Online Learning with Adversaries: A Differential Inclusion Analysis

Swetha Ganesh, Alexandre Reiffers-Masson, Gugan Thoppe

***Abstract*— We consider the measurement model $Y = AX$, where X and, hence, Y are random variables and A is an a priori known tall matrix. At each time instance, a sample of one of Y's coordinates is available, and the goal is to estimate $\mu := \mathbb{E}[X]$ via these samples. However, the challenge is that a small but unknown subset of Y's coordinates are controlled by adversaries with infinite power: they can return any real number each time they are queried for a sample. For such an adversarial setting, we propose the first asynchronous online algorithm that converges to μ almost surely. We prove this result using a novel differential inclusion based two-timescale analysis. Two key highlights of our proof include: (a) the use of a novel Lyapunov function for showing that μ is the unique global attractor for our algorithm's limiting dynamics, and (b) the use of martingale and stopping time theory to show that our algorithm's iterates are almost surely bounded.**

I. Introduction

In this paper, we are interested in incrementally estimating the mean μ of a random variable X when we are only able to sample the random variable $Y = AX$ for some a priori known tall matrix A. The challenge is that some coordinates of Y are controlled by adversaries with infinite power: they can return any value each time they are queried for a sample of their coordinate. Such estimation problems with adversaries naturally are important challenges that commonly appear in different engineering applications such as federated machine learning [6], sensor networks [7], Internet of Battlefield Things [1], or network tomography [5].

This statistical problem has been tackled by two different approaches. First the authors [7] assume that the matrix A is given and the variance of X is 0, i.e., X and Y are deterministic. Even with such simplified assumptions, [7] shows that μ can be retrieved in adversarial settings only under some specific conditions on A. There are some extensions of this work (see [9] for instance) where X is assumed to be a Multivariate normal distribution. In such a line of work, the focus is more on the condition of the recovery than the design of efficient algorithms. The other approach is when it is possible to control the design of the matrix A, as in federated machine learning. In this case, the usual approach is to design a simple A which allows using simple robust statistics such as geometric median or trimmed mean to control the influence of adversaries. However, such approaches require algorithms to be synchronous.

In this work, we presume that the observation matrix A is given to us. We also suppose that, at each instance, we have access only to a sample of a randomly chosen coordinate of Y. In these settings, we propose the first asynchronous online algorithm that almost surely converges to μ. Importantly, our scheme retrieves μ under the same condition that enabled [7] to solve $Y = AX$ when X is deterministic. Note that the setup in [7] is also not online: the adversarial measurements are not queried an infinite number of times.

A. Related works

Our work builds on the findings from [7]. In that paper, the authors suggested an L1-based optimization problem to recover the state of a dynamical system, when an adversary corrupts a sparse number of observations. This work was extended in [9] to a set-up where the observations are also corrupted by Gaussian noise. There, the authors use an extended version of the Kalman filter to estimate the state of the dynamical system.

Our work is also related to distributed machine learning under parameter server architecture, in the presence of adversarial workers. In a parameter server architecture [8], it is assumed that an unattackable parameter server manages the parameter of the machine learning model. Usually, the learning procedure used in such set-up can be described in three steps: (1) the parameter server shares the current parameters of the machine learning model with the workers; (2) the workers, using the current value of the parameters of the machine learning model, and using the data that that they have access to, to compute the gradient associated to the loss function of the model; (3) the workers send the computed gradients to the parameter server which will aggregate them to update the parameters of the model. In an adversarial context, some of the workers can share a corrupted version of their local gradients. Most of the works [13], [15], [4] are focused on the design of $h(\cdot)$ which can handle corrupted local gradient estimation and allows to build a good estimator of the sum of the gradient computed by the workers. To do so, they use aggregator functions such as trimmed mean or geometric median.

The problem tackled in our set-up can be viewed as a way to derive a gradient aggregation method that allows the retrieval of the exact actual gradient even in the presence of adversarial workers. Our algorithm is also asynchronous. The solutions proposed in these papers suffer from two major problems: (1) The problem with the geometric median and trimmed mean is the fact that they are not guaranteed to

S. Ganesh is a Ph.D. student at the Computer Science and Automation Department, Indian Institute of Science, CV Raman Rd, Bengaluru, Karnataka 560012, India. swethaganesh@iisc.ac.in

G. Thoppe is an Asst. professor at the Computer Science and Automation Department, Indian Institute of Science, CV Raman Rd, Bengaluru, Karnataka 560012, India. gthoppe@iisc.ac.in

A. Reiffers-Masson is an associate professor at the Computer Science Department, IMT Atlantique, 655 Av. du Technopôle, 29280 Plouzané, France. alexandre.reiffers-masson@imt-atlantique.fr

retrieve the gradient; and (2) All the algorithms proposed in these papers require a high level of synchronicity.

Such methods are unable to obtain the true value of the gradient because they do not account for the distribution of the data functions among the workers. If we use our method to design the aggregation function, and carefully distribute the data among the workers we will be able to retrieve the exact gradient. To the best of our knowledge, the only work, so far, that has been able to tackle this issue is [6]. In this paper, the authors are interested in solving an empirical risk minimization problem for generalized linear models. The focus is on the design of an encoding matrix such that data is allocated properly between servers and it is possible to estimate the gradient of the machine learning problem even if some workers are adversarial. Moreover, they focus on two distributed set-ups, data parallelism, and model parallelism. They show that the problem can be reduced to adversarial-resilient matrix-vector (MV) multiplication and therefore focus on a specific encoding set-up. Their approach is different from ours and the proposed algorithms in [6] is a synchronous algorithm, as opposed to our method which is an asynchronous algorithm.

More generally, our work is also related to the design of optimization algorithms in presence of adversarial nodes (see [12] and the references therein). Such works use similar techniques as the one proposed in the context of distributed machine learning algorithms (using trimmed mean and median) and share the same weaknesses.

B. Contributions

1) **Algorithm**: We propose the first asynchronous algorithm with convergence guarantees for online learning with adversaries. This algorithm works under the same condition as given in [7]. Loosely, this condition (which is both necessary and sufficient) ensures that the matrix A doesn't put a lot of mass on a small set of coordinates and therefore captures the notion of redundancy. Our algorithm also uses the sign function to ensure that the contribution of every measurement is normalized.
2) **Novel Analysis Framework**: Our work uses a novel Differential Inclusion (DI) based two-timescale analysis to establish convergence of our proposed algorithm. To the best of our knowledge, our work is the first to use a DI to study learning in adversarial settings. There are two key highlights of our proof technique:
   a) *Lyapunov Function*: Our algorithm is based on the gradient descent idea for minimizing $\|Ax - \mathbb{E}[Y]\|_1$. Typically, for such algorithms, the objective function is the natural Lyapunov function. However, in our setup the adversary can make the above function infinite even at μ, and therefore it fails to be Lyapunov. Instead, we show that $\|x - \mu\|_2^2$ acts as a Lyapunov function for our algorithm's limiting dynamics.
   b) *Boundedness of Iterates*: A key step in any ODE/DI based analysis [3] of stochastic algorithms is to show that the algorithm's iterates are stable. In this work, we use a novel martingale and stopping time based approach to show that the algorithm's iterates are almost surely bounded.

II. Setup, Algorithm, and Main Result

We describe here the statistical problem we study, our proposed algorithm to solve it, and our main result that describes the limiting behavior of this algorithm.

Setup: $X \in \mathbb{R}^d$ is a random variable with finite mean and finite covariance matrix entries. There are p agents to collect statistics about X, but an unknown subset M, with $|M| \leq m$, are malicious or adversarial. Specifically, the i-th agent has access to samples of the random variable $Y(i) := a_i^T X$, where $a_i \in \mathbb{R}^d$ is a known deterministic vector. At time $n \geq 1$, a central server picks index i_n uniformly at random from $\{1, \ldots, p\}$ and queries agent i_n for an independent sample of $Y(i_n)$. Agent i_n returns an actual sample if it is non-adversarial, and an arbitrary real number otherwise (the value can change on each query and can depend on the history[1]). In either case, $Y_n(i_n)$ denotes the obtained sample.

Goal: Develop an online algorithm to estimate $\mu := \mathbb{E}[X]$ using the sequence $(Y_n(i_n))$.

Algorithm: Our approach is based on the gradient descent idea for minimizing $\|Ax - \mathbb{E}[Y]\|_1$. Starting from an arbitrary $x_0 \in \mathbb{R}^d$ and $y_0 \in \mathbb{R}^p$, our proposed algorithm to learn μ at the central server is, for $n \geq 0$,

$$\begin{aligned} x_{n+1} &= x_n + \alpha_n a_{i_{n+1}} [\text{sign}(y_n(i_{n+1}) - a_{i_{n+1}}^T x_n)] \\ y_{n+1} &= y_n + \beta_n [Y_{n+1}(i_{n+1}) - y_n(i_{n+1})] u_{i_{n+1}}, \end{aligned} \tag{1}$$

where u_i is i-th column of the $p \times p$-identity matrix and, for any $r \in \mathbb{R}$,

$$\text{sign}(r) = \begin{cases} -1 & \text{if } r < 0, \\ 0 & \text{if } r = 0, \\ 1 & \text{if } r > 0. \end{cases} \tag{2}$$

In (1), the variables indexed by n are known at time n, while the ones by $n+1$ are not. Note that the coordinates of y_n corresponding to malicious nodes are directly fed into x_n's update rule.

Assumptions: Apart from the conditions on $X, (i_n)$, and $Y_n(i_n)$ stated in the setup, we presume that the matrix A and stepsize sequences (α_n) and (β_n) satisfy the following.

1) **Observation matrix**: The matrix A is tall ($p > d$), has full column rank, and satisfies
$$\sum_{i \in K^c} |a_i^T x| > \sum_{i \in K} |a_i^T x| \tag{3}$$
for all $x \in \mathbb{R}^d \setminus \{0\}$ and $K \subseteq \{1, \ldots, p\}$ with $|K| = m$.
2) **Stepsize**: (α_n) and (β_n) are monotonically decreasing positive reals such that $\max\{\alpha_0, \beta_0\} \leq 1$, $\sum_{n \geq 0} \alpha_n = \sum_{n \geq 0} \beta_n = \infty$, $\lim_{n \to \infty} \alpha_n / \beta_n = \lim_{n \to \infty} \beta_n = 0$, and $\max\{\sum_{n \geq 0} \alpha_n^2, \sum_{n \geq 0} \beta_n^2, \sum_{n \geq 0} \alpha_n \gamma_n\} < \infty$, where $\gamma_n = \sqrt{\beta_n \ln(\sum_{k=0}^n \beta_k)}$. An example is $\alpha_n = n^{-\alpha}$, $\alpha \in (2/3, 1]$, and $\beta_n = n^{-\beta}$, $\beta \in (1/2, 1] \cap (2(1-\alpha), \alpha)$.

Our main result is stated below and is derived using a DI-based set-valued analysis. As we discuss in Section II-A,

[1]Such adversaries are commonly referred to as omniscient.

such an analysis is natural for (1) due to its sub-gradient nature and, importantly, the presence of adversaries. Let $h : \mathbb{R}^d \to 2^{\mathbb{R}^d}$ (the power set of $\mathbb{R}^d$) be given by

$$h(x) = \left\{ \frac{1}{p} \sum_{i=1}^{p} a_i \lambda_i : (\lambda_1, \ldots, \lambda_p) \in \Lambda(x) \right\}, \qquad (4)$$

where $\Lambda(x)$ includes all $(\lambda_1, \ldots, \lambda_p)$ for which

$$\lambda_i \in \begin{cases} \{\text{sign}(\mathbb{E}[Y(i)] - a_i^T x)\}, & i \in M^c \text{ and } a_i^T x \neq \mathbb{E}[Y(i)], \\ [-1, +1], & \text{otherwise.} \end{cases}$$

Theorem 1. *The following statements hold.*

1) *μ is the unique Globally Asymptotically Stable Equilibrium (GASE) for the DI*

$$\dot{x}(t) \in h(x(t)). \qquad (5)$$

2) *There exists some constant $\Lambda > 0$ such that*

$$\limsup_{n \to \infty} \frac{\|y_n - \mathbb{E}[Y]\|_{M^c}}{\gamma_n} \leq \Lambda \qquad a.s.,$$

where $\|y\|_{M^c} = \sqrt{\sum_{i \in M^c} y^2(i)}$.

3) *$x_n \to \mu$ a.s.*

The DI in (5) corresponds to the update rule of x_n in (1) with $y_n(i) \equiv \mathbb{E}[Y(i)]$ for $i \in M^c$, and the sign expression replaced with an arbitrary value in $[-1, +1]$, otherwise. Our first result states that every solution of this DI will converge to μ, irrespective of the sign choices made at the adversarial nodes (in a continuous time sense). The second statement provides the asymptotic rate at which $|y_n(i) - \mathbb{E}[Y(i)]| \to 0$, $i \in M^c$, on every sample path. While this result assumes that the stepsizes are square-summable, it can be extended to cover the case of even non-square summable stepsizes; see [11] for details. Our third and final result states that the actual (x_n) iterates in (1) also behave like the solutions of (5) and almost surely converge to μ. However, because the sign function is not continuous, this is not a simple consequence of the first two statements. Instead, we have to rely on a more complex two-timescale DI analysis, and a separate boundedness result for (x_n) based on the theory of martingales and stopping times.

A. Motivation for a DI-based Analysis

In this subsection, we give a simple example on why our algorithm will converge to μ even in the presence of adversarial measurements. We use a simplified set-up to illustrate the necessity of the DI analysis.

Let A be a vector of all ones. This implies that $\mathbb{E}Y(i) = \mu \in \mathbb{R}$, for all i. Our problem setup then reduces to computing $x \in \mathbb{R}$ that minimises $\sum_{i=1}^{p} |x - \mathbb{E}Y(i)|$. The solution to this minimisation problem is called the geometric median [13]. Consider Algorithm (1) in the deterministic setting, where all agents i are given $\mathbb{E}Y(i)$, instead of having to estimate it. Then, $y_n(i)$ will be μ, for $i \in M^c$ and any arbitrary value for $i \in M$. It can be seen that the synchronous version of update (1) can be written as:

$$x_{n+1} = x_n + \alpha_n \left[|M^c| \underbrace{\text{sign}(\mu - x_n)}_{\text{Unperturbed subgradient}} + \sum_{i \in M} \underbrace{\text{sign}(e_n(i))}_{\text{Adversarial noise}} \right].$$

Here, $\text{sign}(\mu - x_n)$ is the subgradient of $|x - \mathbb{E}Y(i)|$ when $i \in M^c$ and $\text{sign}(e_n(i))$ is the perturbed subgradient given by the adversary. The above update rule cannot be analysed using traditional ODE based approaches. Firstly, the update can now take a set of values at each x_n. This is because $\text{sign}(e_n(i))$ can take any value in $[-1, 1]$, regardless of x_n. Moreover, $\text{sign}(\mu - x)$ is discontinuous at $x = \mu$, while ODE approaches require that this function be Lipschitz continuous. Thus, the differential inclusion approach is preferred since it is capable of handling discontinuities and capturing the evolution of a set-valued map. The associated DI for the above update is given by:

$$\dot{x}(t) \in \left\{ |M^c| \text{sign}(\mu - x) + \sum_{i \in M} v_i : v_i \in [-1, 1] \right\},$$

when $x \neq \mu$ and

$$\dot{x}(t) \in \left\{ \sum_{i=1}^{p} v_i : v_i \in [-1, 1] \right\},$$

when $x = \mu$. The DI is modified at $x = \mu$ to make it continuous in a set-valued sense.

Note that if $|M^c| > |M|$ (equivalent to (3)), it follows that $\lim_{t \to +\infty} x(t) = \mu$. The intuition is as follows: if $\mu \neq x$, the sign of $|M^c| \text{sign}(\mu - x) + \sum_{i \in M} v_i$ will be always the same as the $\text{sign}(\mu - x)$ and therefore the drift of the DI is controlled by the $\text{sign}(\mu - x)$ and not by the adversaries. The performance of our algorithm for this problem with $p = 5$ is shown in Figure 1. Here, condition (3) holds if $|M| = 2$, but not when $|M| = 3$. Consequently, our algorithm converges

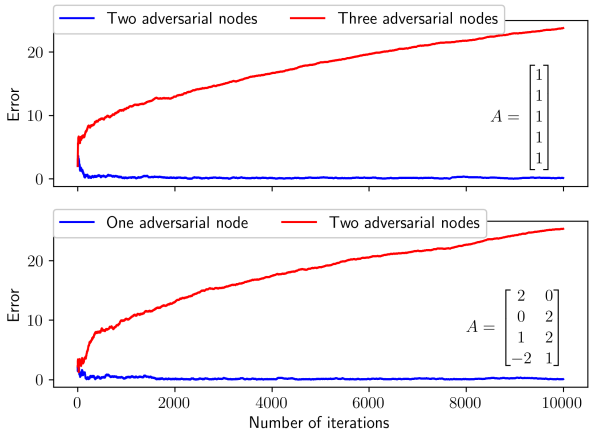

Fig. 1. Error incurred by Algorithm 1 ($\|x_n - \mu\|$) against the number of iterations (n). Within each subplot, the same measurement model is used with the only difference being the number of adversaries. The first subplot concerns the geometric median problem with $p = 5$, while the second considers a generic matrix A (see Section II-A).

in the presence of two advsersaries but diverges in presence of three adversaries.

More generally, condition (3) is necessary and sufficient for our algorithm to converge. We emphasize that this condition is necessary even in the absence of noise and thus cannot be relaxed. A less obvious case where condition (3) is required is shown in Figure 1. For matrix A in this example, the condition holds for $|M| = 1$, but not when $|M| = 2$.

III. Proof of Theorem 1

We first discuss our proof strategy and then provide the details. Since $y_n(i)$'s estimate for $i \in M^c$ is not influenced by the Y samples of other nodes, one would intuitively expect $\|y_n - \mathbb{E}[Y]\|_{M^c} \to 0$. Hence, (5) is the natural object for studying (x_n)'s behaviors. However, because the sign function is discontinuous, (x_n)'s evolution cannot be viewed as a simple perturbation of (5)'s solutions as in [3, pg. 17]. Instead, we rely on a two-timescale DI analysis [14]. Henceforth, $\|\cdot\|$ will denote the Euclidean norm.

A. Informal Outline of Two-timescale Analysis

Our algorithm (1) is of a two-timescale nature because $\alpha_n/\beta_n \to 0$. Thus, the changes in x_n values eventually appear negligible compared to that of y_n, which, in turn, implies (x_n) and (y_n)'s behaviors can be studied in a decoupled fashion. Loosely, our analysis proceeds via the following prescribed steps from [14].

1) (y_n)'s analysis: We set $x_n \equiv x$ for some arbitrary x, and look at $y_n(i)$'s evolution for $i \in M^c$; we ignore what happens at the adversarial nodes. In our case, $y_n(i)$'s evolution is not influenced by the value of x in any way. Further, its limiting ODE can be guessed to be $\dot{z}(t) = \frac{1}{p}(\mathbb{E}[Y(i)] - z(t))$. Since this scalar ODE is linear and has $\mathbb{E}[Y(i)]$ as its unique GASE, it follows from a standard single-timescale stochastic approximation analysis [3, Chapters 2 and 3] that $|y_n(i) - \mathbb{E}[Y(i)]| \to 0$.
2) (x_n)'s analysis: From (x_n)'s perspective, (y_n) would appear to have converged to its limit point. Accordingly, in x_n's update rule, we now set $y_n(i) = \mathbb{E}[Y(i)]$, for $i \in M^c$, and allow for arbitrary values for adversarial i's. This leads to the set-valued DI dynamics (5). In the rest of this section, we formally prove that μ is its only attractor (Section III-B), the original (x_n) sequence in (1) is almost surely bounded (Section III-C), and it almost surely converges to μ (Section III-D).

B. Analysis of the DI in (5)

We first check that (5) is a well-defined DI. Recall that, for an (autonomous) ODE to be well-defined, one sufficient condition is that its driving function be Lipschitz continuous. In particular, this guarantees the existence and uniqueness of a solution for any initial point. Similarly, a DI is well-defined when its set-valued driving function h is Marchaud, i.e., Lipschitz continuous in a set-valued sense (defined below). In general, solutions of a DI from a given starting point are not unique, but the above condition ensures existence.

For $x \in \mathbb{R}^d$, let $Z(x) := M \cup \{i : a_i^T(x - \mu) = 0\}$.

Lemma 1. *The function h defined in (4) is Marchaud, i.e.,*

1) *$h(x)$ is convex and compact for all $x \in \mathbb{R}^d$;*
2) *$\exists K_h > 0$ such that, for all $x \in \mathbb{R}^d$, $\sup_{y \in h(x)} \|y\| \leq K_h(1 + \|x\|)$; and*
3) *h is upper semicontinuous or, equivalently, $\{(x, \theta) \in \mathbb{R}^d \times \mathbb{R}^d : \theta \in h(x)\}$ is closed.*

Hence, the DI in (5) is well-defined.

Proof: The first two conditions are easy. For h's upper semi-continuity, it suffices to check if (x_n) and (θ_n) are such that $x_n \to x$, $\theta_n \in h(x_n)$ $\forall n$, and $\theta_n \to \theta$, then $\theta \in h(x)$.

For $i \in Z(x)^c$, $a_i^T(x - \mu)$ is either > 0 or < 0. This fact along with $x_n \to x$ then implies $\exists n_0 \geq 0$ such that, for $n \geq n_0$, we have $\text{sign}(a_i^T(x_n - \mu)) = \text{sign}(a_i^T(x - \mu))$ for all $i \in Z(x)^c$ and, hence, $Z(x)^c \subseteq Z(x_n)^c$. Consequently, $h(x_n) \subseteq h(x)$ for all $n \geq n_0$, which implies $(\theta_n)_{n \geq n_0} \subseteq h(x)$. The desired result now follows since $h(x)$ is compact.

We now show that μ is (5)'s unique GASE.

Proof of Statement 1, Theorem 1: It suffices to show that $V(x) = \frac{1}{2}\|x - \mu\|^2$ is a Lyapunov function [2] for the DI in (5) with respect to $\{\mu\}$. Clearly, $V(x) = 0$ if and only if $x = \mu$. Further, for any $x \neq \mu$ and $\theta \equiv \frac{1}{p}\sum_{i=1}^p a_i\lambda_i \in h(x)$,

$$\begin{aligned}
&\nabla V(x)^T\theta \\
&= \frac{1}{p}\sum_{i=1}^p \lambda_i a_i^T(x - \mu) \\
&= \frac{1}{p}\left[-\sum_{i \in M^c} |a_i^T(x - \mu)| + \sum_{i \in M} \lambda_i a_i^T(x - \mu)\right] \qquad (6)\\
&\leq \frac{1}{p}\left[-\sum_{i \in M^c} |a_i^T(x - \mu)| + \sum_{i \in M} |a_i^T(x - \mu)|\right] \qquad (7)\\
&< 0, \qquad (8)
\end{aligned}$$

where (6) holds since $\lambda_i \in \text{sign}(-a_i^T(x - \mu))$ for $i \in M^c$, (7) is true because $r \leq |r|$ for any $r \in \mathbb{R}$ and $|\lambda_i| \leq 1$, while (8) follows from (3) since $|M| \leq m$.

The claim now follows from [2, Proposition 3.25]. □

C. Almost Sure Boundedness of (x_n)

We use martingale and stopping time theory to show that (x_n) obtained using (1) is almost surely bounded.

Our proof needs a few intermediate results. In relation to (x_n) and (y_n) in (1), define the following. For $n \geq 0$, let

$$\begin{aligned}
b_n = \frac{1}{p}\sum_{i \in M^c} a_i[&\text{sign}(y_n(i) - a_i^T x_n) \\
&- \text{sign}(\mathbb{E}[Y](i) - a_i^T x_n)], \qquad (9)
\end{aligned}$$

$$\begin{aligned}
g(x_n, y_n) = \frac{1}{p}\sum_{i \in M^c} a_i&\text{sign}(\mathbb{E}[Y](i) - a_i^T x_n) \\
&+ \frac{1}{p}\sum_{i \in M} a_i\text{sign}(y_n(i) - a_i^T x_n),
\end{aligned}$$

and

$$M_{n+1} = a_{i_{n+1}}^T[\text{sign}(y_n(i_{n+1}) - a_{i_{n+1}}^T x_n] - g(x_n, y_n) - b_n. \quad (10)$$

In the above terms, the update rule in (1) can be written as

$$x_{n+1} = x_n + \alpha_n[g(x_n, y_n) + b_n + M_{n+1}]. \quad (11)$$

Note that $g(x_n, y_n) \in h(x_n)$. Therefore, one can view $g(x_n, y_n)$ as the update direction that is prescribed by (5), b_n as a perturbation that arises since, for $i \in M^c$, $y_n(i) \neq \mathbb{E}[Y(i)]$ a.s. for any finite n, and M_{n+1} as the noise.

Lemma 2. *The following statements are true.*

1) *For $x \in \mathbb{R}^d$, let $\phi(x) = \frac{1}{p}\sum_{i\in M^c}|a_i^T x| - \frac{1}{p}\sum_{i\in M}|a_i^T x|$. Then there exists $\eta > 0$ such that $\phi(x) \geq \eta\|x\| \quad \forall x$.*
2) *$|(x_n - \mu)^T b_n| \leq \frac{2\sqrt{|M^c|}}{p}\|y_n - \mathbb{E}[Y]\|_{M^c}$.*
3) *$(x-\mu)^T\theta \leq -\eta\|x-\mu\|$ for any $\theta \in h(x)$.*
4) *Let $C_M := \sup_{1\leq i\leq p}\|a_i\|$. Then, for any $n \geq 0$,*

$$\|x_{n+1} - \mu\|^2 \leq \|x_0 - \mu\|^2 + \sum_{k=0}^{n}\alpha_k(x_k - \mu)^T M_{k+1} + \frac{2}{p}\sum_{k=0}^{n}\alpha_k\|y_k - \mathbb{E}[Y]\|_{M^c} + C_M^2\sum_{k=0}^{n}\alpha_k^2.$$

Proof: The first statement is trivially true for $x = 0$. Hence, suppose $x \neq 0$. It suffices to show that $\exists\,\eta > 0$ such that $\phi(x) \geq \eta$ for any x with unit norm. However, this holds since (a) ϕ is continuous and $\{x \in \mathbb{R}^d : \|x\| = 1\}$ is a compact set: thus, ϕ attains its minimum; and (b) $\phi(x) > 0$ for any $x \neq \mu$ on account of (3).

For the second statement, note that

$$|\text{sign}(r_1 - r_0) - \text{sign}(r_2 - r_0)| \leq 2\delta_{|r_1 - r_2| \geq |r_0 - r_2|}$$

for any $r_0, r_1, r_2 \in \mathbb{R}$, where δ denotes the indicator function. Combining this with the fact that $\mathbb{E}[Y(i)] = a_i^T\mu$, for $i \in M^c$, gives

$$\begin{aligned}
&|(x_n - \mu)^T b_n| \\
&\leq \frac{2}{p}\sum_{i\in M^c}|a_i^T(x_n - \mu)|\delta_{|y_n(i) - \mathbb{E}[Y(i)]| \geq |a_i^T x_n - a_i^T\mu|} \\
&\leq \frac{2}{p}\sum_{i\in M^c}|y_n(i) - \mathbb{E}[Y(i)]|\delta_{|y_n(i) - \mathbb{E}[Y(i)]| \geq |a_i^T x_n - a_i^T\mu|} \\
&\leq \frac{2}{p}\sum_{i\in M^c}|y_n(i) - \mathbb{E}Y(i)| \\
&\leq \frac{2\sqrt{|M^c|}}{p}\|y_n - \mathbb{E}[Y]\|_{M^c},
\end{aligned}$$

as desired.

We now discuss the third statement. Let $\theta \in h(x)$ be arbitrary. Then,

$$\begin{aligned}
&(x-\mu)^T\theta \\
&\leq \frac{1}{p}\left[-\sum_{i\in M^c}|a_i^T(x-\mu)| + \sum_{i\in M}|a_i^T(x-\mu)|\right] \\
&\leq -\phi(x-\mu),
\end{aligned}$$

where the first relation follows as in (7), and the second relation holds from ϕ's definition. The claim now follows from our first statement above.

Finally, we derive the fourth statement. From (11),

$$\|x_{n+1} - \mu\|^2 = \|x_n - \mu\|^2 + \alpha_n^2\|g(x_n, y_n) + b_n + M_{n+1}\|^2 + 2\alpha_n(x_n - \mu)^T[g(x_n, y_n) + b_n + M_{n+1}].$$

Statement 3 along with the fact that $g(x_n, y_n) \in h(x_n)$ shows $(x_n - \mu)^T g(x_n, y_n) \leq -\eta\|x_n - \mu\|$, while Statement 2 gives the bound on $(x_n - \mu)^T b_n$. Separately, $\|g(x_n, y_n) + b_n + M_{n+1}\| = \|a_{i_{n+1}}\| \leq C_M$. It now follows that

$$\|x_{n+1} - \mu\|^2 \leq \|x_n - \mu\|^2 - \alpha_n\eta\|x_n - \mu\| + \frac{2\sqrt{|M^c|}\alpha_n}{p}\|y_n - \mathbb{E}[Y]\|_{M^c} + \alpha_n(x_n - \mu)^T M_{n+1} + C_M^2\alpha_n^2.$$

The desired claim is now easy to see. □

Presuming Statement 2 in Theorem 1 holds, we are now ready to show that (x_n) is bounded almost surely,

Proposition 1. $\sup_{n\geq 0}\|x_n\| < \infty$ *a.s.*

Proof: Let (γ_n) be as in Theorem 1. Fix an arbitrary integer $r \geq 1$, and let $C_r := \frac{2r\sqrt{|M^c|}}{p}\sum_{k=0}^{\infty}\alpha_k\gamma_k + C_M^2\sum_{k=0}^{\infty}\alpha_k^2 < \infty$, and $T(r)$ be the stopping time $\inf\left\{n \geq 0 : \frac{1}{\gamma_n}\|y_n - \mathbb{E}[Y]\|_{M^c} > r\right\}$. Next, for $n \geq 0$, let

$$S_n = \|x_0 - \mu\|^2 + 2\sum_{k=0}^{n-1}\alpha_k(x_k - \mu)^T M_{k+1} + C_r.$$

Clearly, (S_n) and, hence, $(S_n^r) \equiv (S_{n\wedge T(r)})$ is a martingale.

Let $(x_n^r) \equiv (x_{n\wedge T(r)})$. Then Statement 4 of Lemma 2 shows $\|x_n^r - \mu\|^2 \leq S_n^r \;\forall\, n \geq 0$. This implies (S_n^r) is a non-negative martingale and, hence, converges almost surely. Therefore, (x_n^r) is bounded almost surely.

Finally, note that

$$\begin{aligned}
E &:= \left\{\sup_{n\geq 0}\|x_n\| = \infty\right\} \\
&\qquad \cap \left[\bigcup_{r=1}^{\infty}\left\{\sup_{n\geq 0}\frac{\|y_n - \mathbb{E}[Y]\|_{M^c}}{\gamma_n} \leq r\right\}\right] \\
&= \bigcup_{r=1}^{\infty}\left\{\sup_{n\geq 0}\|x_n^r\| = \infty, \sup_{n\geq 0}\frac{\|y_n - \mathbb{E}[Y]\|_{M^c}}{\gamma_n} \leq r\right\} \quad (12) \\
&\subseteq \bigcup_{r=1}^{\infty}\left\{\sup_{n\geq 0}\|x_n^r\| = \infty\right\},
\end{aligned}$$

where (12) follows from the fact that $\sup_{n\geq 0}\frac{\|y_n - \mathbb{E}[Y]\|_{M^c}}{\gamma_n} \leq r$ implies $x_n = x_n^r$ for all n. Since (x_n^r) is almost surely bounded for any $r \geq 1$, we get $\mathbb{P}(E) = 0$. From Statement 2 in Theorem 1, we also have that

$$\mathbb{P}\left(\bigcup_{r=1}^{\infty}\left\{\sup_{n\geq 0}\frac{\|y_n - \mathbb{E}[Y]\|_{M^c}}{\gamma_n} \leq r\right\}\right) = 1.$$

The desired claim now follows since, for any events E_1 and E_2, $\mathbb{P}(E_1) = 1$ and $\mathbb{P}(E_2^c \cap E_1) = 0$ imply $\mathbb{P}(E_2) = 1$. □

D. Rest of the Proof

In this section, we discuss the proofs of Statements 2 and 3 of Theorem 1.

Statement 2 follows from [10, Theorem 1], which provides a law of iterated logarithm type result for generic stochastic approximation algorithms. That work assumes that the iterates almost surely converge, but this can be shown using the results in [3, Chapters 2 and 3], as discussed in Section III-A.

To prove Statement 3, we rely on [14, Theorem 4], which looks at convergence of generic two-timescale algorithms with set-valued limiting dynamics. Specifically, this latter result assumes (x_n)'s limiting DI has a global attractor (see A10 there), and states that, if ten other conditions (labelled A1 - A9 and A11 there) hold, then x_n converges to this global attractor a.s. These ten conditions concern the behaviors of x_n and y_n's driving functions, stepsizes, and noise. Below we provide a brief commentary on why these assumptions hold for (1). The reader should note that the role of x_n and y_n is flipped in [14]: the changes in y_n eventually appear negligible compared to that of x_n. The analysis there also accounts for Markov noise, but it can ignored using the approach suggested in Remark 3 there. Finally, for all of (y_n)'s analysis below, we ignore the evolution at adversarial nodes: instead, we account for them directly in the definition of the DI in (5).

Assumptions A1 and A2 of [14] hold when the limiting DIs associated with x_n and y_n are Marchaud. For (1), this can be established like in the proof of our Lemma 1. Assumptions A3 and A4 concern Markov noise and, hence, trivially hold true in our case. Assumption A5 is on stepsizes and it holds in our case because we also assume those conditions. Assumption A8 there holds if the (x_n) and (y_n) iterates are bounded almost surely. Proposition 1 here proves it for (x_n), while, for (y_n), it follows easily from [3, Chapter 3, Theorem 7] due to its linear nature. Assumptions A6 and A7 hold if the contributions of the additive noise terms are eventually negligible. This can be established as in [3, Chapter 2, (2.19)], which holds in our case because our iterates are bounded a.s. and the noise growth rate condition of (2.13) trivially holds in our context. Assumptions A9 and A11 hold, if for each fixed x, the limiting DI for (y_n) has a unique GASE. As discussed in Section III-A, in our case, the dynamics of (y_n) is not influenced by the value of x and $\{\mathbb{E}[Y(i)] : i \in M^c\}$ is the global attractor for any x. Finally, Assumption A10 requires that (x_n)'s limiting DI has a unique global attractor. We established this in Statement 1 of our Theorem 1.

IV. Generalisations

In this section, we discuss simple extensions of our work, where we can relax certain assumptions.

Non-zero kernel: The condition (3) fails for all matrices A with a non-zero kernel. Thus, Theorem 1 cannot be used for fat matrices or tall matrices with non full rank. However, we can obtain a similar result by relaxing condition (3) to hold only for points outside the kernel of A. Note that in this case, there are several $x \in \mathbb{R}^d$ such that $Ax = \mathbb{E}Y$. Under this modified assumption, it can be shown that the DI always converges to one such point. To see this, the function $\frac{1}{2}\|x-\mu\|_2^2$, with μ as solution of $Ax = \mathbb{E}Y$, would remain a Lyapunov function in this case. Applying a variant of LaSalle's invariance theorem would then give us that the DI converges to an invariant subset of $\{x : Ax = \mathbb{E}Y\}$.

Perturbed samples: Suppose that, instead of being provided samples of $Y(i) = a_i^T X$, we only have access to samples of form $Y(i) = a_i^T X + b(i)$, where $b(i)$ is some random or deterministic perturbation. The only condition imposed on $b(i)$ is that its magnitude remains bounded by some constant B for each i. We can extend the result in Theorem 1 to this setting using similar arguments as discussed in the previous case. However, the Lyapunov function would need to be redefined and may have discontinuous derivatives.

Non-linear function: The results in this work are applicable for solving problems of form $Ax - \mathbb{E}[y] = 0$. An extension to problems of form $f(x) = 0$ could be achieved by replacing $a_{i_n}^T x_n - y(i_n)$ with $a_{i_n}^T f(x_n)$ in (1).

References

[1] Abdelzaher, T., Ayanian, N., Basar, T., Diggavi, S., Diesner, J., Ganesan, D., Govindan, R., Jha, S., Lepoint, T., Marlin, B., et al.: Will distributed computing revolutionize peace? the emergence of battlefield iot. In: 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS). pp. 1129–1138. IEEE (2018)

[2] Benaïm, M., Hofbauer, J., Sorin, S.: Stochastic approximations and differential inclusions. SIAM Journal on Control and Optimization **44**(1), 328–348 (2005)

[3] Borkar, V.S.: Stochastic approximation: a dynamical systems viewpoint, vol. 48. Springer (2009)

[4] Chen, Y., Su, L., Xu, J.: Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. Proceedings of the ACM on Measurement and Analysis of Computing Systems **1**(2), 1–25 (2017)

[5] Chiu, C.C., He, T.: Stealthy dgos attack: Degrading of service under the watch of network tomography. IEEE/ACM Transactions on Networking **29**(3), 1294–1307 (2021)

[6] Data, D., Song, L., Diggavi, S.N.: Data encoding for byzantine-resilient distributed optimization. IEEE Transactions on Information Theory **67**(2), 1117–1140 (2020)

[7] Fawzi, H., Tabuada, P., Diggavi, S.: Secure estimation and control for cyber-physical systems under adversarial attacks. IEEE Transactions on Automatic control **59**(6), 1454–1467 (2014)

[8] Li, M., Andersen, D.G., Smola, A.J., Yu, K.: Communication efficient distributed machine learning with the parameter server. Advances in Neural Information Processing Systems **27** (2014)

[9] Mishra, S., Shoukry, Y., Karamchandani, N., Diggavi, S.N., Tabuada, P.: Secure state estimation against sensor attacks in the presence of noise. IEEE Transactions on Control of Network Systems **4**(1), 49–59 (2016)

[10] Pelletier, M.: On the almost sure asymptotic behaviour of stochastic algorithms. Stochastic processes and their applications **78**(2), 217–244 (1998)

[11] Thoppe, G.C., Kumar, B.: A law of iterated logarithm for multi-agent reinforcement learning. Advances in Neural Information Processing Systems **34**, 17927–17938 (2021)

[12] Wang, R., Liu, Y., Ling, Q.: Byzantine-resilient resource allocation over decentralized networks. IEEE Transactions on Signal Processing **70**, 4711–4726 (2022)

[13] Wu, Z., Ling, Q., Chen, T., Giannakis, G.B.: Federated variance-reduced stochastic gradient descent with robustness to byzantine attacks. IEEE Transactions on Signal Processing **68**, 4583–4596 (2020)

[14] Yaji, V.G., Bhatnagar, S.: Stochastic recursive inclusions in two timescales with nonadditive iterate-dependent markov noise. Mathematics of Operations Research **45**(4), 1405–1444 (2020)

[15] Yin, D., Chen, Y., Kannan, R., Bartlett, P.: Byzantine-robust distributed learning: Towards optimal statistical rates. In: International Conference on Machine Learning. pp. 5650–5659. PMLR (2018)