Communication-Efficient Zeroth-Order Distributed Online Optimization: Algorithm, Theory, and Applications

Ege C. Kaya, M. Berk Sahin*, and Abolfazl Hashemi*

School of Electrical and Computer Engineering, Purdue University

ABSTRACT

This paper focuses on a multi-agent zeroth-order online optimization problem in a federated learning setting for target tracking. The agents only sense their current distances to their targets and aim to maintain a minimum safe distance from each other to prevent collisions. The coordination among the agents and dissemination of collision-prevention information is managed by a central server using the federated learning paradigm. The proposed formulation leads to an instance of distributed online nonconvex optimization problem that is solved via a group of communication-constrained agents. To deal with the communication limitations of the agents, an error feedback-based compression scheme is utilized for agent-to-server communication. The proposed algorithm is analyzed theoretically for the general class of distributed online nonconvex optimization problems. We provide non-asymptotic convergence rates that show the dominant term is independent of the characteristics of the compression scheme. Our theoretical results feature a new approach that employs significantly more relaxed assumptions in comparison to standard literature. The performance of the proposed solution is further analyzed numerically in terms of tracking errors and collisions between agents in two relevant applications.

***Index Terms*—** communication efficiency, compression schemes, federated learning, online optimization, zeroth-order optimization

1. INTRODUCTION

As datasets and machine learning (ML) models continue to grow in size and complexity, training ML models increasingly requires carrying out the optimization process across multiple devices. This is often the result of parallel processing needs or the collaboration of multiple participants in the data acquisition and optimization processes. The federated learning (FL) paradigm [1, 2] addresses this by focusing on the latter scenario and training a global model through the cooperation of multiple clients (or agents), managed by a central server. However, FL is typically carried out by a large number of *communication-constrained* agents, making the transmission of model parameters to the central server a potential bottleneck that needs to be addressed for efficient model training.

In online learning (OL), where decisions are made in real-time with limited information/feedback provided to the decision maker, limited communication resources become even a more severe problem. To address this, first-order FL algorithms like local stochastic gradient descent (SGD) use compression techniques like quantization or sparsification [3–5] to reduce the size of local gradients before transmission, but this causes information loss which may impact the learning performance adversely.

To counteract this loss in information, an error feedback (EF) mechanism can be added. The EF mechanism works by incorporating the error made by compression in the subsequent steps, so that effectively, each gradient is fully utilized, even if at later stages. Moreover, the EF mechanism theoretically achieves the same rate of convergence as the no-compression case, making compression come at no cost. [6].

An additional consideration that we may need to have in practical scenarios is the potentially limited nature of available information. The zeroth-order (ZO) optimization setting presents an example for such limitations. In an optimization problem arising from a real-life scenario, the information to be used in the optimization process may be the sensed values of physical quantities such as sound or light intensity, or relative distance [7]. For instance, assuming that sensing agents may only sense current distances to their targets and other nearby agents, we can consider this to be a ZO setting [8] as agents do not have access to higher-order information, such as velocity or acceleration.

As an example of a practical scenario combining all of the aforementioned considerations, consider delivery robots that are loaded from the same region and aim to find their customers. This situation may be viewed as a *source localization* problem with multiple mobile agents. We adopt the terms *agent* and *source* from the literature on this subject in the upcoming discussion. If the customers are also moving, this

*The first two authors contributed equally to the manuscript.

Ege C. Kaya, M. Berk Sahin, and Abolfazl Hashemi are with College of Engineering, Purdue University, West Lafayette, IN 47907, USA. Emails: {kayae, sahinm, abolfazl}@purdue.edu

becomes a target-tracking problem [9, 10]. In a multi-agent setting, collisions between these delivery robots may occur, which can be solved by establishing communication between the agents using the FL framework, somehow incorporating the information of where nearby agents are. Supposing additionally that the robots are only capable of sensing their current distances to their respective targets and to other nearby robots moves our problem into the field of ZO optimization. However, doing so would also result in an online optimization scenario, seeing as the relative locations of the robots with respect to one another would be continually changing, producing a time-varying sequence of optimization problems to solve. Finally, to overcome the inherent communication bottleneck engendered by the online and FL settings, compression schemes may be used along with the EF mechanism. Our novel formulation of this target tracking problem is illustrated and explained in detail in Section 4.1.

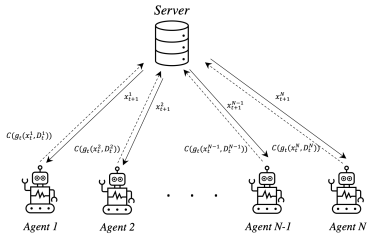


Fig. 1: Illustration of agent-server communication. The agents communicate compressed information to the server, whereas the server transmits back the full information.

1.1. Contribution

Motivated by the previous problem formulation, the purpose of this work is to find an answer to the central question:

Is it possible to devise an algorithm for online, distributed non-convex optimization problems with compressed exchange of zeroth-order information, and with provable convergence guarantees for both single-agent and multi-agent settings?

To address this question, we focus on a general stochastic nonconvex optimization problem, taking into account the following factors: **i)** access to the stochastic cost function is limited to zeroth-order oracle, meaning only function values at current locations and times are available, **ii)** due to communication constraints, only compressed or quantized gradients are exchanged between the agents and the server, **iii)** multiple agents use zeroth-order information to track their targets, and **iv)** the objective functions are time-varying in nature, resulting in an online optimization problem.

We prove the existence of a first-order solution in $\mathbb{R}^d$ that is ξ-accurate with $T = \mathcal{O}\left(\frac{d\sigma^2 ML(\Delta+\bar{\omega})}{\xi^2}\right)$ in the dominant term, where σ^2, L, M, Δ, and $\bar{\omega}$ denote the variance of stochastic gradients, smoothness constant in Assumption 3, bound constant on the stochastic gradients' second moment in Assumption 2, the difference between averages of loss functions for the first and last iterates, and the summation of drift bounds from Assumption 4 respectively. Hence the dominant term in the convergence error is not dependent on the compression ratio. This is achieved while using an EF mechanism and a ZO gradient estimator which uses two function evaluations. In the derivation of this result, we also relax the assumption of bounded second moment commonly found in related literature [6]. Instead of assuming that the second moment of the stochastic gradients are upper-bounded by a constant term greater than or equal to their variance, we adopt the relaxed assumption that it is upper-bounded by the variance plus a term that is proportional to the square of its expected value. In other words, we relax the assumptions on the value of M in Assumption 2, whereas it is commonly assumed in other literature that $M = 0$, uniformly. That is, our upper bound depends on the current sample rather than a uniform bound. Whereas the previous work deals with a single-agent scenario [11], we examine the effectiveness of the proposed approach in a multi-agent target tracking scenario with limited communication where collision avoidance is of paramount importance. The problem of reducing collisions among agents is addressed by incorporating the FL paradigm and a new regularization term. This task is formulated as an online, distributed nonconvex optimization problem that can be solved by a multi-agent variation of the proposed scheme. Theoretical analysis shows that a ξ-accurate first-order solution in $\mathbb{R}^{Nd}$ with $T = \mathcal{O}\left(\dfrac{\sigma^2 dMQ\left(\Delta^2+\bar{\omega}^2\right)+M\left(\sigma^2+Z^4\right)}{\xi^2}\right)$ in the dominant term can be found in a scenario with N agents, where Z^2 and Q are constants that arise from Assumption 6, which effectively places a bound on the norm of the gradients of each client in terms of the average of these gradients over all of the clients [12]. The results of the study are further supported by experimental results

Our preliminary work on single-agent convergence analysis and experiments was accepted to and will be presented at 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing [13]. The current work presents a significantly more thorough analysis of the subject, with an additional part detailing the multi-agent algorithm and its analysis, presented in Theorem 2. The complete proofs of the two theorems are also provided. The experimental section is ameliorated with more descriptive results, and an additional experiment involving an area coverage problem.

1.2. Related Work

Communication-efficient FL. FedAvg is the seminal FL paper in which the central server takes the average of the local gradients transmitted by the clients and distributes the updated parameters to corresponding clients [1]. The crux of this work is the locality of data, in that data is acquired and trained on locally by a multitude of clients, without ever transporting it to a central server. Several difficulties, including privacy concerns [14], heterogeneity of client data [15], and high communication costs in agent-to-server links [16] arise in relation to this paradigm. Variations of FedAvg have been developed to mitigate these problems. For instance, to deal with high communication costs, [17, 18] propose a sparsification algorithm in communication for time-varying decentralized learning and optimization. Reference [19] proposes utilizing adaptive learning rate for aggregation, which is relevant to both the client data heterogeneity and communication efficiency issues. Reference [20] suggests using a novel aggregation technique which first quantizes gradients, and then skips communicating less impactful quantized gradients in favor of reusing previous ones. Reference [21] proposes using a momentum-based global update at the server, which promotes communication efficiency through variance reduction. Reference [22] proposes a derivative-free federated ZO optimization (FedZO) algorithm, and to improve its communication efficiency over wireless networks, they propose an over-air computation assisted variant. Reference [23] proposes a multiple local update strategy and a decentralized ZO algorithm to improve the communication efficiency and convergence rate in the decentralized FL scenario, in which there is no access to first-order derivatives. Reference [24] promotes the use of multiple gossip steps for communication efficiency. Various compression schemes such as Top-k [6], Rand-k [5], Biased and Unbiased Dropout-p [4], Quantized SGD (QSGD) [3], and their variants/generalizations are used to achieve the communication efficiency of FL algorithms. Compression schemes can be divided into contractive and non-contractive methods. With contractive compression schemes, which are our focus in this work, it is common to introduce an EF mechanism to compensate for the error due to compression by accumulating compression error in memory and adding it back as feedback for subsequent rounds. In [6], it is shown that such a method used in conjunction with SGD has a comparable rate of convergence to non-compressed SGD. In this study, we relax the assumption of having stochastic-first order oracle with bounded noise required in [6] by meticulously characterizing the impact of such relaxation on convergence. Furthermore, instead of the single-agent case as investigated in [6], we consider multiple agents with the additional contingency of preventing their collisions, which make the theoretical side more challenging.

Multi-agent target tracking. In our setting, agents are limited to ZO information, since they are assumed to only be able to sense their distance to their targets and other nearby agents. As a result of this consideration, our method is applicable to different practical scenarios such as [25, 26]. In these kinds of scenarios, gradients of the loss function can still be estimated by finite differences [27] but doing so in a multi-agent setting under communication constraints still remains an open challenge. Reference [11] describes a setup comparable to online optimization employing ZO oracles, applied to a target tracking problem. In that work, the authors focus on the case where there is a single source pursued by a single agent, which we generalize to the multi-agent setting as part of our contribution. We further investigate an effective approach via nonconvex regularization for collision avoidance. The λ parameter we refer to as the *regularization parameter* is in essence similar to the penalty and augmented Lagrangian methods used in functional constrained optimization [28, 29]. However, these methods aim to adaptively tune the λ parameter on-the-go, which is out of the scope of our work. It should be noted also that this line of research is very relevant to the area of *safe reinforcement learning*, see e.g. [30–33].

In [34], a cooperative, mobile multi-agent source localization problem is tackled via using a distributed algorithm. Compared to our setting, the agents sense first-order information and their neighboring agents benefit from collaboration between agents to avoid collision. Reference [35] deals with a source localization problem in a single-agent and single-source setting, where the source is stationary or near-stationary. Reference [36] studies the problem of OL using ZO information with convex cost functions. They extend the problem out of the conventional Euclidean setting onto Riemannian manifolds. In [37] a ZO source localization problem is considered using distance information, where the agents are essentially multiple sensors of known position. References [38] and [39] deal with an online optimization problem using a decentralized network of multiple agents which have access to ZO information, and propose the usage of local information and information from neighboring agents in the network. In [38], an iterative algorithm with guarantees is proposed for time-varying online loss functions. The theoretical result there is established by assuming a certain *bounded drift in time* assumption which is standard in the literature, see, e.g., [11].

Online optimization and online target tracking. In the general online optimization setting, we focus on literature within or similar to the online convex optimization framework, which we can consider as a sequential decision making game in the presence of time-varying loss functions [40]. In [41], a distributed online optimization problem with multiple agents is considered, and the local loss function of each agent is convex and time-varying. The authors propose a randomized gradient-free distributed projected gradient descent procedure, where agents estimate the gradient of their local loss functions in a random direction using information from a locally-built ZO oracle. In [42], a similar setting is considered, and a multi-agent distributed optimization problem is studied in continuous-time, with time-varying convex loss functions. Reference [11] deals with a setting where zeroth-order oracles are used for optimization in the presence of time-varying cost functions. Besides the general online optimization setting, there is an abundance of literature focusing on the online optimization aspect of the target tracking problem. A large number of them also involve a swarm of multiple agents working in coordination. Usually, literature on this area tends to consider the problem in the context of unmanned aerial vehicles (UAV), or unmanned surface vehicles (USV). As pointed out in [43], the approaches to the problem may be separated into three broad categories: those using *filtering based, control theory based and machine learning based* approaches. For instance, [43] examines the problem within the domain of reinforcement learning by formulating it as a constrained Markov decision process, with application to autonomous target tracking using a swarm of UAVs. The authors provide an algorithm with provable guarantees. In [44], the authors again consider a multi-agent multi-target pursuit evasion scenario, where they propose the usage of a recurrent neural net work for target trajectory prediction, in conjunction with a multi-agent deep deterministic policy gradient formulation for decision making. Reference [45] deals with a robust formulation of a similar scenario in the domain of supervised learning, using a game theoretic approach. Reference [46] deals with a multi-target following scenario with consideration of external threats. The authors treat the problem as an online path planning problem and adopt a control-theoretic approach. In [47], an online adaptive Kalman filter is used in a target tracking problem where the sensor signals of the agents are assumed to have unknown noise statistics, to formulate a solution that is robust to noise. Lastly, [48] considers a decentralized control problem involving multiple agents with multiple control objectives, among which target tracking is one. The authors make use of a scheme based on adaptive dynamic programming, and feedback from a critic neural network which approximates the control objectives in online fashion.

1.3. Novelty w.r.t. Existing Works

Our work is focused on a nonconvex online distributed optimization problem with compressed exchange of zeroth-order information, along with the error feedback mechanism. Although these concepts were investigated individually in prior works [6, 11, 22, 27, 34] to the best of our knowledge, we are the first to combine them in a single framework and propose an algorithm with its theoretical analysis and convergence guarantee.

Furthermore, we may compare our theoretical results in Section 3 with the result of the analysis in [49], which derives an upper bound for the offline, first-order case with contractive compressors and the error feedback mechanism for the optimization of a smooth, nonconvex function in the FL paradigm. Their result establishes, in this setting, an iteration complexity of $\mathcal{O}(1/N\xi^2)$ to produce a $\xi-$accurate first-order solution. Our result agrees with this result on the $\xi-$dependency. However, the analysis in this result is obtained in an offline setting with access to the first-order derivatives and is hence able to derive a convergence rate that is inversely proportional to the number of agents N. As opposed to that setting, we consider a more challenging online setting in which agents lack access to first-order derivatives but have access to finite differences. Thus, the convergence rate we establish is independent of N. Moreover, reference [49] assumes a uniform bound on the second moment of the gradients. We adopt a more relaxed version of this assumption, which does not assume a uniform bound on the second moment (Assumption 2), and this makes our analysis more involved. We make a similar relaxation of a standard assumption used in distributed optimization [50–56] in the same vein, by lifting the assumption of a uniform bound (Assumption 6).

The rest of the paper is organized as follows: In Section 2, we present the related background on stochastic gradient descent in zeroth-order oracle setting and necessary assumptions for our theoretical analysis. In Section 3, we propose the EF-ZO-SGD and FED-EF-ZO-SGD algorithms and present two theorems for their convergence along with sketches of proofs. Experimental results for two different settings are presented in Section 4 followed by the conclusion in Section 5. The complete proofs of the theorems along with the statements of relevant lemmas are presented in the Appendix.

2. PRELIMINARIES AND BACKGROUND

We start by providing a description of the problem in the single-agent setting. We deal with a sequence of time-varying optimization problems: $\min_{x\in\mathbb{R}^d} \ell_t(x)$, $t \in \mathbb{Z}^+$. Each $\ell_t : \mathbb{R}^d \to \mathbb{R}$ is a continuous loss function and $\ell_t(x) := \mathbb{E}_z[\tilde{\ell}_t(x)]$. We denote $\tilde{\ell}_t(x) := \ell_t(x, z)$ where z is a random variable representing data points coming from the unknown distribution P_z, so $z \sim P_z$. In our application, the target tracking problem, it is the position vector of targets. We aim to find a sequence of solutions $\{x_t\}_{t=1}^T$ such that $\frac{1}{T}\sum_{t=1}^T \|\nabla \ell_t(x_t)\|^2 \leq \xi$ for some small $\xi > 0$. Suppose that, at time t, we have somehow generated a (possibly non-optimal) solution x_t to the problem $\min_{x\in\mathbb{R}^d} \ell_t(x)$. As we are motivated by online and time-critical missions, we would like to generate a solution x_{t+1} to the problem $\min_{x\in\mathbb{R}^d} \ell_{t+1}(x)$ applying a simple update rule which is similar to SGD to x_t:

$$x_{t+1} = x_t - \eta_t \nabla \tilde{\ell}_t(x_t), \tag{1}$$

where η_t is the step size or learning rate adopted at time t. As discussed in Section 1, we cannot directly apply such an update since we are in the ZO setting, that is, we only have access to evaluation of $\tilde{\ell}_t$ and not to its gradient or stochastic gradient. To overcome this limitation, we resort to a ZO estimator of the gradient:

$$\tilde{g}_{\mu,t}(x_t) := \frac{\tilde{\ell}_t(x_t + \mu u_t) - \tilde{\ell}_t(x_t)}{\mu} u_t, \tag{2}$$

where $\mu \in \mathbb{R}$ is the so-called smoothing parameter, and each $u_t \sim \mathcal{N}(0, I_d)$. Note that $\tilde{g}_{\mu,t}$ can be thought of as an approximation to the stochastic gradient of a Gaussian smoothing of $\tilde{\ell}_t$, i.e., $\tilde{\ell}_{\mu,t}(x) := \mathbb{E}_u[\tilde{\ell}_t(x + \mu u)]$. A final modification to the update rule arises due to the aforementioned communication constraints. We apply compression to the ZO estimator and use the resulting quantity in the update rule. To mitigate the negative effect of compression on the convergence of the method, we employ the error feedback mechanism. Essentially, this serves in each time step to partially recover information discarded in the previous compression steps. The details of our approach may be seen in EF-ZO-SGD.

In the multi-agent setting, we generalize the problem as follows: There are now N sequences of continuous loss functions where $t \in \mathbb{Z}^+$ and each $\ell_t : \mathbb{R}^{Nd} \to \mathbb{R}$, which we denote $\ell_t^1, \dots, \ell_t^N$, belonging to agents 1 through N. Similar to the previous part, $\ell_t^i(x) := \mathbb{E}_z[\tilde{\ell}_t^i(x)]$, $\tilde{\ell}_t^i(x) := \ell_t^i(x, z)$ and $z \sim P_z$. We name these the *local loss functions*, since they represent the loss of each specific agent. The objective is to find a sequence of solutions $\{x_t^{1:N}\}_{t=1}^T \subset \mathbb{R}^{Nd}$ that minimizes the *global loss function* $\tilde{\bar{\ell}}_t = \frac{1}{N}\sum_{i=1}^N \tilde{\ell}_t^i$. Akin to the single-agent setting, each agent computes a compressed version of the ZO estimator, corrected to some extent by feedback of the error generated due to compression in the previous steps. The result of this computation is then transmitted to the central server, where they are aggregated and used to update the locations of each agent. The full algorithm entailed by this approach can be seen in FED-EF-ZO-SGD.

Next, we state the assumptions adopted in the forthcoming analyses of the single- and multi-agent settings.

Assumption 1. *(Unbiased Stochastic Zeroth-Order Oracle) For any $t \in \mathbb{Z}^+$, $i \in \{1, \dots, N\}$ and $x \in \mathbb{R}^d$, we have*

$$\mathbb{E}_z\left[\tilde{\ell}_t^i(x)\right] = \ell_t^i(x). \tag{3}$$

Although we do not explicitly utilize the stochastic gradient $\nabla\tilde{\ell}_t$ in the forthcoming algorithm, our analysis still requires a certain regulatory assumption on it.

Assumption 2. *(Bounded Stochastic Gradients) For any $t \in \mathbb{Z}^+$, $i \in \{1, \dots, N\}$ and $x \in \mathbb{R}^d$, there exist $\sigma, M > 0$ such that*

$$\mathbb{E}_z\left[\|\nabla\tilde{\ell}_t^i(x)\|^2\right] \le \sigma^2 + M\|\nabla\ell_t^i(x)\|^2. \tag{4}$$

We note that this assumption is significantly more relaxed compared to the assumption typically used in stochastic optimization [57] and EF-based compression [6]. In particular, [6] requires $M = 0$ which effectively imposes a uniform bound on the gradient of ℓ_t. As part of our contribution, we carry out the analysis under the relaxed assumption stated above.

Assumption 3. *(L-smoothness) Each $\tilde{\ell}_t^i(x)$ is continuously differentiable and L-smooth over x on $\mathbb{R}^d$, that is, there exists an $L \ge 0$ such that for all $x, y \in \mathbb{R}^d$, $t \in \mathbb{Z}^+$ and $i \in \{1, \dots, N\}$, we have*

$$\|\nabla\tilde{\ell}_t^i(x) - \nabla\tilde{\ell}_t^i(y)\| \le L\|x - y\|. \tag{5}$$

We denote this by $\tilde{\ell}_t^i(x) \in C_L^{1,1}(\mathbb{R}^d)$. Note that this assumption implies $\ell_t^i(x) \in C_L^{1,1}(\mathbb{R}^d)$.

Assumption 4. *(Bounded Drift in Time) There exist N bounded sequences $\{\omega_t^1\}_{t=1}^T, \dots, \{\omega_t^N\}_{t=1}^T$ such that for all $t \in \mathbb{Z}^+$ and $i \in \{1, \dots, N\}$, $|\ell_t^i(x) - \ell_{t+1}^i(x)| \le \omega_t^i$ for any $x \in \mathbb{R}^d$. Note that in the case where $\ell_{t+1}^i = \ell_t^i$, this assumption holds with $\omega_t^i = 0$.*

Assumption 4 is standard in the literature on time-varying optimization [11, 58]. Since we work in the online optimization setting where our loss function is time-varying, this assumption upper-bounds the change in the loss function uniformly with a different constant value at each time step.

The next assumption has to do with the aforementioned compression of the gradient estimator $g_{\mu,t}$. We assume that the schemes used for the compression satisfy the following assumption.

Assumption 5. *(Contractive Compression [6]) The compression function $\mathcal{C}$ is a contraction mapping, that is,*

$$\mathbb{E}_{\mathcal{C}}\left[\|\mathcal{C}(x) - x\|^2 \mid x\right] \le (1 - \delta)\,\|x\|^2 \tag{6}$$

for all $x \in \mathbb{R}^d$ where $0 < \delta \le 1$, and the expectation is over the randomness generated by compression $\mathcal{C}$.

One can see that δ effectively controls the scale of the compression. $\delta = 1$ corresponds to the case of no compression and the amount of compression increases as $\delta \to 0$.

The compression operators we use in the numerical experiments are as follows:

- top_k: We fix a parameter $k \in \{0, \dots, d\}$. $\mathrm{top}_k : \mathbb{R}^d \to \mathbb{R}^d$ is defined as:

$$(\mathrm{top}_k(x))_i := \begin{cases} (x)_{\pi(i)} & i \le k, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

  where $\pi(i)$ is a permutation of $\{1, \dots, d\}$ such that $(|x|)_{\pi(i)} \ge (|x|)_{\pi(i+1)}$ for every $i \in \{1, \dots, d-1\}$ [5]. In other words, top_k preserves the k elements of x that are largest in magnitude, and assigns 0 to the rest.

- rand_k: We fix a parameter $k \in \{0, \dots, d\}$. $\mathrm{rand}_k : \mathbb{R}^d \times \Omega_k \to \mathbb{R}^d$ is defined as:

$$(\mathrm{rand}_k(x, \omega_0))_i := \begin{cases} x_i & i \in \omega_0, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

  where $\Omega_k = \{\omega : \omega \subseteq \{1, \dots, d\}, |\omega| = k\}$ and ω_0 is chosen uniformly at random from Ω_k [5]. In other words, rand_k preserves k random elements of x, and assigns 0 to the rest.

- dropout-b$_p$: We fix a parameter $p \in [0,1]$. dropout-b$_p$: $\mathbb{R}^d \to \mathbb{R}^d$ is defined as:

$$(\text{dropout-b}_p(x))_i := \begin{cases} (x)_i & u_i \le p, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

  where each $u_i \sim U[0,1]$. Note that $\text{dropout-b}_p(x)$ is a biased estimator of x.

- dropout-u$_p$: We fix a parameter $p \in [0,1]$. dropout-u$_p$: $\mathbb{R}^d \to \mathbb{R}^d$ is defined as:

$$(\text{dropout-u}_p(x))_i := \begin{cases} \frac{1}{p}(x)_i & u_i \le p, \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

  where each $u_i \sim U[0,1]$. Note that $\text{dropout-u}_p(x)$ is an unbiased estimator of x.

- qsgd$_b$: We fix a parameter $b \in \mathbb{N}$ and perform b-bit random quantization (where 2^b is the quantization level):

$$\text{qsgd}_b(x) = \frac{\text{sign}(x)\|x\|_2}{2^b w}\left[2^b \frac{|x|}{\|x\|_2} + u\right] \tag{11}$$

  where $w = 1 + \min(\sqrt{d}/2^b, d/2^{2b})$, $u \sim (U[0,1])^d$, and $\text{qsgd}_b(0) = 0$ [3].

It is worth noting that all of these compression schemes respect Assumption 5, with the sole exception of dropout-u$_p$.

Our final assumption concerns only the analysis of the multi-agent case:

Assumption 6. *(Bounded Gradients) For any $x_t^{1:N} \in \mathbb{R}^{Nd}$, there exist $Z, Q > 0$ such that*

$$\mathbb{E}_z\left[\|\nabla \ell_t^i(x_t^{1:N})\|^2\right] \le Z^2 + Q\|\nabla \bar{\ell}_t(x_t^{1:N})\|^2 \tag{12}$$

for all $i \in \{1,\dots,N\}$, where $\nabla \bar{\ell}_t(x_t^{1:N}) = \frac{1}{N}\sum_{i=1}^N \nabla \ell_t^i(x_t^{1:N})$.

We note that this is a relaxation of the standard assumption capturing the effect of data heterogeneity, commonly employed in the analyses of decentralized optimization algorithms [12,59,60] and in the analysis of FedAvg-like methods in particular [50–56]. The standard assumption poses a uniform bound: $\mathbb{E}_{z_{1:T}}\left[\|\nabla \ell_t^i(x_t^{1:N}) - \nabla \bar{\ell}_t(x_t^{1:N})\|^2\right] \le Z^2$. In [61], it is argued that this form usually holds in practice, and may even be considered too pessimistic. However, one can easily come up with a counterexample where it does not, e.g., with $\ell_t^i(x) = (ix)^2$ for all $t \in \mathbb{Z}^+$. We note that this relaxation of the assumption is akin to the one adopted with Assumption 2.

3. PROPOSED METHOD

In this section, we present our EF-ZO-SGD and FED-EF-ZO-SGD algorithms along with their convergence results and provide sketches of the proofs for these results. The complete proofs may be found in Appendix.

3.1. EF-ZO-SGD

We now present EF-ZO-SGD, an algorithm which uses compression along with the EF mechanism in addition to the ZO estimator in (2) to achieve a communication-efficient method of approaching the presented problem in the single-agent scenario. The complete algorithm is demonstrated in EF-ZO-SGD. Given an initial solution $x_0 \in \mathbb{R}^d$, which for our problem represents the initial position of the agent, the algorithm works iteratively to construct subsequent solutions to the sequence of optimization problems. It first samples a random vector in $\mathbb{R}^d$ from the standard Gaussian distribution and uses this to construct a ZO estimator to the gradient (steps 3 and 4). Then, the error feedback vector, which keeps track of information discarded during compression in previous communication rounds (step 7) is added to this ZO estimator to produce the augmented estimator (step 5). In this manner, information previously lost to compression is re-utilized. The augmented estimator is the quantity used in the update rule to produce the subsequent solution (step 6), and it is further used to update the error feedback vector (step 7). This process is repeated for $t = 1, ..., T$ to produce solutions to all terms of the sequence of optimization problems.

Algorithm 1 EF-ZO-SGD

Input: Number of time steps $T \in \mathbb{Z}^+$, smoothing parameter $\mu \in \mathbb{R}$, initial agent position $x_0 \in \mathbb{R}^d$, learning rate $\eta \in \mathbb{R}$, sequence of target positions $\{z_t\}_{t=1}^T \subset \mathbb{R}^d$.

Output: Sequence of optimal agent positions $\{x_t\}_{t=1}^T \subset \mathbb{R}^d$.

```
1: $e_0 = 0$
2: for $t = 1, \dots, T$ do
3:     $u_t \sim \mathcal{N}(0, I_d)$
4:     $\tilde{g}_{\mu,t}(x_t) = \frac{\tilde{\ell}_t(x_t + \mu u_t) - \tilde{\ell}_t(x_t)}{\mu} u_t$
5:     $p_t = \tilde{g}_{\mu,t}(x_t) + e_t$
6:     $x_{t+1} = x_t - \eta \mathcal{C}(p_t)$
7:     $e_{t+1} = p_t - \mathcal{C}(p_t)$
8: end for
```

The convergence properties of EF-ZO-SGD are analyzed next. For the convergence of EF-ZO-SGD in a single-agent setting, we establish Theorem 1.

Note that although the EF-ZO-SGD algorithm can be thought of as a SGD-type scheme, the analysis – due the interaction of EF and ZO estimation – is involved. In the proof, we leverage a new *intertwined perturbation analysis*, wherein we analyze the convergence of a virtual solution sequence to the smoothed functions $\ell_{\mu,t}$ and tie that to the performance of the real iterates x_t to ℓ_t, while utilizing the relaxed bounded stochastic gradient assumption.

Theorem 1. *Suppose Assumptions 1–2 hold. Consider EF-ZO-SGD algorithm. Then, if $\eta = \dfrac{1}{\sigma\sqrt{(d+4)MTL}}$ and $\mu =$*

$\dfrac{1}{(d+4)\sqrt{T}}$, *it holds that*

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\ell_t(x_t)\|^2\right] \le{}& \frac{8\Delta\sigma(d+4)^{\frac{1}{2}}M^{\frac{1}{2}}L^{\frac{1}{2}}}{T^{\frac{1}{2}}}\\ &+\frac{8\sigma dL^{\frac{3}{2}}M^{\frac{1}{2}}}{T^{\frac{3}{2}}(d+3)^{\frac{3}{2}}}+\frac{2(d+6)^{\frac{3}{2}}L^{\frac{5}{2}}}{\sigma(d+4)^{\frac{5}{2}}T^{\frac{3}{2}}M^{\frac{1}{2}}}+\frac{8\sigma(d+4)^{\frac{1}{2}}L^{\frac{1}{2}}}{M^{\frac{1}{2}}T^{\frac{1}{2}}}\\ &+\frac{(d+3)^3L^2}{(d+2)^2T}+\frac{32L}{\delta^2\sigma^2MT}+\frac{8(d+6)^3L^3}{\delta^2\sigma^2(d+4)^3MT^2}\\ &+\frac{8\bar{\omega}\sigma(d+4)^{\frac{1}{2}}M^{\frac{1}{2}}L^{\frac{1}{2}}}{T^{\frac{1}{2}}},\end{aligned} \tag{13}$$

where $x^*_{T+1}\in\operatorname{argmin}_{x\in\mathbb{R}^d}\ell_{T+1}(x)$, $\Delta=\ell_1(x_1)-\ell_{T+1}(x^*_{T+1})$, $\bar{\omega}=\sum_{t=1}^{T}\omega_t$, *and* $\mathbb{E}[\,\cdot\,]$ *denotes* $\mathbb{E}_{z_{1:T}}[\,\cdot\,]$. *Furthermore, the number of time steps* T *to obtain a* ξ*-accurate first order solution is*

$$T=\mathcal{O}\left(\frac{d\sigma^2L\Delta M}{\xi^2}+\frac{dL\Delta}{\delta^2\xi}+\frac{\bar{\omega}\sigma^2dML}{\xi^2}\right). \tag{14}$$

Sketch of proof. We begin by defining the perturbed quantity $\tilde{x}_t:=x_t-\eta e_t$. Then, using assumptions 3 and 4, we obtain the inequality

$$\begin{aligned}\ell_{\mu,t+1}(\tilde{x}_{t+1})\le \ell_{\mu,t}(\tilde{x}_t)-\eta\langle\tilde{g}_{\mu,t}(x_t),\nabla\ell_{\mu,t}(\tilde{x}_t)\rangle\\ +\frac{L\eta^2}{2}\|\tilde{g}_{\mu,t}(x_t)\|^2+\omega_t.\end{aligned} \tag{15}$$

Taking expectations and performing algebraic manipulations produce the main inequality with the four terms:

$$\begin{aligned}\underbrace{\frac{\eta}{2}\|\nabla\ell_{\mu,t}(x_t)\|^2}_{\text{Term I}}\le \underbrace{\left[\ell_{\mu,t}(\tilde{x}_t)-\ell_{\mu,t+1}(\tilde{x}_{t+1})\right]}_{\text{Term II}}+\\ \underbrace{\frac{L\eta^2}{2}\mathbb{E}_{u_t,z_t}\left[\|\tilde{g}_{\mu,t}(x_t)\|^2\right]}_{\text{Term III}}+\underbrace{\frac{L^2\eta^3}{2}\|e_t\|^2}_{\text{Term IV}}+\omega_t.\end{aligned} \tag{16}$$

We can upper-bound Term II by means of a telescoping sum. Then, using assumptions 5 and 2, Term I can be lower-bounded and Terms III and IV can be upper-bounded by quantities involving $\mathbb{E}_{z_{1:T}}[\|\nabla\ell_t(x_t)\|^2]$. Rearranging this, inserting the values for η and μ and introducing ξ to obtain an expression for the time complexity lead directly to the result. The complete proof may be found in the Appendix. ■

We further note that (14) demonstrates that the dominant term in the complexity is independent of the compression parameter δ. Therefore, for long sequences of time-varying optimization problems where T is very large, the contribution of compression to the convergence error is negligible. Also notable is the fact that the complexity scales with dimension d. While this dependence is undesirable, in the worst case, it is unavoidable even without compression as shown in [62].

Algorithm 2 FED-EF-ZO-SGD

```
Input: Number of time steps $T \in \mathbb{Z}^+$, number of agents $N \in \mathbb{Z}^+$, smoothing parameter $\mu \in \mathbb{R}$, initial agent positions $x_0^{1:N} \in \mathbb{R}^{Nd}$, learning rate $\eta \in \mathbb{R}$, sequence of target positions $\{z_t^{1:N}\}_{t=1}^T \subset \mathbb{R}^{Nd}$.
Output: Sequence of optimal target positions $\{x_t^{1:N}\}_{t=1}^T \subset \mathbb{R}^{Nd}$.
for $i = 1, \dots, N$ do
    $e_0^i = 0$
end for
for $t = 1, \dots, T$ do
    Runs on each agent:
    for $i = 1, \dots, N$ do
        $u_t^i \sim \mathcal{N}(0, I_{Nd})$
        $\tilde{g}_{\mu,t}^i(x_t^{1:N}) = \frac{\tilde{\ell}_t^i(x_t^{1:N} + \mu u_t^i) - \tilde{\ell}_t^i(x_t^{1:N})}{\mu} u_t^i$
        $p_t^i = \tilde{g}_{\mu,t}^i(x_t^{1:N}) + e_t^i$
        $e_{t+1}^i = p_t^i - \mathcal{C}(p_t^i)$
        transmit_to_server $(\mathcal{C}(p_t^i))$
    end for
    Runs on the server:
    $\mathcal{G}_t = \frac{1}{N}\sum_{i=1}^N \mathcal{C}(p_t^i)$
    $x_{t+1}^{1:N} = x_t^{1:N} - \eta \mathcal{G}_t$
    transmit_to_clients $(x_{t+1}^{1:N})$
end for
```

We may discuss the implication of our results to the setting of learning parameters of an overparameterized model, e.g., a deep learning predictor. It has been argued, see, e.g. [63, 64], such models typically satisfy a so-called *strong growth condition* which implies $\sigma=0$ in Assumption 2. That is, as the EF-ZO-SGD algorithm converges to a stationary solution, it enters into a virtuous cycle wherein the noise in the stochastic gradient reduces. As our analysis demonstrates, in such settings we can modify η and μ accordingly (in particular set η independent of T) to improve the complexity of the proposed algorithm to $T=\mathcal{O}(\frac{1}{\xi})$.

3.2. FED-EF-ZO-SGD

FED-EF-ZO-SGD algorithm is a generalization of EF-ZO-SGD to multi-agent and multi-target setting. In addition to the compression, EF mechanism, and ZO estimator, agents are coordinated with the central server and their compressed gradients are averaged in the server as in [1]. The complete algorithm is shown in FED-EF-ZO-SGD. Given an initial solution $x_0^{1:N}\in\mathbb{R}^{Nd}$, which in our problem represents the concatenation of the initial position of the agents, the FED-EF-ZO-SGD algorithm works iteratively on both the agent side and the server side to generate the consecutive solutions to the sequence of optimization problems. The agent side is similar to EF-ZO-SGD except for the content of the solution vectors. In our setting, without loss of generality, we consider agents which can sense the position of nearby agents called "neighbors" and merge the position vectors with their current position to obtain $x_t^{1:N}$. Entries that correspond to the other agents which are not neighbors are set to 0. The same algorithm can be implemented for the agents having no knowledge of the nearby agents' positions. For every agent, the algorithm first samples a random vector in $\mathbb{R}^{Nd}$ from the standard Gaussian distribution and the entries that do not correspond to i^{th} agent's position are set to 0 (step 6). Thus, only i^{th} agents position vector is perturbed to approximate the noisy gradient with finite differences (step 7). Steps 8 and 9 are the same as EF-ZO-SGD. Lastly, each agent sends its compressed augmented estimator to the central server. After the server collects all the estimators from every agent, it takes their average (step 12). Then, this average is used in the update (step 13) and the new positions are transmitted to the agents. This procedure is followed for $t = 1, ..., T$ to produce solutions to all terms of the sequence of optimization problems.

Now, we proceed with the analysis extended to the multi-agent case, which involves FED-EF-ZO-SGD. We state the following theorem:

Theorem 2. *Suppose Assumptions 1–6 hold. Consider FED-EF-ZO-SGD algorithm. Then, if* $\eta = \dfrac{1}{\sigma\sqrt{(d+4)MQTL}}$ *and* $\mu = \dfrac{1}{(d+4)\sqrt{T}}$, *it holds that*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2\right] \leq \frac{8\Delta\sigma\left(d+4\right)^{\frac{1}{2}}M^{\frac{1}{2}}Q^{\frac{1}{2}}L^{\frac{1}{2}}}{T^{\frac{1}{2}}}$$
$$+\frac{8L^{\frac{3}{2}}d\sigma M^{\frac{1}{2}}Q^{\frac{1}{2}}}{\left(d+4\right)^{\frac{3}{2}}T^{\frac{3}{2}}}+\frac{8L^{\frac{1}{2}}\left(d+4\right)^{\frac{1}{2}}M^{\frac{1}{2}}Z^2}{\sigma Q^{\frac{1}{2}}T^{\frac{1}{2}}}$$
$$+\frac{8L^{\frac{1}{2}}\left(d+4\right)^{\frac{1}{2}}\sigma}{M^{\frac{1}{2}}Q^{\frac{1}{2}}T^{\frac{1}{2}}}+\frac{2L^{\frac{5}{2}}\left(d+6\right)^3}{\left(d+4\right)^{\frac{3}{2}}T^{\frac{3}{2}}\sigma M^{\frac{1}{2}}Q^{\frac{1}{2}}}$$
$$+\frac{32LZ^2}{\sigma^2QT\delta^2}+\frac{32L}{MQT\delta^2}+\frac{8L^3\left(d+6\right)^3}{\left(d+4\right)^3T^2\sigma^2MQ}$$
$$+\frac{8\bar{\omega}\sigma\left(d+4\right)^{\frac{1}{2}}M^{\frac{1}{2}}Q^{\frac{1}{2}}L^{\frac{1}{2}}}{T^{\frac{1}{2}}}, \tag{17}$$

where $\bar{\ell}_t(x) = \frac{1}{N}\sum_{i=1}^N \ell_t^i(x)$, $\bar{\omega} := \sum_{t=1}^T \omega_t$, $x_{T+1}^* = \min_{i\in\{1,...,N\}} \operatorname{argmin}_x \ell_{T+1}^i(x)$, $\Delta = \bar{\ell}_1(x_1^{1:N}) - \bar{\ell}_{T+1}(x_{T+1}^*)$, *and* $\mathbb{E}[\,\cdot\,]$ *denotes* $\mathbb{E}_{z_{1:T}^{1:N}}[\,\cdot\,]$. *Furthermore, the number of time steps* T *to obtain a* ξ*-accurate first order solution is*

$$T = \mathcal{O}\left(\frac{\sigma^2dMQ\left(\Delta^2+\bar{\omega}^2\right)+M\left(\sigma^2+Z^4\right)}{\xi^2}+\frac{L^{\frac{5}{3}}}{\xi^{\frac{2}{3}}}+\frac{1}{\delta^2\xi}\right). \tag{18}$$

Sketch of proof. The general outline of the proof is very similar to that of the single-agent case. We define and work with the perturbed quantity $\tilde{x}_t^{1:N} := x_t^{1:N} - \eta\bar{e}_t$, where $\bar{e}_t := \dfrac{1}{N}\sum_{i=1}^N e_t^i$. Additionally, our global loss function in this scenario is $\bar{\tilde{\ell}}_t\left(x_t^{1:N}\right) = \dfrac{1}{N}\sum_{i=1}^N \tilde{\ell}_t^i\left(x_t^{1:N}\right)$. Using Assumptions 3 and 4, we obtain

$$\begin{aligned}\bar{\ell}_{\mu,t+1}\left(\tilde{x}_{t+1}^{1:N}\right) \leq{}& \bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\\ &-\eta\left\langle\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right),\nabla\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\right\rangle\\ &+\frac{L\eta^2}{2}\left\|\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2+\omega_t,\end{aligned} \tag{19}$$

where $\omega_t = \max\{w_t^1, ..., w_t^N\}$. Taking expectations and algebraic manipulations lead to the main inequality with four terms:

$$\underbrace{\frac{\eta}{2}\left\|\nabla\bar{\ell}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2}_{\text{Term I}} \leq \underbrace{\left[\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)-\bar{\ell}_{\mu,t+1}\left(\tilde{x}_{t+1}^{1:N}\right)\right]}_{\text{Term II}}$$
$$\underbrace{+\frac{L\eta^2}{2}\mathbb{E}_{u_t^{1:N},z_t^{1:N}}\left[\left\|\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2\right]}_{\text{Term III}}+\underbrace{\frac{L^2\eta^3}{2}\|\bar{e}_t\|^2}_{\text{Term IV}}+\omega_t. \tag{20}$$

Term II may be upper-bounded by means of a telescoping sum. Term I may be lower-bounded and Terms III and IV upper-bounded by quantities involving $\mathbb{E}_{z_{1:T}^{1:N}}[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2]$, using assumptions 5, 2 and 6. Rearranging this, inserting the values for η and μ and introducing ξ to obtain an expression for the time complexity lead directly to the result. The complete proof may be found in the Appendix. ■

Much like in the single-agent analysis, we note that the dominant term in the complexity is independent of the compression ratio δ.

4. EXPERIMENTAL RESULTS

In this section, we explore two applications of the proposed method to multi-agent target tracking under communication constraints. The first application deals with the main focus of the work, i.e., multi-agent target tracking. The second is an alternative view of the problem involving an area-coverage problem. Our code used for the experiments is available online with the simulation video [65].

4.1. Target Tracking

We begin with the application of our proposed FED-EF-ZO-SGD multi-agent target tracking scenario detailed in the previous sections. In all experiments, we instantiate a central server, N agents $\{\mathcal{A}_i\}_{i=1}^N$ and N sources $\{\mathcal{S}_i\}_{i=1}^N$. The initial location of each agent is chosen uniformly at random from $[-100, 100]^2$ and each source from $[200, 400]^2$. Hence,

$d = 2$, i.e., we consider the target tracking problem on a $2-$dimensional plane, which is reasonable for the motivating example of delivery robots. Also, we instantiate the agents and sources in two separate clusters, with some initial distance between them. Each agent $\mathcal{A}_i$ aims to track source $\mathcal{S}_i$ and each $\mathcal{S}_i$ actively evades its tracker with maximum speed. This setting generalizes that of [11] to a scenario with multiple agents and sources. We use x_t^i, z_t^i to denote the positions of $\mathcal{A}_i$ and $\mathcal{S}_i$ at time step t. Each $\mathcal{S}_i$ aims to maximize its distance to $\mathcal{A}_i$, by setting its velocity at each step to $\zeta_t^i = \beta(z_t^i - x_t^i)/\|z_t^i - x_t^i\|$, i.e., moving directly away from $\mathcal{A}_i$ with speed $\beta = 0.1$. An illustration of the movements of agents and sources is given in Fig. 2.

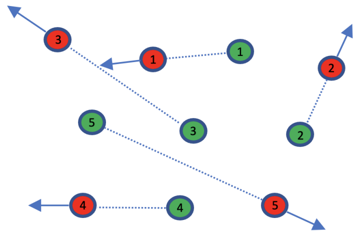


Fig. 2: Illustration of 5 agents tracking 5 sources. Sources evade the agents by moving directly away from them.

As explained in the previous sections, an additional contingency we introduce to the above setting is the requirement of a collision avoidance mechanism to prevent agents from unsafe maneuvers. To this end, we propose a two-level approach: **i)** on the local level within each agent, by means of local neighbor detection leveraging a judicious regularization term, and **ii)** coordination via the FL paradigm. With regard to **i)**, at every step of the simulation, we calculate the set of neighbors of each $\mathcal{A}_i$ as $D_t^i := \{j \neq i : \|x_t^i - x_t^j\| \leq r\}$, where we set $r = 10$. These neighbor sets determine the local loss function ℓ_t^i of $\mathcal{A}_i$ at time step t, which we define as:

$$\ell_t^i(x_t^{1:N}, z_t^i) = \frac{1}{2}\|x_t^i - z_t^i\|^2 - \lambda \sum_{j \in D_t^i} \left(\|x_t^i - x_t^j\|^2 - r^2\right), \tag{21}$$

where $x_t^{1:N} = [(x_t^1)^T \cdots (x_t^N)^T]^T \in \mathbb{R}^{(Nd)}$ and λ is the predetermined *regularization parameter*. We note that the time-varying nature of these neighbor sets introduce time-variance to the loss functions, which is exactly the setting we examine in the theoretical analysis. We divide the local loss function into two terms in order to simplify the notation in the subsequent calculation of the local ZO gradient estimator $g_{\mu,t}^i$:

$$\ell_t^i(x_t^{1:N}, z_t^i) = s_t^i(x_t^i, z_t^i) - \sum_{j \in D_t^i} r_t^{i,j}(x_t^i, x_t^j), \tag{22}$$

where the loss due to source s_t^i is given by $s_t^i(x_t^i, z_t^i) = \frac{1}{2}\|x_t^i - z_t^i\|^2$ and the loss due to regularization between agents $\mathcal{A}_i$ and $\mathcal{A}_j$, $r_t^{i,j}$, by $r_t^{i,j}(x_t^i, x_t^j) = \lambda(\|x_t^i - x_t^j\|^2 - r^2)$. In terms of the scenario, one could see the regularization term as agents being able to sense other agents within a radius r around its position. With regard to **ii)**, collision avoidance is ensured by means of federated aggregation of the local gradient estimators. The global loss function at t is defined as $\bar{\ell}_t(x_t^{1:N}, z_t^{1:N}) = \frac{1}{N}\sum_{i=1}^N \ell_t^i(x_t^{1:N}, z_t^i)$, where $z_t^{1:N}$ is defined similarly to $x_t^{1:N}$.

Defining the neighborhood of two agents $\mathcal{A}_i$ and $\mathcal{A}_j$ in the above manner results in a symmetric relation. To make the setting more interesting, we also introduce the concept of *neighbor dropout* which aims to capture practical considerations such as imperfection in communication links and sensing capabilities. At each t, if $\mathcal{A}_i$ is to be added to $\mathcal{D}_t^j$, a random number X is sampled from $U[0, 1]$. If $X > p$, i is added to $\mathcal{D}_t^j$, otherwise, it is dropped out. This leads to a more realistic scenario and opens up room for more meaningful collaboration between agents by breaking the symmetricity of the relation. If, for example, $\mathcal{A}_i$ is a neighbor of $\mathcal{A}_j$ but fails to detect it, we would expect $\mathcal{A}_j$ to compensate for this. Or worse, if both $\mathcal{A}_i$ and $\mathcal{A}_j$ fail to detect each other, we would expect $\mathcal{A}_k$ such that $k \in \mathcal{D}_t^i \cap \mathcal{D}_t^j$ to compensate for these detection failures. With the local loss function defined in (21), every agent $\mathcal{A}_i$ calculates a ZO gradient estimator $g_{\mu,t}^i$. Following the setup in [11], we slightly modify the computation of the ZO estimator, by introducing a small change in the argument of the first function evaluation. Let

$$\begin{aligned}\ell_{t+}^i(x_t^{1:N}, z_t^i) &:= \frac{1}{2}\|x_t^i + \mu u_t^{i,i} - (z_t^i + 0.5\zeta_t^i)\|^2 \\ -\lambda &\sum_{j \in \mathcal{D}_t^i} \left(\|x_t^i + \mu u_t^{i,j} - (x_t^j + 0.5\xi_t^j)\|^2 - r^2\right),\end{aligned} \tag{23}$$

where $u_t^{i,j}$ for all $j \in \mathcal{D}_t^i$ are drawn from $\mathcal{N}(0, I_d)$ at time t and ζ_t^i, ξ_t^i denote the velocities of agent $\mathcal{A}_i$ and source $\mathcal{S}_i$ at time t, respectively. Similar to ℓ_t^i and (22), we divide ℓ_{t+}^i into two terms:

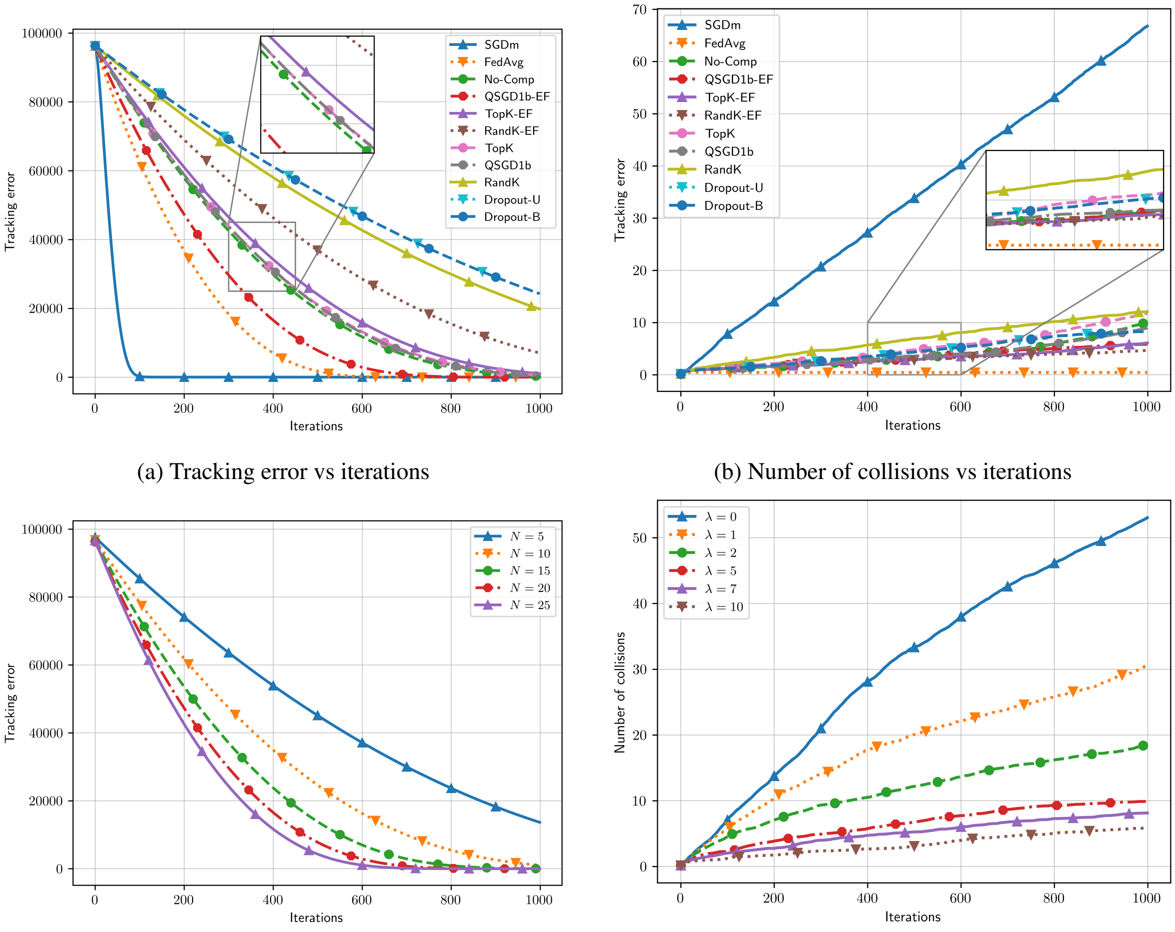
$$\ell_{t+}^i(x_t^{1:N}, z_t^i) = s_{t+}^i(x_t^i, z_t^i) - \sum_{j \in \mathcal{D}_t^i} r_{t+}^{i,j}(x_t^i, x_t^j) \tag{24}$$

where $r_{t+}^{i,j}(x_t^i, x_t^j) = \lambda(\|x_t^i + \mu u_t^{i,j} - (x_t^j + 0.5\xi_t^j)\|^2 - r^2)$ and $s_{t+}^i(x_t^i, z_t^i) = \frac{1}{2}\|x_t^i + \mu u_t^{i,i} - (z_t^i + 0.5\zeta_t^i)\|^2$.

Now, we define $g_{\mu,t}^i = [(g_{\mu,t}^{i,1})^T \cdots (g_{\mu,t}^{i,N})^T]^T \in \mathbb{R}^{(Nd)}$ where

$$g_{\mu,t}^{i,j} = \begin{cases} \dfrac{s_{t+}^i(x_t^i, z_t^i) - s_t^i(x_t^i, z_t^i)}{\mu} u_t^{i,i} & j = i, \\ -\dfrac{r_{t+}^{i,j}(x_t^i, x_t^j, \xi_t^j) - r_t^{i,j}(x_t^i, x_t^j)}{\mu} u_t^{i,j} & j \in \mathcal{D}_t^i, \\ 0 \in \mathbb{R}^d & \text{otherwise.} \end{cases} \tag{25}$$

In practice, it usually holds that for any $i \in \{1, \dots, N\}$, $|\mathcal{D}_t^i| \ll N$, which results in a sparse $g_{\mu,t}^i$. Each agent then



(a) Tracking error vs iterations

(b) Number of collisions vs iterations

(c) Effect of varying N on tracking error

(d) Effect of varying λ on number of collisions

Fig. 3: Results of FED-EF-ZO-SGD via the proposed scheme: (a) shows the average tracking error over 100 runs of the simulation with different compression schemes and EF combinations in the FL paradigm, with SGDm being the non-FL benchmark algorithm, in which SGD with momentum is run locally on each agent with no communication, and FedAvg with $1-$bit QSGD compression and error feedback term being the FL benchmark algorithm. The difference of this benchmark algorithm with FED-EF-ZO-SGD is the fact that FedAvg uses first-order information. (b) shows collision numbers for the same experiment. (c) shows the average tracking errors over 100 runs of the simulation with varying number of agents N, using the best-performing model of QSGD1b-EF (1-bit QSGD with error feedback term). The learning rate η is set proportionally to $\sqrt{N}$. (d) shows the average numbers of collisions over 100 runs of the simulation for varying values of the regularization parameter λ, using the best-performing model of QSGD1b-EF.

transmits its local ZO gradient estimator $g^i_{\mu,t}$ to the server. In scenarios with compression, each agent applies compression before transmission and transmits $\mathcal{C}(g^i_{\mu,t} + e^i_t)$ (see step 10 in FED-EF-ZO-SGD). The possible compression schemes that are used in the experiments are the ones that are detailed in Section 2. The server collects all of the transmitted (and possibly compressed) local gradient estimators and averages them, producing the aggregated global gradient estimator $\mathcal{G}_t$ of $\bar{\ell}_t$: $\mathcal{G}_t = \dfrac{1}{N}\sum_{i=1}^{N}\mathcal{C}(g^i_{\mu,t} + e^i_t)$. Then, to keep the speed of the agents bounded in order to maintain a practically plausible simulation, the server normalizes $\mathcal{G}_t$ and then computes its estimation to the optimal position of every agent by $x^{1:N}_{t+1} = x^{1:N}_t - \eta\mathcal{G}_t$ where η is the learning rate. With this formulation, η determines the speed of the agents in the practical sense, since $\|\mathcal{G}_t\| = 1$, therefore it only plays a role in determining the directions of the agents. The subsequent positions of agents are transmitted to the agents, without compression, and agents move to these positions. This process is illustrated on Fig. 1. To gauge the performance of the model with respect to the number of collisions, we keep track of the number of collisions between agents by checking whether the position of any two agents $\mathcal{A}_i$ and $\mathcal{A}_j$ are close in Euclidean norm, the measure of closeness depends on the radii of the agents in the simulation. In all experiments, we set the collision radius $R = 3$, i.e., we increment the collision counter whenever $\|x_t^i - x_t^j\| \leq 3$ for any two agents $\mathcal{A}_i$ and $\mathcal{A}_j$ such that $i \neq j$.

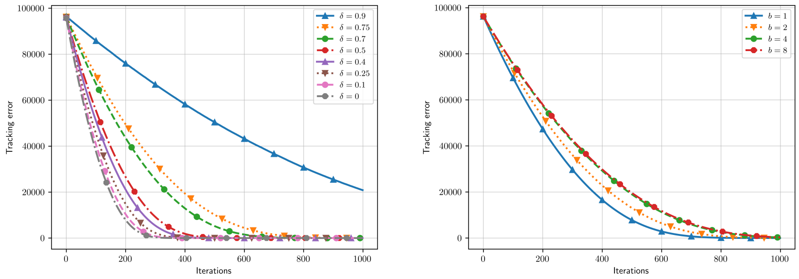
We conduct 3 types of experiments and depict the results on the 4 plots of Fig. 3: In Fig. 3 (a) and Fig. 3 (b) we test the FED-EF-ZO-SGD algorithms' performance in terms of loss and number of collisions with various compression schemes. Fig. 3 (c) compares the convergence of FED-EF-ZO-SGD for different numbers of agents N while scaling the learning rate in proportion with $\sqrt{N}$, since the application bears theoretical resemblance to mini-batch SGD. Fig. 3 (d) demonstrates the effect of varying the regularization parameter λ on the number of collisions. Unless otherwise stated, the parameters used in the experiments are $K = 0.5$ for TopK and RandK, $p = 0.5$ for Dropout, $\eta = 1$, $\beta = 0.1$, $p_N = 0.5$, $d = 2$, $N = 20$, $r = 10$ and $steps = 1000$. 100 instances of the simulation are run for each experiment, with the same fixed random seeds across different methods. In the first experiment, we also run SGD with momentum (SGDm) locally on each agent with no communication, i.e., without the FL paradigm as a benchmark algorithm. Additionally, a benchmark algorithm within the FL paradigm, we also look at the performance of FedAvg with 1-bit QSGD and error feedback mechanism, its key difference from FED-EF-ZO-SGD being that it uses first-order information.

As Fig. 3 (a) demonstrates, the variant that leverages EF along with the QSGD compression scheme with $1-$bit quantization (QSGD1b-EF) enjoys the fastest convergence and even outperforms the setting with no compression (No-Comp). This might be explained by the inherent noise introduced by quantization helping convergence. TopK with error feedback (TopK-EF), $1-$bit QSGD without error feedback (QSGD1b) and TopK without error feedback (TopK) perform virtually on par with the no compression setting. It is interesting to note that TopK seems to slightly outperform TopK-EF. These are followed in performance by RandK with error feedback (RandK-EF), and then RandK without error feedback (RandK). These are finally followed by Unbiased Dropout (Dropout-U) and Biased Dropout (Dropout-B), which perform equally well, but with a large gap to the best performers. It is expected for RandK-EF, RandK, Dropout-U and Dropout-B to take longer to converge, due to the high compression error that they inject in the communicated gradient estimators. Although, it appears that the error feedback helps the convergence of RandK significantly. We note that all of QSGD1b-EF, No-Comp, QSGD1b, TopK and TopK-EF converge within 1000 iterations, with RandK-EF also coming very close. The non-FL benchmark algorithm SGDm outperforms all FL-based methods in terms of iterations needed for convergence, however the rate of convergence appears to be of the same order, and the results are comparable. The first-order FL benchmark algorithm FO-QSGD1b-EF enjoys slightly faster convergence than FED-EF-ZO-SGD, but the performance difference is marginal.

To evaluate the effectiveness of collaboration, we compare the number of collisions vs iterations for the same experiment in Fig. 3 (b). The results show that all of our FL-based methods far outperform the non-FL benchmark method of SGDm in terms of number of collisions. SGDm, which has no regard for collision prevention causes on average about 70 collisions whereas all of our schemes, even the ones that do not achieve good convergence results such as RandK and Dropout-B cause at most about 10 collisions on the average. This demonstrates the efficiency of the proposed *regularization term.* In Fig. 3 (c), we show the results of the second experiment, where we use the best-performing scheme in the first experiment, (QSDGD1b-EF) and test the convergence results with varying numbers of agents N. We make the observation that increasing the number of agents in the described multi-agent scenario is akin to increasing the batch size in mini-batch SGD, as by aggregating the messages received from agents, the server performs an update on the global objective. Thus, motivated by the theoretical studies of mini-batch SGD (see, e.g., [66]), to see the effect of varying the number of agents, we set the η parameter proportional to $\sqrt{N}$. The values we use for N are $5, 10, 15, 20$, and 25, with the respective η values being $0.5, 0.71, 0.87, 1$, and 1.12. The main lines in the plot show the average tracking errors averaged over 100 runs of the simulation. It can be seen that the comparison to mini-batch SGD may be justified, as the model converges roughly around the same iteration for all N values except for 5, when η values are set proportionally.

In Fig. 4, we demonstrate the effect of the compression parameter δ on the convergence of the tracking error. Although the theoretical analysis shows that the dominant term in the convergence bound is independent from the compression parameter δ, the transient behavior of the convergence still depends on δ. This is reflected in the experimental results. In Fig. 4 (a), we consider the effect of varying δ in the Dropout-B compression scheme. Here, δ corresponds to the probability that a gradient component will be dropped. The case of $\delta = 0$ corresponds to when there is no compression. In these experiments, we set the step size $\eta = 3$, to facilitate convergence in the highly compressed regime when $\delta = 0.9$. It can be seen that even in the presence of extreme compression, convergence can be achieved by increasing the step size. Similarly, in In Fig. 4 (b), we consider the effect of varying the number of bits used in QSGD on the convergence of the tracking error. We simulate the experiment with number of bits $1, 2, 4$, and 8 and plot the results.

Finally, in Fig. 3 (d), we compare the effect of the regularization parameter λ on the number of collisions. Similar to the second experiment, we use the best-performing scheme in the first experiment, QSDGD1b-EF. The values tested for λ are $0, 1, 2, 5, 7$, and 10. We observe that, as expected, increasing the λ parameter has a significant effect on decreasing the number of collisions. In the $\lambda = 0$ scenario, which practically



(a) Effect of varying δ on tracking error with Dropout-B compression with error feedback term

(b) Effect of varying number of bits in QSGD compression with error feedback term on tracking error

Fig. 4: Convergence results of FED-EF-ZO-SGD under different compression rates: (a) shows the tracking error over 100 runs of the simulation with different values for δ under the Dropout-B compression scheme with error feedback term. (b) shows the tracking error over 100 runs of the simulation with different numbers of bits used for the QSGD compression scheme with error feedback term.

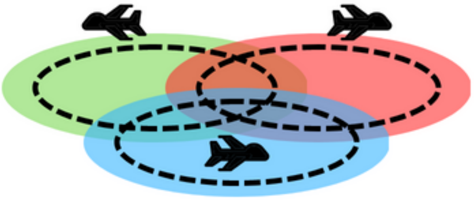


Fig. 5: Illustration of the agents in the area coverage experiment. Each agent has the main objective of patrolling its designated area, following a circular route (indicated with the dashed curves). However, there is overlap between the areas, and the secondary objective is to discourage the agents from moving towards area that is already covered by another agent.

corresponds to no communication with regards to collision prevention among the agents, we observe on the average up to more than 50 collisions, similar to the numbers observed in the first experiment with the benchmark SGDm method. Even the small value of $\lambda = 1$ drops the number of collisions on average by almost half. We observe a drop in the number of collisions with each increment in λ, with $\lambda = 10$ achieving less than 5 collisions on average. This is naturally expected, since as we increase λ, the agents are more severely penalized when they get close to each other; hence, they maintain a safe distance to ensure a lower collision likelihood. It is intuitively clear that this effect demonstrates a diminishing marginal gain effect, in that the decrease in the number of collisions beyond the value of $\lambda = 5$ seems to slow down.

4.2. Area Coverage

For the second application, we consider a scenario where multiple agents patrol a designated area by following a fixed trajectory, illustrated in Fig. 5. The goal of each agent is to maintain maximum total area coverage by avoiding crossing into areas already covered by other agents, while generally maintaining its fixed trajectory. A motivating example might be one where the agents are UAVs carrying out a ground coverage task of their designated areas, where the areas overlap in certain regions. Ideally, to have the maximum amount of ground coverage at any given time, we would want to discourage a UAV from approaching an overlapping region of its designated coverage area if it is already being covered by another UAV, since employing multiple UAVs for covering the same area would reduce the total amount of area covered. We claim that this can be seen akin to the first experiment in the following manner: If we increase the r parameter of the agents to a suitable value, the collision prevention mechanism works in the way that the agents try not to cross into territories that are already covered by other agents. Also, the trajectories of the agents in their patrolling area can be modelled as perpetually tracking a target that follows said trajectory. In the experiments, we investigate 3 scenarios: the central server assigns agents new locations under compressed gradients with error feedback and nonzero regularization term, the same scenario but with the regularization term set to 0 (which corresponds to a scenario without communication), and a scenario with no central aggregation, where agents run SGDm locally. We model the intended coverage areas of each agent as a disk of radius 5, with overlapping regions ranging between 10% to 25%. We report the number of “collisions”, which in this case represents the number of area violations between agents, and present them in Table 1.

In the experiments, in addition to the 3-agent scenario illustrated in Fig. 5, we also test 2 and 4 agent cases. We run each experiment for 7000 iterations, with values $\lambda = 100$, $N = 2, 3, 4$, and the rest of the parameters have the same values as in the first experiment of the target-tracking problem. Running the simulation for 7000 iterations corresponds to about 4 full cycles of the agents around their circular tra-

N	SGDm	No-Comp	QSGD3b	TopK	Dropout-B	RandK
2	0.8	0.0	0.0	0.0	0.0	0.0
3	9.0	0.8	0.2	1.2	0.6	0.4
4	12.4	1.0	2.59	0.8	0.2	0.2

Table 1: Average number of collisions over 5 runs of the simulation for varying N with methods SGDm without central server; FED-EF-ZO-SGD with QSGD3b, TopK, Dropout-B and RandK, and No-Comp.

jectory. Again, we observe that the number of collisions reduces significantly by our FED-EF-ZO-SGD algorithm and the results obtained using a compressed gradient with error feedback are very close to the case where no compression is used. In some cases, compression with error feedback leads to even better results. This can be explained similarly to before in that owing to compressed gradients, we inject more noise to the gradients, introducing randomness to the trajectories of the agents, which helps avoid collisions.

5. CONCLUSION

In this study, we tackled a problem of distributed online optimization with communication limitations, where multiple agents collaborate to track targets in a federated learning setting, limited to only zeroth-order information. The communication from the agents to the server was assumed to be constrained, and we addressed this constraint by compressing the communicated information along with an error feedback term. Our analysis showed that in the single-agent scenario, after $\mathcal{O}(\frac{d\sigma^2}{\xi^2})$ steps in the dominant term, the EF-ZO-SGD algorithm will reach a ξ-accurate first-order solution. In the multi-agent scenario, the FED-EF-ZO-SGD algorithm will converge to a ξ-accurate first-order solution after $\mathcal{O}(\frac{\sigma^2 dMQ(\Delta^2+\bar{\varpi}^2)+M\sigma^2+Z^4)}{\xi^2})$ steps in the dominant term. The dominant term in these convergence results are independent of the compression ratio δ. The convergence of the FED-EF-ZO-SGD algorithm was confirmed through simulations.

As future work, one can investigate the collision constraints of each agent from safe reinforcement learning where in addition to maximizing rewards, agents must satisfy some constraints. This framework can be incorporated into our setting and can be analyzed from an optimization perspective. Additionally, rather than doing simple averaging at the central server, our work can be extended to a personalized federated learning setting where the losses are minimized by considering one step further of each agent. Further avenues for research include the examination of how the adaptive tuning of step sizes and the regularization parameters might change the convergence analysis. We note that the tuning of the regularization parameter is very related to dual formulations and Lagrangian methods in the general functional constrained optimization context. Finally, following the intuition presented in the experimental section, the effect of the number of agents on the variance of the stochastic gradients of the local loss functions may be studied. In this manner, as in mini-batch SGD, one might discover that incorporating a factor of $\sqrt{N}$ in the selection of the step size might accelerate convergence in a multi-agent scenario with N agents.

6. APPENDIX. PROOFS

6.1. Lemmas

We state several lemmas from [67], mainly related to the zeroth-order method, which will be used in the main proofs. Suppose $f(x) \in C_L^{1,1}(\mathbb{R}^d)$. Then, the following hold:

Lemma 1. *$f_\mu(x) \in C_{L_\mu}^{1,1}(\mathbb{R}^d)$, where $L_\mu \leq L$ [67].*

Lemma 2. *$f_\mu(x)$ has the following gradient with respect to x:*

$$\nabla f_\mu(x) = \frac{1}{(2\pi)^{d/2}} \int \frac{f(x+\mu u) - f(x)}{\mu} u e^{(-\frac{1}{2}\|u\|^2)} \mathrm{d}u, \tag{26}$$

where $u \sim \mathcal{N}(0, I_d)$ [67].

Lemma 3. *For any $x \in \mathbb{R}^d$, we have*

$$|f_\mu(x) - f(x)| \leq \frac{\mu^2 L d}{2}, \tag{27}$$

[67].

Lemma 4. *For any $x \in \mathbb{R}^d$, we have*

$$\|\nabla f_\mu(x) - \nabla f(x)\| \leq \frac{\mu}{2} L (d+3)^{\frac{3}{2}} \tag{28}$$

[67].

Lemma 5. *For any $x \in \mathbb{R}^d$, we have*

$$\mathbb{E}_u\left[\|g_\mu(x)\|^2\right] \leq \frac{\mu^2}{2} L^2 (d+6)^3 + 2(d+4)\|\nabla f(x)\|^2, \tag{29}$$

where $u \sim \mathcal{N}(0, I_d)$ and $g_\mu(x) = \frac{f(x+\mu u) - f(x)}{\mu} u$ [67].

Lemma 6. *(Young's inequality) For any $x, y \in \mathbb{R}^d$ and $\lambda > 0$, we have*

$$\langle x, y \rangle \leq \frac{\|x\|^2}{2\lambda} + \frac{\|y\|^2 \lambda}{2} \tag{30}$$

[67].

6.2. Proof of Theorem 1

Proof. We assume that $z_t \in \mathbb{R}^d$ are *i.i.d.* random variables for all $t \in \mathbb{Z}^+$. Furthermore, we drop the superscript notation present in the assumptions, since i is always 1 for the single-agent case. Let $\tilde{x}_t$ be defined as follows (following the analysis in [6]):

$$\tilde{x}_t := x_t - \eta e_t. \tag{31}$$

From EF-ZO-SGD, we know that $e_{t+1} = p_t - \mathcal{C}(p_t)$ and $p_t = \tilde{g}_{\mu,t}(x_t) + e_t$, so we can rewrite $\tilde{x}_{t+1}$ as

$$\begin{aligned} \tilde{x}_{t+1} &= x_{t+1} - \eta p_t + \eta\mathcal{C}(p_t) \\ &= x_t - \eta\mathcal{C}(p_t) - \eta\tilde{g}_{\mu,t}(x_t) - \eta e_t + \eta\mathcal{C}(p_t) \\ &= x_t - \eta e_t - \eta\tilde{g}_{\mu,t}(x_t) \\ &= \tilde{x}_t - \eta\tilde{g}_{\mu,t}(x_t), \end{aligned} \tag{32}$$

where $\tilde{g}_{\mu,t}(x_t) = \frac{\tilde{\ell}_t(x_t+\mu u_t)-\tilde{\ell}_t(x_t)}{\mu}u_t$ and $u_t \sim \mathcal{N}(0, I_d)$. By Assumption 3, we can write the following:

$$\begin{aligned} \ell_{\mu,t}(\tilde{x}_{t+1}) \leq \ell_{\mu,t}(\tilde{x}_t) &+ \langle \nabla\ell_{\mu,t}(\tilde{x}_t), \tilde{x}_{t+1} - \tilde{x}_t\rangle \\ &+ \frac{L}{2}\|\tilde{x}_{t+1} - \tilde{x}_t\|^2. \end{aligned} \tag{33}$$

Now by Assumption 4, we get:

$$\begin{aligned} \ell_{\mu,t+1}(\tilde{x}_{t+1}) \leq \ell_{\mu,t}(\tilde{x}_t) &- \eta\langle \tilde{g}_{\mu,t}(x_t), \nabla\ell_{\mu,t}(\tilde{x}_t)\rangle \\ &+ \frac{L\eta^2}{2}\|\tilde{g}_{\mu,t}(x_t)\|^2 + \omega_t. \end{aligned} \tag{34}$$

Since $\nabla\ell_{\mu,t}(x_t) = \mathbb{E}_{u_t,z_t}\left[\tilde{g}_{\mu,t}(x_t)\right]$, taking the expectation of both sides with respect to u_t and z_t, we have the following:

$$\mathbb{E}_{u_t,z_t}\left[\langle \tilde{g}_{\mu,t}(x_t), \nabla\ell_{\mu,t}(\tilde{x}_t)\rangle\right] = \langle \nabla\ell_{\mu,t}(x_t), \nabla\ell_{\mu,t}(\tilde{x}_t)\rangle, \tag{35}$$

and

$$\begin{aligned} \langle \nabla\ell_{\mu,t}(x_t), \nabla\ell_{\mu,t}(\tilde{x}_t)\rangle &= \frac{1}{2}\|\nabla\ell_{\mu,t}(x_t)\|^2 + \frac{1}{2}\|\nabla\ell_{\mu,t}(\tilde{x}_t)\|^2 \\ &- \frac{1}{2}\|\nabla\ell_{\mu,t}(x_t) - \nabla\ell_{\mu,t}(\tilde{x}_t)\|^2. \end{aligned} \tag{36}$$

In the last step, we use the fact that $2\langle a, b\rangle = \|a\|^2 + \|b\|^2 - \|a-b\|^2$. Inserting this into (34), we get:

$$\begin{aligned} \ell_{\mu,t+1}(\tilde{x}_{t+1}) \leq \ell_{\mu,t}(\tilde{x}_t) &- \frac{\eta}{2}\|\nabla\ell_{\mu,t}(x_t)\|^2 \\ &- \frac{\eta}{2}\|\nabla\ell_{\mu,t}(\tilde{x}_t)\|^2 + \frac{L^2\eta}{2}\|x_t - \tilde{x}_t\|^2 \\ &+ \frac{L\eta^2}{2}\mathbb{E}_{u_t,z_t}\left[\|\tilde{g}_{\mu,t}(x_t)\|^2\right] + \omega_t. \end{aligned} \tag{37}$$

Note that $\|\nabla\ell_{\mu,t}(x_t) - \nabla\ell_{\mu,t}(\tilde{x}_t)\|^2 \leq L^2\|x_t - \tilde{x}_t\|^2$ by Assumption 3, with subsequent application of Lemma 1. Also, we can drop -$\frac{\eta}{2}\|\nabla\ell_{\mu,t}(\tilde{x}_t)\|^2$ because it is nonpositive. Using the fact that $\tilde{x}_t - x_t = \eta e_t$, we get the main inequality:

$$\begin{aligned} \underbrace{\frac{\eta}{2}\|\nabla\ell_{\mu,t}(x_t)\|^2}_{\text{Term I}} \leq &\underbrace{\left[\ell_{\mu,t}(\tilde{x}_t) - \ell_{\mu,t+1}(\tilde{x}_{t+1})\right]}_{\text{Term II}} \\ &+ \underbrace{\frac{L\eta^2}{2}\mathbb{E}_{u_t,z_t}\left[\|\tilde{g}_{\mu,t}(x_t)\|^2\right]}_{\text{Term III}} \\ &+ \underbrace{\frac{L^2\eta^3}{2}\|e_t\|^2}_{\text{Term IV}} + \omega_t. \end{aligned} \tag{38}$$

We will put an upper bound to the Terms II, III and IV and a lower bound to Term I. Starting with **Term III**, by Lemma 5, we know that

$$\begin{aligned} \mathbb{E}_{u_t,z_{1:T}}\left[\|\tilde{g}_{\mu,t}(x_t)\|^2\right] \leq 2(d+4)&\mathbb{E}_{z_{1:T}}\left[\|\tilde{\nabla}\ell_t(x_t)\|^2\right] \\ &+ \frac{\mu^2L^2}{2}(d+6)^3, \end{aligned} \tag{39}$$

where $\mathbb{E}_{z_{1:T}}[\|\tilde{\nabla}\ell_t(x_t)\|^2] \leq M\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right] + \sigma^2$ by Assumption 2. Note that, in this step, we use the the principle of causality and the fact that z_t are *i.i.d.* random variables. We can put the following upper bound to **Term II** by means of a telescoping sum and subsequent application of Lemma 3:

$$\sum_{t=1}^{T}\left[\ell_{\mu,t}(\tilde{x}_t) - \ell_{\mu,t+1}(\tilde{x}_{t+1})\right] = \ell_{\mu,1}(\tilde{x}_1) - \ell_{\mu,T+1}(\tilde{x}_{T+1}), \tag{40}$$

and

$$\begin{aligned} \ell_{\mu,1}(\tilde{x}_1) - \ell_{\mu,T+1}(\tilde{x}_{T+1}) &\leq \mu^2Ld + \ell_1(\tilde{x}_1) - \ell_{T+1}(\tilde{x}_{T+1}) \\ &= \mu^2Ld + \ell_1(x_1) - \ell_{T+1}(\tilde{x}_{T+1}), \end{aligned} \tag{41}$$

where we use the fact that $\ell(x_1) = \ell_1(\tilde{x}_1)$, since $\tilde{x}_1 = x_1$ by definition. Then, we can do the following:

$$\begin{aligned} \sum_{t=1}^{T}\left[\ell_{\mu,t}(\tilde{x}_t) - \ell_{\mu,t+1}(\tilde{x}_{t+1})\right] &\leq \mu^2Ld + \ell_1(x_1) \\ &\quad - \ell_{T+1}(\tilde{x}_{T+1}) \\ &\leq \mu^2Ld + \ell_1(x_1) \\ &\quad - \ell_{T+1}(x^*_{T+1}), \end{aligned} \tag{42}$$

where $x^*_{T+1} \in \operatorname{argmin}_x \ell_{T+1}(x)$. We can put the following lower bound to **Term I** by using Lemmas 4 and 6:

$$\frac{1}{2}\|\nabla\ell_t(x_t)\|^2 - \frac{\mu^2L^2}{4}(d+3)^3 \leq \|\nabla\ell_{\mu,t}(x_t)\|^2. \tag{43}$$

Lastly, we can put the following upper bound to **Term IV** by Assumption 5 and Lemma 6. (Due to space considerations, in the remainder of the proof, we denote the total expectation $\mathbb{E}_{u_{1:T},z_{1:T},\mathcal{C}_{1:T}}[\,\cdot\,]$ as $\mathbb{E}[\,\cdot\,]$.)

$$\begin{aligned} \mathbb{E}\left[\|e_{t+1}\|^2\right] &= \mathbb{E}\left[\|p_t - \mathcal{C}_t(p_t)\|^2\right] \\ &\leq (1-\delta)\mathbb{E}\left[\|p_t\|^2\right] \\ &= (1-\delta)\mathbb{E}\left[\|e_t + \tilde{g}_{\mu,t}(x_t)\|^2\right] \\ &\leq (1-\delta)(1+\varphi)\mathbb{E}\left[\|e_t\|^2\right] + (1-\delta)\left(1+\frac{1}{\varphi}\right) \\ &\quad \mathbb{E}_{u_{1:T},z_{1:T}}\left[\|\tilde{g}_{\mu,t}(x_t)\|^2\right], \end{aligned} \tag{44}$$

which we can write as,

$$\begin{aligned}&\sum_{i=1}^{t}\left[(1-\delta)(1+\varphi)\right]^{t-i}(1-\delta)(1+\frac{1}{\varphi})\\&\qquad\qquad\mathbb{E}_{u_i,z_{1:T}}\left[\|\tilde{g}_{\mu,i}(x_i)\|^2\right],\end{aligned}\tag{45}$$

for some $\varphi > 0$, $z_t, u_t, \mathcal{C}_t$ are *i.i.d.*, and $\mathbb{E}_{\mathcal{C}_t}[\,\cdot\,]$ denotes the expectation over the randomness at time t due to the compression used. Note that by using Lemma 5 and Assumption 2,

$$\mathbb{E}_{u_t,z_{1:T}}[\|\tilde{g}_{\mu,t}(x_t)\|^2] \leq A\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right] + B,\tag{46}$$

where

$$\begin{aligned}&B = 2\sigma^2(d+4) + \frac{\mu^2L^2}{2}(d+6)^3 \text{ and}\\&A = 2M(d+4).\end{aligned}\tag{47}$$

So we can rewrite (44) as follows:

$$\begin{aligned}\mathbb{E}\left[\|e_{t+1}\|^2\right] \leq &\sum_{i=1}^{t}\left[(1-\delta)(1+\varphi)\right]^{t-i}(1-\delta)(1+\frac{1}{\varphi})\\&\qquad\left[A\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_i(x_i)\|^2\right]+B\right].\end{aligned}\tag{48}$$

If we set $\varphi := \frac{\delta}{2(1-\delta)}$, then $1+\frac{1}{\varphi} \leq \frac{2}{\delta}$ and $(1-\delta)(1+\varphi) = (1-\frac{\delta}{2})$, so we get:

$$\begin{aligned}\mathbb{E}\left[\|e_{t+1}\|^2\right] \leq \sum_{i=1}^{t}\left(1-\frac{\delta}{2}\right)^{t-i}&\left[A\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_i(x_i)\|^2\right]+B\right]\\&\qquad\frac{2(1-\delta)}{\delta}.\end{aligned}\tag{49}$$

If we sum through all $\mathbb{E}[\|e_t\|^2]$, we get:

$$\begin{aligned}\sum_{t=1}^{T}\mathbb{E}\left[\|e_t\|^2\right] &\leq \sum_{t=1}^{T}\sum_{i=1}^{t-1}\left(1-\frac{\delta}{2}\right)^{t-i}\\&\qquad\left[A\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_i(x_i)\|^2\right]+B\right]\frac{2(1-\delta)}{\delta}\\&\leq \sum_{t=1}^{T}\left[A\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right]+B\right]\\&\qquad\sum_{i=0}^{\infty}\left(1-\frac{\delta}{2}\right)^{i}\frac{2(1-\delta)}{\delta}\\&\leq \sum_{t=1}^{T}\left[A\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right]+B\right]K,\end{aligned}\tag{50}$$

where $K = \frac{2(1-\delta)}{\delta}\frac{2}{\delta} \leq \frac{4}{\delta^2}$. If we define $\Delta := \ell_1(x_1) - \ell_{T+1}(x^*_{T+1})$, where $x^*_{T+1} \in \text{argmin}_x\, \ell_{T+1}(x)$, and combine the upper bounds derived in (39), (40), (44), and the lower bound derived in (43) and insert them into (38), we get the following:

$$\begin{aligned}&\sum_{t=1}^{T}\frac{\eta}{4}\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right] - \frac{\eta\mu^2L^2}{8}(d+3)^3T\\&\leq \mu^2Ld + \Delta + \frac{T\mu^2L^3\eta^2}{4}(d+6)^3 + \frac{L\eta^2}{2}\sigma^2T2(d+4)\\&+\frac{L\eta^2}{2}\times 2M(d+4)\sum_{t=1}^{T}\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right] + \frac{\eta^3L^2}{2}\\&\times\frac{4}{\delta^2}T\left[2\sigma^2(d+4) + \frac{\mu^2L^2}{2}(d+6)^3\right] + \frac{\eta^3L^2}{2}\\&\times\frac{4}{\delta^2}\sum_{t=1}^{T}2M(d+4)\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right] + \sum_{t=1}^{T}\omega_t.\end{aligned}\tag{51}$$

Now, since z_t's are *i.i.d.* for all $t \in \mathbb{Z}^+$, we have:

$$\begin{aligned}&\frac{E}{T}\sum_{t=1}^{T}\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right]\\&\leq \frac{\mu^2Ld+\Delta}{T} + \frac{\eta^2L^3\mu^2(d+6)^3}{4} + L\eta^2\sigma^2(d+4)\\&+\frac{\eta\mu^2L^2(d+3)^3}{8} + \frac{\eta^3L^2}{\delta^2}4\sigma^2(d+4)\\&+\frac{\eta^3L^2}{\delta^2}\mu^2L^2(d+6)^3 + \frac{1}{T}\sum_{t=1}^{T}\omega_t,\end{aligned}\tag{52}$$

where

$$\begin{aligned}E &= \frac{\eta}{4} - LM\eta^2(d+4) - \frac{L^2\eta^3}{\delta^2}4M(d+4)\\&= \eta\left[\frac{1}{4} - LM\eta(d+4)\left(1+\frac{4L\eta}{\delta^2}\right)\right].\end{aligned}\tag{53}$$

If $\eta \leq \frac{1}{4L}$, the first upper bound will instead be:

$$1 + \frac{4L\eta}{\delta^2} \leq 1 + \frac{1}{\delta^2} = \frac{\delta^2+1}{\delta^2} \leq \frac{2}{\delta^2}.\tag{54}$$

We proceed to find an η such that

$$\frac{2}{\delta^2}LM\eta(d+4) \leq \frac{1}{8}.\tag{55}$$

Then, we get

$$\eta \leq \frac{\delta^2}{16LM(d+4)},\tag{56}$$

which implies $E \geq \frac{\eta}{8}$. Multiplying all terms in the bound by $\frac{8}{\eta}$,

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{z_{1:T}}\left[\|\nabla\ell_t(x_t)\|^2\right] &\leq \frac{8\Delta}{(\eta T)} + \frac{8\mu^2 Ld}{\eta T}\\ &+ 2\eta L^3\mu^2(d+6)^3 + 8L\eta\sigma^2(d+4) + \mu^2L^2(d+3)^3\\ &+ \frac{32\eta^2L^2}{\delta^2}\sigma^2(d+4) + \frac{8\eta^2L^4\mu^2(d+6)^3}{\delta^2} + \frac{8}{\eta T}\sum_{t=1}^{T}\omega_t.\end{aligned} \tag{57}$$

Let

$$\eta = \frac{1}{\sigma\sqrt{(d+4)MTL}} \quad \text{and} \quad \mu = \frac{1}{(d+4)\sqrt{T}}. \tag{58}$$

Putting these values into (57), we get (13) as follows:

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla\ell_t(x_t)\|^2 &\leq \frac{8\Delta\sigma(d+4)^{\frac{1}{2}}M^{\frac{1}{2}}L^{\frac{1}{2}}}{T^{\frac{1}{2}}}\\ &+ \frac{8\sigma dL^{\frac{3}{2}}M^{\frac{1}{2}}}{T^{\frac{3}{2}}(d+3)^{\frac{3}{2}}} + \frac{2(d+6)^{\frac{3}{2}}L^{\frac{5}{2}}}{\sigma(d+4)^{\frac{5}{2}}T^{\frac{3}{2}}M^{\frac{1}{2}}} + \frac{8\sigma(d+4)^{\frac{1}{2}}L^{\frac{1}{2}}}{M^{\frac{1}{2}}T^{\frac{1}{2}}}\\ &+ \frac{(d+3)^3L^2}{(d+2)^2T} + \frac{32L}{\delta^2\sigma^2MT} + \frac{8(d+6)^3L^3}{\delta^2\sigma^2(d+4)^3MT^2}\\ &+ \frac{8\bar{\omega}\sigma(d+4)^{\frac{1}{2}}M^{\frac{1}{2}}L^{\frac{1}{2}}}{T^{\frac{1}{2}}}.\end{aligned} \tag{59}$$

Defining $\bar{\omega} := \sum_{t=1}^{T}\omega_t$, the number of times steps T to obtain a ξ-accurate first order solution is

$$T = \mathcal{O}\left(\frac{d\sigma^2L\Delta M}{\xi^2} + \frac{dL\Delta}{\delta^2\xi} + \frac{\bar{\omega}\sigma^2dML}{\xi^2}\right). \tag{60}$$

■

6.3. Proof of Theorem 2

Proof. We assume in the following that $z_t^{1:N} \in \mathbb{R}^{Nd}$ are *i.i.d.* random variables for all $t \in \mathbb{Z}^+$. Similar to the analysis in the single-agent case, we begin by defining:

$$\bar{e}_t := \frac{1}{N}\sum_{i=1}^{N} e_t^i, \tag{61}$$

and

$$\tilde{x}_t^{1:N} := x_t^{1:N} - \eta\bar{e}_t. \tag{62}$$

Additionally, our global loss function in this scenario is:

$$\bar{\tilde{\ell}}_t\left(x_t^{1:N}\right) = \frac{1}{N}\sum_{i=1}^{N}\tilde{\ell}_t^i\left(x_t^{1:N}\right). \tag{63}$$

Now, we have:

$$\begin{aligned}\tilde{x}_{t+1}^{1:N} &= x_{t+1}^{1:N} - \eta\bar{e}_{t+1}\\ &= x_{t+1}^{1:N} - \eta\frac{1}{N}\sum_{i=1}^{N}\left[p_t^i - \mathcal{C}\left(p_t^i\right)\right]\\ &= x_t^{1:N} - \eta\mathcal{G}_t - \eta\frac{1}{N}\sum_{i=1}^{N}\left[p_t^i - \mathcal{C}\left(p_t^i\right)\right]\\ &= x_t^{1:N} - \eta\frac{1}{N}\sum_{i=1}^{N}p_t^i\\ &= x_t^{1:N} - \eta\frac{1}{N}\sum_{i=1}^{N}\left[\tilde{g}_{\mu,t}^i\left(x_t^{1:N}\right) + e_t^i\right]\\ &= \tilde{x}_t^{1:N} - \eta\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right),\end{aligned} \tag{64}$$

where we define $\bar{\tilde{g}}_{\mu,t}(x_t^{1:N}) := \frac{1}{N}\sum_{i=1}^{N}\tilde{g}_{\mu,t}^i\left(x_t^{1:N}\right)$. Now, we have by Assumption 3 that each ℓ_t^i is $L-$smooth, therefore, our global loss function $\bar{\ell}_t$ is also $L-$smooth. Using Lemma 1, we write

$$\begin{aligned}\bar{\ell}_{\mu,t}\left(\tilde{x}_{t+1}^{1:N}\right) \leq \bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right) + \left\langle\nabla\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right), \tilde{x}_{t+1}^{1:N} - \tilde{x}_t^{1:N}\right\rangle\\ + \frac{L}{2}\left\|\tilde{x}_{t+1}^{1:N} - \tilde{x}_t^{1:N}\right\|^2.\end{aligned} \tag{65}$$

By Assumption 4, this implies

$$\begin{aligned}\bar{\ell}_{\mu,t+1}\left(\tilde{x}_{t+1}^{1:N}\right) &\leq \bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\\ &- \eta\left\langle\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right), \nabla\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\right\rangle\\ &+ \frac{L\eta^2}{2}\left\|\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2 + \omega_t,\end{aligned} \tag{66}$$

where $\omega_t = \max\{w_t^1, ..., w_t^N\}$. Now, since we have

$$\begin{aligned}\mathbb{E}_{u_t^{1:N}}\left[\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right)\right] &= \mathbb{E}_{u_t^{1:N}}\left[\frac{1}{N}\sum_{i=1}^{N}\tilde{g}_{\mu,t}^i\left(x_t^{1:N}\right)\right]\\ &= \frac{1}{N}\sum_{i=1}^{N}\nabla\tilde{\ell}_{\mu,t}^i\left(x_t^{1:N}\right)\\ &= \nabla\bar{\tilde{\ell}}_{\mu,t}\left(x_t^{1:N}\right),\end{aligned} \tag{67}$$

the following holds:

$$\begin{aligned}&\mathbb{E}_{u_t^{1:N}, z_t^{1:N}}\left[\left\langle\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right), \nabla\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\right\rangle\right]\\ &= \left\langle\nabla\bar{\ell}_{\mu,t}\left(x_t^{1:N}\right), \nabla\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\right\rangle = \frac{1}{2}\left\|\nabla\bar{\ell}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2\\ &+ \frac{1}{2}\left\|\nabla\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\right\|^2\\ &- \frac{1}{2}\left\|\nabla\bar{\ell}_{\mu,t}\left(x_t^{1:N}\right) - \nabla\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\right\|^2,\end{aligned} \tag{68}$$

since $\mathbb{E}_{z_t^{1:N}}[\nabla\bar{\tilde{\ell}}(x_t^{1:N})] = \nabla\bar{\ell}(x_t^{1:N})$. Now, combining this with (66) and using $L-$smoothness, we obtain:

$$\begin{aligned}\bar{\ell}_{\mu,t+1}\left(\tilde{x}_{t+1}^{1:N}\right) \leq \bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right) &- \frac{\eta}{2}\left\|\nabla\bar{\ell}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2 \\ &- \frac{\eta}{2}\left\|\nabla\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right)\right\|^2 \\ &+ \frac{L^2\eta}{2}\left\|x_t^{1:N} - \tilde{x}_t^{1:N}\right\|^2 \\ &+ \frac{L\eta^2}{2}\mathbb{E}_{u_t^{1:N},z_t^{1:N}}\left[\left\|\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2\right] + \omega_t\end{aligned} \tag{69}$$

Note that the third term at the right-hand side of the inequality can be dropped because it is nonpositive. Using the definition of $\tilde{x}_t^{1:N}$, and taking the expectation of both sides with respect to $u_t^{1:N}$ and $z_t^{1:N}$, we have the following main inequality:

$$\begin{aligned}\underbrace{\frac{\eta}{2}\left\|\nabla\bar{\ell}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2}_{\text{Term I}} \leq \underbrace{\left[\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right) - \bar{\ell}_{\mu,t+1}\left(\tilde{x}_{t+1}^{1:N}\right)\right]}_{\text{Term II}} \\ + \underbrace{\frac{L\eta^2}{2}\mathbb{E}_{u_t^{1:N},z_t^{1:N}}\left[\left\|\bar{\tilde{g}}_{\mu,t}\left(x_t^{1:N}\right)\right\|^2\right]}_{\text{Term III}} \\ + \underbrace{\frac{L^2\eta^3}{2}\|\bar{e}_t\|^2}_{\text{Term IV}} + \omega_t.\end{aligned} \tag{70}$$

We will continue the proof by putting an upper bound to Terms II, III, and IV and a lower bound to Term I. Starting with **Term III**, using Jensen's inequality, we get

$$\begin{aligned}&\mathbb{E}_{u_t^{1:N},z_t^{1:N}}\left[\left\|\bar{\tilde{g}}_{\mu,t}(x_t^{1:N})\right\|^2\right] \\ &= \mathbb{E}_{u_t^{1:N},z_t^{1:N}}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\tilde{g}_{\mu,t}^i(x_t^{1:N})\right\|^2\right] \\ &\leq \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{u_t^{1:N},z_t^{1:N}}\left[\left\|\tilde{g}_{\mu,t}^i(x_t^{1:N})\right\|^2\right].\end{aligned} \tag{71}$$

Then, by Lemma 5 we know

$$\begin{aligned}\mathbb{E}_{u_{1:T}^{1:N},z_{1:T}^{1:N}}\left[\|\tilde{g}_{\mu,t}^i(x_t^{1:N})\|^2\right] \leq 2(d+4) \\ \mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\tilde{\ell}_t^i(x_t^{1:N})\|^2\right] \\ + \frac{\mu^2L^2}{2}(d+6)^3.\end{aligned} \tag{72}$$

Using Assumption 2, we have $\mathbb{E}_{z_{1:T}^{1:N}}[\|\nabla\tilde{\ell}_t^i(x_t^{1:N})\|^2] \leq M\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\ell_t^i(x_t^{1:N})\|^2\right] + \sigma^2$. Then, through application of Assumption 6 and Lemma 6, we have:

$$\begin{aligned}\mathbb{E}_{u_{1:T}^{1:N},z_{1:T}^{1:N}}\left[\|\tilde{g}_{\mu,t}^i(x_t^{1:N})\|^2\right] \leq 2(d+4)(MZ^2+\sigma^2) \\ +2(d+4)MQ\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2\right] \\ +\frac{\mu^2L^2}{2}(d+6)^3.\end{aligned} \tag{73}$$

For **Term II**, if we do a summation on both sides of (70) from $t = 1$ to T, we get a telescoping sum:

$$\begin{aligned}&\sum_{t=1}^{T}\left[\bar{\ell}_{\mu,t}\left(\tilde{x}_t^{1:N}\right) - \bar{\ell}_{\mu,t+1}\left(\tilde{x}_{t+1}^{1:N}\right)\right] \\ &= \bar{\ell}_{\mu,1}\left(\tilde{x}_1^{1:N}\right) - \bar{\ell}_{\mu,T+1}\left(\tilde{x}_{T+1}^{1:N}\right).\end{aligned} \tag{74}$$

By adding and subtracting $\bar{\ell}_1(\tilde{x}_1^{1:N})$ and $\bar{\ell}_{T+1}(\tilde{x}_{T+1}^{1:N})$ on both sides and using Lemma 3, we have:

$$\begin{aligned}&\bar{\ell}_{\mu,1}\left(\tilde{x}_1^{1:N}\right) - \bar{\ell}_{\mu,T+1}\left(\tilde{x}_{T+1}^{1:N}\right) \\ &\leq \mu^2Ld + \bar{\ell}_1(x_1^{1:N}) - \bar{\ell}_{T+1}(\tilde{x}_{T+1}^{1:N}). \\ &\leq \mu^2Ld + \bar{\ell}_1(x_1^{1:N}) - \bar{\ell}_{T+1}(x_{T+1}^*) \\ &= \mu^2Ld + \Delta,\end{aligned} \tag{75}$$

where $x_{T+1}^* = \min_{i\in\{1,\ldots,N\}}\operatorname{argmin}_x \ell_{T+1}^i(x)$ and $\Delta = \bar{\ell}_1(x_1^{1:N}) - \bar{\ell}_{T+1}(x_{T+1}^*)$. Note that we use $\tilde{x}_1^{1:N} = x_1^{1:N}$. For **Term I**, one should note that if $\ell_t^i(x) \in C_L^{1,1}$, then $\ell_{\mu,t}^i(x) \in C_L^{1,1}$ by Lemma 1. This implies that $\bar{\ell}_{\mu,t}(x) \in C_L^{1,1}$ because $\bar{\ell}_{\mu,t}(x) = \frac{1}{N}\sum_{i=1}^{N}\ell_{\mu,t}^i(x)$. Thus, using Lemmas 4 and 6, we get

$$\frac{1}{2}\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2 - \frac{\mu^2L^2(d+3)^2}{4} \leq \|\nabla\bar{\ell}_{\mu,t}(x_t^{1:N})\|^2. \tag{76}$$

Finally, for **Term IV**, we use the recursive summation similar to the one in the single-agent proof. We want to put an upper bound to $\|\bar{e}_t\|^2$. We can do so by taking the expectation of both sides in (70) with respect to $u_{1:T}^{1:N}, z_{1:T}^{1:N}, C_{1:T}$ and put an upper bound to $\mathbb{E}_{u_{1:T}^{1:N},z_{1:T}^{1:N},C_{1:T}}\left[\|\bar{e}_t\|^2\right]$ instead. (Due to space considerations, in the remainder of the proof, we denote the total expectation $\mathbb{E}_{u_{1:T}^{1:N},z_{1:T}^{1:N},\mathcal{C}_{1:T}}[\,\cdot\,]$ as $\mathbb{E}[\,\cdot\,]$.) By Jensen's inequality, we can do the following:

$$\begin{aligned}\mathbb{E}\left[\|\bar{e}_t\|^2\right] = \mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}e_t^i\right\|^2\right] \leq \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\left\|e_t^i\right\|^2\right] \\ = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}\left[\|e_t^i\|^2\right]\end{aligned} \tag{77}$$

Note that putting an upper bound to the terms inside summation is nothing but putting an upper bound to the single-agent case, which we have done in Proof 6.2 of the single-agent setting. Hence, we know

$$\begin{aligned}\mathbb{E}\left[\|e_{t-1}^i\|^2\right] \leq \sum_{j=1}^{t-1}[(1-\delta)(1+\varphi)]^{t-1-j}(1-\delta)\left(1+\frac{1}{\varphi}\right) \\ \left[A\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\ell_j^i(x_j^{1:N})\|^2\right] + B\right].\end{aligned} \tag{78}$$

Using this fact in (77), we obtain

$$\begin{aligned}\mathbb{E}\left[\|e_t^{1:N}\|^2\right] &\le \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{t-1}[(1-\delta)(1+\varphi)]^{t-1-j}\\ (1-\delta)&\left(1+\frac{1}{\varphi}\right)\left[A\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\ell_j^i(x_j^{1:N})\|^2\right]+B\right].\end{aligned} \tag{79}$$

Using the same procedure in (50), if we sum both sides through $t = 1$ to $t = T$, we get the following inequality:

$$\begin{aligned}&\sum_{t=1}^{T}\mathbb{E}\left[\|e_t^{1:N}\|^2\right]\\ &\le \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\left[A\mathbb{E}_{z_{1:T}^{1:N}}\|\nabla\ell_t^i(x_t^{1:N})\|^2+B\right]K,\end{aligned} \tag{80}$$

where $A = 2M(d+4), B = 2\sigma^2(d+4) + \frac{\mu^2L^2(d+6)^3}{2}$ and $K = \frac{4(1-\delta)}{\delta^2} \le \frac{4}{\delta^2}$. Another way of expressing (80) is:

$$\begin{aligned}&\sum_{t=1}^{T}\mathbb{E}\left[\|e_t^{1:N}\|^2\right]\\ &\le \sum_{t=1}^{T}\left[A\left(\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\ell_t^i(x_t^{1:N})\|^2\right]\right)+B\right]K.\end{aligned} \tag{81}$$

Using Assumption 6, we can write this as:

$$\begin{aligned}&\sum_{t=1}^{T}\mathbb{E}\left[\|e_t^{1:N}\|^2\right]\\ &\le \sum_{t=1}^{T}\left[A\left(Z^2+Q\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2\right]\right)+B\right]K.\end{aligned} \tag{82}$$

If we now combine the upper bounds derived for Terms I, II and IV, and the lower bound derived for Term III and insert them into (70), we get the following inequality:

$$\begin{aligned}\frac{\eta}{4}\sum_{t=1}^{T}&\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2\right]-\frac{T\eta\mu^2L^2(d+3)^2}{8}\\ &\le \mu^2Ld+\Delta+TL\eta^2(d+4)(MZ^2+\sigma^2)\\ &+L\eta^2(d+4)MQ\left(\sum_{t=1}^{T}\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2\right]\right)\\ &+\frac{TL^2\mu^2\eta^2(d+6)^3}{4}+\frac{2TL^2\eta^3K}{\delta^2}\\ &+\frac{4L^2\eta^3(d+4)MQ}{\delta^2}\left(\sum_{t=1}^{T}\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2\right]\right)\\ &+\sum_{t=1}^{T}\omega_t\end{aligned} \tag{83}$$

where $K = 2M(d+4)Z^2 + 2\sigma^2(d+4) + \frac{\mu^2L^2(d+6)^3}{2}$. After rearranging the terms and dividing both sides by T, we have the following inequality:

$$\begin{aligned}\frac{E}{T}\sum_{t=1}^{T}\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2\right] \le \frac{\mu^2Ld+\Delta}{T}&\\ +L\eta^2(d+4)(MZ^2+\sigma^2)&\\ +\frac{L^3\mu^2\eta^2(d+6)^3}{4}&\\ +\frac{2L^2\eta^3K}{\delta^2}+\frac{\bar{\omega}}{T}&,\end{aligned} \tag{84}$$

where $\bar{\omega} := \sum_{t=1}^{T}\omega_t$, and

$$\begin{aligned}E &= \frac{\eta}{4} - LMQ\eta^2(d+4) - \frac{4L^2\eta^3MQ(d+4)}{\delta^2}\\ &= \eta\left[\frac{1}{4} - LM\eta(d+4)\left(1+\frac{4L\eta}{\delta^2}\right)\right].\end{aligned} \tag{85}$$

If $\eta < \frac{1}{4L}$, the first upper bound will instead be:

$$1+\frac{4L\eta}{\delta^2} \le 1+\frac{1}{\delta^2} \le \frac{2}{\delta^2}. \tag{86}$$

We proceed to find an η such that

$$\frac{2L\eta(d+4)MQ}{\delta^2} \le \frac{1}{8}. \tag{87}$$

Then, we get

$$\eta \le \frac{\delta^2}{16LMQ(d+4)}, \tag{88}$$

which implies $E \ge \frac{\eta}{8}$. Multiplying all the terms in the bound by $\frac{8}{\eta}$,

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{z_{1:T}^{1:N}}\left[\|\nabla\bar{\ell}_t(x_t^{1:N})\|^2\right] &\le \frac{8\Delta}{\eta T}+\frac{8\mu^2Ld}{\eta T}\\ +8L\eta(d+4)(MZ^2+\sigma^2)&+2L^3\mu^2\eta(d+6)^3\\ +\frac{32L^2\eta^2M(d+4)Z^2}{\delta^2}&+\frac{32L^2\eta^2\sigma^2(d+4)}{\delta^2}\\ +\frac{8L^4\mu^2\eta^2(d+6)^3}{\delta^2}&+\frac{8\bar{\omega}}{\eta T}.\end{aligned} \tag{89}$$

Let

$$\eta = \frac{1}{\sigma\sqrt{(d+4)MQTL}} \quad \text{and} \quad \mu = \frac{1}{(d+4)\sqrt{T}}. \tag{90}$$

Then, the number of times steps T to obtain a ξ-accurate first order solution is:

$$\begin{aligned}&T =\\ &\mathcal{O}\left(\frac{\sigma^2dMQ\left(\Delta^2+\bar{\omega}^2\right)+M\left(\sigma^2+Z^4\right)}{\xi^2}+\frac{L^{\frac{5}{3}}}{\xi^{\frac{2}{3}}}+\frac{1}{\delta^2\xi}\right).\end{aligned} \tag{91}$$

■

7. REFERENCES

[1] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. 2016.

[2] Reza Shokri and Vitaly Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, CCS '15, page 1310–1321. ACM, 2015.

[3] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding, 2016.

[4] Anastasia Koloskova, Sebastian U. Stich, and Martin Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication, 2019.

[5] Sebastian U. Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified sgd with memory, 2018.

[6] Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian U. Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes, 2019.

[7] Baris Fidan, Soura Dasgupta, and Brian D. O. Anderson. Guaranteeing practical convergence in algorithms for sensor and source localization. *IEEE Transactions on Signal Processing*, 56(9):4458–4469, 2008.

[8] Anthony Nguyen and Krishnakumar Balasubramanian. Stochastic zeroth-order functional constrained optimization: Oracle complexity and applications. *INFORMS Journal on Optimization*, 2022.

[9] Jean-Marc Valin, François Michaud, and Jean Rouat. Robust localization and tracking of simultaneous moving sound sources using beamforming and particle filtering. *Robotics and Autonomous Systems*, 55(3):216–228, 2007.

[10] Dongming Luan, Yongjian Yang, En Wang, Qiyang Zeng, Zhaohui Li, and Li Zhou. An efficient target tracking approach through mobile crowdsensing. *IEEE Access*, 7:110749–110760, 2019.

[11] Iman Shames, Daniel Selvaratnam, and Jonathan H. Manton. Online optimization using zeroth order oracles. *IEEE Control Systems Letters*, 4(1):31–36, 2020.

[12] Kun Yuan, Qing Ling, and Wotao Yin. On the convergence of decentralized gradient descent. 2013.

[13] Ege C. Kaya, M. Berk Sahin, and Abolfazl Hashemi. Communication-constrained exchange of zeroth-order information with application to collaborative target tracking. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023.

[14] Truc Nguyen and My T. Thai. Preserving privacy and security in federated learning, 2022.

[15] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. 2018.

[16] Cong Xie, Sanmi Koyejo, and Indranil Gupta. Asynchronous federated optimization, 2019.

[17] Yiyue Chen, Abolfazl Hashemi, and Haris Vikalo. Communication-efficient variance-reduced decentralized stochastic optimization over time-varying directed graphs. *IEEE Transactions on Automatic Control*, 2021.

[18] Yiyue Chen, Abolfazl Hashemi, and Haris Vikalo. Decentralized optimization on time-varying directed graphs under communication constraints. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3670–3674. IEEE, 2021.

[19] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H. Brendan McMahan. Adaptive federated optimization, 2020.

[20] Jun Sun, Tianyi Chen, Georgios B. Giannakis, and Zaiyue Yang. Communication-efficient distributed learning via lazily aggregated quantized gradients, 2019.

[21] Rudrajit Das, Anish Acharya, Abolfazl Hashemi, Sujay Sanghavi, Inderjit S Dhillon, and Ufuk Topcu. Faster non-convex federated learning via global and local momentum. In *Uncertainty in Artificial Intelligence*, pages 496–506. PMLR, 2022.

[22] Wenzhi Fang, Ziyi Yu, Yuning Jiang, Yuanming Shi, Colin N. Jones, and Yong Zhou. Communication-efficient stochastic zeroth-order optimization for federated learning. *IEEE Transactions on Signal Processing*, 70:5058–5073, 2022.

[23] Zan Li and Li Chen. Communication-efficient decentralized zeroth-order method on heterogeneous data. In *2021 13th International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1–6, 2021.

[24] Abolfazl Hashemi, Anish Acharya, Rudrajit Das, Haris Vikalo, Sujay Sanghavi, and Inderjit Dhillon. On the benefits of multiple gossip steps in communication-constrained decentralized federated learning. *IEEE Transactions on Parallel and Distributed Systems*, 33(11):2727–2739, 2021.

[25] Seung-Jun Kim and Geogios B. Giannakis. An online convex optimization approach to real-time energy pricing for demand response. *IEEE Transactions on Smart Grid*, 8(6):2784–2793, 2017.

[26] Tianyi Chen and Georgios B. Giannakis. Bandit convex optimization for scalable and dynamic IoT management. *IEEE Internet of Things Journal*, 6(1):1276–1286, feb 2019.

[27] Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Found. Comput. Math.*, 17(2):527–566, apr 2017.

[28] Jorge Nocedal and Stephen J Wright. *Penalty and Augmented Lagrangian Methods*, page 497–524. Springer Series in Operations Research. Springer Science+Business Media, 2nd edition, 2006.

[29] Songtao Lu. A single-loop gradient descent and perturbed ascent algorithm for nonconvex functional constrained optimization, 2022.

[30] David Isele, Reza Rahimi, Akansel Cosgun, Kaushik Subramanian, and Kikuo Fujimura. Navigating occluded intersections with autonomous vehicles using deep reinforcement learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2034–2039, 2018.

[31] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. Safe, multi-agent, reinforcement learning for autonomous driving. *CoRR*, abs/1610.03295, 2016.

[32] Meixin Zhu, Yinhai Wang, Ziyuan Pu, Jingyun Hu, Xuesong Wang, and Ruimin Ke. Safe, efficient, and comfortable velocity control based on reinforcement learning for autonomous driving. *Transportation Research Part C: Emerging Technologies*, 117:102662, 2020.

[33] Songtao Lu, Kaiqing Zhang, Tianyi Chen, Tamer Başar, and Lior Horesh. Decentralized policy gradient descent ascent for safe multi-agent reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(10):8767–8775, May 2021.

[34] Mansoor Shaukat and Mandar Chitre. Adaptive behaviors in multi-agent source localization using passive sensing. *Adaptive Behavior*, 24(6):446–463, 2016. PMID: 28018121.

[35] Sandra H. Dandach, Baris Fidan, Soura Dasgupta, and Brian D. O. Anderson. Adaptive source localization by mobile agents. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 2045–2050, 2006.

[36] Alejandro I. Maass, Chris Manzie, Dragan Nešić, Jonathan H. Manton, and Iman Shames. Tracking and regret bounds for online zeroth-order euclidean and riemannian optimization. *SIAM Journal on Optimization*, 32(2):445–469, 2022.

[37] Barış Fidan, Soura Dasgupta, and Brian. D. O. Anderson. Guaranteeing practical convergence in algorithms for sensor and source localization. *IEEE Transactions on Signal Processing*, 56:4458–4469, 2008.

[38] Elad Michael, Daniel Zelazo, Tony A. Wood, Chris Manzie, and Iman Shames. Optimisation with zeroth-order oracles in formation. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 5354–5359, 2020.

[39] Yipeng Pang and Guoqiang Hu. Randomized gradient-free distributed optimization methods for a multiagent system with unknown cost function. *IEEE Transactions on Automatic Control*, 65(1):333–340, 2020.

[40] Elad Hazan. Introduction to online convex optimization. *CoRR*, abs/1909.05207, 2019.

[41] Yipeng Pang and Guoqiang Hu. Randomized gradient-free distributed online optimization with time-varying cost functions. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 4910–4915, 2019.

[42] Salar Rahili and Wei Ren. Distributed continuous-time convex optimization with time-varying cost functions. *IEEE Transactions on Automatic Control*, 62(4):1590–1605, 2017.

[43] Yu-Jia Chen, Deng-Kai Chang, and Cheng Zhang. Autonomous tracking using a swarm of uavs: A constrained multi-agent reinforcement learning approach. *IEEE Transactions on Vehicular Technology*, 69(11):13702–13717, 2020.

[44] Ruilong Zhang, Qun Zong, Xiuyun Zhang, Liqian Dou, and Bailing Tian. Game of drones: Multi-uav pursuit-evasion game with online motion planning by deep reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–10, 2022.

[45] Luis Rodolfo Garcia Carrillo and Kyriakos G. Vamvoudakis. Deep-learning tracking for autonomous flying systems under adversarial inputs. *IEEE Transactions on Aerospace and Electronic Systems*, 56(2):1444–1459, 2020.

[46] Hao Jiang and Yueqian Liang. Online path planning of autonomous uavs for bearing-only standoff multi-target following in threat environment. *IEEE Access*, 6:22531–22544, 2018.

[47] Yuming Chen, Wei Li, and Yuqiao Wang. Online adaptive kalman filter for target tracking with unknown noise statistics. *IEEE Sensors Letters*, 5(3):1–4, 2021.

[48] Xuejing Lan, Lei Liu, and Yongji Wang. Adp-based intelligent decentralized control for multi-agent systems moving in obstacle environment. *IEEE Access*, 7:59624–59630, 2019.

[49] Shuai Zheng, Ziyue Huang, and James T. Kwok. Communication-efficient distributed blockwise momentum sgd with error-feedback, 2019.

[50] Hao Yu, Rong Jin, and Sen Yang. On the linear speedup analysis of communication efficient momentum sgd for distributed non-convex optimization, 2019.

[51] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U. Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning, 2019.

[52] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local sgd on identical and heterogeneous data, 2019.

[53] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian U. Stich. A unified theory of decentralized sgd with changing topology and local updates, 2020.

[54] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H. Brendan McMahan. Adaptive federated optimization, 2020.

[55] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization, 2020.

[56] Jianyu Wang, Vinayak Tantia, Nicolas Ballas, and Michael Rabbat. Slowmo: Improving communication-efficient distributed sgd with slow momentum, 2019.

[57] Guanghui Lan. *First-order and stochastic optimization methods for machine learning*. Springer, 2020.

[58] Andrea Simonetto, Emiliano Dall'Anese, Santiago Paternain, Geert Leus, and Georgios B Giannakis. Time-varying convex optimization: Time-structured algorithms and applications. *Proceedings of the IEEE*, 108(11):2032–2048, 2020.

[59] Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent, 2017.

[60] Mahmoud Assran, Nicolas Loizou, Nicolas Ballas, and Michael Rabbat. Stochastic gradient push for distributed deep learning. 2018.

[61] Jianyu Wang, Rudrajit Das, Gauri Joshi, Satyen Kale, Zheng Xu, and Tong Zhang. On the unreasonable effectiveness of federated averaging with heterogeneous data, 2022.

[62] John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

[63] Sharan Vaswani, Francis Bach, and Mark Schmidt. Fast and faster convergence of sgd for over-parameterized models and an accelerated perceptron. In *The 22nd international conference on artificial intelligence and statistics*, pages 1195–1204. PMLR, 2019.

[64] Anish Acharya, Abolfazl Hashemi, Prateek Jain, Sujay Sanghavi, Inderjit S Dhillon, and Ufuk Topcu. Robust training in high dimensions via block coordinate geometric median descent. In *International Conference on Artificial Intelligence and Statistics*, pages 11145–11168. PMLR, 2022.

[65] Supplementary material. `https://github.com/Sunses-hub/FED-EF-ZO-SGD.git`, 2022.

[66] Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtarik. Sgd: General analysis and improved rates. 2019.

[67] Guanghui. Lan. *First-order and stochastic optimization methods for machine learning*. Springer Series in the Data Sciences. Springer, Cham, 2020.