2021 American Control Conference (ACC)
New Orleans, USA, May 25-28, 2021

# An Incremental Gradient Method for Large-scale Distributed Nonlinearly Constrained Optimization

Harshal D. Kaushik[1] and Farzad Yousefian[2]

***Abstract*— Motivated by applications arising from sensor networks and machine learning, we consider the problem of minimizing a finite sum of nondifferentiable convex functions where each component function is associated with an agent and a hard-to-project constraint set. Among well-known avenues to address finite sum problems is the class of incremental gradient (IG) methods where a single component function is selected at each iteration in a cyclic or randomized manner. When the problem is constrained, the existing IG schemes (including projected IG, proximal IAG, and SAGA) require a projection step onto the feasible set at each iteration. Consequently, the performance of these schemes is afflicted with costly projections when the problem includes: (1) nonlinear constraints, or (2) a large number of linear constraints. Our focus in this paper lies in addressing both of these challenges. We develop an algorithm called averaged iteratively regularized incremental gradient (aIR-IG) that does not involve any hard-to-project computation. Under mild assumptions, we derive non-asymptotic rates of convergence for both suboptimality and infeasibility metrics. Numerically, we show that the proposed scheme outperforms the standard projected IG methods on distributed soft-margin support vector machine problems.**

## I. Introduction

We consider a finite sum minimization subject to nonlinear inequality and linear equality functional constraints as follows:

$$\begin{aligned} \underset{x\in\mathbb{R}^n}{\text{minimize}} \quad & f(x) \triangleq \textstyle\sum_{i=1}^m f_i(x) && (P)\\ \text{subject to} \quad & h_i(x) \le 0 && \text{for all } i \in \{1,\ldots,m\},\\ & A_i x = b_i && \text{for all } i \in \{1,\ldots,m\},\\ & x^{(j)} \ge 0 && \text{for all } j \in J,\\ & x \in X, \end{aligned}$$

where the component functions $f_i : \mathbb{R}^n \to \mathbb{R}$ and $h_i : \mathbb{R}^n \to \mathbb{R}$ are nonsmooth convex, $A_i \in \mathbb{R}^{d_i \times n}$, and $b_i \in \mathbb{R}^{d_i}$, for all $i \in \{1,\ldots,m\}$. Also, $X \subseteq \mathbb{R}^n$ is an easy-to-project convex set and $J \subseteq \{1,\ldots,n\}$. The information about $f_i$, $h_i$, $A_i$, and $b_i$ is only known by agent $i$, while the sets $X$ and $J$ are known by all the agents. Parameters $n$, $m$, and $p \triangleq \sum_{i=1}^m d_i$ are possibly large. Problem $(P)$ arises in a breadth of applications including expected loss minimization in statistical learning [1] where $f_i$ is associated with a data block, as well as distributed optimization in wireless sensor networks where $f_i$ represents the local performance measure of the $i^{\text{th}}$ agent [2]. One of the popular methods in addressing finite sum problems, in particular, in the unconstrained regime, is the class of incremental gradient (IG) methods where utilizing the additive structure of the problem, the algorithm cycles through the data blocks and updates the local estimates of the optimal solution in a sequential manner [3]. While the first variants of IG schemes find their roots in addressing neural networks as early as in the '80s [4], the complexity analysis of these schemes has been a trending research topic in the fields of control and machine learning in the past two decades. In addressing constrained problems with easy-to-project constraint sets, the projected incremental gradient (P-IG) method and its subgradient variant were developed [5]. In the smooth case, it is described as follows: given an initial point $x_{0,1} \in X$, where $X \subseteq \mathbb{R}^n$ denotes the constraint set, for each $k \ge 1$, consider the following update rule:

$$\begin{aligned} x_{k,i+1} &:= \mathcal{P}_X\left(x_{k,i} - \gamma_k \nabla f_i\left(x_{k,i}\right)\right) \quad \text{for all } i = 1,\ldots,m,\\ x_{k+1,1} &:= x_{k,m+1} \qquad\qquad \text{for all } k \ge 0, \end{aligned}$$

where $\mathcal{P}$ denotes the Euclidean projection operator and is defined as $\mathcal{P}_X(z) \triangleq \operatorname{argmin}_{x\in X}\|x - z\|_2$ and $\gamma_k > 0$ is the stepsize parameter. Recently, under the assumption of strong convexity and twice continuous differentiability of the objective function, the standard IG method was proved to converge with the rate $\mathcal{O}(1/k)$ in the unconstrained case [6]. This is an improvement to the previously known rate of $\mathcal{O}(1/\sqrt{k})$ for the merely convex case. Accelerated variants of IG schemes with provable convergence speeds were also developed, including the incremental aggregated gradient method (IAG) [7], [8], SAG [1], and SAGA [9]. While addressing the merely convex case, SAGA using averaging achieves a sublinear convergence rate, assuming strong convexity and smoothness, this is improved for non-averaging variants of SAGA and IAG to a linear rate.

**Existing gap.** Despite the faster rates of convergence in comparison with the standard IG method, the aforementioned methods require an excessive memory of $\mathcal{O}\left(mn\right)$ which limits their applications in the large-scale settings. Another existing challenge in the implementation of these schemes lies in addressing the hard-to-project constraints. Contending with the presence of constraints, projected (and more generally proximal) variants of the aforementioned IG schemes have been developed. However, the performance of these schemes is afflicted with costly projections when the problem includes: (1) nonlinear constraints, or (2) a large number of linear constraints. In the area of distributed optimization over networks, addressing constraints has been done to a limited

[1]Doctoral Candidate in the School of Industrial Engineering & Management, Oklahoma State University, Stillwater, OK 74074, USA harshal.kaushik@okstate.edu

[2]Assistant Professor in the School of Industrial Engineering & Management, Oklahoma State University, Stillwater, OK 74074, USA farzad.yousefian@okstate.edu

Farzad Yousefian gratefully acknowledges the support of the NSF through CAREER grant ECCS−1944500.

extent through employing duality theory, projection, or penalty methods (see [10], [11], [12], [13], [14]). We also note that a celebrated variant of the dual based schemes is the alternating direction method of multipliers (ADMM) (e.g., see [15], [16], [17], [18], [19]). Despite the recent advancements in this area, most ADMM methods cannot address inequality constraints with a separable structure as in $(P)$. Also, ADMM schemes often work under the premise that the communication graph is undirected. Indeed, despite the wide-spread application of the theory of duality and Lagrangian relaxation in addressing constrained problems in centralized regimes, there have been a limited work in the area of distributed optimization that can cope with hard-to-project constraints (see [20], [11], [12] and the references therein). Nevertheless, the problem formulation $(P)$ is not addressed in the aforementioned articles. Recently, primal-dual algorithms are proposed for finite sum convex optimization problems with conic constraints [11], [21]. A recent work [22] introduced primal-dual incremental gradient method for nonsmooth convex optimization problems. Moreover, iterative regularization (IR) has been employed as a new constraint-relaxation strategy in regimes where addressing the constraints are challenging (e.g., see [23], [24], [25], [26]). Our work in this paper has been motivated by the recent success of the IR approach. To this end, our goal lies in employing the IR approach to develop an IG algorithm that can address formulation $(P)$ without requiring any hard-to-project computation.

**Main contributions.** This work enables IG methods to address large-scale nonlinearly constrained optimization problems efficiently. Our main contributions are as follows:

(i) We develop an algorithm called averaged iteratively regularized incremental gradient (aIR-IG) where at each iteration, a suitably defined stepsize and a regularization parameter are updated. Importantly, the proposed algorithm circumvents the hard-to-project computation. It is an incremental gradient scheme in the sense that at each iteration, only the local information of $f_i$, $h_i$, $A_i$, and $b_i$ is used by agent $i$ and agents communicate through a cycle graph.

(ii) Under mild assumptions, we derive non-asymptotic rates of convergence for both suboptimality and infeasibility metrics. This is done through a careful choice of the stepsize and the regularization parameter that are updated iteratively. Importantly, the rate analysis in this paper is done under much weaker assumptions for functions $f_i$ in comparison with standard IG methods.

**Outline.** The remainder of the paper is organized as follows. Section II includes the algorithm outline for addressing problem $(P)$. We also provide the main assumptions and the required preliminaries. Section III includes the convergence analysis of the proposed scheme. Section IV contains the numerical implementation where we compare the proposed algorithm with the standard IG methods.

**Notation and preliminary definitions.** A function $f:\mathbb{R}^n\to\mathbb{R}$ is said to be in the class $C^{k,r}_{\mu,L}$ if $f$ is $\mu$-strongly convex in $\mathbb{R}^n$, $k$ times continuously differentiable, and its $r^{\text{th}}$ derivative is Lipschitz continuous with constant $L$. A nondifferentiable $\mu$-strongly convex function $f:\mathbb{R}^n\to\mathbb{R}$ is in the class $C^0_\mu$. For any vector $x\in\mathbb{R}^n$, we use $\|x\|$ to denote the $\ell_2$-norm and $x^{(j)}$ is used for denoting the j$^{\text{th}}$ component of $x$. For problem $(P)$, we define matrix $A\in\mathbb{R}^{p\times n}$ as $A\triangleq\left(A_1^T,A_2^T,\dots,A_m^T\right)^T$ and vector $b\in\mathbb{R}^{p\times 1}$ as $b\triangleq\left(b_1^T,b_2^T,\dots,b_m^T\right)^T$. To produce a diagonal matrix in $\mathbb{R}^{n\times n}$ from vector $x$, we use the notation $diag(x)$. For a convex function $f:\mathbb{R}^n\to\mathbb{R}$ with the domain $\text{dom}(f)$ and any $x\in\text{dom}(f)$, vector $\tilde\nabla f(x)\in\mathbb{R}^n$ with $f(x)+\tilde\nabla f(x)^T(y-x)\le f(y)$ for all $y\in\text{dom}(f)$, is called a subgradient of $f$ at $x$. We let $\partial f(x)$ denote the subdifferential set of function $f$ at $x$. Euclidean projection of vector $x$ onto a closed convex set $X$ is denoted by $\mathcal{P}_X(x)$. We let $[m]$ abbreviate the set $\{1,\dots,m\}$.

## II. Algorithm Outline

In this section, we first provide the main assumptions on problem $(P)$ and present the outline of the algorithm. Then, we present a few preliminary results to be used in the analysis.

**Assumption 1** (Properties of problem $(P)$)**.** *Suppose:*
*(a) Component function $f_i:\mathbb{R}^n\to\mathbb{R}$ is merely convex and subdifferentiable with bounded subgradients for all $i\in[m]$.*
*(b) Function $h_i:\mathbb{R}^n\to\mathbb{R}$ is convex and subdifferentiable with bounded subgradients for all $i\in[m]$.*
*(c) The set $X$ is compact and convex.*
*(d) The feasible set of problem $(P)$ is nonempty.*

An underlying idea in development of Algorithm 1 is to define a regularized error metric.

**Definition 1.** Consider the following term for measuring infeasibility for an agent $i$:

$$\phi_i(x)\triangleq\tfrac{1}{2}\|A_ix-b_i\|^2+h_i^+(x)+\textstyle\sum_{j\in J}\frac{\max\{-x^{(j)},0\}}{m},$$

where $h_i^+(x)\triangleq\max\{0,h_i(x)\}$ for $i\in[m]$ and all $x\in\mathbb{R}^n$. Further, we define $\phi(x)=\sum_{i=1}^m\phi_i(x)$.

Then, for each agent $i$, we consider a regularized metric defined as $\phi_i(x)+\eta_kf_i(x)$ at iteration $k$. This metric captures both infeasibility and objective component function of the agent. Next, we derive a subgradient to this metric.

Let $\partial h_i^+(x)$ denote the subdifferential set of the function $h_i^+$ at $x$. Consider the vector $\tilde\nabla h_i^+(x)$ defined as $\tilde\nabla h_i^+(x)\triangleq h_i^+(x)\tilde\nabla h_i(x)$ where $\tilde\nabla h_i(x)$ denotes a subgradient of function $h_i$ at $x$. Then, from the definition of subgradient mapping and the definition of $h_i^+(x)$, we have that $\tilde\nabla h_i^+(x)\in\partial h_i^+(x)$. Next, consider the function $\frac{1}{m}\sum_{j\in J}\max\left\{0,-x^{(j)}\right\}$. A subgradient to this function is the vector $\frac{\mathbb{1}^-(x)}{m}$ where $\mathbb{1}^-(x)$ is defined a column vector $\in\mathbb{R}^n$ and the value of any component $i\in\{1,\dots,n\}$ is $-1$ when $x^{(i)}<0$ and $i\in J$, otherwise that component is 0. Let $x_{k,i}$ in $\mathbb{R}^n$ denote the iterate of agent $i$ at iteration $k$. From the above discussion, we can conclude that the subgradient of the regularized error metric for agent $i$, is given as follows:

$$A_i^T\left(A_ix_{k,i}-b_i\right)+\tilde\nabla h_i^+(x_{k,i})+\tfrac{\mathbb{1}^-(x_{k,i})}{m}+\eta_k\tilde\nabla f_i\left(x_{k,i}\right).$$

We are now ready to present the outline of aIR-IG scheme presented by Algorithm 1. At each iteration, agents update their iterates in a cyclic manner by employing

the aforementioned subgradient. Each agent uses its local information including subgradients of functions $f_i$, $h_i$, as well as matrix $A_i$ and vector $b_i$. Here $\gamma_k$ and $\eta_k$ are the stepsize and regularization parameters, respectively. These parameters are updated at each iteration. This, indeed, is important because the convergence and rate analysis mainly depend on the choice of $\gamma_k$ and $\eta_k$. The key research question lies in finding suitable update rules for the two sequences so that we can achieve convergence and rate results. For the rate analysis, we employ averaging which is characterized by stepsize $\gamma_k$ and a scalar $0 \leq r < 1$.

**Algorithm 1** Averaged Iteratively Regularized Incremental Gradient (aIR-IG)

1: **Input**: $x_0 \in \mathbb{R}^n$, $\bar{x}_0 := x_0$, $S_0 := \gamma_0^r$, and $0 \leq r < 1$.
2: **for** $k = 0, 1, \ldots, N-1$ **do**
3: Let $x_{k,1} := x_k$ and select $\gamma_k > 0$, $\eta_k > 0$
4: **for** $i = 1, \ldots, m$ **do**

$$x_{k,i+1} := \mathcal{P}_X\left(x_{k,i} - \gamma_k\left(A_i^T\left(A_i x_{k,i} - b_i\right) + \tilde{\nabla} h_i^+(x_{k,i}) + \frac{\mathbb{1}^-(x_{k,i})}{m} + \eta_k \tilde{\nabla} f_i\left(x_{k,i}\right)\right)\right)$$

5: **end for**
6: Set $x_{k+1} \triangleq x_{k,m+1}$.
7: Update the weighted average iterate as

$$\bar{x}_{k+1} := \frac{S_k \bar{x}_k + \gamma_{k+1}^r x_{k+1}}{S_{k+1}}, \text{ where } S_{k+1} := S_k + \gamma_{k+1}^r.$$

8: **end for**
9: **return**: $\bar{x}_N$.

In the following, we claim the boundedness of the subgradients $\tilde{\nabla}\phi_i(x)$ and $\tilde{\nabla} f_i(x)$ which will be used in the rate analysis in the next section.

**Remark 1.** Under Assumption 1, from compactness of the set $X$, the term $A_i^T\left(A_i x - b_i\right)$ is bounded. Also, from the boundedness of subgradients of function $h_i$ and continuity of the function $h_i$ that is implied from convexity of $h_i$, we can claim that the subgradient $\tilde{\nabla} h_i^+(x) \triangleq h_i^+(x)\tilde{\nabla} h_i(x)$ is bounded on the set $X$. Consequently, we have that $\tilde{\nabla}\phi_i(x) \triangleq A_i^T\left(A_i x - b_i\right) + \tilde{\nabla} h_i^+(x) + \frac{\mathbb{1}^-(x)}{m}$ is a bounded subgradient of $\phi_i$ for all $x \in X$. This implies that there exists a scalar $C > 0$ such that for all $x \in X$, we have:

$$\textstyle\sum_{i=1}^m \tilde{\nabla}\phi_i(x) \leq C \quad \text{and} \quad \tilde{\nabla}\phi_i(x) \leq \frac{C}{m} \text{ for all } i \in [m].$$

**Remark 2.** From Assumption 1, taking into account the subdifferentiability and boundedness of subgradient of function $f_i$, there exists a scalar $C_f > 0$ such that for all $x \in X$,

$$\textstyle\sum_{i=1}^m \left\|\tilde{\nabla} f_i\left(x\right)\right\| \leq C_f \text{ and } \left\|\tilde{\nabla} f_i\left(x\right)\right\| \leq \frac{C_f}{m} \text{ for all } i \in [m].$$

**Remark 3.** Taking into account Assumption 1, from Theorem 3.61 in [27], functions $f_i$ and $\phi_i$ are Lipschitz continuous over set $X$. Therefore for $x, y \in X$, and $i \in [m]$, $|f_i(x) - f_i(y)| \leq \frac{C_f}{m}\|x - y\|$ and $|\phi_i(x) - \phi_i(y)| \leq \frac{C}{m}\|x - y\|$.

Next, we show that the sequence $\bar{x}_k$, employed in Algorithm 1, is a well-defined weighted average.

**Remark 4.** From Algorithm 1, the average of the iterate can be written as $\bar{x}_{k+1} = \sum_{t=0}^k \lambda_{t,k} x_t$, where $\lambda_{t,k} \triangleq \frac{\gamma_t^r}{\sum_{j=0}^k \gamma_j^r}$ for $t \in \{0, \ldots, k\}$ denote the weights. This can be shown using induction on $k \geq 0$.

In this work, the average of the $m^{\text{th}}$ agent's iterate is taken. We believe the rate results also hold for the average iterates of the other agents. This remains a future direction to analyze.

The next result will be employed in the rate analysis.

**Lemma 1** (Lemma 2.14 in [26])**.** *For any scalar $\alpha \in [0, 1)$ and integer $N$ such that $N \geq 2^{\frac{1}{1-\alpha}} - 1$, we have:*

$$\tfrac{(N+1)^{1-\alpha}}{2(1-\alpha)} \leq \textstyle\sum_{k=0}^N (k+1)^{-\alpha} \leq \tfrac{(N+1)^{1-\alpha}}{1-\alpha}.$$

## III. Convergence Analysis

We begin with obtaining an error bound that will be employed later in the construction of bounds on the objective value and infeasibility metrics for Algorithm 1. The proof is presented in an extended version of the paper [28].

**Lemma 2.** *Let the sequence $\{x_k\}$ be generated by Algorithm 1 and $\{\gamma_k\}$ and $\{\eta_k\}$ be nonincreasing positive sequences. Let Assumption 1 hold, $0 \leq r < 1$, and scalars $C, C_f > 0$ be defined as in Remarks 1 and 2, respectively. Then, for any $y \in X$ and $k \geq 0$, we have:*

$$\begin{aligned} 2\gamma_k^r \eta_k \left(f(x_k) - f(y)\right) + 2\gamma_k^r \left(\phi\left(x_k\right) - \phi\left(y\right)\right) \\ \leq \gamma_k^{r-1}\left\|x_k - y\right\|^2 - \gamma_k^{r-1}\left\|x_{k+1} - y\right\|^2 \\ + \left(1 + \tfrac{1}{m}\right)\gamma_k^{r+1}\left(C + \eta_k C_f\right)^2. \end{aligned} \tag{1}$$

Next we construct the error bounds for Algorithm 1 in terms of the sequences $\{\gamma_k\}$ and $\{\eta_k\}$.

**Proposition 1** (Error bounds for Algorithm 1)**.** *Consider problem $(P)$. Let $\bar{x}_N$ be generated by Algorithm 1 after $N$ iterations and $\{\gamma_k\}$ and $\{\eta_k\}$ be nonincreasing and strictly positive sequences. Further, let Assumption 1 hold, scalars $C_f, C > 0$, and parameter $0 \leq r < 1$. Let scalars $M, M_f > 0$ be defined such that we have: $\|x\| \leq M$ and $|f(x)| \leq M_f$ for all $x \in X$. Then for any optimal solution $x^*$ to $(P)$, we have the following:*

$$\begin{aligned} (a)\ f(\bar{x}_N) - f(x^*) \leq \left(\textstyle\sum_{k=0}^N \gamma_k^r\right)^{-1}\left(\tfrac{2M^2\gamma_N^{r-1}}{\eta_N}\right. \\ \left. + \left(1 + \tfrac{1}{m}\right)\tfrac{(C+\eta_0 C_f)^2}{2}\textstyle\sum_{k=0}^N \tfrac{\gamma_k^{r+1}}{\eta_k}\right). \end{aligned}$$

$$\begin{aligned} (b)\ \phi\left(\bar{x}_N\right) \leq \left(\textstyle\sum_{k=0}^N \gamma_k^r\right)^{-1}\left(2M^2\gamma_N^{r-1} + 2M_f\textstyle\sum_{k=0}^N \gamma_k^r\eta_k\right. \\ \left. + \left(1 + \tfrac{1}{m}\right)\tfrac{(C+\eta_0 C_f)^2}{2}\textstyle\sum_{k=0}^N \gamma_k^{r+1}\right). \end{aligned}$$

*Proof.* Consider relation (1) from Lemma 2, for any $y \in X$. Substituting $y$ by $x^*$ and taking into account the feasibility of the vector $x^*$ to problem $(P)$, we obtain:

$$\begin{aligned} 2\gamma_k^r\eta_k\left(f(x_k) - f(x^*)\right) + 2\gamma_k^r\phi\left(x_k\right) \leq \gamma_k^{r-1}\left(\left\|x_k - x^*\right\|^2\right. \\ \left. - \left\|x_{k+1} - x^*\right\|^2\right) + \left(1 + \tfrac{1}{m}\right)\gamma_k^{r+1}\left(C + \eta_k C_f\right)^2. \end{aligned}$$

Taking into account the nonnegativity of $2\gamma_k^r\phi(x_k)$ and dividing both sides by $2\eta_k$, we have:

$$\gamma_k^r\left(f(x_k)-f(x^*)\right) \leq \frac{\gamma_k^{r-1}}{2\eta_k}\left(\|x_k-x^*\|^2-\|x_{k+1}-x^*\|^2\right) + \left(1+\tfrac{1}{m}\right)\frac{\gamma_k^{r+1}(C+\eta_k C_f)^2}{2\eta_k}. \quad (2)$$

Adding and subtracting $\frac{\gamma_{k-1}^{r-1}}{2\eta_{k-1}}\|x_k-x^*\|^2$ in the above,

$$\gamma_k^r\left(f(x_k)-f(x^*)\right) \leq \frac{\gamma_{k-1}^{r-1}}{2\eta_{k-1}}\|x_k-x^*\|^2-\frac{\gamma_k^{r-1}}{2\eta_k}\|x_{k+1}-x^*\|^2 + \underbrace{\left(\frac{\gamma_k^{r-1}}{2\eta_k}-\frac{\gamma_{k-1}^{r-1}}{2\eta_{k-1}}\right)}_{\text{term 1}}\|x_k-x^*\|^2 + \underbrace{\left(1+\tfrac{1}{m}\right)\frac{\gamma_k^{r+1}(C+\eta_k C_f)^2}{2\eta_k}}_{\text{term 2}}.$$

Recalling the definition for scalar $M$, we have:

$$\|x_k-x^*\|^2 \leq 2\|x_k\|^2+2\|x^*\|^2 \leq 4M^2. \quad (3)$$

Taking into account $r<1$ and the nonincreasing property of the sequences $\{\gamma_k\}$ and $\{\eta_k\}$, we have: term $1\geq 0$. Bounding term 2, we have:

$$\gamma_k^r\left(f(x_k)-f(x^*)\right) \leq \frac{\gamma_{k-1}^{r-1}}{2\eta_{k-1}}\|x_k-x^*\|^2 - \frac{\gamma_k^{r-1}}{2\eta_k}\|x_{k+1}-x^*\|^2+\left(\frac{\gamma_k^{r-1}}{2\eta_k}-\frac{\gamma_{k-1}^{r-1}}{2\eta_{k-1}}\right)4M^2 + \left(1+\tfrac{1}{m}\right)\frac{(C+\eta_0 C_f)^2\gamma_k^{r+1}}{2\eta_k}.$$

Next, taking summations over $k=1,\ldots,N$, we obtain:

$$\sum_{k=1}^N\gamma_k^r\left(f(x_k)-f(x^*)\right) \leq \frac{\gamma_0^{r-1}}{2\eta_0}\|x_1-x^*\|^2 - \frac{\gamma_N^{r-1}}{2\eta_N}\|x_{N+1}-x^*\|^2+\left(\frac{\gamma_N^{r-1}}{2\eta_N}-\frac{\gamma_0^{r-1}}{2\eta_0}\right)4M^2 + \left(1+\tfrac{1}{m}\right)\frac{(C+\eta_0 C_f)^2}{2}\sum_{k=1}^N\frac{\gamma_k^{r+1}}{\eta_k}. \quad (4)$$

Rewriting equation (2) for $k=0$, we have:

$$\gamma_0^r\left(f(x_0)-f(x^*)\right) \leq \frac{\gamma_0^{r-1}}{2\eta_0}\left(\|x_0-x^*\|^2-\|x_1-x^*\|^2\right) + \left(1+\tfrac{1}{m}\right)\frac{\gamma_0^{r+1}\left(C^2+\eta_0 C_f\right)^2}{2\eta_0}.$$

Adding the preceding relation with (4), we obtain:

$$\sum_{k=0}^N\gamma_k^r\left(f(x_k)-f(x^*)\right) \leq 2M^2\left(\frac{\gamma_N^{r-1}}{\eta_N}-\frac{\gamma_0^{r-1}}{\eta_0}\right) - \frac{\gamma_N^{r-1}}{2\eta_N}\|x_{N+1}-x^*\|^2+\frac{\gamma_0^{r-1}\|x_0-x^*\|^2}{2\eta_0} + \left(1+\tfrac{1}{m}\right)\frac{(C+\eta_0 C_f)^2}{2}\sum_{k=0}^N\frac{\gamma_k^{r+1}}{\eta_k}.$$

Further from (3), and neglecting the nonpositive term,

$$\sum_{k=0}^N\gamma_k^r\left(f(x_k)-f(x^*)\right) \leq 2M^2\gamma_N^{r-1}/\eta_N + \left(1+\tfrac{1}{m}\right)\frac{(C+\eta_0 C_f)^2}{2}\sum_{k=0}^N\frac{\gamma_k^{r+1}}{\eta_k}.$$

Next, dividing both sides by $\sum_{k=0}^N\gamma_k^r$, taking into account the convexity of $f$, and Remark 4 we obtain the result.

(b) Consider equation (1). Writing it for $y:=x^*\in X$,

$$2\gamma_k^r\phi(x_k) \leq 2\gamma_k^r\eta_k\left(f(x^*)-f(x_k)\right)+\gamma_k^{r-1}\left(\|x_k-x^*\|^2-\|x_{k+1}-x^*\|^2\right)+\left(1+\tfrac{1}{m}\right)\gamma_k^{r+1}\left(C+\eta_k C_f\right)^2.$$

Recalling the definition of $M_f$, we have, $|f(x^*)-f(x_k)|\leq 2M_f$. Bounding the preceding inequality,

$$2\gamma_k^r\phi(x_k) \leq \gamma_k^{r-1}\left(\|x_k-x^*\|^2-\|x_{k+1}-x^*\|^2\right) + 4\gamma_k^r\eta_k M_f+\left(1+\tfrac{1}{m}\right)\gamma_k^{r+1}\left(C+\eta_k C_f\right)^2. \quad (5)$$

Adding and subtracting $\gamma_{k-1}^{r-1}\|x_k-x^*\|^2$ in the above,

$$2\gamma_k^r\phi(x_k) \leq \gamma_{k-1}^{r-1}\|x_k-x^*\|^2-\gamma_k^{r-1}\|x_{k+1}-x^*\|^2 + 4\gamma_k^r\eta_k M_f+\underbrace{\left(\gamma_k^{r-1}-\gamma_{k-1}^{r-1}\right)\|x_k-x^*\|^2}_{\text{term 3}} + \underbrace{\left(1+\tfrac{1}{m}\right)\gamma_k^{r+1}\left(C+\eta_k C_f\right)^2}_{\text{term 4}}.$$

Using the nonincreasing property of $\{\gamma_k\}$ and $\{\eta_k\}$, recalling $0\leq r<1$, we have $\gamma_k^{r-1}-\gamma_{k-1}^{r-1}>0$, and $\left(1+\frac{1}{m}\right)\gamma_k^{r+1}>0$. Further, from the boundedness of set $X$, we have: term 3 $<\left(\gamma_k^{r-1}-\gamma_{k-1}^{r-1}\right)4M^2$, and term 4 $<\left(1+\frac{1}{m}\right)\gamma_k^{r+1}\left(C+\eta_0 C_f\right)^2$. Next, taking summations over $k=1,\ldots,N$, and dropping the nonpositive terms, we obtain:

$$2\sum_{k=1}^N\gamma_k^r\phi(x_k) \leq \gamma_0^{r-1}\|x_1-x^*\|^2+4M^2\left(\gamma_N^{r-1}-\gamma_0^{r-1}\right) + \left(1+\tfrac{1}{m}\right)\left(C+\eta_0 C_f\right)^2\sum_{k=1}^N\gamma_k^{r+1}+4M_f\sum_{k=1}^N\gamma_k^r\eta_k. \quad (6)$$

Writing equation (5) for $k=0$, we have:

$$2\gamma_0^r\phi(x_0) \leq \gamma_0^{r-1}\left(\|x_0-x^*\|^2-\|x_1-x^*\|^2\right)+4\gamma_0^r\eta_0 M_f + \left(1+\tfrac{1}{m}\right)\gamma_0^{r+1}\left(C+\eta_0 C_f\right)^2.$$

Adding this into equation (6), we have:

$$2\sum_{k=0}^N\gamma_k^r\phi(x_k) \leq \gamma_0^{r-1}\|x_0-x^*\|^2+4M^2\left(\gamma_N^{r-1}-\gamma_0^{r-1}\right) + \left(1+\tfrac{1}{m}\right)\left(C+\eta_0 C_f\right)^2\sum_{k=0}^N\gamma_k^{r+1}+4M_f\sum_{k=0}^N\gamma_k^r\eta_k.$$

Bounding $\|x_0-x^*\|^2$ from equation (3), dividing both sides by $\sum_{k=0}^N\gamma_k^r$, taking into account the convexity of $\phi(x_k)$, and from Remark 4, we obtain the required result. □

Next, we present the suboptimality and infeasibility convergence rate statements for the proposed algorithm.

**Theorem 1** (Suboptimality and infeasibility rate results)**.** *Consider Algorithm 1. Let Assumption 1 hold. Consider scalars $M, M_f>0$ such that $\|x\|\leq M$ and $|f(x)|\leq M_f$ for all $x\in X$. Let $\bar{x}_N$ be generated by Algorithm 1 after $N$ iterations. Let $\{\gamma_k\}$ and $\{\eta_k\}$ be the stepsize and regularization parameter sequences generated using $\gamma_k=\frac{\gamma_0}{\sqrt{1+k}}$, $\eta_k=\frac{\eta_0}{(1+k)^b}$, where $\gamma_0,\eta_0>0$, and $0<b<0.5$. Then, for any optimal solution $x^*$ to problem $(P)$, we have:*

$$(a)\ f(\bar{x}_N)-f(x^*) \leq \frac{2-r}{\gamma_0^r(N+1)^{0.5-b}}\left(\frac{2M^2}{\eta_0\gamma_0^{1-r}} + \frac{(m+1)\gamma_0^{1+r}(C+\eta_0 C_f)^2}{2m\eta_0(0.5-0.5r+b)}\right). \quad (7)$$

$$(b)\ \phi(\bar{x}_N) \leq \frac{(2-r)}{(N+1)^b}\left(\frac{2M^2}{\gamma_0}+\frac{2M_f\eta_0}{(1-0.5r-b)} + \frac{(m+1)(C+\eta_0 C_f)^2\gamma_0}{2m(0.5-0.5r)}\right). \quad (8)$$

*Proof.* Taking Proposition 1 (a) and (b) into account, let us define the following terms:

$$\Lambda_{N,1} \triangleq \sum_{k=0}^{N} \gamma_k^r, \quad \Lambda_{N,2} \triangleq \frac{2M^2\gamma_N^{r-1}}{\eta_N},$$
$$\Lambda_{N,3} \triangleq \left(1+\tfrac{1}{m}\right)\frac{(C+\eta_0 C_f)^2}{2}\sum_{k=0}^{N}\eta_k^{-1}\gamma_k^{r+1},$$
$$\Lambda_{N,4} \triangleq 2M^2\gamma_N^{r-1}, \quad \Lambda_{N,5} \triangleq 2M_f\sum_{k=0}^{N}\eta_k\gamma_k^r,$$
$$\Lambda_{N,6} \triangleq \left(1+\tfrac{1}{m}\right)\frac{(C+\eta_0 C_f)^2}{2}\sum_{k=0}^{N}\gamma_k^{r+1}.$$

From Proposition 1 (a) and (b), we have:

$$f(\bar{x}_N) - f(x^*) \leq (\Lambda_{N,2}+\Lambda_{N,3})/\Lambda_{N,1},$$
$$\phi(\bar{x}_N) \leq (\Lambda_{N,4}+\Lambda_{N,5}+\Lambda_{N,6})/\Lambda_{N,1}. \quad (9)$$

Next, applying Lemma 1 and substituting $\{\gamma_k\}$ and $\{\eta_k\}$ by their update rules, we obtain:

$$\Lambda_{N,1} = \sum_{k=0}^{N}\frac{\gamma_0^r}{(k+1)^{0.5r}} \geq \frac{\gamma_0^r(N+1)^{1-0.5r}}{2(1-0.5r)}.$$
$$\Lambda_{N,2} = \frac{2M^2(N+1)^{0.5(1-r)+b}}{\eta_0\gamma_0^{1-r}}. \qquad \Lambda_{N,4} = \frac{2M^2(N+1)^{0.5(1-r)}}{\gamma_0^{1-r}}.$$
$$\Lambda_{N,3} = \left(1+\tfrac{1}{m}\right)\frac{(C+\eta_0 C_f)^2}{2}\sum_{k=0}^{N}\frac{\gamma_0^{1+r}}{\eta_0(k+1)^{0.5(1+r)-b}}$$
$$\leq \frac{(m+1)\gamma_0^{1+r}(C+\eta_0 C_f)^2(N+1)^{1-0.5(1+r)+b}}{2m\eta_0(1-0.5(1+r)+b)}.$$
$$\Lambda_{N,5} = \sum_{k=0}^{N}\frac{2M_f\eta_0\gamma_0^r}{(k+1)^{0.5r+b}} \leq \frac{2M_f\eta_0\gamma_0^r(N+1)^{1-0.5r-b}}{1-0.5r-b}.$$
$$\Lambda_{N,6} \leq \frac{(m+1)(C+\eta_0 C_f)^2\gamma_0^{r+1}(N+1)^{1-0.5(1+r)}}{2m(1-0.5(1+r))}.$$

For these inequalities to hold, we need to ensure that conditions of Lemma 1 are met. Accordingly, we must have $0 \leq 0.5r < 1$, $0 \leq 0.5(1+r)-b < 1$, $0 \leq 0.5r+b < 1$, and $0 \leq 0.5(1+r) < 1$. These relations hold because $0 \leq r < 1$ and $0 < b < 0.5$. Another set of conditions when applying Lemma 1 includes $N \geq \max\{2^{1/(1-0.5r)}, 2^{1/(1-0.5(1+r)+b)}, 2^{1/(1-0.5r-b)}, 2^{1/(1-0.5(1+r))}\} - 1$. Note that this condition is satisfied as a consequence of $N \geq 2^{\frac{2}{1-r}} - 1$, $b > 0$, and $0 \leq r < 1$. We conclude that all the necessary conditions for applying Lemma 1 and obtaining the aforementioned bounds for the terms $\Lambda_{N,i}$ are satisfied. To show that the inequalities (7) and (8), it suffices to substitute the preceding bounds of $\Lambda_{N,i}$, in the inequalities (9).

Inequality (7) is obtained by rearranging the terms in the preceding relation. Next, consider the following:

$$\phi(\bar{x}_N) \leq (2-r)\left(\frac{2M^2}{\gamma_0(N+1)^{0.5}} + \frac{2M_f\eta_0}{(1-0.5r-b)(N+1)^b} + \frac{(m+1)(C+\eta_0 C_f)^2\gamma_0}{2m(0.5-0.5r)(N+1)^{0.5}}\right).$$

Taking into account $0 < b < 0.5$, equation (8) is obtained by rearranging the terms in the preceding inequality. □

## IV. Numerical Results

In this section, we present the simulations for the proposed algorithm on a distributed soft-margin support vector machine (SVM). We compare the performance of aIR-IG with the state-of-the-art IG schemes, including the projected IG, proximal IAG, and SAGA. The schemes are compared in terms of CPU time. For these numerical experiments, we use the soft-margin formulation of SVM, as follows:

$$\begin{aligned} \underset{w,b,z}{\text{minimize}} \quad & \tfrac{1}{2}\|w\|^2 + \tfrac{1}{\lambda}\sum_{i=1}^{N} z_i \\ \text{subject to} \quad & v_i(w^T u_i + b) \geq 1 - z_i \quad \text{for } i = 1,\ldots,N. \\ & z_i \geq 0 \quad \text{for } i = 1,\ldots,N. \end{aligned} \quad (10)$$

Here, $(u_1,v_1),(u_2,v_2),\ldots,(u_N,v_N)$ denote the dataset such that $u \in \mathbb{R}^n$ and $v \in \{-1,+1\}$. The goal here is to find a classifier given by $w^T u + b$ to separate the two classes of $v := +1$ and $v := -1$, whereas $w \in \mathbb{R}^n$ and $b \in \mathbb{R}$. For a distributed implementation, we define the objective for agent $i \in \{1,\ldots,m\}$ as follows:

$$f_i(w,z_i) = \sum_{j=\frac{N}{m}\times(i-1)+1}^{\frac{N}{m}\times i} \tfrac{1}{2N}\|w\|^2 + \tfrac{1}{\lambda}z_j.$$

Recall that Algorithm 1 does not require any projection onto the feasible set. However, in other schemes including IG, proximal IAG, and SAGA a projection (more generally a proximal step) is needed. For convenience, define $x \triangleq (w^T, b, z^T)^T$. Now for evaluating the projection of vector $x_1 \triangleq (w_1^T, b_1, z_1^T)^T$, we solve the following optimization:

$$\min_{w,b,z}\left\{\frac{\|x-x_1\|^2}{2}\middle| v_i(w^T u_i + b) \geq 1 - z_i, z_i \geq 0 \ \forall i \in [N]\right\}. \quad (11)$$

**Set up.** The simulations were performed for $m = 20$ agents, $\lambda = 10$, $\gamma_0 = \eta_0 = 1$, and $b = 0.25$. For this experiment, time was fixed to 200 seconds and the performance of each scheme is recorded. Figure 1 shows the performance of Algorithm 1, projected IG, proximal IAG, and SAGA for the different choices of dimensionality $n$ and the total number of samples $N$. Performance is recorded in terms of suboptimality and infeasibility where suboptimality is $\frac{1}{2}\|w\|^2 + \frac{1}{\lambda}\sum_{i=1}^{N} z_i$ and infeasibility is the violation of constraints of problem (10). Suboptimality is shown in a logarithmic scale in Figure 1. We use the Gurobi-Python interface to solve problem (11).

**Insights.** With increasing the dimension and the number of samples, the projection evaluations take longer and consequently, the performance of the projected variant of the aforementioned IG schemes is deteriorated. This is the case in particular when $N = 500$. Note that the other schemes, namely Proj IG, Prox IAG, and SAGA do not show any update for $N = 200$ and 500 after about 70 and 20 seconds, respectively. This is because of the interruption in their last update due to reaching the time limit of 200 seconds.

## V. Concluding Remarks

We consider the problem of minimizing the finite sum with separable (agent-wise) nonlinear inequality and linear equality and inequality constraints. Our work is motivated by the computational challenges in the projected incremental gradient schemes under the presence of hard-to-project constraints. We develop an averaged iteratively regularized incremental gradient scheme where we employ a novel regularization-based relaxation technique. The proposed algorithm is designed in a way that it does not require a hard-to-project

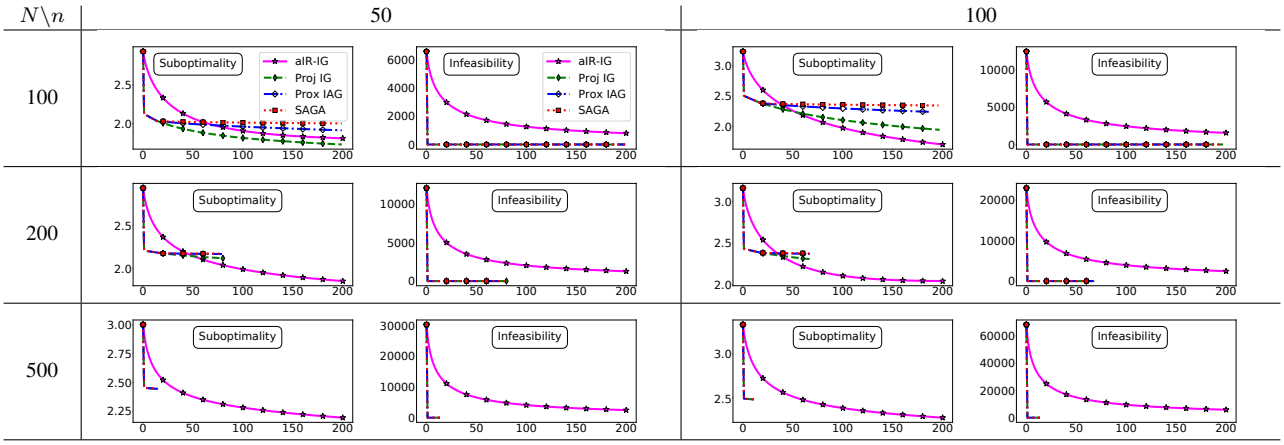


Figure 1: Comparison of suboptimality and infeasibility of Algorithm 1, projected IG, proximal IAG, and SAGA over time.

computation. We establish the rates of convergence for the objective function value and the infeasibility of the generated iterates. We compare the proposed scheme with the state-of-the-art incremental gradient schemes including projected IG, proximal IAG, and SAGA. We observe that the proposed scheme outperforms the projected schemes as the number of samples or the dimension of the solution space increases.

## REFERENCES


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

[2] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," *Third International Symposium on Information Processing in Sensor Networks, 2004. IPSN 2004*, pp. 20–27, 2004.

[3] D. P. Bertsekas, *Nonlinear Programming*, 3rd ed. Bellmont, MA: Athena Scientific, 2016.

[4] ——, "Incremental gradient, subgradient, and proximal methods for convex optimization: A survey," *arXiv:1507.01030v2*, 2017.

[5] A. Nedić and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM Journal on Optimization*, vol. 12, no. 1, pp. 109–138, 2001.

[6] M. Gürbüzbalaban, A. Ozdaglar, and P. A. Parrilo, "Convergence rate of incremental gradient and incremental Newton methods," *SIAM Journal on Optimization*, vol. 29, no. 4, pp. 2542–2565, 2019.

[7] D. Blatt, A. O. Hero, and H. Gauchman, "A convergent incremental gradient method with a constant stepsize," *SIAM Journal on Optimization*, vol. 18, no. 1, pp. 29–51, 2007.

[8] M. Gürbüzbalaban, A. Ozdaglar, and P. A. Parrilo, "On the convergence rate of incremental aggregated gradient algorithms," *SIAM Journal on Optimization*, vol. 27, no. 2, pp. 1035–1048, 2017.

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

[10] T.-H. Chang, A. Nedić, and A. Scaglione, "Distributed constrained optimization by consensus-based primal-dual perturbation method," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1524 – 1538, 2014.

[11] N. S. Aybat and E. Y. Hamedani, "A primal-dual method for conic constrained distributed optimization problems," *Advances in Neural Information Processing Systems*, pp. 5049–5057, 2016.

[12] E. Y. Hamedani and N. S. Aybat, "A primal-dual algorithm for general convex-concave saddle point problems," *arXiv:1803.01401*, 2019.

[13] G. Scutari and Y. Sun, "Distributed nonconvex constrained optimization over time-varying digraphs," *Mathematical Programming*, vol. 176, pp. 497–544, 2019.

[14] A. Nedić and T. Tatarenko, "Convergence rate of a penalty method for strongly convex problems with linear constraints," *59th IEEE Conference on Decision and Control (CDC), Jeju, Korea (South)*, pp. 372–377, 2020.

[15] A. Makhdoumi and A. Ozdaglar, "Convergence rate of distributed ADMM over networks," *IEEE Transactions on Automatic Control*, vol. 62, no. 10, pp. 5082 – 5095, 2017.

[16] V. Khatana and M. V. Salapaka, "DC-DistADMM: ADMM algorithm for constrained distributed optimization over directed graphs," *arXiv:2003.13742*, 2020.

[17] N. S. Aybat and E. Y. Hamedani, "A distributed ADMM-like method for resource sharing over time-varying networks," *SIAM Journal on Optimization*, vol. 29, no. 4, pp. 3036–3068, 2019.

[18] K. Sun and X. A. Sun, "A two-level distributed algorithm for general constrained nonconvex optimization with global convergence," *arXiv:1902.07654v3*, 2020.

[19] W. Tang and P. Daoutidis, "Distributed nonlinear model predictive control through accelerated parallel ADMM," *American Control Conference (ACC), Philadelphia, PA, USA*, pp. 1406–1411, 2019.

[20] D. P. Bertsekas, "Incremental aggregated proximal and augmented Lagrangian algorithms," *arXiv:1509.09257*, 2015.

[21] E. Y. Hamedani and N. S. Aybat, "A decentralized primal-dual method for constrained minimization of a strongly convex function," *arXiv:1908.11835v2*, 2020.

[22] A. Jalilzadeh, "Primal-dual incremental gradient method for nonsmooth and convex optimization problems," *arXiv:2011.02059v4*, 2021.

[23] M. Amini and F. Yousefian, "An iterative regularized incremental projected subgradient method for a class of bilevel optimization problems," *American Control Conference (ACC), Philadelphia, PA, USA*, pp. 4069–4074, 2019.

[24] F. Yousefian, A. Nedić, and U. V. Shanbhag, "On smoothing, regularization, and averaging in stochastic approximation methods for stochastic variational inequality problems," *Mathematical Programming*, vol. 165, no. 1, pp. 391–431, 2017.

[25] ——, "On stochastic and deterministic quasi-Newton methods for nonstrongly convex optimization: Asymptotic convergence and rate analysis," *SIAM Journal on Optimization*, vol. 30, no. 2, pp. 1144–1172, 2020.

[26] H. D. Kaushik and F. Yousefian, "A method with convergence rates for optimization problems with variational inequality constraints," *arXiv:2007.15845v2*, 2021.

[27] A. Beck, *First-Order Methods in Optimization*. Philadelphia, PA: MOS-SIAM Series on Optimization, 2017.

[28] H. D. Kaushik and F. Yousefian, "An incremental gradient method for large-scale distributed nonlinearly constrained optimization," *arXiv:2006.07956v4*, 2021.