2023 IEEE 9th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP) | 979-8-3503-4452-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/CAMSAP58249.2023.10403485

A DISTRIBUTED FIRST-ORDER OPTIMIZATION METHOD FOR STRONGLY CONCAVE-CONVEX SADDLE-POINT PROBLEMS

Muhammad I. Qureshi and Usman A. Khan

Tufts University, Medford, MA, USA

ABSTRACT

In this paper, we propose a first-order optimization method for solving saddle-point problems when the data is distributed over a strongly connected weight-balanced network of nodes. Our solution is based on the gradient descent-ascent where each node iteratively computes partial gradients of its local cost function to implement the corresponding steps. The proposed method further uses gradient tracking for both descent and ascent updates to tackle the local versus global cost gaps. We show that the proposed method converges linearly to the unique saddle-point when the global problem is strongly concave-convex. The numerical experiments illustrate the performance comparison of the proposed method with related work for different classes of problems.

***Index Terms*—** Distributed min-max optimization, first-order methods, saddle-point problems

1. INTRODUCTION AND RELATED WORK

Many applications related to signal processing, machine learning, and robust optimization naturally take the form of min-max problems [1–8]. The objective is to simultaneously minimize and maximize the cost function with respect to specific variables such that we find the point of equilibrium (or the saddle-point). In this paper, we consider the cost function $F : \mathbb{R}^{p_x} \times^{p_y} \to \mathbb{R}$; and assume that F is convex in $\mathbf{x}$ and concave in $\mathbf{y}$, where $\mathbf{x} \in \mathbb{R}^{p_x}$ and $\mathbf{y} \in \mathbb{R}^{p_y}$. Thus, the objective is to find the saddle-point $(\mathbf{x}^*, \mathbf{y}^*)$ by minimizing F with respect to $\mathbf{x}$ and maximizing F with respect to $\mathbf{y}$, i.e.,

$$\min_{\mathbf{x}\in\mathbb{R}^{p_x}} \max_{\mathbf{y}\in\mathbb{R}^{p_y}} F(\mathbf{x}, \mathbf{y}).$$

A natural way to solve the above problem is by using a gradient-based method: gradient descent-ascent (**GDA**) [8–10]. **GDA** is well studied in literature due to its extensive applications in constrained optimization, robust regression, image reconstruction, and generative adversarial networks [4–7]. For each iteration, **GDA** requires the evaluation of partial gradients of F with respect to $\mathbf{x}$ and with respect to $\mathbf{y}$, i.e., $\nabla_x F$ and $\nabla_y F$ respectively. However, for very large-scale applications, data is often divided over a network of geographically distributed nodes. Therefore, the evaluation of $\nabla_x F$ and $\nabla_y F$ is practically not possible, and we can only evaluate the partial gradients with respect to the local data, i.e., for each node i, $\nabla_x f_i$ and $\nabla_y f_i$ (where f_i is the local cost function). Moreover, heterogeneous data setting leads to local versus global cost gaps. Therefore, the local partial gradients are often very different from the global partial gradients, i.e., $\forall i, \|\nabla_x f_i - \nabla_x F\| \neq 0$ and $\|\nabla_y f_i - \nabla_y F\| \neq 0$, which leads to inexact convergence.

Existing work on distributed optimization has mainly focused on minimization problems [11–14]. Early work includes [11], which converges to an error ball around the optimal solution at a linear rate using a constant step size (and to the exact solution with a decaying step size at a sublinear rate). This error arises due to data dissimilarity across nodes. To eliminate this problem, the methods described in [12, 13] use gradient tracking and converge to the optimal solution at a linear rate; see [14] for a detailed overview. [15–17] are also of interest where the authors propose distributed stochastic optimization methods for online problems.

Due to the distributed nature of data in several practical applications, some distributed variants of **GDA** are proposed recently [18–21]. In [18], the authors consider **GDA** in federated settings and use bounds on the dissimilarity of local cost functions possessed by the nodes using dissimilarity constants. [19] generalizes the distributed gradient descent-ascent method to any strongly connected undirected network topology but still uses similarity constants. [20, 21] use gradient tracking to eliminate this problem but require strong assumptions on the cost functions (quadratic in $\mathbf{x}$ and $\mathbf{y}$).

In this paper, we propose a distributed first-order gradient descent-ascent method (**GT−GDA**) that uses gradient tracking for both descent and ascent updates. The main contributions are: (i) **GT−GDA** addresses the data heterogeneity using gradient tracking; (ii) We use a unique analysis methodology that leverages the matrix perturbation theory of semi-simple eigenvalues to show that for (small enough) constant stepsizes, **GT−GDA** linearly converges to the unique saddle-point when the global cost function is strongly concave-convex.

We now describe the rest of the paper. Section 2 provides the problem formulation and the necessary assumptions. Section 3 discusses the algorithm development. Section 4 provides the main results along with the convergence analysis. Section 5 illustrates the performance of **GT−GDA** in different numerical experiments. Section 6 concludes the paper.

2. PROBLEM FORMULATION

In this paper, we consider a saddle-point problem distributed over a strongly connected network of n nodes, i.e.,

$$\mathbf{P}: \min_{\mathbf{x}\in\mathbb{R}^{p_x}} \max_{\mathbf{y}\in\mathbb{R}^{p_y}} \left\{ F(\mathbf{x},\mathbf{y}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x},\mathbf{y}) \right\},$$

where the local cost function, at node i, is defined as:

$$f_i(\mathbf{x},\mathbf{y}) := g_i(\mathbf{x}) + \langle \mathbf{y}, P_i\mathbf{x}\rangle - h_i(\mathbf{y}), \qquad \forall i,$$

for $P_i \in \mathbb{R}^{p_y\times p_x}$ and, the global cost is $F(\mathbf{x},\mathbf{y}) := G(\mathbf{x}) + \langle \mathbf{y}, \overline{P}\mathbf{x}\rangle - H(\mathbf{y})$, for $\overline{P} := \frac{1}{n}\sum_{i=1}^{n} P_i$. We next describe some assumptions necessary to ensure the convergence of **GT-GDA**.

Assumption 1 (Network connectivity)**.** *The nodes communicate over a strongly connected weight-balanced graph. The corresponding weight matrices $W := \{w_{i,j}\}$ associated with the network are primitive and doubly stochastic, i.e., $W\mathbf{1}_n = \mathbf{1}_n$ and $\mathbf{1}_n^\top W = \mathbf{1}_n^\top$, where $\mathbf{1}_n$ is a vector of n ones.*

Assumption 2 (Smoothness and convexity)**.** *Each local g_i is ℓ_1-smooth and each h_i is ℓ_2-smooth, where ℓ_1, ℓ_2 are arbitrary positive constants. Furthermore, the global G is convex and the global H is μ-strongly convex.*

Assumption 3 (Full ranked coupling matrix)**.** *The global coupling matrix $\overline{P} := \frac{1}{n}\sum_i P_i$ has full column rank.*

The above-mentioned assumptions are common in the literature on distributed optimization. Assumption 1 is used to ensure average consensus [14] when the nodes communicate over a network. Assumptions 2 and 3 are necessary for the existence of a unique saddle-point [9] in min-max problems.

3. ALGORITHM DEVELOPMENT

To motivate the significance of gradient tracking, we first describe a well-known distributed optimization method: distributed gradient descent (**DGD**) [11]. For a strongly convex cost function $G(\mathbf{x}) := 1/n\sum_{i=1}^{n} g_i(\mathbf{x})$, distributed over a network of n nodes, **DGD** aims to find the unique minimum $\mathbf{x}^*$ (of G) iteratively by implementing:

$$\mathbf{x}_i^{k+1} = \sum_{r=1}^{n} w_{i,r}(\mathbf{x}_r^k - \alpha\nabla g_r^k), \qquad \forall k \geq 0.$$

However, due to data heterogeneity, there exists a local versus global cost gap, i.e., $\forall i$, $\|\nabla g_i(\mathbf{x}) - \nabla G(\mathbf{x})\| \neq 0$. Therefore, **DGD** converges to an error ball around $\mathbf{x}^*$. To eliminate this error, **GT-DGD** uses a gradient tracking technique [12, 13] which replaces ∇g_r^k with a gradient tracking term $\mathbf{q}_i^k$ evaluated as follows:

$$\mathbf{q}_i^k = \sum_{r=1}^{n} w_{ir}(\mathbf{q}_r^{k-1} + \nabla g_r^k - \nabla g_r^{k-1}), \qquad \forall k > 0.$$

It can be verified that for all i, $\mathbf{q}_i^k \to \nabla G(\mathbf{x})$. Thus, **GT-DGD** converges to the unique minimum $\mathbf{x}^*$. In this paper, we use a similar gradient tracking technique for both descent and ascent updates. Algorithm 1 formally describes **GT-GDA**.

Algorithm 1 **GT-GDA** at each node i

Require: $\mathbf{x}_i^0 \in \mathbb{R}^{p_x}, \mathbf{y}_i^0 \in \mathbb{R}^{p_y}, P_i^0 = P_i, \{w_{ir}\}_{r=1}^n, \alpha > 0,$ $\beta > 0, \mathbf{q}_i^0 = \nabla_x f_i(\mathbf{x}_i^0, \mathbf{y}_i^0), \mathbf{t}_i^0 = \nabla_y f_i(\mathbf{x}_i^0, \mathbf{y}_i^0)$

1: **for** $k = 0, 1, 2, \ldots,$ **do**,
2: $\quad P_i^{k+1} \leftarrow \sum_{r=1}^n w_{ir} P_r^k$
3: $\quad \mathbf{x}_i^{k+1} \leftarrow \sum_{r=1}^n w_{ir}(\mathbf{x}_r^k - \alpha\cdot\mathbf{q}_r^k)$
4: $\quad \mathbf{q}_i^{k+1} \leftarrow \sum_{r=1}^n w_{ir}(\mathbf{q}_r^k + \nabla_x f_r^{k+1} - \nabla_x f_r^k)$
5: $\quad \mathbf{y}_i^{k+1} \leftarrow \sum_{r=1}^n w_{ir}(\mathbf{y}_r^k + \beta\cdot\mathbf{t}_r^k)$
6: $\quad \mathbf{t}_i^{k+1} \leftarrow \sum_{r=1}^n w_{ir}(\mathbf{t}_r^k + \nabla_y f_r^{k+1} - \nabla_y f_r^k)$
7: **end for**

At each node i, the two state vectors $\mathbf{x}_i^k$ and $\mathbf{y}_i^k$ are initialized randomly. For some positive constant step-sizes α and β, **GT-GDA** iteratively evaluates: (i) P_i^k, the estimate of global coupling matrix $\overline{P}$; (ii) gradient descent step for $\mathbf{x}_i^k$ updates and the corresponding gradient tracking term $\mathbf{q}_i^k$; (iii) gradient ascent step for $\mathbf{y}_i^k$ updates and the corresponding gradient tracking term $\mathbf{t}_i^k$. It can be verified that for all i, $P_i^k \to \overline{P}$, $\mathbf{q}_i^k \to \nabla_x F$, and $\mathbf{t}_i^k \to \nabla_y F$. Therefore, the state vectors $\mathbf{x}_i^k$ and $\mathbf{y}_i^k$ are updated using the estimates of global partial gradients $\mathbf{q}_i^k$ and $\mathbf{t}_i^k$ respectively, and not the local partial gradients $\nabla_x f_i$ and $\nabla_y f_i$.

4. MAIN RESULTS AND CONVERGENCE ANALYSIS

Now we describe the main theorem that establishes the convergence properties of **GT-GDA**.

Theorem 1. *Consider* $\mathbf{P}$ *under Assumptions* $1, 2,$ *and* 3*. For small enough positive step-sizes $\alpha, \beta > 0$,* ***GT-GDA*** *converges linearly to the unique saddle-point $(\mathbf{x}^*, \mathbf{y}^*)$ of F.*

Theorem 1 states that **GT-GDA** converges to the exact solution $(\mathbf{x}^*, \mathbf{y}^*)$ at a linear rate. To establish this result, we first formulate an LTI system that governs the error dynamics of **GT-GDA**. Then we show that the errors decay to zero at a linear rate. For evaluating the error quantities, we first define the state vectors $\mathbf{x}^k, \mathbf{q}^k \in \mathbb{R}^{np_x}$ and $\mathbf{y}^k, \mathbf{t}^k \in \mathbb{R}^{np_y}$ that concatenate the local vectors $\mathbf{x}_i^k, \mathbf{q}_i^k$ and $\mathbf{y}_i^k, \mathbf{t}_i^k$ for all i. Moreover, we define the average vectors $\overline{\mathbf{x}}^k := \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^k$, $\overline{\mathbf{y}}^k := \frac{1}{n}\sum_{i=1}^n \mathbf{y}_i^k$, $\overline{\mathbf{q}}^k := \frac{1}{n}\sum_{i=1}^n \mathbf{q}_i^k$, and $\overline{\mathbf{t}}^k := \frac{1}{n}\sum_{i=1}^n \mathbf{t}_i^k$. We now describe the error quantities.

(i) Agreement errors, $\|\mathbf{x}^k - \mathbf{1}_n \otimes \overline{\mathbf{x}}^k\|$ and $\|\mathbf{y}^k - \mathbf{1}_n \otimes \overline{\mathbf{y}}^k\|$: This error quantifies the gap between the network's estimate and the agreement (where $\otimes$ denotes the Kronecker product);

(ii) Optimality gaps, $\|\overline{\mathbf{x}}^k - \mathbf{x}^*\|$ and $\|\overline{\mathbf{y}}^k - \nabla H^*(\overline{P}\overline{\mathbf{x}}^k)\|$: This error quantifies the discrepancy between the network average and the optimal solution $(\mathbf{x}^*, \mathbf{y}^*)$;

(iii) Gradient tracking error terms, $\|\mathbf{q}^k - \mathbf{1}_n \otimes \overline{\mathbf{q}}^k\|$ and $\|\mathbf{t}^k - \mathbf{1}_n \otimes \overline{\mathbf{t}}^k\|$: This error quantifies the gap between the gradient tracking estimates and the global gradients.

We now describe Lemma 1 to establish the system dynamics of **GT-GDA** using these error quantities.

Lemma 1. *Consider **GT-GDA** described in Algorithm 1 under Assumptions 1, 2, and 3. Let $\mathbf{s}^k, \mathbf{e}^k \in \mathbb{R}^6$ be defined as*

$$\mathbf{s}^k := \begin{bmatrix} \|\mathbf{x}^k - \mathbf{1}_n \otimes \overline{\mathbf{x}}^k\| \\ \sqrt{n}\|\overline{\mathbf{x}}^k - \mathbf{x}^*\| \\ \ell^{-1}\|\mathbf{q}^k - \mathbf{1}_n \otimes \overline{\mathbf{q}}^k\| \\ \|\mathbf{y}^k - \mathbf{1}_n \otimes \overline{\mathbf{y}}^k\| \\ \sqrt{n}\|\overline{\mathbf{y}}^k - \nabla H^*(\overline{P}\overline{\mathbf{x}}^k)\| \\ \ell^{-1}\|\mathbf{t}^k - \mathbf{1}_n \otimes \overline{\mathbf{t}}^k\| \end{bmatrix}, \qquad \mathbf{e}^k := \begin{bmatrix} \|\mathbf{x}^k\| \\ \|\mathbf{y}^k\| \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix};$$

*and $N_{\alpha,\beta} \in \mathbb{R}^{6\times 6}$ be such that it has α and β at the $(2,1)$ and $(1,5)$ locations, respectively, and zeros everywhere else. Furthermore, for some positive $\sigma_M \geq \sigma_m > 0$, we define $\sigma_m \leq \|\!|\, P_i \,|\!\| \leq \sigma_M$ for all i. Then $\forall k > 0$ and some positive step-sizes $0 < \alpha \leq \frac{\beta\mu^2}{\Omega\sigma_M^2}$, (where $\Omega > 1$) and $\beta > 0$, the LTI system governing the error dynamics of **GT-GDA** can be described by the following relation:*

$$\mathbf{s}^{k+1} \leq (M_0 + \beta M)\,\mathbf{s}^k + N_{\alpha,\beta}\mathbf{e}^k\lambda^k\nu, \tag{1}$$

where $\nu := \sqrt{\frac{1}{n}\sum_{i=1}^n \|\!|\, P_i - \overline{P} \,|\!\|^2}$, the matrix $M_0 \in \mathbb{R}^{6\times 6}$ is defined as:

$$M_0 := \begin{bmatrix} \lambda & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \lambda & 0 & \lambda & \frac{\lambda\sigma_M}{\ell} & 0 & 0 \\ 0 & 0 & 0 & \lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ \frac{\lambda\sigma_M}{\ell} & 0 & 0 & \lambda & 0 & \lambda \end{bmatrix},$$

and $M \in \mathbb{R}^{6\times 6}$ takes the form:

$$M := \begin{bmatrix} 0 & 0 & \times & 0 & 0 & 0 \\ \times & -\frac{\sigma_m^2}{\omega\ell_2} & 0 & 0 & \frac{\mu^2}{\Omega\sigma_M} & 0 \\ \times & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \\ \times & \frac{\mu\ell_1}{\Omega\sigma_M} + \frac{\sigma_M}{\Omega} & 0 & \times & \frac{\mu}{\Omega} - \mu & 0 \\ \times & \times & \times & \times & \times & \times \end{bmatrix},$$

where '$\times$' are the "don't care" terms (will be clarified later), $\lambda := \rho(W - \lim_{k\to\infty} W^k)$, $\ell = \max\{\ell_1, \ell_2\}$, and ω is a positive constant such that $0 < \beta/\omega \leq \alpha$.

We note that Lemma 1 establishes the system dynamics of **GT-GDA**. The proof is omitted due to space limitations, see [22] for details. Next, we describe a useful lemma that is essential for proving the convergence properties of **GT-GDA** using matrix perturbation theory for semi-simple eigenvalues.

Lemma 2. *[23] Consider a matrix $M_\beta \in \mathbb{R}^{n\times n}$, of the form $M_\beta := M_0 + \beta M$, depends smoothly on a real parameter $\beta \geq 0$. Assume M_0 has $l < n$ equal eigenvalues, $\lambda_1 = \cdots = \lambda_l$, associated with independent left and right eigenvectors $\mathbf{u}_1, \cdots, \mathbf{u}_l$ and $\mathbf{v}_1, \cdots, \mathbf{v}_l$, respectively, i.e.*

$$\begin{bmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_l \end{bmatrix} \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_l \end{bmatrix} = I_l,$$

such that $I_l \in \mathbb{R}^{l\times l}$ is an identity matrix. Let $\lambda_i(\beta)$ denote the i-th eigenvalue of M_β, as a function of β, corresponding to λ_i, where $i \in \{1, \ldots, l\}$, and the matrix $M := dM_\beta/d\beta|_{\beta=0}$. Then $d\lambda_i/d\beta|_{\beta=0}$ is the i-th eigenvalue of the following $l \times l$ matrix:

$$S := \begin{bmatrix} \mathbf{u}_1^\top M\mathbf{v}_1 & \cdots & \mathbf{u}_1^\top M\mathbf{v}_l \\ \vdots & \ddots & \vdots \\ \mathbf{u}_l^\top M\mathbf{v}_1 & \cdots & \mathbf{u}_l^\top M\mathbf{v}_l \end{bmatrix}.$$

Proof of Theorem 1. Now we analyze the error dynamics of **GT-GDA** described in Lemma 1 and evaluate the convergence rate. Equation (1) shows the evolution of error with time. If $\|\mathbf{s}^k\| \to 0$ at a linear rate, it essentially proves that **GT-GDA** converges to $(\mathbf{x}^*, \mathbf{y}^*)$ linearly. We observe that the second term on the right-hand side of (1) converges to $\mathbf{0}_6$ (a vector of six zeros) exponentially because $\lambda \in [0, 1)$ and $\lambda^k \to 0$ exponentially. Therefore, it is sufficient to show that the spectral radius $\rho(M_0 + \beta M) < 1$. By observing the structure of M_0, it can be verified that the eigenvalues of the matrix are the elements at the main diagonal. Therefore, when $\beta = 0$, the spectral radius $\rho(M_0 + \beta M) = 1$ (and is governed by the two semi-simple eigenvalues of M_0). To understand the perturbation effect of the step-size β on these repeated eigenvalues, we use the result from Lemma 2. It can be verified that the left eigenvectors corresponding to the two semi-simple eigenvalues of M_0 are $\mathbf{u}_1^\top = [0\,1\,0\,0\,0\,0]$ and $\mathbf{u}_2^\top = [0\,0\,0\,0\,1\,0]$, and the right eigenvectors are $\mathbf{v}_1 = \mathbf{u}_1$ and $\mathbf{v}_2 = \mathbf{u}_2$. Thus, the $d\lambda_1/d\beta|_{\beta=0}$ and $d\lambda_2/d\beta|_{\beta=0}$ are the eigenvalues of the following 2×2 matrix:

$$S := \begin{bmatrix} \mathbf{u}_1^\top M\mathbf{v}_1 & \mathbf{u}_1^\top M\mathbf{v}_2 \\ \mathbf{u}_2^\top M\mathbf{v}_1 & \mathbf{u}_2^\top M\mathbf{v}_2 \end{bmatrix} = \begin{bmatrix} -\frac{\sigma_m^2}{\omega\ell_2} & \frac{\mu^2}{\Omega\sigma_M} \\ \frac{\mu\ell_1}{\Omega\sigma_M} + \frac{\sigma_M}{\Omega} & \frac{\mu}{\Omega} - \mu \end{bmatrix}.$$

If both eigenvalues of S matrix are negative, it is sufficient proof to show that both semi-simple eigenvalues become less than 1 with a small change in β. It can be verified that both diagonal terms of S are negative. Thus, the sum of eigenvalues of S is negative (trace is equal to the sum of eigenvalues). This implies that at least one eigenvalue of S is negative. Next, we observe the determinant of S. If the determinant is positive, then both eigenvalues must possess the same (nega-

tive in our case) sign. To establish that, we find a bound on Ω:

$$\frac{\sigma_m^2\mu}{\omega\ell_2}\left(1-\frac{1}{\Omega}\right)-\frac{\mu^2}{\Omega\sigma_M}\left(\ell_1+\frac{\sigma_M^2}{\mu}\right)\frac{\mu}{\Omega\sigma_M}>0,$$
$$\iff \frac{\sigma_m^2\mu}{\omega\ell_2}\left(1-\frac{1}{\Omega}\right)>\frac{\mu^3}{\Omega^2\sigma_M^2}\left(\ell_1+\frac{\sigma_M^2}{\mu}\right),$$
$$\impliedby \Omega>\max\left\{\frac{\omega\mu^2\ell_1\ell_2}{\sigma_M^2\sigma_m^2}+\frac{\omega\mu\ell_2}{\sigma_m^2},2\right\}.$$

When Ω satisfies the above bound, both eigenvalues of S are negative. Thus, the spectral radius $\rho(M_0+\beta M)<1$ and the theorem follows. □

5. NUMERICAL EXPERIMENTS

In this section, we describe some numerical experiments to compare the performance of **GT-GDA** with **D-GDA** (a distributed variant of **GDA** that does not use gradient tracking). For the generality of results, we consider two types of networks based on the structure and connectivity of nodes: (i) directed exponential graph with $n=16$ nodes (see Figure 1, left) and (ii) undirected geometric graph with $n=200$ nodes (see Figure 1, right).

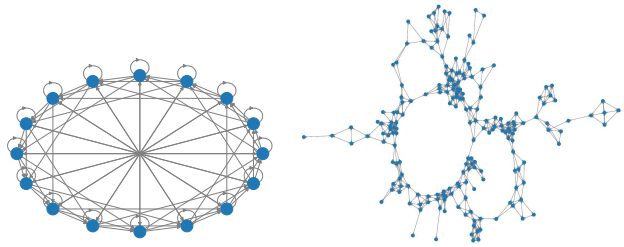

Fig. 1. (left) Directed exponential graph with $n=16$ nodes and (right) undirected geometric graph with $n=200$ nodes.

We consider the saddle-point equivalent of the regression problem such that each node i has access to its private cost: $f_i(\mathbf{x},\mathbf{y}):=\langle\mathbf{y},\mathbf{b}_i\rangle-\frac{1}{2}\|\mathbf{y}\|^2-\langle\mathbf{y},P_i\mathbf{x}\rangle+\lambda_R R_i(\mathbf{x})$, where $\mathbf{b}_i\in\mathbb{R}^{p_y}$ is the local vector, $P_i\in\mathbb{R}^{p_y\times p_x}$ is the local data matrix, $R_i(\mathbf{x})$ is the local regularizer, and λ_R is the regularization constant. The global objective is to evaluate the saddle-point by computing:

$$\min_{\mathbf{x}\in\mathbb{R}^{p_x}}\max_{\mathbf{y}\in\mathbb{R}^{p_y}}F(\mathbf{x},\mathbf{y})=\min_{\mathbf{x}\in\mathbb{R}^{p_x}}\max_{\mathbf{y}\in\mathbb{R}^{p_y}}\frac{1}{n}\sum_{i=1}^{n}f_i(\mathbf{x},\mathbf{y}).$$

In this section, we consider two different regularizers and evaluate the performance by computing the optimality gap, i.e., $\|\overline{\mathbf{x}}^k-\mathbf{x}^*\|+\|\overline{\mathbf{y}}^k-\mathbf{y}^*\|$ for all $k>0$.

Smooth and strongly convex regularizer: We first use a smooth and strongly convex regularizer for every node i, i.e., $R_i(\mathbf{x})=\|\mathbf{x}\|^2$. It can be verified that the global cost F is strongly convex and strongly concave. Figure 2 shows the performance results of **GT-GDA** and **D-GDA**. The left figure shows the evolution of optimality gap when the problem is distributed over a strongly connected directed exponential network of $n=16$ nodes. The figure on the right shows the performance when the underlying graph is geometric with $n=200$ nodes. For both cases, it can be seen that the optimality gap of **D-GDA** initially decreases and then stays constant. This steady-state error is caused by the global versus local cost gaps due to heterogeneous data distribution. However, **GT-GDA** converges to the unique saddle-point using gradient tracking.

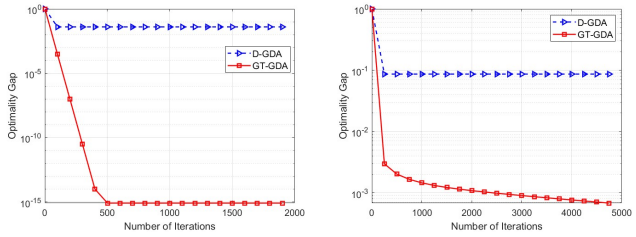

Fig. 2. Performance comparison of **D-GDA** and **GT-GDA** over a network of $n=32$ nodes (left) and $n=200$ nodes (right).

Smooth and convex regularizer: Next, we consider a smooth and convex regularizer (but *not strongly* convex) such that $R_i(\mathbf{x}):=\frac{1}{a_i}\sum_{j=1}^{p_x}[\log(1+e^{a_i x_j})+\log(1+e^{-a_i x_j})]$, $\forall i$. Figure 3 shows the performance results of **GT-GDA** and **D-GDA**. For both graphs (exponential on the left and geometric on the right), it can be observed that **D-GDA** converges to an error ball around the unique saddle-point of F. However, **GT-GDA** converges to the exact solution $(\mathbf{x}^*,\mathbf{y}^*)$.

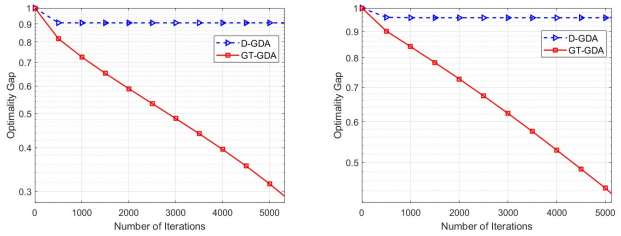

Fig. 3. Performance comparison of **D-GDA** and **GT-GDA** for smooth and convex regularizer.

6. CONCLUSION

In this paper, we propose a first-order optimization method that solves distributed saddle-point problem when the global cost can be described as $F(\mathbf{x},\mathbf{y}):=G(\mathbf{x})+\langle\mathbf{y},\overline{P}\mathbf{x}\rangle-H(\mathbf{y})$. The proposed method **GT-GDA** uses gradient tracking to deal with the error caused by data heterogeneity and converges to the unique saddle-point $(\mathbf{x}^*,\mathbf{y}^*)$ at a linear rate when F is strongly concave-convex. We develop an LTI system governing the error dynamics of **GT-GDA** and use matrix perturbation theory for semi-simple eigenvalues to prove the convergence results. Numerical experiments illustrate the performance of **GT-GDA** and compare it with related work that does not use gradient tracking.

7. REFERENCES

[1] E. L. Hall, J. J. Hwang, and F. A. Sadjadi, "Computer Image Processing And Recognition," in *Optics in Metrology and Quality Assurance*, Harvey L. Kasdan, Ed. International Society for Optics and Photonics, 1980, vol. 0220, pp. 2 – 10, SPIE.

[2] Y. Malitsky and M. K. Tam, "A forward-backward splitting method for monotone inclusions without cocoercivity," *SIAM Journal on Optimization*, vol. 30, no. 2, pp. 1451–1472, 2020.

[3] C. Y. Chen and P. P. Vaidyanathan, "Quadratically constrained beamforming robust against direction-of-arrival mismatch," *IEEE Transactions on Signal Processing*, vol. 55, no. 8, pp. 4139–4150, 2007.

[4] A. Sinha, H. Namkoong, and J. Duchi, "Certifiable distributional robustness with principled adversarial training," in *International Conference on Learning Representations*, 2018.

[5] M. Benzi, G. H. Golub, and J. Liesen, "Numerical solution of saddle point problems," *Acta Numerica*, vol. 14, pp. 1–137, 2005.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, Eds. 2014, vol. 27, Curran Associates, Inc.

[7] T. Liang and J. Stokes, "Interaction matters: A note on non-asymptotic local convergence of generative adversarial networks.," *CoRR*, vol. abs/1802.06132, 2018.

[8] T. Lin, C. Jin, and M. Jordan, "On gradient descent ascent for nonconvex-concave minimax problems," in *Proceedings of the 37th International Conference on Machine Learning*. 13–18 Jul 2020, vol. 119 of *Proceedings of Machine Learning Research*, pp. 6083–6093, PMLR.

[9] S. S. Du and W. Hu, "Linear convergence of the primal-dual gradient method for convex-concave saddle point problems without strong convexity," 2019.

[10] A. Mokhtari, A. Ozdaglar, and S. Pattathil, "A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach," in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. 26–28 Aug 2020, vol. 108, pp. 1497–1507, PMLR.

[11] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48, 2009.

[12] P. Di Lorenzo and G. Scutari, "Distributed nonconvex optimization over networks," in *6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*. IEEE, 2015, pp. 229–232.

[13] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes," in *54th IEEE Conference on Decision and Control*, 2015, pp. 2055–2060.

[14] R. Xin, S. Pu, A. Nedić, and U. A. Khan, "A general framework for decentralized optimization with first-order methods," *Proceedings of the IEEE*, vol. 108, no. 11, pp. 1869–1889, 2020.

[15] R. Xin, U. A. Khan, and S. Kar, "Variance-reduced decentralized stochastic optimization with gradient tracking," *IEEE Transactions on Signal Processing*, vol. 68, pp. 6255–6271, 2020.

[16] R. Xin, S. Kar, and U. A. Khan, "Decentralized stochastic optimization and machine learning," *IEEE Signal Processing Magazine*, May 2020.

[17] M. I. Qureshi, R. Xin, S. Kar, and U. A. Khan, "Variance reduced stochastic optimization over directed graphs with row and column stochastic weights," 2022, arXiv: 2202.03346.

[18] Y. Deng and M. Mahdavi, "Local stochastic gradient descent ascent: Convergence analysis and communication efficiency," 2021, arXiv: 2102.13152.

[19] A. Beznosikov, G. Scutari, A. Rogozin, and A. Gasnikov, "Distributed saddle-point problems under similarity," 2021, arXiv: 2107.10706.

[20] H. Wai, Z. Yang, Z. Wang, and M. Hong, "Multi-agent reinforcement learning via double averaging primal-dual optimization," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2018, NIPS'18, p. 9672–9683, Curran Associates Inc.

[21] J. Ren, J. Haupt, and Z. Guo, "Communication-efficient hierarchical distributed optimization for multi-agent policy evaluation," *Journal of Computational Science*, vol. 49, pp. 101280, 2021.

[22] M. I. Qureshi and U. A. Khan, "Distributed saddle point problems for strongly concave-convex functions," *IEEE Transactions on Signal and Information Processing over Networks*, pp. 1–12, 2023.

[23] A.P. Seyranian and A.A. Mailybaev, *Multiparameter Stability Theory With Mechanical Applications*, World Scientific Publishing Company, 2003.