

# Zeroth-Order Non-Convex Optimization for Cooperative Multi-Agent Systems with Diminishing Step Size and Smoothing Radius

Xinran Zheng, Tara Javidi, and Behrouz Touri

Electrical and Computer Engineering Department, University of California San Diego

**Abstract**—We study a class of zeroth-order distributed optimization problems, where each agent can control a partial vector and observe a local cost that depends on the joint vector of all agents, and the agents can communicate with each other with time delay. We propose and study a gradient descent-based algorithm using two-point gradient estimators with diminishing smoothing parameters and diminishing step-size and we establish the convergence rate to a first-order stationary point for general nonconvex problems. A byproduct of our proposed method with diminishing step size and smoothing parameters, as opposed to the fixed-parameter scheme, is that our proposed algorithm does not require any information regarding the local cost functions. This makes the solution appealing in practice as it allows for optimizing an unknown (black-box) global function without prior knowledge of its smoothness parameters. At the same time, the performance will adaptively match the problem instance parameters.

## I. INTRODUCTION

Distributed optimization has found broad applications, such as distributed learning [1], network source allocation [2], and wind farm control [3]. Generally, such problems are related to minimizing the sum  $\sum_{i=1}^n f_i(\mathbf{x})$  of local functions  $f_i(\mathbf{x})$ s where  $f_i(\mathbf{x})$  is only known to agent  $i$ . There are two classes of distributed optimization problems. Extensive research has been done on consensus-based distributed optimization, where each agent controls the full vector  $\mathbf{x}$  and needs to achieve consensus with other agents on an optimal or stationary decision vector [4]–[8]. This paper investigates the other class of problems, which we call cooperative multi-agent systems. In these problems, each agent can only partially control the decision vector, but the local cost  $f_i$  observed by agent  $i$  reflects the impact of all agents’ decisions.

In many real-world situations, objective functions are available only as the output of a black-box, or the relationship between the objective function and the variables is so complicated that the direct calculation of the derivatives could be expensive or infeasible. This has led to a research interest in black-box optimization. The well-known Kiefer-Wolfowitz scheme [9] uses  $2d$  ( $d$  is the dimension of the variable) function evaluations to construct a gradient estimator, but this does not scale up with high-dimensional problems. Therefore, many existing works propose and analyze two-point [10] and single-point [11] gradient estimators and show that the convergence rates of two-point estimator algorithms are comparable to their first-order counterparts [10].

This research is supported by NSF grant CNS-2148313, AutoCOMBOT MURI Grant, and AFOSR FA9550-23-1-0057, emails: {x7zheng, tjavidi, btouri}@ucsd.edu

Recently zeroth-order methods have been investigated to solve cooperative multi-agent problems. For example, [12] and [13] proposed algorithms based on two-point gradient estimator and analyzed the convergence rates in both convex and non-convex, and both noiseless and noisy settings. In [14] an asynchronous method is proposed based on single-point gradient estimator and showed that their method outperforms two-point methods under asynchronous updating. However, both papers use constant step-size and smoothing radius that depend on the total number of iterations and the smoothness and Lipschitz parameters of objective functions, which might not be known in practical applications.

Inspired by [12], we analyze a cooperative multi-agent optimization problem with a (slightly) more generalized communication scheme between agents. As opposed to [12], we propose to use diminishing step-size and smoothing radius schedules, which have been used in a similar manner in [15] and [16]. This leads to convergence guarantees without any prior knowledge of the parameters of objective functions (which are required in [12]). This also establishes a rigorous asymptotic bound that was not established previously. We demonstrate that for a class of diminishing step-sizes and smoothing radius, our algorithm can obtain  $\mathcal{O}(T^{-(\frac{1}{2}-\varepsilon)})$  convergence rate, for sufficiently small  $\varepsilon > 0$ .

**Notations.** We denote the set of real numbers by  $\mathbb{R}$  and the vector space of  $d$ -dimensional real-valued row vectors by  $\mathbb{R}^d$ . We use bold letters to denote row vectors. We use  $I_d$  to denote the identity matrix of dimension  $d$ . We use  $\mathcal{N}(0, \Sigma)$  to denote the multi-dimensional Gaussian distribution with zero mean and covariance  $\Sigma$ . We use  $[n]$  to denote  $\{1, \dots, n\}$ . We use  $\|\cdot\|$  to denote the standard Euclidean norm, and  $\langle \cdot, \cdot \rangle$  to denote the standard Euclidean inner product.

## II. PROBLEM FORMULATION

We consider a cooperative multi-agent optimization problem among  $n$  agents, where the agents seek to find

$$\min_{\mathbf{x}^i \in \mathbb{R}^{d_i}} f(\mathbf{x}^1, \dots, \mathbf{x}^n) := \frac{1}{n} \sum_{i=1}^n f_i(\mathbf{x}^1, \dots, \mathbf{x}^n). \quad (1)$$

We assume that  $\mathbf{x}^i \in \mathbb{R}^{d_i}$  is the decision vector controlled by agent  $i \in [n]$ , and  $f_i : \mathbb{R}^{d_1} \times \dots \times \mathbb{R}^{d_n} \rightarrow \mathbb{R}$  is the local cost function observed by agent  $i \in [n]$ .

In this paper, we consider the setting that each agent  $i \in [n]$  can only access the (noiseless) value of its local cost function  $f_i$ , and the information regarding higher order derivatives of  $f_i$  is unavailable to the agent. In addition,

agents are able to exchange information over a possibly time-varying network. Therefore, we assume that the exchange of information incurs a delay. More precisely, for any two agents  $i, j \in [n]$ , when agent  $i$  receives some information from agent  $j$  (with some delay), it also receives a time stamp that denotes the time when agent  $j$  sends this information.

### III. THE MAIN RESULTS

In this section, we introduce our algorithm to solve problem (1). Our algorithm operates on the discrete-time instants  $t = 0, 1, 2, \dots$ . At  $t = 0$ , each agent initializes a guess at an arbitrary point  $\mathbf{x}^i(0) = \mathbf{x}_0^i \in \mathbb{R}^{d_i}$ . At time  $t = 0, 1, 2, \dots$ , each agent  $i \in [n]$  generates a random vector  $\mathbf{z}^i(t) \sim \mathcal{N}(0, I_{d_i})$ , adjusts its decision vector to be  $\mathbf{x}^i(t) + u(t)\mathbf{z}^i(t)$ , where  $u(t)$  is a positive scalar we refer to as *smoothing radius*. Once, every agent does that, agent  $i \in [n]$  observes the corresponding local cost  $f_i(\mathbf{x}(t) + u(t)\mathbf{z}(t))$ , where  $\mathbf{x}(t), \mathbf{z}(t)$  are the concatenated vectors  $(\mathbf{x}^1(t), \dots, \mathbf{x}^n(t))$ ,  $(\mathbf{z}^1(t), \dots, \mathbf{z}^n(t))$ , respectively. Then, in the same way, each agent  $i$  adjusts its vector to be  $\mathbf{x}^i(t) - u(t)\mathbf{z}^i(t)$ , and observes the corresponding local cost  $f_i(\mathbf{x}(t) - u(t)\mathbf{z}(t))$ . Then, using these two observations, the agent  $i$  computes the approximate derivative of  $f_i(\mathbf{x}(t))$  along the vector  $\mathbf{z}(t)$

$$D_i(t) = \frac{f_i(\mathbf{x}(t) + u(t)\mathbf{z}(t)) - f_i(\mathbf{x}(t) - u(t)\mathbf{z}(t))}{2u(t)}$$

and sends it out. At the same time, each agent  $i$  may receive the derivative information of agent  $j \in [n]$ , and only stores the latest  $D_j(t)$  and the latest time stamp from each agent  $j$ . To be specific, at each time  $t$ , we denote the latest time  $s$  that  $D_j(s)$  (calculated by agent  $j$ ) is received by agent  $i$  by time  $t$  to be  $\tau_j^i(t)$ . Thus,  $t - \tau_j^i(t) \geq 0$  represents the communication time delay between agent  $j$  and  $i$ . If agent  $i$  has not received any  $D_j$  from agent  $j$ , we denote  $\tau_j^i(t) = -1$ . For  $i = j$ , we have  $\tau_j^i(t) = t$ . Then, each agent  $i$  computes a partial gradient estimator as

$$\mathbf{g}^i(t) = \frac{1}{n} \sum_{j: \tau_j^i(t) \geq 0} D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)).$$

Finally, each agent  $i$  updates its decision vector as

$$\mathbf{x}^i(t+1) = \mathbf{x}^i(t) - \eta(t)\mathbf{g}^i(t). \quad (2)$$

#### A. Analysis

Here, we present our main result concerning the convergence guarantee for our algorithm. In order to do so, we need to assume some regularity conditions and assumptions on various objects in our work. First, we discuss the assumptions on local cost functions  $f_i(\cdot)$ .

*Assumption 1 (Assumption on Objective Functions):* We assume  $f^* := \inf_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x}) > -\infty$ . Further, for  $i \in [n]$ ,  $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$  is  $G$ -Lipschitz and  $L$ -smooth, i.e.,

$$\|f_i(\mathbf{x}) - f_i(\mathbf{y})\| \leq G\|\mathbf{x} - \mathbf{y}\|, \quad (3)$$

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \quad (4)$$

for all  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ .

We may comment that the above assumptions can be replaced by requiring that the trajectories of the dynamics are bounded in a set  $E$ , and the  $f_i$ s satisfy the above conditions only on the set  $E$ .

The following is the assumption on the communication delay between the agents.

*Assumption 2 (Assumption on Communication Delay):*

The communication delay between agents  $i, j \in [n]$  is a uniformly bounded non-negative integer, i.e.,  $0 \leq t - \tau_j^i(t) \leq B$  for some constant  $B$ .

Note that Assumption 2 subsumes the assumption on time delay in [12], where information is flooding over a (time-invariant) network connecting all agents and there, the time delay between each pair of agents relates to their distance over the network. Here, the network can be time-varying and we only need bounded delay  $B$  between any two nodes when the information is flooded over the network. This is equivalent to the strongly  $B$ -connected assumption that is commonly assumed in the distributed optimization literature (see e.g. [4], [6]).

Finally, we discuss the assumption regarding the step-size sequences and the smoothing radius sequences.

*Assumption 3 (Assumption on Step-size Sequences):* The step-size  $\eta(t)$  and the smoothing radius  $u(t)$  take the form  $\eta(t) = \frac{\eta_0}{(t+1)^\alpha}$ , and  $u(t) = \frac{u_0}{(t+1)^\beta}$ , respectively, where  $\eta_0 > 0$ ,  $u_0 > 0$ ,  $\alpha$  and  $\beta$  are positive exponents satisfying  $0 < \alpha < 1$  and  $\beta > 0$ .

A practical and important caveat here is that in practice, if  $u(t)$  is too small, noise could dominate the function difference, which leads to a bad approximation of the gradient [17].

We are now ready to present our main result that characterizes the convergence rate of our algorithm.

*Theorem 1:* Under assumptions 1-3,

$$\lim_{T \rightarrow \infty} \frac{1}{T+1} \sum_{t=0}^T \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] = 0.$$

In particular, if  $\alpha = \frac{1}{2}$ ,  $\beta = \frac{1}{4}$ , for any  $0 < \varepsilon < \frac{1}{2}$ , we have

$$\frac{1}{T+1} \sum_{t=0}^T \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] = \mathcal{O}(T^{-(\frac{1}{2}-\varepsilon)}).$$

In a closely related work [12], the step-size and smoothing radius were assumed to be constants given by  $\eta = \frac{\alpha_\eta}{L\sqrt{b}\sqrt{nd}\sqrt{T-B+1}}$ , where  $G, L$  are the Lipschitz and smoothness parameters of  $f_i$ s, and the number of iterations  $T$  is predetermined. They show that for these fixed parameters, given the number of iterations  $T > B$ , they have  $\frac{1}{T-B+1} \sum_{t=B}^T \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] \leq K\sqrt{\frac{b\sqrt{nd}}{T-B+1}}$  for some constant  $K > 0$ . Our diminishing step-size approach achieves  $\mathcal{O}(T^{-(\frac{1}{2}-\varepsilon)})$  where  $\varepsilon > 0$  can be arbitrarily close to 0. Therefore, we show that using diminishing step-size does not slow down the convergence in a significant manner. However, in our result, we do not need to know any of these parameters to design our step-size and smoothing radius.

#### IV. NUMERICAL EXPERIMENT

Here, we demonstrate the performance of our algorithm on a wind farm power maximization problem studied in [3], [12]. This example consists of  $n = 80$  wind turbines, each of which can adjust its own axial induction factor denoted by decision variable  $x^i \in \mathbb{R}$ . The power generated by turbine  $i$ , denoted by  $f_i$ , depends on its own axial induction as well as those of the wind turbines upstream. We denote  $\mathbf{x} = (x^1, \dots, x^n)$ , so the wind farm power maximization problem can be written as  $\max_{\mathbf{x}} f(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n f_i(\mathbf{x})$ .

In our experiment, we start from an initial point  $\mathbf{x}(0) = (\frac{1}{3}, \dots, \frac{1}{3})$ , and normalize the power  $f(\mathbf{x})$  by the optimal power  $f^* = f(\mathbf{x}^*)$ , where  $\mathbf{x}^*$  is the optimal action profile. We compare the performance of our algorithm with the algorithm using constant  $\eta$  and  $u$  proposed in [12]. For our algorithm, we choose  $\eta_0 = 0.1$ ,  $u_0 = 0.01$ , and two different  $(\alpha, \beta)$  pairs  $(\alpha, \beta) = (0.4, 0.5)$  or  $(\alpha, \beta) = (0.51, 0.25)$ . For constant-step-size algorithm in [12], we choose three different values for step size,  $\eta_1 = 0.05$ ,  $\eta_2 = 0.01$ ,  $\eta_3 = 0.005$ , both with  $u = 0.001$ . We run the algorithm for  $T = 8000$  iterations and repeat this for 10 trials.

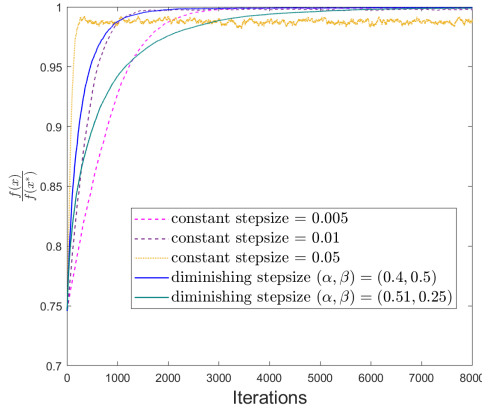


Fig. 1. Optimal wind farm power extraction using diminishing step size vs constant step size algorithm.

In Figure 1, we note that for a constant step-size algorithm, once we choose a step size  $\eta$ , there exists a  $T'$  such that when  $t > T'$ ,  $\mathbf{x}(t)$  will stay in a neighborhood of  $\mathbf{x}^*$ . When the constant step size  $\eta$  is large, this neighborhood is large. When  $\eta$  is small, this neighborhood is small, but the convergence to the neighborhood is slow. By contrast, for our diminishing step size algorithm, we observe convergence to the optimal point and choose proper parameters to achieve a balance between accuracy and convergence rate.

#### V. PROOF OF MAIN RESULT

To prove the main result, first we establish a recursive inequality for  $\mathbb{E}[f(\mathbf{x}(t))]$ . Using that, we can bound  $\sum_{t=B}^T \eta(t) \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2]$ . Finally, we translate such a bound to a bound for the (empirical) average expected squared norm of the gradients, i.e.,

$\frac{1}{T+1} \sum_{t=0}^T \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2]$ . We establish this using intermediate results as discussed below.

First, we show that our gradient estimator is nearly unbiased and its proof follows from a similar argument as the derivation of (21) in [10] and the proof of Lemma 5(b)- [18].

*Lemma 1:* Consider the smoothed version of  $f$  and  $f_j$  defined by  $f^u(\mathbf{x}) := \mathbb{E}_{\mathbf{z}}[f(\mathbf{x} + u\mathbf{z})]$  and  $f_j^u(\mathbf{x}) := \mathbb{E}_{\mathbf{z}}[f_j(\mathbf{x} + u\mathbf{z})]$ , respectively, where  $\mathbf{z} \sim \mathcal{N}(0, I_d)$ . Then, if  $f_j$  is  $G$ -Lipschitz and  $L$ -smooth (see Assumption 1) for  $j \in [n]$ , so will be the smoothed function  $f_j^u : \mathbb{R}^d \rightarrow \mathbb{R}$ . Moreover, for each  $\mathbf{x} \in \mathbb{R}^d$ ,

$$\mathbb{E}_{\mathbf{z}} \left[ \frac{f_j(\mathbf{x} + u\mathbf{z}) - f_j(\mathbf{x} - u\mathbf{z})}{2u} \mathbf{z} \right] = \nabla f_j^u(\mathbf{x}), \text{ and} \quad (5)$$

$$\|\nabla f(\mathbf{x}) - \nabla f^u(\mathbf{x})\| \leq uL\sqrt{d}. \quad (6)$$

Our next intermediate result provides a bound on the second moment of our gradient estimator, whose proof follows the proof of Lemma 3 in [12] and Lemma 3 in [13].

*Lemma 2:* Let  $\mathbf{z} \sim \mathcal{N}(0, I_d)$ , and assume  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  is  $G$ -Lipschitz. Then

$$\mathbb{E}_{\mathbf{z}} \left[ \left| \frac{f(\mathbf{x} + u\mathbf{z}) - f(\mathbf{x} - u\mathbf{z})}{2u} z_i \right|^2 \right] \leq 2\sqrt{6}G^2,$$

for any  $i = 1, \dots, d$ , where  $z_i$  denotes the  $i$ 'th entry of  $\mathbf{z}$ . In particular, if for all  $j \in [n]$ ,  $f_j$  is  $G$ -Lipschitz, then for any  $t \geq 0$  in the evolution of our algorithm, we have

$$\mathbb{E}[\|D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t))\|^2] \leq 2\sqrt{6}G^2 d_i,$$

$$\mathbb{E}[\|\mathbf{g}(t)\|^2] \leq 2\sqrt{6}G^2 d.$$

Our next result quantifies the effect of the delay on our algorithm.

*Lemma 3:* Suppose that the assumptions of Theorem 1 holds. Then, for any  $i, j \in [n]$  and any  $t \geq B$ , we have

$$\begin{aligned} \mathbb{E}[\|\nabla^i f_j^u(\mathbf{x}(t)) - \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t)))\|^2] \\ \leq 2\sqrt{6}G^2 L^2 B d \sum_{\tau=-B}^{-1} \eta^2(t + \tau), \end{aligned}$$

and similarly,  $\mathbb{E}[\|\nabla^i f(\mathbf{x}(t)) - \nabla^i f(\mathbf{x}(\tau_j^i(t)))\|^2] \leq 2\sqrt{6}G^2 L^2 B d \sum_{\tau=-B}^{-1} \eta^2(t + \tau)$ .

*Proof:* Since  $f_j^u$  is  $L$ -smooth, we have

$$\begin{aligned} \mathbb{E}[\|\nabla^i f_j^u(\mathbf{x}(t)) - \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t)))\|^2] \\ \leq L^2 \mathbb{E}[\|\mathbf{x}(t) - \mathbf{x}(\tau_j^i(t))\|^2] \\ \leq L^2 \mathbb{E}[\left( \sum_{\tau=-B}^{-1} \|\eta(t + \tau) \mathbf{g}(t + \tau)\| \right)^2] \\ \leq L^2 \sum_{\tau=-B}^{-1} \eta^2(t + \tau) \sum_{\tau=-B}^{-1} \mathbb{E}[\|\mathbf{g}(t + \tau)\|^2] \\ \leq 2\sqrt{6}G^2 L^2 B d \sum_{\tau=-B}^{-1} \eta^2(t + \tau). \end{aligned}$$

Here, the penultimate inequality is due to the Cauchy-Schwartz inequality (31 in [19]), and the last inequality uses Lemma 2. The second inequality follows similarly.  $\blacksquare$

Next, we introduce Lemma 4, 5, and 6 to bound  $\mathbb{E}[-\sum_{i=1}^n \langle \nabla^i f(\mathbf{x}(t)), \mathbf{g}^i(t) \rangle]$ .

*Lemma 4:* Let  $p(t) = \sqrt{\sum_{\tau=-B}^{-1} \eta^2(t+\tau)}$ . Then for any  $t \geq B$ , we have

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(t)) - \nabla^i f(\mathbf{x}(\tau_j^i(t))), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \leq 2\sqrt{6}G^2 L d \sqrt{nB} p(t). \end{aligned}$$

*Proof:* Using  $ab \leq \frac{1}{2}(a^2 + b^2)$ , we have

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(t)) - \nabla^i f(\mathbf{x}(\tau_j^i(t))), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \leq \frac{1}{2n} \frac{1}{\sqrt{nB} L p(t)} \sum_{i,j} \mathbb{E}[\|\nabla^i f(\mathbf{x}(t)) - \nabla^i f(\mathbf{x}(\tau_j^i(t)))\|^2] \\ & \quad + \frac{1}{2n} \sqrt{nB} L p(t) \sum_{i,j} \mathbb{E}[\|D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t))\|^2] \\ & \leq \frac{1}{2n} \frac{1}{\sqrt{nB} L p(t)} \sum_{i,j} \left( 2\sqrt{6}G^2 L^2 B d \sum_{\tau=-B}^{-1} \eta^2(t+\tau) \right) \\ & \quad + \frac{1}{2n} \sqrt{nB} L p(t) \sum_{i,j} 2\sqrt{6}G^2 d_i = 2\sqrt{6}G^2 L d \sqrt{nB} p(t), \end{aligned}$$

where the last inequality follows from Lemma 3 and Lemma 2.  $\blacksquare$

Similar to Lemma 4, the following result is used to prove Lemma 6.

*Lemma 5:* For any  $t \geq B$ , we have

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))) - \nabla^i f(\mathbf{x}(t)), \right. \\ & \quad \left. \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) - D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \leq 2\sqrt{6}G^2 L d \sqrt{nB} p(t). \end{aligned}$$

*Proof:* Similar to the proof of Lemma 4, we have

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))) - \nabla^i f(\mathbf{x}(t)), \right. \\ & \quad \left. \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) - D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \leq \frac{1}{2n} \frac{1}{\sqrt{nB} L p(t)} \sum_{i,j} \mathbb{E}[\|\nabla^i f(\mathbf{x}(t)) - \nabla^i f(\mathbf{x}(\tau_j^i(t)))\|^2] \\ & \quad + \frac{1}{2n} \sqrt{nB} L p(t) \sum_{i,j} \mathbb{E}[\|D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) - \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t)))\|^2] \stackrel{(7)}{\leq} \frac{1}{2n} \mathbb{E} \left[ \sum_{i,j} \|\nabla^i f(\mathbf{x}(t))\|^2 \right] \end{aligned}$$

Since  $\mathbb{E}[D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t))] = \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t)))$ , we have  $\mathbb{E}[\|D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) - \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t)))\|^2] \leq \mathbb{E}[\|D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t))\|^2]$ . Finally, the proof of the lemma follows by using this inequality in (7) and following the same argument as of the proof of Lemma 4.  $\blacksquare$

Now we can introduce our last lemma.

*Lemma 6:* For any  $t \geq B$ , we have

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \leq -\frac{1}{3} \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] + 4\sqrt{6}G^2 L d \sqrt{nB} p(t) \\ & \quad + \sqrt{6}nG^2 L^2 B d p^2(t) + \frac{3}{2} u^2(t) L^2 d. \end{aligned}$$

*Proof:* Let  $\mathcal{F}_\tau = \sigma(\mathbf{x}(\tau_j^i(t)))$  be the  $\sigma$ -algebra generated by  $\mathbf{x}(\tau_j^i(t))$ . Then, using the law of total expectation (Theorem 4.1.13 in [20]), we have

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & = \mathbb{E}\left[\mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle \middle| \mathcal{F}_\tau\right]\right] \\ & = \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))), \mathbb{E}[D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) | \mathcal{F}_\tau] \rangle\right] \\ & = \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))), \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) \rangle\right] \\ & = \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))) - \nabla^i f(\mathbf{x}(t)), \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) \rangle\right] \\ & \quad + \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(t)), \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) - \nabla^i f_j^u(\mathbf{x}(t)) \rangle\right] \\ & \quad + \mathbb{E}[-\langle \nabla f(\mathbf{x}(t)), \nabla f^u(\mathbf{x}(t)) - \nabla f(\mathbf{x}(t)) \rangle] \\ & \quad - \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2]. \end{aligned} \tag{8}$$

To bound the first term in (8), we use Lemma 4 and Lemma 5 and get

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))) - \nabla^i f(\mathbf{x}(t)), \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) \rangle\right] \\ & = \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))) - \nabla^i f(\mathbf{x}(t)), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \quad + \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))) - \nabla^i f(\mathbf{x}(t)), \right. \\ & \quad \left. \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) - D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \leq 4\sqrt{6}G^2 L d \sqrt{nB} p(t). \end{aligned} \tag{9}$$

To bound the second term in (8), we use  $ab \leq \frac{1}{2}(a^2 + b^2)$  and Lemma 3 and get

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(t)), \nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) - \nabla^i f_j^u(\mathbf{x}(t)) \rangle\right] \\ & \quad + \frac{1}{2n} \mathbb{E} \left[ \sum_{i,j} \|\nabla^i f(\mathbf{x}(t))\|^2 \right] \\ & \quad + \frac{1}{2n} \mathbb{E} \left[ \sum_{i,j} \|\nabla^i f_j^u(\mathbf{x}(\tau_j^i(t))) - \nabla^i f_j^u(\mathbf{x}(t))\|^2 \right] \\ & \leq \frac{1}{2} \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] + \sqrt{6}nG^2 L^2 B d p^2(t). \end{aligned} \tag{10}$$

To bound the third term in (8), we use  $ab \leq \frac{1}{2}(a^2 + b^2)$  Lemma 1 and get

$$\begin{aligned} & \mathbb{E}[-\langle \nabla f(\mathbf{x}(t)), \nabla f^u(\mathbf{x}(t)) - \nabla f(\mathbf{x}(t)) \rangle] \\ & \leq \frac{1}{2} \mathbb{E} \left[ \frac{1}{3} \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] + 3\|\nabla f^u(\mathbf{x}(t)) - \nabla f(\mathbf{x}(t))\|^2 \right] \\ & \leq \frac{1}{6} \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] + \frac{3}{2} u^2(t) L^2 d. \end{aligned} \tag{11}$$

Plugging (9), (10), and (11) into (8), we get

$$\begin{aligned} & \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \leq -\frac{1}{3} \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] + 4\sqrt{6}G^2 Ld\sqrt{nB}p(t) \\ & \quad + \sqrt{6n}G^2 L^2 Bdp^2(t) + \frac{3}{2}u^2(t)L^2d. \end{aligned}$$

Now we are ready to prove the main result.

*Proof:* [Proof of Theorem 1] Our algorithm (2) can be compactly written as  $\mathbf{x}(t+1) = \mathbf{x}(t) - \eta(t)\mathbf{g}(t)$ , where  $\mathbf{g}(t)$  is the  $d$ -dimensional vector that is obtained by concatenation of  $\mathbf{g}^1(t), \dots, \mathbf{g}^n(t)$  vectors. Using  $L$ -smooth condition (Lemma 5.7 in [21]) of  $f$ , we have

$$\begin{aligned} f(\mathbf{x}(t+1)) & \leq f(\mathbf{x}(t)) - \langle \nabla f(\mathbf{x}(t)), \eta(t)\mathbf{g}(t) \rangle + \frac{L}{2} \|\eta(t)\mathbf{g}(t)\|^2 \\ & = f(\mathbf{x}(t)) - \eta(t) \sum_{i=1}^n \langle \nabla^i f(\mathbf{x}(t)), \mathbf{g}^i(t) \rangle + \frac{L}{2} \eta^2(t) \|\mathbf{g}(t)\|^2. \end{aligned}$$

Taking the expected value of both sides, we have

$$\begin{aligned} \mathbb{E}[f(\mathbf{x}(t+1))] & \leq \mathbb{E}[f(\mathbf{x}(t))] \\ & \quad - \eta(t) \mathbb{E}\left[\sum_{i=1}^n \langle \nabla^i f(\mathbf{x}(t)), \mathbf{g}^i(t) \rangle\right] + \frac{L}{2} \eta^2(t) \mathbb{E}[\|\mathbf{g}(t)\|^2]. \end{aligned} \quad (12)$$

From Lemma 4 and Lemma 6 we have

$$\begin{aligned} & \mathbb{E}\left[-\sum_{i=1}^n \langle \nabla^i f(\mathbf{x}(t)), \mathbf{g}^i(t) \rangle\right] \\ & = \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(t)) - \nabla^i f(\mathbf{x}(\tau_j^i(t))), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \quad + \mathbb{E}\left[-\frac{1}{n} \sum_{i,j} \langle \nabla^i f(\mathbf{x}(\tau_j^i(t))), D_j(\tau_j^i(t)) \mathbf{z}^i(\tau_j^i(t)) \rangle\right] \\ & \leq -\frac{1}{3} \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] + 6\sqrt{6}G^2 Ld\sqrt{nB}p(t) \\ & \quad + \sqrt{6n}G^2 L^2 Bdp^2(t) + \frac{3}{2}u^2(t)L^2d. \end{aligned}$$

Using the above inequality and Lemma 2, (12) becomes

$$\begin{aligned} \mathbb{E}[f(\mathbf{x}(t+1))] & \leq \mathbb{E}[f(\mathbf{x}(t))] - \frac{1}{3} \eta(t) \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] \\ & \quad + 6\sqrt{6}G^2 Ld\sqrt{nB}\eta(t)p(t) + \sqrt{6n}G^2 L^2 Bd\eta(t)p^2(t) \\ & \quad + \frac{3}{2}u^2(t)L^2d\eta(t) + \sqrt{6}G^2 Ld\eta^2(t). \end{aligned}$$

By taking the telescoping sum, we get

$$\begin{aligned} & \sum_{t=B}^T \eta(t) \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2] \\ & \leq 3 \mathbb{E}[f(\mathbf{x}(B)) - f^*] + 18\sqrt{6}G^2 Ld\sqrt{nB} \sum_{t=B}^T \eta(t)p(t) \\ & \quad + 3\sqrt{6n}G^2 L^2 Bd \sum_{t=B}^T \eta(t)p^2(t) \\ & \quad + \frac{9}{2}L^2d \sum_{t=B}^T u^2(t)\eta(t) + 3\sqrt{6}G^2 Ld \sum_{t=B}^T \eta^2(t). \end{aligned} \quad (13)$$

Since  $f$  is  $G$ -Lipschitz, we have

$$\begin{aligned} \mathbb{E}[f(\mathbf{x}(B))] & \leq f(\mathbf{x}_0) + G \mathbb{E}[\|\mathbf{x}(B) - \mathbf{x}(0)\|] \\ & \leq f(\mathbf{x}_0) + G \sum_{t=0}^{B-1} \eta(t) \mathbb{E}[\|\mathbf{g}(t)\|] \\ & \leq f(\mathbf{x}_0) + \sqrt{2\sqrt{6}G^2} \sqrt{d} \sum_{t=0}^{B-1} \eta(t) := C_0. \end{aligned}$$

Therefore, using (13) and letting  $S(T) := \sum_{t=B}^T \eta(t) \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2]$ , we have

$$\begin{aligned} S(T) & \leq 18\sqrt{6}G^2 Ld\sqrt{nB} \sum_{t=B}^T \eta(t)p(t) \\ & \quad + 3\sqrt{6n}G^2 L^2 Bd \sum_{t=B}^T \eta(t)p^2(t) \\ & \quad + \frac{9}{2}L^2d \sum_{t=B}^T u^2(t)\eta(t) + 3\sqrt{6}G^2 Ld \sum_{t=B}^T \eta^2(t) + C_1, \end{aligned}$$

where  $C_1 = 3(C_0 - f^*)$  is a constant.

Since  $\eta(t) = \frac{\eta_0}{(t+1)^\alpha}$  and  $u(t) = \frac{u_0}{(t+1)^\beta}$ , we have  $p(t) = \mathcal{O}(\frac{1}{t^\alpha})$ . Therefore, using Riemann sum approximation [22], we have

$$S(T) = \begin{cases} \mathcal{O}(T^{1-2\alpha} + T^{1-\alpha-2\beta}) & \text{if } 0 < \alpha < \frac{1}{2} \text{ and } \alpha + 2\beta < 1, \\ \mathcal{O}(T^{1-2\alpha}) & \text{if } 0 < \alpha < \frac{1}{2} \text{ and } \alpha + 2\beta \geq 1, \\ \mathcal{O}(T^{1-\alpha-2\beta}) & \text{if } \frac{1}{2} \leq \alpha < 1 \text{ and } \alpha + 2\beta < 1, \\ \mathcal{O}(\log T) & \text{if } \alpha = \frac{1}{2} \text{ and } \beta \geq \frac{1}{4}, \\ & \text{or } \frac{1}{2} < \alpha < 1 \text{ and } \alpha + 2\beta = 1, \\ \mathcal{O}(1) & \text{if } \frac{1}{2} < \alpha < 1 \text{ and } \alpha + 2\beta > 1. \end{cases} \quad (14)$$

Next, we establish an upper bound for

$$M(T) := \frac{1}{T-B+1} \sum_{t=B}^T \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2].$$

For any  $\theta \in (0, 1)$ , define

$$M_\theta(T) = \left[ \frac{1}{T-B+1} \sum_{t=B}^T (\mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2])^\theta \right]^{\frac{1}{\theta}}.$$

Note that by Hölder's inequality (Theorem 6.2 in [23]), for any  $p, q > 1$  with  $\frac{1}{p} + \frac{1}{q} = 1$ , and non-negative sequences  $\{a_t\}_{t=1}^T$  and  $\{b_t\}_{t=1}^T$ , we have

$$\left( \sum_{t=1}^T a_t b_t \right)^q \leq \left( \sum_{t=1}^T a_t^p \right)^{\frac{q}{p}} \sum_{t=1}^T b_t^q. \quad (15)$$

Let

$$\begin{aligned} a_t & = \left( \frac{1}{\eta(t)} \right)^\theta = \frac{1}{\eta_0^\theta} (t+1)^{\alpha\theta}, \\ b_t & = (\eta(t) \mathbb{E}[\|\nabla f(\mathbf{x}(t))\|^2])^\theta, \end{aligned}$$

and  $(p, q) = \left(\frac{1}{1-\theta}, \frac{1}{\theta}\right)$ . Noting  $\sum_{t=B}^T b_t^{\frac{1}{\theta}} = S(T)$ , and using (15), we have

$$\begin{aligned} M_\theta(T) &= \left(\frac{1}{T-B+1} \sum_{t=B}^T a_t b_t\right)^{\frac{1}{\theta}} \\ &\leq (T-B+1)^{-\frac{1}{\theta}} \left(\sum_{t=B}^T a_t^{\frac{1}{1-\theta}}\right)^{\frac{1-\theta}{\theta}} \sum_{t=B}^T b_t^{\frac{1}{\theta}} \\ &= (T-B+1)^{-\frac{1}{\theta}} \left(\sum_{t=B}^T \eta_0^{-\frac{\theta}{1-\theta}} (t+1)^{\frac{\alpha\theta}{1-\theta}}\right)^{\frac{1-\theta}{\theta}} S(T) \\ &= \mathcal{O}(T^{-\frac{1}{\theta}} (T^{\frac{\alpha\theta}{1-\theta}+1})^{\frac{1-\theta}{\theta}} S(T)) = \mathcal{O}(T^{\alpha-1} S(T)). \end{aligned}$$

Using a similar argument as in the proof of Proposition 1 in [24], it can be shown that for any  $\theta \in (0, 1)$

$$\sum_{t=B}^T \mathbb{E}[\|\nabla f(x(t))\|^2] \leq \left[\sum_{t=B}^T (\mathbb{E}[\|\nabla f(x(t))\|^2])^\theta\right]^{\frac{1}{\theta}}.$$

Now for a given  $\varepsilon > 0$ , if we let  $\theta = \frac{1}{1+\varepsilon/2}$ , we have

$$\begin{aligned} M(T) &\leq (T-B+1)^{\frac{1}{\theta}-1} M_\theta(T) = (T-B+1)^{\frac{\varepsilon}{2}} M_\theta(T) \\ &= \mathcal{O}(T^{\alpha-1+\frac{\varepsilon}{2}} S(T)). \end{aligned}$$

Therefore, using (14), if  $0 < \varepsilon/2 < \min\{\alpha, 2\beta, 1-\alpha\}$ , we have  $M(T) \rightarrow 0$  as  $T \rightarrow \infty$  for  $0 < \alpha < 1$  and  $\beta > 0$ .

Finally, since the sum of the first  $B$  terms is finite, we have  $\lim_{T \rightarrow \infty} \frac{1}{T+1} \sum_{t=0}^T \mathbb{E}[\|\nabla f(x(t))\|^2] = 0$  and for a special case of  $\alpha = \frac{1}{2}$  and  $\beta = \frac{1}{4}$ , we have

$$\begin{aligned} \frac{1}{T+1} \sum_{t=0}^T \mathbb{E}[\|\nabla f(x(t))\|^2] &= \mathcal{O}(\log(T) T^{-(\frac{1}{2}-\frac{\varepsilon}{2})}) \\ &= \mathcal{O}(T^{-\frac{1}{2}+\varepsilon}). \end{aligned}$$

## VI. CONCLUSION

We studied a zeroth order cooperative multi-agent optimization problem with communication delay. We showed that for a class of diminishing smoothing radius and step sizes if the time delay between any two agents is uniformly bounded, the objective functions are  $L$ -smooth, Lipschitz, then our algorithm will converge to a first-order stationary point of the underlying problem, in particular, the established convergence rates are robust to the objective functions' smoothness parameters.

## REFERENCES

- [1] S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian, "Deep decentralized multi-task multi-agent reinforcement learning under partial observability," in *International Conference on Machine Learning*, pp. 2681–2690, PMLR, 2017.
- [2] G. Tychogiorgos, A. Gkelias, and K. K. Leung, "A non-convex distributed optimization framework and its application to wireless ad-hoc networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4286–4296, 2013.
- [3] J. R. Marden, S. D. Ruben, and L. Y. Pao, "A model-free approach to wind farm control using game theoretic methods," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1207–1214, 2013.

- [4] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.
- [5] W. Shi, Q. Ling, G. Wu, and W. Yin, "Extra: An exact first-order algorithm for decentralized consensus optimization," *SIAM Journal on Optimization*, vol. 25, no. 2, pp. 944–966, 2015.
- [6] T. Tatarenko and B. Touri, "Non-convex distributed optimization," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3744–3757, 2017.
- [7] J. Zeng and W. Yin, "On nonconvex decentralized gradient descent," *IEEE Transactions on signal processing*, vol. 66, no. 11, pp. 2834–2848, 2018.
- [8] H. Reisizadeh, B. Touri, and S. Mohajer, "Distributed optimization over time-varying graphs with imperfect sharing of information," *IEEE Transactions on Automatic Control*, pp. 1–8, 2022.
- [9] J. Kiefer and J. Wolfowitz, "Stochastic estimation of the maximum of a regression function," *The Annals of Mathematical Statistics*, pp. 462–466, 1952.
- [10] Y. Nesterov and V. Spokoiny, "Random gradient-free minimization of convex functions," *Foundations of Computational Mathematics*, vol. 17, no. 2, pp. 527–566, 2017.
- [11] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, "Online convex optimization in the bandit setting: gradient descent without a gradient," *arXiv preprint cs/0408007*, 2004.
- [12] Y. Tang, Z. Ren, and N. Li, "Zeroth-order feedback optimization for cooperative multi-agent systems," in *2020 59th IEEE Conference on Decision and Control (CDC)*, pp. 3649–3656, 2020.
- [13] Y. Tang, Z. Ren, and N. Li, "Zeroth-order feedback optimization for cooperative multi-agent systems," *Automatica*, vol. 148, p. 110741, 2023.
- [14] Y. Shen, Y. Zhang, S. Nivison, Z. I. Bell, and M. M. Zavlanos, "Asynchronous zeroth-order distributed optimization with residual feedback," in *2021 60th IEEE Conference on Decision and Control (CDC)*, pp. 3349–3354, IEEE, 2021.
- [15] D. Drusvyatskiy, M. Fazel, and L. J. Ratliff, "Improved rates for derivative free gradient play in strongly monotone games," in *2022 IEEE 61st Conference on Decision and Control (CDC)*, pp. 3403–3408, IEEE, 2022.
- [16] M. Bravo, D. Leslie, and P. Mertikopoulos, "Bandit learning in concave n-person games," *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [17] S. Liu, P.-Y. Chen, B. Kailkhura, G. Zhang, A. O. Hero III, and P. K. Varshney, "A primer on zeroth-order optimization in signal processing and machine learning: Principals, recent advances, and applications," *IEEE Signal Processing Magazine*, vol. 37, no. 5, pp. 43–54, 2020.
- [18] D. Malik, A. Pananjady, K. Bhatia, K. Khamaru, P. Bartlett, and M. Wainwright, "Derivative-free methods for policy optimization: Guarantees for linear quadratic systems," in *The 22nd international conference on artificial intelligence and statistics*, pp. 2916–2925, PMLR, 2019.
- [19] G. Strang, *Linear algebra and its applications*. Belmont, CA: Thomson, Brooks/Cole, 2006.
- [20] R. Durrett, *Probability: theory and examples*, vol. 49. Cambridge university press, 2019.
- [21] A. Beck, *First-order methods in optimization*. SIAM, 2017.
- [22] H. J. Keisler, *Elementary calculus: An infinitesimal approach*. Courier Corporation, 2013.
- [23] G. B. Folland, *Real analysis: modern techniques and their applications*, vol. 40. John Wiley & Sons, 1999.
- [24] H. Reisizadeh, B. Touri, and S. Mohajer, "Dimix: Diminishing mixing for sloppy agents," *arXiv preprint arXiv:2204.10974*, 2022.