $\mathcal{O}(1/(KL\log(QK)))$, the PIAG algorithm is guaranteed to return an ϵ-optimal solution after $\mathcal{O}(QK^2\log^2(QK)\log(1/\epsilon))$ iterations, or equivalently after $\widetilde{\mathcal{O}}(QK^2\log(1/\epsilon))$ iterations, where the tilde is used to hide the logarithmic terms in Q and K.

COROLLARY 3.10. *Suppose that Assumptions 1 and 2 hold. Then, the PIAG algorithm in* (2.1) *with step size* $\eta = \frac{1}{3L(K+1)}\widetilde{\eta}_a$, *where* $\widetilde{\eta}_a \triangleq \frac{1}{a}\left(\frac{1}{12(K+1)Q}\right)^{\frac{2}{a-2}}$ *and* $a = \lceil \log(12(K+1)Q)\rceil + 2$, *is guaranteed to return an* ϵ*-optimal solution after*

$$k \geq Ma^2(K+1)^2Q\log(c/\epsilon) + aK \tag{3.33}$$

iterations, where $M \triangleq 54e^2$ *and* $c = \max_{0\leq j\leq aK} F_j$ *denotes the initial suboptimality in the function values.*

Proof. We begin the proof of this corollary by showing that $\widetilde{\eta}_a \leq \overline{\eta}_a$, i.e., $\eta = \frac{1}{3L(K+1)}\widetilde{\eta}_a$ is a valid step size satisfying (3.25) of Theorem 3.9. Clearly, we have $(aK+1)K = a(K+\frac{1}{a})K \leq a(K+1)^2$ as $a \geq 3$. Therefore, we obtain

$$\overline{\eta}_a = \left(\frac{1}{144(aK+1)KQ^2}\right)^{\frac{1}{a-2}} \geq \left(\frac{1}{a}\right)^{\frac{1}{a-2}}\left(\frac{1}{12(K+1)Q}\right)^{\frac{2}{a-2}}. \tag{3.34}$$

As $a \geq 3$, we also have $\left(\frac{1}{a}\right)^{\frac{1}{a-2}} \geq \frac{1}{a}$, which indicates that $\overline{\eta}_a \geq \widetilde{\eta}_a$. Therefore, we can apply Theorem 3.9 with step size $\eta = \frac{1}{3L(K+1)}\widetilde{\eta}_a$, which results in the following linear convergence

$$F_k \leq c\left(1-\eta\frac{\mu}{18}\right)^{\frac{k+1}{aK+1}-1}, \tag{3.35}$$

for any $k \geq aK+1$. Using the inequality $(1-x)^\gamma \leq 1-\gamma x$ for any $\gamma, x \in [0,1]$ in (3.35) and noting that $a(K+1) \geq aK+1$, we obtain

$$F_k \leq c\left(1-\eta\frac{\mu}{18a(K+1)}\right)^{k-aK}.$$

Taking the logarithm of both sides yields

$$\begin{aligned}\log(F_k) &\leq \log(c) + (k-aK)\log\left(1-\eta\frac{\mu}{18a(K+1)}\right)\\ &\leq \log(c) - (k-aK)\eta\frac{\mu}{18a(K+1)},\end{aligned} \tag{3.36}$$

where the last line follows since $\log(1+x) \leq x$ for any $x > -1$. Therefore, in order to achieve an ϵ-optimal solution, the right-hand side of (3.36) should be upper bounded by $\log(\epsilon)$, which implies

$$\begin{aligned}k &\geq \frac{18a(K+1)}{\eta\mu}\log(c/\epsilon) + aK\\ &= \frac{54a(K+1)^2Q}{\widetilde{\eta}_a}\log(c/\epsilon) + aK,\end{aligned} \tag{3.37}$$

where the equality follows since $\eta = \frac{1}{3L(K+1)}\widetilde{\eta}_a$. Picking $a = \lceil \log(12(K+1)Q) \rceil + 2$, we can lower bound $\widetilde{\eta}_a$ as follows

$$\begin{aligned}\log(\widetilde{\eta}_a) &= \log\left(\frac{1}{a}\right) + \frac{2}{a-2}\log\left(\frac{1}{12(K+1)Q}\right) \\ &= -\log(a) - \frac{2}{\lceil \log(12(K+1)Q) \rceil}\log(12(K+1)Q) \\ &\geq -\log(a) - 2,\end{aligned}$$

which yields $\widetilde{\eta}_a \geq 1/(ae^2)$. Using this result in (3.37), we conclude that the PIAG algorithm is guaranteed to return an ϵ-optimal solution after $k \geq 54e^2a^2(K+1)^2Q\log(c/\epsilon) + aK$ iterations. □

4. Concluding Remarks. In this paper, we studied the PIAG method for additive composite optimization problems of the form (1.1). We showed the first linear convergence rate result for the PIAG method and provided explicit convergence rate estimates that highlight the dependence on the condition number of the problem and the size of the window K over which outdated component gradients are evaluated (under the assumptions that $f(x)$ is strongly convex and each $f_i(x)$ is smooth with Lipschitz gradients). Our results hold for any deterministic order (in processing the component functions) in contrast to the existing work on stochastic variants of our algorithm, which presents convergence results in expectation.

REFERENCES

[1] D. P. Bertsekas. Incremental gradient, subgradient, and proximal methods for convex optimization: a survey. *Optimization for Machine Learning*, 2010:1–38, 2011.

[2] D. P. Bertsekas. Incremental proximal methods for large scale convex optimization. *Mathematical Programming*, 129(2):163–195, 2011.

[3] D. P. Bertsekas. *Convex Optimization Algorithms*. Athena Scientific, 2015.

[4] D. P. Bertsekas. Incremental aggregated proximal and augmented lagrangian algorithms. *arXiv preprint arXiv:1509.09257*, 2015.

[5] 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.

[6] S. Bubeck. Theory of Convex Optimization for Machine Learning. *arXiv preprint arXiv:1405.4980*, May 2014.

[7] A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems 27*, pages 1646–1654, 2014.

[8] A. J. Defazio, T. S. Caetano, and J. Domke. Finito: A faster, permutable incremental gradient method for big data problems. In *Advances in Neural Information Processing Systems 26*, pages 315–323, 2013.

[9] D. Drusvyatskiy and A. S. Lewis. Error bounds, quadratic growth, and linear convergence of proximal methods. *CoRR*, abs/1602.06661, 2016.

[10] H. R. Feyzmahdavian, A. Aytekin, and M. Johansson. A delayed proximal gradient method with linear convergence rate. In *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2014.

[11] F. Guo, C. Wen, J. Mao, and Y. D. Song. Distributed economic dispatch for smart grids with random wind power. *IEEE Transactions on Smart Grid*, 7(3):1572–1583, May 2016.

[12] M. Gurbuzbalaban, A. Ozdaglar, and P. Parrilo. On the convergence rate of incremental aggregated gradient algorithms. *arXiv preprint arXiv:1506.02081*, 2015.

[13] M. Gurbuzbalaban, A. Ozdaglar, and P. Parrilo. Why random reshuffling beats stochastic gradient descent. *arXiv preprint arXiv:1510.08560*, 2015.

[14] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems 26*, pages 315–323, 2013.

[15] H. Karimi and M. Schmidt. Linear convergence of proximal-gradient methods under the Polyak-Lojasiewicz condition. In *NIPS Workshop on Optimization*. 2015.

[16] H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems 28*, pages 3384–3392. 2015.

[17] J. Mairal. Incremental majorization-minimization optimization with application to large-scale machine learning. *SIAM Journal on Optimization*, 25(2):829–855, 2015.

[18] A. Nedic and D. P. Bertsekas. Incremental subgradient methods for nondifferentiable optimization. *SIAM Journal on Optimization*, 12(1):109–138, 2001.

[19] A. Nedic, D. P. Bertsekas, and V. S. Borkar. Distributed asynchronous incremental subgradient methods. *Studies in Computational Mathematics*, 8:381–407, 2001.

[20] Y. Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical Programming*, 120(1):221–259, 2009.

[21] F. Niu, B. Recht, C. Re, and S. Wright. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *Advances in Neural Information Processing Systems 24*, pages 693–701, 2011.

[22] H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

[23] N. L. Roux, M. Schmidt, and F. R. Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in Neural Information Processing Systems 25*, pages 2663–2671, 2012.

[24] M. V. Solodov. Incremental gradient algorithms with stepsizes bounded away from zero. *Computational Optimization and Applications*, 11(1):23–35, 1998.

[25] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1-2):387–423, 2009.

[26] P. Tseng and S. Yun. Incrementally updated gradient methods for constrained and regularized optimization. *Journal of Optimization Theory and Applications*, 160(3):832–853, 2014.

[27] J. Tsitsiklis, D. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31(9):803–812, Sep 1986.