

GTAdam: Gradient Tracking With Adaptive Momentum for Distributed Online Optimization

Guido Carnevale , *Graduate Student Member, IEEE*, Francesco Farina ,
Ivano Notarnicola , *Member, IEEE*, and Giuseppe Notarstefano , *Member, IEEE*

Abstract—This article deals with a network of computing agents aiming to solve an online optimization problem in a distributed fashion, i.e., by means of local computation and communication, without any central coordinator. We propose the gradient tracking with an adaptive momentum estimation (GTAdam) distributed algorithm, which combines a gradient tracking mechanism with first- and second-order momentum estimates of the gradient. The algorithm is analyzed in the online setting for strongly convex cost functions with Lipschitz continuous gradients. We provide an upper bound for the dynamic regret given by a term related to the initial conditions and another term related to the temporal variations of the objective functions. Moreover, a linear convergence rate is guaranteed in the static setup. The algorithm is tested on a time-varying classification problem, on a (moving) target localization problem, and in a stochastic optimization setup from image classification. In these numerical experiments from multiagent learning, GTAdam outperforms state-of-the-art distributed optimization methods.

Index Terms—Distributed optimization, multiagent systems, networked systems, optimization algorithms.

I. INTRODUCTION

IN THIS article, we deal with online optimization problems over networks and propose a new distributed algorithm. In this framework, interconnected computing agents have only partial knowledge of the problem to solve but can exchange information with neighbors according to a given communication graph and without any central unit. In particular, we consider networks represented by a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$, where $\mathcal{V} = \{1, \dots, N\}$ is the set of agents, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges (or communication links), and $\mathcal{W} \in \mathbb{R}^{N \times N}$ is the (weighted) adjacency matrix of the graph. The matrix \mathcal{W} is compliant with the topology described by \mathcal{E} , i.e., being w_{ij} the (i, j) -entry of \mathcal{W} , then $w_{ij} > 0$ if $(i, j) \in \mathcal{E}$ and $w_{ij} = 0$ otherwise. We denote $\mathcal{N}_i = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\}$ the set of (in-)neighbors of agent i .

Manuscript received 3 August 2022; revised 18 November 2022; accepted 29 November 2022. Date of publication 27 December 2022; date of current version 17 August 2023. This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme Grant 638992 - OPT4SMART. Recommended by Associate Editor A. Olshevsky. (*Corresponding author: Guido Carnevale.*)

The authors are with the Department of Electrical, Electronic and Information Engineering, University of Bologna, 40126 Bologna, Italy (e-mail: guido.carnevale@unibo.it; farina.francesco@unibo.it; ivano.notarnicola@unibo.it; giuseppe.notarstefano@unibo.it).

Digital Object Identifier 10.1109/TCNS.2022.3232519

The aim of the network is to cooperatively solve the online optimization problem

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^N f_i^t(x), \quad t \geq 0 \quad (1)$$

where each $f_i^t : \mathbb{R}^n \rightarrow \mathbb{R}$ is a local function revealed only to agent i at time t . In the following, we let $f^t(x) \triangleq \sum_{i=1}^N f_i^t(x)$. This distributed optimization framework captures a variety of estimation and learning problems over networks, including distributed data classification and localization in smart sensor networks (see the recent survey [1] for an overview).

In this article, we address the distributed solution of the online optimization problem (1) in terms of *dynamic regret* (see, e.g., [1]). In particular, let x_i^t be the solution estimate of the problem at time t maintained by agent i , and let x_\star^t be a minimizer of $\sum_{i=1}^N f_i^t$. Then, the agents want to minimize the dynamic regret defined as

$$R_T \triangleq \sum_{t=1}^T f^t(\bar{x}^t) - \sum_{t=1}^T f^t(x_\star^t) \quad (2)$$

for a finite value $T > 1$ with $\bar{x}^t \triangleq \frac{1}{N} \sum_{i=1}^N x_i^t$. Another possible performance metric is the so-called static regret (see, e.g., [1]). The dynamic regret (2) is known to be more challenging than the static one [1] and, for this reason, consistently with the majority of the recent papers in the literature, this work focuses on the dynamic regret (2). As it is customary in the distributed setting, we also complement these measures with the consensus metric $\sum_{i=1}^N \|x_i^T - \bar{x}^T\|^2$, quantifying how far from consensus the local decisions are.

Related work: The proposed distributed algorithm combines a gradient tracking mechanism with adaptive estimation of first- and second-order momenta.

We organize the literature review into the following three main parts:

- 1) distributed algorithms for online optimization;
- 2) gradient tracking distributed schemes (mainly suited for static optimization);
- 3) centralized methods for online and stochastic optimization based on adaptive momentum estimation.

Online optimization problems, characterized by time-varying cost functions, have been originally addressed in the centralized

framework, see, e.g., [2], [3], and references therein, but recently have received significant attention also in the distributed optimization literature. In [4], an online optimization algorithm based on a distributed subgradient scheme is proposed. In [5], an adaptive diffusion algorithm is proposed to address changes regarding both the cost function and the constraints characterizing the problem. A class of coordination algorithms that generalize distributed online subgradient descent and saddle-point dynamics is proposed in [6] for network scenarios modeled by jointly connected graphs. An algorithm consisting of a subgradient flow combined with a push-sum consensus is studied in [7] for time-varying directed graphs. Cost uncertainties and switching communication topologies are addressed in [8] by using a distributed algorithm based on dual subgradient averaging. A distributed version of the mirror descent algorithm is proposed in [9] to address online optimization problems. In [10], an online algorithm based on the alternating direction method of multipliers is proposed, and in [11], time-varying inequality constraints are also considered. Online optimization is strictly related to stochastic optimization. Regarding distributed algorithms for stochastic optimization, in [12], Ram et al. investigate the convergence properties of a distributed algorithm dealing with subgradients affected by stochastic errors. In [13], a blockwise method is proposed to deal with high-dimensional stochastic problems, whereas in [14], a distributed gradient tracking method is analyzed in a stochastic setup.

The gradient tracking scheme, which we extend in this article, has been proposed in several variants in recent years and studied under different problem assumptions [15], [16], [17], [18], [19], [20], [21], [22]. This algorithm leverages a “signal tracking action” based on the dynamic average consensus (see [23] and [24]) in order to let the agents obtain a local estimate of the gradient of the whole cost function. Recently, in [25], the gradient tracking algorithm has been applied to online optimization problems. Finally, in [26], a dynamic gradient tracking update is combined with a recursive least squares scheme to address a distributed way the (centralized) personalized optimization framework introduced in [27].

The other algorithm inspiring our work is Adam, a centralized method originally proposed in [28]. Adam is an optimization algorithm based on adaptive estimates of first- and second-order gradient momenta that have been successfully employed in many online and stochastic optimization frameworks. Additional insights about Adam are given in [29], [30], and [31], where some frameworks in which the algorithm is not able to reach the optimal solution are also shown. This limitation is addressed in [32], where an effective extension of Adam, namely, AdaShift, is proposed. In [33], Nazari et al. proposed an enhanced version of the distributed gradient method with adaptive estimates of first- and second-order gradient momenta.

Contribution: The main contribution of this article is the design of a new distributed algorithm to solve online optimization problems for multiagent learning over networks. This novel scheme builds on the recently proposed gradient tracking distributed algorithm. Specifically, in the gradient tracking, the agents update their local solution estimates using a consensus averaging scheme perturbed with a local variable representing a

descent direction. This variable is concurrently updated using a dynamic consensus scheme aiming at reconstructing the total cost function gradient in a distributed way. Inspired by the centralized Adam algorithm, we accelerate the basic gradient tracking scheme by enhancing the descent direction resorting to first- and second-order momenta of the cost function gradient. The use of momenta turned out to be very effective in the centralized Adam to solve online optimization problems at a fast rate. Therefore, we design our novel gradient tracking with an adaptive momentum estimation (GTAdam) distributed algorithm to solve online optimization problems over networks. The algorithm relies on local estimators for the two momenta, in which the total gradient is replaced by a (local) gradient tracker. Although the intuition behind the construction of GTAdam is clear and consists of mimicking the centralized Adam in a distributed setting by using a gradient tracking scheme, its analysis presents several additional challenges with respect to both the gradient tracking and Adam. Indeed, being the descent direction a nonlinear combination of the local states updated through consensus averaging, the proof approach of the gradient tracking needs to be carefully reworked. We provide an upper bound about the dynamic regret for strongly convex online optimization problems. This bound consists of a constant term, related to the initial conditions of the algorithm, and another term depending on the temporal variations of both the optimal solution of the problem and the gradients of the objective functions. Thus, if the latter variations are sublinear with respect to time, then our bound about the dynamic regret is sublinear too. A similar result is also guaranteed for an agent-specific dynamic regret. Moreover, we show that in the static case, our algorithm reaches the optimal solution with a linear rate. Finally, we perform extensive numerical simulations on the following three application scenarios from distributed machine learning:

- 1) a classification problem via logistic regression;
- 2) a source localization problem in smart sensor networks;
- 3) an image classification task.

We show that GTAdam outperforms in all cases the current state-of-the-art algorithms in terms of convergence rate.

Organization and Notation: The article is organized as follows. In Section II, we recall the two algorithms that inspired the novel distributed algorithm proposed in this article. In Section III, GTAdam is presented with its convergence properties that are proved in Section IV. Finally, Section V shows numerical examples highlighting the advantages of GTAdam.

The vertical concatenation of the vectors v_1 and v_2 is $\text{col}(v_1, v_2)$. We use $\text{diag}(v)$ to denote the diagonal matrix with diagonal elements given by the components of v . The Hadamard product is denoted with \odot while the Kronecker product with \otimes . The identity matrix in $\mathbb{R}^{m \times m}$ is I_m while 0_m is the zero matrix in $\mathbb{R}^{m \times m}$. The column vector of N ones is denoted by $\mathbf{1}_N$ and we define $\mathbf{1} \triangleq \mathbf{1}_N \otimes I_n$. The spectral radius of a square matrix M is denoted as $\rho(M)$.

II. INSPIRING ALGORITHMS

In this section, we briefly recall two existing algorithms that represent the building blocks for GTAdam.

Algorithm 1: Adam.

initialization: x^0 arbitrary, $m^0 = v^0 = 0$, $g^0 = \nabla f^0(x^0)$
for $t = 1, 2, \dots$ **do**
 $\mathbf{m}^{t+1} = \beta_1 \mathbf{m}^t + (1 - \beta_1) \mathbf{g}^t$
 $\mathbf{v}^{t+1} = \beta_2 \mathbf{v}^t + (1 - \beta_2) \mathbf{g}^t \odot \mathbf{g}^t$
 $\mathbf{x}^{t+1} = \mathbf{x}^t - \alpha \frac{\sqrt{1-\beta_2}}{1-\beta_1} \frac{\mathbf{m}^{t+1}}{\sqrt{\mathbf{v}^{t+1} + \epsilon}}$
 $\mathbf{g}^{t+1} = \nabla f^{t+1}(\mathbf{x}^{t+1})$
end for

A. Adam Centralized Algorithm

Adam [28] is an optimization algorithm that solves problems in the form (1) in a *centralized* computation framework. It is an iterative gradientlike procedure in which, at each iteration t , a solution estimate \mathbf{x}^t is updated by means of a descent direction, which is enhanced by a proper use of the gradient history, i.e., through estimates of their first- and second-order momenta. Specifically, the (time-varying) gradient $\mathbf{g}^t = \nabla f^t(\mathbf{x}^t)$ of the function drives two exponential moving average estimators. The two estimates, denoted by \mathbf{m}^t and \mathbf{v}^t , represent, respectively, mean and variance (first and second momenta) of the gradient sequence and are nonlinearly combined to build the descent direction. A pseudocode of the Adam algorithm is reported in Algorithm 1 in which $\alpha > 0$ is the step size, the constant $0 < \epsilon \ll 1$ is introduced to guarantee the numerical robustness of the scheme while the hyperparameters $\beta_1, \beta_2 \in (0, 1)$ control the exponential-decay rate of the moving average dynamics.

We point out that in the algorithm above, the ratio $\frac{\mathbf{m}^{t+1}}{\sqrt{\mathbf{v}^{t+1} + \epsilon}}$ is meant elementwise. Typical choices for the algorithmic parameters are $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$.

B. Gradient Tracking Distributed Algorithm

The gradient tracking is a distributed algorithm mainly tailored to *static* instances of problem (1). Agents in a network maintain and update two local states x_i^t and s_i^t by iteratively combining a perturbed average consensus and a dynamic tracking mechanism. Consensus is used to enforce agreement among the local agents' estimates x_i^t . The agreement is also locally perturbed in order to steer the local estimates toward a (static) optimal solution of the problem. The perturbation is obtained by using a tracking scheme that allows agents to locally reconstruct a progressively accurate estimate of the whole gradient of the (static) cost function in a distributed way. A pseudocode of the gradient tracking distributed algorithm is reported in Algorithm 2, in which \mathcal{N}_i denotes the set of (in-)neighbors of agent i while $\alpha > 0$ is the step size. The protocol is shown from the perspective of agent i only.

III. GRADIENT TRACKING WITH ADAPTIVE MOMENTUM ESTIMATION

In this section, we present the main contribution of this article, i.e., the GTAdam distributed algorithm. GTAdam is designed to address in a distributed fashion problem (1), taking inspiration

Algorithm 2: Gradient Tracking (For Agent i).

initialization: x_i^0 arbitrary, $s_i^0 = g_i^0 = \nabla f_i(x_i^0)$
for $t = 1, 2, \dots$ **do**
 $x_i^{t+1} = \sum_{j \in \mathcal{N}_i} w_{ij} x_j^t - \alpha s_i^t$
 $g_i^{t+1} = \nabla f_i(x_i^{t+1})$
 $s_i^{t+1} = \sum_{j \in \mathcal{N}_i} w_{ij} s_j^t + g_i^{t+1} - g_i^t$
end for

Algorithm 3: GTAdam (For Agent i).

initialization: x_i^0 arbitrary, $s_i^0 = g_i^0 = \nabla f_i^0(x_i^0)$,
 $m_i^0 = v_i^0 = 0$
for $t = 1, \dots, T$ **do**
 $m_i^{t+1} = \beta_1 m_i^t + (1 - \beta_1) s_i^t$
 $v_i^{t+1} = \min\{\beta_2 v_i^t + (1 - \beta_2) s_i^t \odot s_i^t, G\}$
 $x_i^{t+1} = \sum_{j \in \mathcal{N}_i} w_{ij} x_j^t - \alpha \frac{m_i^{t+1}}{\sqrt{v_i^{t+1} + \epsilon}}$
 $g_i^{t+1} = \nabla f_i^{t+1}(x_i^{t+1})$
 $s_i^{t+1} = \sum_{j \in \mathcal{N}_i} w_{ij} s_j^t + g_i^{t+1} - g_i^t$
end for

both from Adam and from the gradient tracking distributed algorithm.

Along the evolution of the algorithm, each agent i maintains the following four local states:

- 1) a local estimate x_i^t of the current optimal solution x_*^t ;
- 2) an auxiliary variable s_i^t , whose role is to track the gradient of the whole cost function;
- 3) an estimate m_i^t of the first momentum of s_i^t ;
- 4) an estimate v_i^t of the second momentum of s_i^t .

The momentum estimates of s_i^t are initialized as $m_i^0 = v_i^0 = 0$ while the tracker of the gradient is initialized as $s_i^0 = \nabla f_i^0(x_i^0)$.

The algorithm works as follows. At each time instant t , each agent i performs the following operations:

- 1) it updates the moving averages m_i^t and v_i^t ;
- 2) it computes a weighted average of the solution estimates of its neighbors and, starting from this point, it uses the updated direction $\frac{m_i^{t+1}}{\sqrt{v_i^{t+1} + \epsilon}}$ to compute the new solution estimate x_i^{t+1} ;
- 3) it updates the local gradient tracker s_i^t via a ‘‘dynamic consensus’’ mechanism.

A pseudocode of GTAdam is reported in Algorithm 3.

Some remarks are in order. The algorithm proposed in this article is different from [33]. In fact, although they both use a similar strategy involving first- and second-order momenta, in that work, only local gradients are considered, without resorting to any tracking mechanism. Note that a saturation term $G \gg 0$ is introduced in the update of v_i^t , where the min operator is to be intended elementwise. The value of G guarantees a bound for the scaling factor that multiplies the descent direction. Such a bound will turn out to be important for analysis purposes. We suggest to take it proportional to the initial estimates v_i^0 .

We now state some regularity requirements on problem (1). We first make two assumptions regarding each f_i^t .

Assumption 1 (Lipschitz continuous gradients): The functions f_i^t have L -Lipschitz continuous gradients for all $i \in \{1, \dots, N\}$ and $t \geq 0$.

Assumption 2 (Strong convexity): The functions f_i^t are μ -strongly convex for all $i \in \{1, \dots, N\}$ and $t \geq 0$.

We point out that, in light of Assumption 2, the minimizer x_\star^t is unique for all $t \geq 0$. Finally, the following characterizes the communication structure.

Assumption 3 (Network Structure): The weighted graph \mathcal{G} is connected with doubly stochastic matrix \mathcal{W} stochastic.

In order to analyze GTAdam, we rewrite it into an aggregate form. Given the variables $\{x_i^t\}_{i=1}^N$, we define $\mathbf{x}^t \triangleq \text{col}(x_1^t, \dots, x_N^t)$ and their average as $\bar{x}^t \triangleq \frac{1}{N} \sum_{i=1}^N x_i^t$. Similar definitions apply to the quantities $\mathbf{m}^t, \mathbf{v}^t, \mathbf{d}^t, \mathbf{g}^t, \mathbf{s}^t$ and their averages $\bar{m}^t, \bar{v}^t, \bar{d}^t, \bar{s}^t$. With these definitions at hand, GTAdam can be rephrased from a global perspective as

$$\mathbf{m}^{t+1} = \beta_1 \mathbf{m}^t + (1 - \beta_1) \mathbf{s}^t \quad (3a)$$

$$\mathbf{v}^{t+1} = \min\{\beta_2 \mathbf{v}^t + (1 - \beta_2) \mathbf{s}^t \odot \mathbf{1}G\} \quad (3b)$$

$$\mathbf{d}^{t+1} = (\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{m}^{t+1} \quad (3c)$$

$$\mathbf{x}^{t+1} = W \mathbf{x}^t - \alpha \mathbf{d}^{t+1} \quad (3d)$$

$$\mathbf{s}^{t+1} = W \mathbf{s}^t + \mathbf{g}^{t+1} - \mathbf{g}^t \quad (3e)$$

where we set $W \triangleq \mathcal{W} \otimes I_n$, $\mathbf{V}^t \triangleq \text{diag}(\mathbf{v}^t)$, and $\bar{v}^t \triangleq \text{diag}(\bar{v}^t)$. Moreover, the averaged quantities of (3) satisfy

$$\bar{m}^{t+1} = \beta_1 \bar{m}^t + (1 - \beta_1) \bar{s}^t \quad (4a)$$

$$\bar{v}^{t+1} = \min\{\beta_2 \bar{v}^t + (1 - \beta_2) \bar{s}^t \odot \mathbf{1}G\} \quad (4b)$$

$$\bar{d}^{t+1} = \frac{1}{N} \mathbf{1}^\top \mathbf{d}^{t+1} \quad (4c)$$

$$\bar{x}^{t+1} = \bar{x}^t - \alpha \bar{d}^{t+1} \quad (4d)$$

$$\bar{s}^{t+1} = \bar{s}^t + \frac{1}{N} \sum_{i=1}^N (g_i^{t+1} - g_i^t). \quad (4e)$$

Our analysis is based on studying the aggregate dynamical evolution of the following: average first momentum $\|\bar{m}^t\|$, average tracking momentum difference $\|\bar{s}^t - \bar{m}^t\|$, first momentum error $\|\mathbf{m}^t - \mathbf{1}\bar{m}^t\|$, gradient tracking error $\|\mathbf{s}^t - \mathbf{1}\bar{s}^t\|$, consensus error $\|\mathbf{x}^t - \mathbf{1}\bar{x}^t\|$, and solution error $\|\bar{x}^t - x_\star^t\|$. Let y^t be the vector stacking the above quantities at iterations t

$$y^t \triangleq \begin{bmatrix} \|\bar{m}^t\| \\ \|\bar{s}^t - \bar{m}^t\| \\ \|\mathbf{m}^t - \mathbf{1}\bar{m}^t\| \\ \|\mathbf{s}^t - \mathbf{1}\bar{s}^t\| \\ \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| \\ \|\bar{x}^t - x_\star^t\| \end{bmatrix}. \quad (5)$$

Notice that due to the distributed context and no assumptions on the boundedness of the gradients, we need to take into account all these quantities to study the convergence. Let us introduce two useful variables that will be used to provide the main result

of the article, namely

$$\begin{aligned} \eta^t &\triangleq \sup_i \sup_{x \in \mathbb{R}^n} \|\nabla f_i^{t+1}(x) - \nabla f_i^t(x)\| \\ \zeta^t &\triangleq \|x_\star^{t+1} - x_\star^t\|. \end{aligned} \quad (6)$$

Then, the main result of this article is stated as follows.

Theorem 1: Consider GTAdam as given in Algorithm 3. Let Assumptions 1, 2, and 3 hold. Then, for a sufficiently small step-size $\alpha > 0$, there exists a constant $0 < \tilde{\rho} < 1$, such that

$$R_T \leq \frac{L\lambda^2}{2} \left(\frac{\|y^0\|^2}{1 - \tilde{\rho}^2} + 2\|y^0\| S_T + Q_T \right) \quad (7)$$

where R_T is defined in (2), the constant λ is defined in the proof [cf. (18)] and

$$S_T \triangleq \sum_{t=1}^T \sum_{k=0}^{t-1} \tilde{\rho}^{t+k} \left(\frac{N+1}{\sqrt{N}} \|\eta^{t-k-1}\| + \|\zeta^{t-k-1}\| \right) \quad (8a)$$

$$Q_T \triangleq \sum_{t=1}^T \left(\sum_{k=0}^{t-1} \tilde{\rho}^k \frac{N+1}{\sqrt{N}} \|\eta^{t-k-1}\| + \|\zeta^{t-k-1}\| \right)^2 \quad (8b)$$

where η^t, ζ^t are defined in (6) and we assume that they are finite. Moreover, it holds

$$\lim_{T \rightarrow \infty} \sum_{i=1}^N \|x_i^T - \bar{x}^T\|^2 \leq \frac{\lambda^2}{(1-\tilde{\rho})^2} \max_t \left\{ \frac{N^2+1}{N} \eta^t + \zeta^t \right\}. \quad (9)$$

As it requires several intermediate results, the proof of Theorem 1 is carried out in Section IV.

There is evidence in the literature, see, e.g., [1], [9], [26], [34], [35], [36] that the bound on the dynamic regret cannot be sublinear with respect to T . As stated, e.g., in [1], when the objective functions are strongly convex and have bounded gradients, the bound on dynamic regret is $O(1 + \eta^t)$. Our work does not assume gradient boundedness and, thus, our bound has additional terms due to variations over time of the gradients. Specifically, Theorem 1 shows that R_T is upper bounded by a constant depending on the initial conditions and by the other two terms. The latter involves S_T and Q_T , which capture the time-varying nature of the problem itself. Indeed, suppose that the problem varies linearly, i.e., there exists $C > 0$ so that $\eta^t, \zeta^t \leq C$ for all $t \geq 0$. Then, being $\tilde{\rho} \in (0, 1)$, we can exploit the geometric series properties to write the following:

$$S_T \leq \frac{(N + \sqrt{N} + 1)(\tilde{\rho} - \tilde{\rho}^{T+1})(1 - \tilde{\rho}^T)C}{\sqrt{N}(1 - \tilde{\rho})^2}$$

$$Q_T \leq \frac{(N + \sqrt{N} + 1)^2 (1 - \tilde{\rho}^T)^2 C^2 T}{N(1 - \tilde{\rho})^2}.$$

In this case, (7) ensures that the average regret R_T/T asymptotically approaches a constant when $T \rightarrow \infty$, specifically

$$\lim_{T \rightarrow \infty} \frac{R_T}{T} \leq \frac{L\lambda^2(N^2 + \sqrt{N} + 1)^2 C^2}{2N(1 - \tilde{\rho})^2}.$$

The key point of the proof consists in showing that the error vector y^t (5) evolves according to a linear system with state matrix $A(\alpha)$ (whose entries depend on the problem parameters,

e.g., the strong convexity function or the network connectivity), which is perturbed by an input q^t related to the variations of the problem over time (11). Notice that the parameter $\tilde{\rho}$ is related to the spectral radius of $A(\alpha)$ and, thus, depends also on the network topology.

Agent Regret: We may also consider a regret for each agent i defined as $R_{T,i} \triangleq \sum_{t=1}^T f^t(x_i^t) - \sum_{t=1}^T f^t(x_\star^t)$.

Corollary 1: Under the same assumptions of Theorem 1, for all $i \in \{1, \dots, N\}$, it holds

$$R_{T,i} \leq 2L\lambda^2 \left(\frac{\|y^0\|^2}{1-\tilde{\rho}^2} + 2\|y^0\|S_T + Q_T \right)$$

where λ , $\tilde{\rho}$, S_T , and Q_T are defined as in Theorem 1.

The proof is given in Appendix G.

Static Set-up: We provide an additional corollary of Theorem 1 asserting theoretical guarantees in a static scenario. Specifically, for this special case, the GTAdam distributed algorithm converges to the optimal solution with a linear rate.

Corollary 2 (Static Set-up): Under the same assumptions of Theorem 1, if additionally holds $f^t = f$ for all $t \geq 0$, then, for a sufficiently small step-size $\alpha > 0$, there exists a constant $0 < \tilde{\rho} < 1$ such that

$$f(\bar{x}^t) - f(x_\star^t) \leq \tilde{\rho}^{2t} \frac{L\lambda^2}{2} \|y^0\|^2 \quad (10)$$

where the constant λ is defined in (18).

The proof is given in Appendix H.

IV. ANALYSIS

This section is devoted to provide the proof of Theorem 1.

A. Preparatory Lemmas

We now give a sequence of intermediate results, providing proper bounds on the components of y^t [defined in (5)], that are then used as building blocks for proving Theorem 1.

Lemma 1 (Average first momentum magnitude): Let Assumption 1 holds. Then, for all $t \geq 0$, it holds

$$\|\bar{m}^{t+1}\| \leq \beta_1 \|\bar{m}^t\| + \frac{(1-\beta_1)L}{\sqrt{N}} \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + (1-\beta_1)L \|\bar{x}^t - x_\star^t\|.$$

The proof is given in Appendix A.

Lemma 2 (First momentum error): For all $t \geq 0$, it holds

$$\|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\| \leq \beta_1 \|\mathbf{m}^t - \mathbf{1}\bar{m}^t\| + (1-\beta_1) \|\mathbf{s}^t - \mathbf{1}\bar{s}^t\|.$$

The proof of Lemma 2 follows by combining (3a) and (4a) with the triangle inequality.

Lemma 3 (Input signal error): For all $t \geq 0$, it holds

$$\begin{aligned} \|\mathbf{d}^{t+1} - \mathbf{1}\bar{d}^{t+1}\| &\leq \frac{\beta_1\sqrt{N}}{\sqrt{\epsilon}} \|\bar{m}^t\| + \frac{\beta_1}{\sqrt{\epsilon}} \|\mathbf{m}^t - \mathbf{1}\bar{m}^t\| + \frac{(1-\beta_1)}{\sqrt{\epsilon}} \|\mathbf{s}^t - \bar{s}^t\| \\ &\quad + \frac{(1-\beta_1)L}{\sqrt{\epsilon}} \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + \frac{(1-\beta_1)\beta_1 L\sqrt{N}}{\sqrt{\epsilon}} \|\bar{x}^t - x_\star^t\|. \end{aligned}$$

The proof is given in Appendix B.

Lemma 4 (Tracking error): Let Assumptions 1, 2, and 3 hold. Then, for all $t \geq 0$, it holds

$$\begin{aligned} \|\mathbf{s}^{t+1} - \mathbf{1}\bar{s}^{t+1}\| &\leq \left(\sigma_W + \alpha \frac{2(1-\beta_1)L}{\sqrt{\epsilon}} \right) \|\mathbf{s}^t - \mathbf{1}\bar{s}^t\| \\ &\quad + \alpha \frac{2\beta_1 L\sqrt{N}}{\sqrt{\epsilon}} \|\bar{m}^t\| + \alpha \frac{2\beta_1 L}{\sqrt{\epsilon}} \|\mathbf{m}^t - \mathbf{1}\bar{m}^t\| \\ &\quad + \left(L\|W-I\| + \alpha \frac{2(1-\beta_1)\beta_1 L^2}{\sqrt{\epsilon}} \right) \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| \\ &\quad + \alpha \frac{(1-\beta_1)(1+\beta_1)L^2\sqrt{N}}{\sqrt{\epsilon}} \|\bar{x}^t - x_\star^t\| + \sqrt{N}\eta^t \end{aligned}$$

where $\sigma_W \in (0, 1)$ is the spectral radius of $W - \frac{1}{N}\mathbf{1}\mathbf{1}^\top$ and η^t has been defined in (6).

The proof is given in Appendix C.

Lemma 5 (Consensus error): Let Assumptions 1 and 3 hold. Then, for all $t \geq 0$, it holds

$$\begin{aligned} \|\mathbf{x}^{t+1} - \mathbf{1}\bar{x}^{t+1}\| &\leq \left(\sigma_W + \alpha \frac{(1-\beta_1)L}{\sqrt{\epsilon}} \right) \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| \\ &\quad + \alpha \frac{\beta_1\sqrt{N}}{\sqrt{\epsilon}} \|\bar{m}^t\| + \alpha \frac{\beta_1}{\sqrt{\epsilon}} \|\mathbf{m}^t - \mathbf{1}\bar{m}^t\| \\ &\quad + \alpha \frac{(1-\beta_1)}{\sqrt{\epsilon}} \|\mathbf{s}^t - \bar{s}^t\| + \alpha \frac{(1-\beta_1)\beta_1 L\sqrt{N}}{\sqrt{\epsilon}} \|\bar{x}^t - x_\star^t\|. \end{aligned}$$

The proof is given in Appendix D.

Lemma 6 (Tracking momentum difference magnitude): Let Assumptions 1, 2, and 3 hold. Then, for all $t \geq 0$, it holds

$$\begin{aligned} \|\bar{s}^{t+1} - \bar{m}^{t+1}\| &\leq \beta_1 \|\bar{s}^t - \bar{m}^t\| + \alpha \frac{\beta_1 L}{\sqrt{\epsilon}} \|\bar{m}^t\| \\ &\quad + \alpha \frac{2\beta_1 L}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{m}^t - \mathbf{1}\bar{m}^t\| + \alpha \frac{L}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{s}^t - \mathbf{1}\bar{s}^t\| \\ &\quad + \left(\sigma_W \frac{L}{\sqrt{N}} + \frac{L}{\sqrt{N}} + \alpha \frac{(1-\beta_1)L^2}{\sqrt{\epsilon}\sqrt{N}} \right) \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| \\ &\quad + \alpha \frac{(1-\beta_1)L^2}{\sqrt{\epsilon}} \|\bar{x}^t - x_\star^t\| + \frac{1}{\sqrt{N}}\eta^t. \end{aligned}$$

The proof is given in Appendix E.

Lemma 7 (Solution error): Let Assumptions 1, 2, and 3 hold. Then, for all $t \geq 1$, it holds

$$\begin{aligned} \|\bar{x}^{t+1} - x_\star^{t+1}\| &\leq (1-\alpha\delta) \|\bar{x}^t - x_\star^t\| + \alpha \frac{\beta_1}{\sqrt{\epsilon}} \|\bar{s}^t - \bar{m}^t\| + \alpha \frac{L}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| \\ &\quad + \alpha \frac{\beta_1}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{m}^t - \mathbf{1}\bar{m}^t\| + \alpha \frac{(1-\beta_1)}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{s}^t - \mathbf{1}\bar{s}^t\| + \zeta^t \end{aligned}$$

where ζ^t is defined in (6) and $\delta \triangleq \min\{\frac{\mu}{\sqrt{\epsilon}+G}, \frac{L}{\sqrt{\epsilon}}\}$.

The proof is given in Appendix F.

B. Proof of Theorem 1

By recalling the definition of y^t given in (5) and combining Lemma 1, 2, 4, 5, 6, and 7, it is possible to write

$$y^{t+1} \leq A(\alpha)y^t + q^t \quad (11)$$

where $q^t \triangleq \text{col}(0, \frac{1}{\sqrt{N}}\eta^t, 0, \sqrt{N}\eta^t, 0, \zeta^t)$. The matrix $A(\alpha)$ can be decomposed in $A(\alpha) \triangleq A_0 + \alpha E$, with

$$A_0 \triangleq \begin{bmatrix} \beta_1 & 0 & 0 & 0 & \beta_1 c_1 & (1-\beta_1)L \\ 0 & \beta_1 & 0 & 0 & \sigma_W c_1 + c_1 & 0 \\ 0 & 0 & \beta_1 & 1-\beta_1 & 0 & 0 \\ 0 & 0 & 0 & \sigma_W & c_2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_W & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and

$$E \triangleq \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\beta_1 L}{\sqrt{\epsilon}} & 0 & \frac{2\beta_1 c_1}{\sqrt{\epsilon}} & \frac{c_1}{\sqrt{\epsilon}} & \frac{(1-\beta_1)c_1 L}{\sqrt{\epsilon}} & \frac{(1-\beta_1)L^2}{\sqrt{\epsilon}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{2\beta_1 L\sqrt{N}}{\sqrt{\epsilon}} & 0 & \frac{2\beta_1 L}{\sqrt{\epsilon}} & \frac{2(1-\beta_1)L}{\sqrt{\epsilon}} & c_3 & c_4 \\ \frac{\beta_1\sqrt{N}}{\sqrt{\epsilon}} & 0 & \frac{\beta_1}{\sqrt{\epsilon}} & \frac{1-\beta_1}{\sqrt{\epsilon}} & 0 & c_5 \\ 0 & \frac{\beta_1}{\sqrt{\epsilon}} & \frac{\beta_1}{\sqrt{\epsilon\sqrt{N}}} & 0 & \frac{(1-\beta_1)}{\sqrt{\epsilon\sqrt{N}}} & -\delta \end{bmatrix}$$

where we used the following shorthands:

$$c_1 \triangleq \frac{L}{\sqrt{N}}, \quad c_2 \triangleq L\|W - I\|, \quad c_3 \triangleq \frac{2(1-\beta_1)\beta_1 L^2}{\sqrt{\epsilon}}$$

$$c_4 \triangleq \frac{(1-\beta_1)(1+\beta_1)L^2\sqrt{N}}{\sqrt{\epsilon}}, \quad c_5 \triangleq \frac{(1-\beta_1)\beta_1 L\sqrt{N}}{\sqrt{\epsilon}}.$$

Being A_0 triangular, it is easy to see that its spectral radius is 1 since both β_1 and σ_W are in $(0, 1)$. We want to study how the perturbation matrix αE affects the simple eigenvalue 1 of A_0 . Hence, we denote by $\chi(\alpha)$ such eigenvalue of $A(\alpha)$ as a function of α . Call w and v , respectively, the left and right eigenvectors of A_0 associated with the eigenvalue 1, then $w = \text{col}(0, 0, 0, 0, 0, 1)$ and $v = \text{col}(L, 0, 0, 0, 0, 1)$. Since the eigenvalue 1 is simple, from [37, Th. 6.3.12], it holds

$$\left. \frac{d\chi(\alpha)}{d\alpha} \right|_{\alpha=0} = \frac{w^\top E v}{w^\top v} = -\delta < 0.$$

Then, by continuity of eigenvalues with respect to the matrix entries, $\chi(\alpha)$ is strictly less than 1 for sufficiently small $\alpha > 0$. Then, it is always possible to choose $\alpha > 0$ so as the remaining eigenvalues stay in the unit circle. Therefore, the spectral radius is $\rho(A(\alpha)) < 1$. Moreover, since $A(\alpha)$ and q^t have only nonnegative entries, one can use (11) to write

$$y^t \leq A(\alpha)^t y^0 + \sum_{k=0}^{t-1} A(\alpha)^{t-1-k} q^k. \quad (12)$$

From [37, Lemma 5.6.10], we have that for any $\gamma > 0$, there exists a matrix norm, say $\|\cdot\|_\gamma$, such that

$$\|A(\alpha)\|_\gamma \leq \rho(A(\alpha)) + \gamma. \quad (13)$$

Let us pick $\gamma \in (0, 1 - \rho(A(\alpha)))$ and define $\tilde{\rho} \triangleq \rho(A(\alpha)) + \gamma$. Then, in light of (13) it holds $\|A(\alpha)\|_\gamma \leq \tilde{\rho} < 1$. Moreover, by applying [37, Th. 5.7.13], there exists a vector norm $\|\cdot\|_\gamma$ such that $\|Mv\|_\gamma \leq \|M\|_\gamma \|v\|_\gamma$ for any matrix $M \in \mathbb{R}^{6 \times 6}$ and

$v \in \mathbb{R}^6$. Hence, we can manipulate (12) taking the norm and using the triangle inequality to write

$$\|y^t\|_\gamma \leq \|A(\alpha)^t y^0\|_\gamma + \left\| \sum_{k=0}^{t-1} A(\alpha)^{t-1-k} q^k \right\|_\gamma$$

$$\leq \tilde{\rho}^t \|y^0\|_\gamma + \sum_{k=0}^{t-1} \tilde{\rho}^k \|q^{t-1-k}\|_\gamma \quad (14)$$

which shows that first term decreases linearly with rate $\tilde{\rho} < 1$ while the second one is bounded. By using the Lipschitz continuity of the gradients of f^t (cf. Assumption 1), we have

$$f^t(\bar{x}^t) - f^t(x_\star^t) \leq \frac{L}{2} \|\bar{x}^t - x_\star^t\|^2 \stackrel{(a)}{\leq} \frac{L}{2} \|y^t\|^2 \quad (15)$$

where, in (a), we use the fact that $\|\bar{x}^t - x_\star^t\|$ represents a component of y^t leading to the trivial bound $\|\bar{x}^t - x_\star^t\| \leq \|y^t\|$. Recalling that all norms are equivalent on finite-dimensional vector spaces, there always exist $\lambda_1 > 0$ and $\lambda_2 > 0$ such that

$$\|\cdot\| \leq \lambda_1 \|\cdot\|_\gamma \quad (16a)$$

$$\|\cdot\|_\gamma \leq \lambda_2 \|\cdot\|. \quad (16b)$$

Thus, by applying (16a), we bound (15) as

$$f^t(\bar{x}^t) - f^t(x_\star^t) \leq \frac{L\lambda_1}{2} \|y^t\|_\gamma^2$$

which, combined with the definition of R_T [cf. (2)] and the result (14), leads to

$$R_T \leq \frac{L\lambda_1^2}{2} \left(\sum_{t=1}^T \tilde{\rho}^{2t} \|y^0\|_\gamma^2 + 2\|y^0\|_\gamma \sum_{t=1}^T \sum_{k=0}^{t-1} \tilde{\rho}^{t+k} \|q^{t-1-k}\|_\gamma \right.$$

$$\left. + \sum_{t=1}^T \left(\sum_{k=0}^{t-1} \tilde{\rho}^k \|q^{t-1-k}\|_\gamma \right)^2 \right)$$

$$\stackrel{(a)}{\leq} \frac{L\lambda_1^2 \lambda_2^2}{2} \left(\frac{\|y^0\|^2}{1-\tilde{\rho}^2} + 2\|y^0\| \sum_{t=1}^T \sum_{k=0}^{t-1} \tilde{\rho}^{t+k} \|q^{t-1-k}\| \right.$$

$$\left. + \sum_{t=1}^T \left(\sum_{k=0}^{t-1} \tilde{\rho}^k \|q^{t-1-k}\| \right)^2 \right) \quad (17)$$

where, in (a), we use the geometric property series and the relation (16b). The proof follows by using the triangle inequality, the definitions of U_T and Q_T [cf. (8)], and by setting

$$\lambda \triangleq \lambda_1 \lambda_2. \quad (18)$$

Finally, in order to prove (9), we notice that $\sum_{i=1}^N \|x_i^T - \bar{x}^T\|^2 \leq \|y^T\|^2 \leq \lambda_1^2 \|y^T\|_\gamma^2$, in which we apply (16a). By applying the bound (14) for $t = T$, we get

$$\|y^T\|_\gamma \leq \tilde{\rho}^T \|y^0\|_\gamma + \sum_{k=0}^{T-1} \tilde{\rho}^k \|q^{T-k-1}\|_\gamma.$$

The first term of the latter inequality vanishes as $T \rightarrow \infty$ while the second one can be bounded by relying on the geometric series property and $\max_k \{\|q^k\|^2\}$. By exploiting these arguments, we

can write

$$\begin{aligned} \lim_{T \rightarrow \infty} \sum_{i=1}^N \|x_i^T - \bar{x}^T\|^2 &\leq \frac{\lambda_1^2}{(1-\rho)^2} \max_t \left\{ \|q^t\|_\gamma^2 \right\} \\ &\stackrel{(a)}{\leq} \frac{\lambda^2}{(1-\rho)^2} \max_t \left\{ \|q^t\|^2 \right\} \end{aligned} \quad (19)$$

where, in (a), we apply (16b) and the definition (18) of λ . The result (9) follows by noticing that:

$$\max_t \left\{ \|q^t\|^2 \right\} = \max_t \left\{ \frac{N^2+1}{N} \eta^t + \zeta^t \right\}.$$

V. NUMERICAL EXPERIMENTS

In this section, we consider three multiagent distributed learning problems to show the effectiveness of GTAdam. The first scenario regards the computation of a linear classifier via a regularized logistic regression function for a set of points that change over time. The second scenario involves the localization of a moving target. The third example is a stochastic optimization problem arising in a distributed image classification task. In all the examples, the parameters of GTAdam are chosen as $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. Moreover, we compare GTAdam with the gradient tracking distributed algorithm (GT) (cf. Algorithm 2 in Section II), the distributed gradient descent (DGD) (see [38]), and the distributed Adam (DAdam) (see [33]) described by

$$\begin{aligned} m_i^{t+1} &= \beta_1 m_i^t + (1 - \beta_1) \nabla f_i^t(x_i^t) \\ v_i^{t+1} &= \beta_2 v_i^t + (1 - \beta_2) \nabla f_i^t(x_i^{t+1}) \odot \nabla f_i^t(x_i^{t+1}) \\ \tilde{v}_i^{t+1} &= \beta_3 \tilde{v}_i^t + (1 - \beta_3) \max\{v_i^t, v_i^{t+1}\} \\ x_i^{t+1} &= \sum_{j \in \mathcal{N}_i} w_{ij} x_j^t + \gamma^t \frac{m_i^{t+1}}{\tilde{v}_i^{t+1}} \end{aligned}$$

for all $i \in \{1, \dots, N\}$. As suggested in [33], we set $\beta_1 = \beta_3 = 0.9$, $\beta_2 = 0.999$, and a diminishing step-size $\gamma^t = (\frac{\alpha}{t})^{-1/2}$ for some $\alpha > 0$.

A. Distributed Classification via Logistic Regression

Consider a network of agents that want to cooperatively train a linear classifier for a set of (moving) points in a given feature space. At time $t \geq 0$, each agent i is equipped with $m_i \in \mathbb{N}$ points $p_{i,1}^t, \dots, p_{i,m_i}^t \in \mathbb{R}^d$ with binary labels $l_{i,k} \in \{-1, 1\}$ for all $k \in \{1, \dots, m_i\}$. The problem consists of building a linear classification model from the given points, also called training samples. In particular, we look for a separating hyperplane described by a pair $(w, b) \in \mathbb{R}^d \times \mathbb{R}$ given by $\{p \in \mathbb{R}^d \mid w^\top p + b = 0\}$. This *online* classification problem can be posed at each time $t \geq 0$, as a minimization problem described by

$$\min_{w,b} \sum_{i=1}^N \sum_{k=1}^{m_i} \log \left(1 + e^{-l_{i,k} (w^\top p_{i,k}^t + b)} \right) + \frac{C}{2} (\|w\|^2 + b^2) \quad (20)$$

where $C > 0$ is the so-called regularization parameter. Notice that the presence of the regularization makes the cost function strongly convex. Each point $p_{i,k}^t \in \mathbb{R}^2$ moves along a circle of

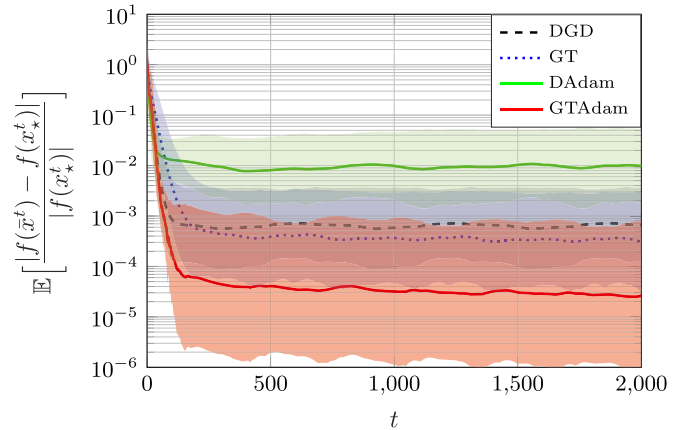


Fig. 1. Distributed classification via logistic regression. Mean of the relative cost errors and 1-standard deviation band obtained with Monte Carlo simulations consisting of 100 trials in which each of the $N = 50$ agents is equipped with $m = 5$ points.

radius $r = 1$ according to the following law:

$$p_{i,k}^t = p_{i,k}^c + r \begin{bmatrix} \cos(t/100) \\ \sin(t/100) \end{bmatrix}$$

where $p_{i,k}^c \in \mathbb{R}^2$ represents the randomly generated center of the considered circle. We consider a network of $N = 50$ agents and pick $m_i = 5$ (for all i). We performed an experimental tuning to optimize the step sizes to enhance the convergence properties of each algorithm. In particular, we selected $\alpha = 0.1$ for GTAdam, $\alpha = 0.05$ for GT, $\alpha = 0.1$ for DGD, and $\alpha = 0.1$ for DAdam. We performed Monte Carlo simulations consisting of 100 trials, in which we alternatively consider an undirected, connected Erdős-Rényi graph with connectivity parameter 0.5, and a ring graph. In Fig. 1, we plot the average across the trials of the relative cost error, namely $\frac{f^t(\bar{x}^t) - f^t(x_*^t)}{f^t(x_*^t)}$, with x_*^t being the minimum of f^t for all t . The plot highlights that GTAdam exhibits a faster convergence compared to the other algorithms and achieves a smaller tracking error.

Finally, we consider a static instance of problem (20), i.e., with fixed objective function $f_i^t = f_i$ for all $t \geq 0$ and $i \in \{1, \dots, N\}$. We consider a network of $N = 50$ agents in a ring topology. We take $\alpha = 0.001$ for GTAdam, $\alpha = 0.01$ for GT, $\alpha = 0.1$ for DGD, and $\alpha = 0.5$ for DAdam. In Fig. 2, we plot the error $\|\bar{x}^t - x_*\|$ achieved by the considered methods, where $x_* \in \mathbb{R}^d$ is the (fixed) optimal solution of the problem. Fig. 2 clearly shows the benefit of the tracking mechanism, which allows GTAdam and GT to achieve the exact problem solution. The plot also shows that GTAdam is faster than GT.

B. Distributed Source Localization in Smart Sensor Networks

The estimation of the exact position of a source is a key task in several applications in multiagent distributed estimation and learning. Here, we consider an online version of the static localization problem considered in [39, Sec. 4.2]. An acoustic source is positioned at an unknown and time-varying location $\theta_{\text{target}}^t \in \mathbb{R}^2$. A network of N sensors is capable to measure an

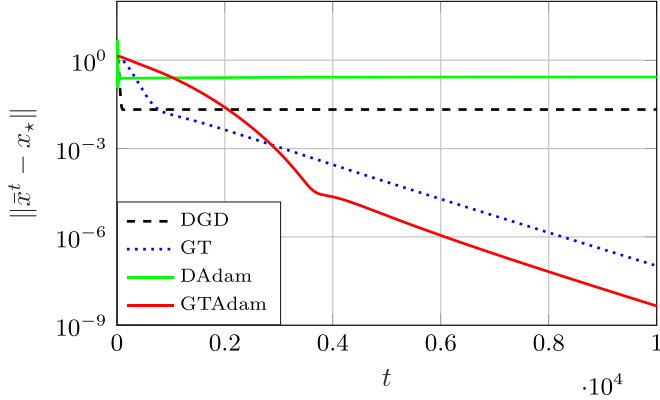


Fig. 2. Distributed classification via logistic regression. Static set-up in which each of the $N = 50$ agents is equipped with $m = 5$ points.

isotropic signal related to such location and aims at cooperatively estimating θ_{target}^t . Each sensor is placed at a fixed location $c_i \in \mathbb{R}^2$ and takes, at each time instant, a noisy measurement according to an isotropic propagation model $\omega_i^t \triangleq \frac{A}{\|\theta_{\text{target}}^t - c_i\|^\gamma} + \epsilon_i^t$, where $A > 0$, $\gamma \geq 1$ describes the attenuation characteristics of the medium through which the signal propagates, and ϵ_i^t is a zero-mean Gaussian noise with variance σ^2 . With these data, each node i at each time $t \geq 0$ addresses a nonlinear least-squares online problem

$$\min_x \sum_{i=1}^N \left(\omega_i^t - \frac{A}{\|x - c_i\|^\gamma} \right)^2.$$

We consider a network of $N = 50$ agents randomly located according to a two-dimensional Gaussian distribution with zero mean and variance $a^2 I_2 = 100 I_2$. The agents want to track the location of a moving target, which starts at a random location $\theta_{\text{target}}^0 \in \mathbb{R}^2$ generated according to the same distribution of the agents. The target moves along a circle of radius $r = 0.5$ according to the following law:

$$\theta_{\text{target}}^t = \theta_{\text{center}} + r \begin{bmatrix} \cos(t/200) \\ \sin(t/200) \end{bmatrix}$$

where $\theta_{\text{center}} \in \mathbb{R}^2$ represents the randomly generated circle center. We pick $\gamma = 1$, $A = 100$, and a noise variance $\sigma^2 = 0.001$. We take $\alpha = 0.05$ for GTAdam, $\alpha = 0.02$ for GT, $\alpha = 0.05$ for DGD, and $\alpha = 0.0725$ for DAdam. The agents communicate according to a ring graph. In Fig. 3, we compare the algorithm performance in terms of the (instantaneous) cost function evolution. Fig. 4 shows that the best performance in terms of average dynamic regret is obtained by GTAdam. GTAdam seems to achieve a smaller error with respect to the other algorithms. We make these comparisons by using θ_{target}^t as the optimal estimate associated with the iteration t but we note that the actual optimal solution may be slightly different since the noise ϵ_i^t affects the measurement of each agent.

C. Distributed Image Classification via Neural Networks

In this example, we consider an image classification problem in which N nodes have to cooperatively learn how to correctly classify images. We pick the Fashion-MNIST dataset [40]

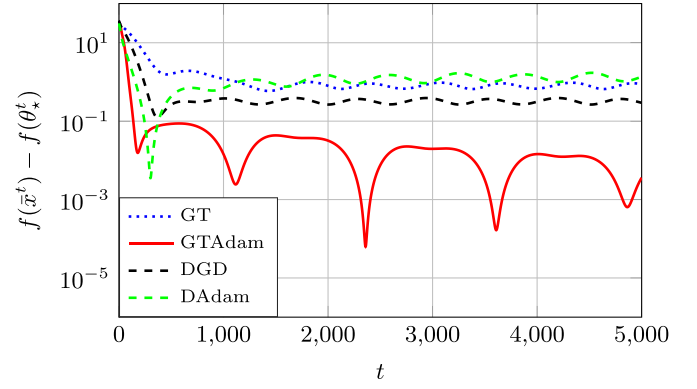


Fig. 3. Distributed source localization. Cost function values obtained for a network of $N = 50$ agents.

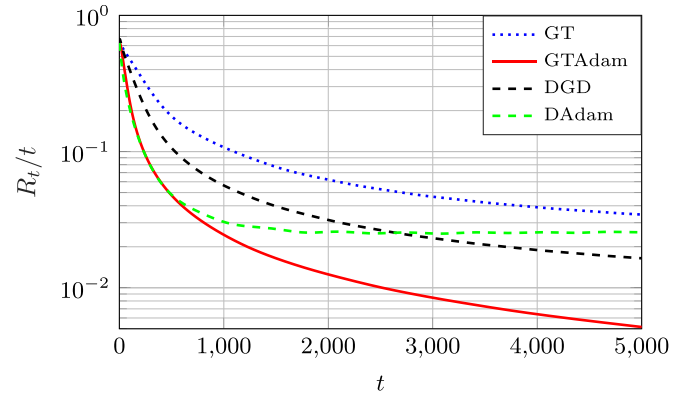


Fig. 4. Distributed source localization. Average regret values obtained for a network of $N = 50$ agents.

consisting of black-and-white 28×28 -pixels images of clothes belonging to 10 different classes. Each agent i has a local dataset $\mathcal{D}_i = \{(p_{i,k}, y_{i,k})\}_{k=1}^{m_i}$ consisting of m_i images $p_{i,k} \in \mathbb{R}^{28 \times 28}$ and their associated labels $y_{i,k} \in \{1, \dots, 10\}$. The goal of the agents is to learn the parameters x_* of a function $h(p; x_*)$ so that $h(p_{i,k}; x_*)$ gives the correct label for $p_{i,k}$. The resulting optimization problem is

$$\min_x \sum_{i=1}^N \frac{1}{m_i} \sum_{k=1}^{m_i} V(y_{i,k}, h(p_{i,k}, x)) + C \|x\|^2$$

where $V(\cdot)$ is the categorical cross-entropy loss, and $C > 0$ is a regularization parameter. The local cost function is

$$f_i(x | \mathcal{D}_i) \triangleq \mathbb{E}_{\mathcal{D}_i} [\ell_i(x)] = \frac{1}{m_i} \sum_{k=1}^{m_i} V(y_{i,k}, h(p_{i,k}, x)) + \frac{C}{N} \|x\|^2.$$

We represent $h(\cdot)$ by a neural network with one hidden layer (with 300 units with ReLU activation function) and an output layer with 10 units. Moreover, we pick $N = 16$ agents and associate each of them $m_i = 3750$ labeled images for all i . We performed Monte Carlo simulations consisting of 100 trials and each trial lasts 10 epochs over the local datasets. The results are reported In Figs. 5 and 6, in terms of the global training loss $f(\{\bar{x}_{ep}, \mathcal{D}_1, \dots, \mathcal{D}_N\}) \triangleq \sum_{i=1}^N f_i(\bar{x}_{ep} |$

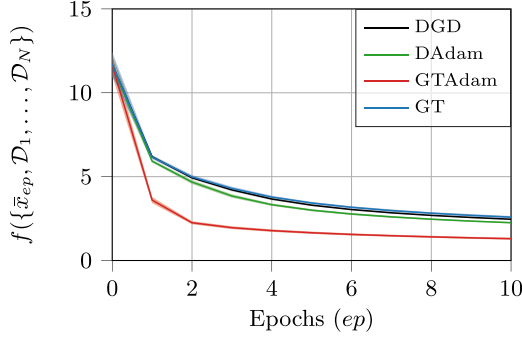


Fig. 5. Distributed image classification. Mean and 3-standard deviation band of the training loss.

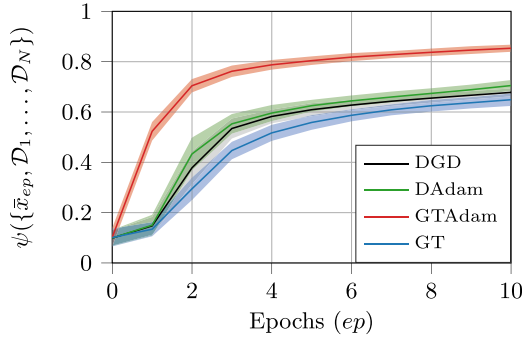


Fig. 6. Distributed image classification. Mean and 3-standard deviation band of the training accuracy.

\mathcal{D}_i), with $\bar{x}_{ep} \triangleq \frac{1}{N} \sum_{i=1}^N x_{i,ep}$, and the average training accuracy $\psi(\{\bar{x}_{ep}, \mathcal{D}_1, \dots, \mathcal{D}_N\}) \triangleq \frac{1}{N} \sum_{i=1}^N \psi_i(\bar{x}_{ep} | \mathcal{D}_i)$, where $\psi_i(\bar{x}_{ep} | \mathcal{D}_i)$ is the accuracy achieved with \bar{x}_{ep} on the local dataset of the agent i at the end of epoch ep . We take $\alpha = 0.001$ for GTAdam, and $\alpha = 0.1$ for DGD, GT, and DAdam. As it can be appreciated from Figs. 5 and 6, in both cases GTAdam outperforms the other algorithms.

VI. CONCLUSION

We proposed GTAdam, a novel distributed optimization algorithm tailored for multiagent online learning. Inspired by the popular Adam algorithm, our novel GTAdam is based on the gradient tracking distributed scheme, which is enhanced with adaptive first- and second-order momentum estimates of the gradient. We provided theoretical bounds on the convergence of the proposed algorithm. Moreover, we tested GTAdam in three different scenarios showing a performance improvement with respect to state-of-the-art algorithms.

APPENDIX

We report a lemma that will be used in the proof of Lemma 7 (cf. Appendix F).

Lemma 8: Let $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ be σ -strongly convex and with L -Lipschitz continuous gradient. Moreover, let $D \in \mathbb{R}^{n \times n}$ be a positive definite diagonal matrix such that $D_{ii} \in [\epsilon, M]$ for all $i = 1, \dots, n$ with $M \geq \epsilon > 0$ and $M < \infty$. Let $\bar{L} =$

ML and $\mathbf{s} = \epsilon\sigma$. Let $\mathbf{x}^{t+1} = \mathbf{x}^t - \alpha D \nabla f(\mathbf{x}^t)$, with $\alpha \in (0, \frac{2}{\bar{L}}]$. Then $\|\mathbf{x}^{t+1} - x_\star\| \leq \max\{(1 - \alpha\mathbf{s}), (1 - \alpha\bar{L})\} \|\mathbf{x}^t - x_\star\|$.

Proof: Let $h(x)$ be a function such that $\nabla h(x) = D \nabla f(x)$ for all x . It can be easily shown that h has \bar{L} -Lipschitz continuous gradients, in fact

$$\begin{aligned} \|\nabla h(x) - \nabla h(y)\| &= \|D \nabla f(x) - D \nabla f(y)\| \\ &\leq \|D\| \|\nabla f(x) - \nabla f(y)\| \leq \|D\| L \|x - y\| \leq ML \|x - y\|. \end{aligned}$$

Moreover h is \mathbf{s} -strongly convex, since $\nabla^2 h(x) = D \nabla^2 f(x) \succeq D \sigma I \geq \epsilon \sigma I$. Define $g(x) = h(x) - \frac{\mathbf{s}}{2} \|x\|^2$. Notice that, by definition, g is convex and with $(\bar{L} - \mathbf{s})$ -Lipschitz continuous gradient. Thus, by definition, we have

$$\langle \nabla g(x) - \nabla g(y), x - y \rangle \geq \frac{1}{\bar{L} - \mathbf{s}} \|\nabla g(x) - \nabla g(y)\|^2. \quad (21)$$

Now, by using the definition of g , one has

$$\begin{aligned} \langle \nabla h(x) - \mathbf{s}x - \nabla h(y) + \mathbf{s}y, x - y \rangle \\ = \langle \nabla h(x) - \nabla h(y), x - y \rangle - \mathbf{s} \|x - y\|^2. \end{aligned} \quad (22)$$

Moreover

$$\begin{aligned} \|\nabla g(x) - \nabla g(y)\|^2 &= \|\nabla h(x) - \mathbf{s}x - \nabla h(y) + \mathbf{s}y\|^2 \\ &= \|\nabla h(x) - \nabla h(y)\|^2 + \mathbf{s}^2 \|x - y\|^2 \\ &\quad - 2\mathbf{s} \langle \nabla h(x) - \nabla h(y), x - y \rangle. \end{aligned} \quad (23)$$

By combining (21), (22), and (23), we get

$$\begin{aligned} \langle \nabla h(x) - \nabla h(y), x - y \rangle \\ \geq \frac{\mathbf{s}\bar{L}}{\mathbf{s} + \bar{L}} \|x - y\|^2 + \frac{1}{\mathbf{s} + \bar{L}} \|\nabla h(x) - \nabla h(y)\|^2. \end{aligned} \quad (24)$$

Now, by using the update rule, one has

$$\begin{aligned} \|\mathbf{x}^{t+1} - x_\star\|^2 &= \|\mathbf{x}^t - \alpha D \nabla f(\mathbf{x}^t) - x_\star\|^2 \\ &= \|\mathbf{x}^t - x_\star\|^2 - 2\alpha \langle D \nabla f(\mathbf{x}^t), \mathbf{x}^t - x_\star \rangle + \alpha^2 \|D \nabla f(\mathbf{x}^t)\|^2 \\ &= \|\mathbf{x}^t - x_\star\|^2 - 2\alpha \langle D \nabla f(\mathbf{x}^t) - D \nabla f(x_\star), \mathbf{x}^t - x_\star \rangle \\ &\quad + \alpha^2 \|D \nabla f(\mathbf{x}^t) - D \nabla f(x_\star)\|^2. \end{aligned}$$

By using the result (24) with $\nabla h(x) = D \nabla f(x)$, we have

$$\begin{aligned} \|\mathbf{x}^{t+1} - x_\star\|^2 &\leq \|\mathbf{x}^t - x_\star\|^2 + \alpha^2 \|D \nabla f(\mathbf{x}^t) - D \nabla f(x_\star)\|^2 \\ &\quad - 2\alpha \frac{\mathbf{s}\bar{L}}{\mathbf{s} + \bar{L}} \|\mathbf{x}^t - x_\star\|^2 - \frac{2\alpha}{\mathbf{s} + \bar{L}} \|D \nabla f(\mathbf{x}^t) - D \nabla f(x_\star)\|^2 \\ &= \left(1 - 2\alpha \frac{\mathbf{s}\bar{L}}{\mathbf{s} + \bar{L}}\right) \|\mathbf{x}^t - x_\star\|^2 \\ &\quad + \alpha \left(\alpha - \frac{2}{\mathbf{s} + \bar{L}}\right) \|D \nabla f(\mathbf{x}^t) - D \nabla f(x_\star)\|^2 \\ &\leq \left(1 - 2\alpha \frac{\mathbf{s}\bar{L}}{\mathbf{s} + \bar{L}}\right) \|\mathbf{x}^t - x_\star\|^2 + \alpha \left(\alpha \bar{L}^2 - \frac{2\mathbf{s}^2}{\mathbf{s} + \bar{L}}\right) \|\mathbf{x}^t - x_\star\|^2 \\ &\leq \max\{(1 - \alpha\mathbf{s})^2, (1 - \alpha\bar{L})^2\} \|\mathbf{x}^t - x_\star\|^2. \end{aligned}$$

The proof follows by taking the square root of both sides. \blacksquare

A. Proof of Lemma 1

By using the update (4a), we can write

$$\|\bar{m}^{t+1}\| = \|\beta_1 \bar{m}^t + (1 - \beta_1) \bar{s}^t\| \leq \beta_1 \|\bar{m}^t\| + (1 - \beta_1) \|\bar{s}^t\| \quad (25)$$

in which we use the triangle inequality. Regarding the term $\|\bar{s}^t\|$, we use the relation $\bar{s}^t = \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t)$, and we add $\frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_\star^t) = 0$, thus obtaining

$$\begin{aligned} \|\bar{s}^t\| &= \left\| \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) - \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_\star^t) \right\| \stackrel{(a)}{\leq} \frac{L}{N} \sum_{i=1}^N \|x_i^t - x_\star^t\| \\ &\stackrel{(b)}{\leq} \frac{L}{\sqrt{N}} \|\mathbf{x}^t - \mathbf{1}x_\star^t\| \stackrel{(c)}{\leq} \frac{L}{\sqrt{N}} \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + L \|\bar{x}^t - x_\star^t\| \end{aligned} \quad (26)$$

where, in (a), we exploit the Lipschitz continuity of the gradients of the cost functions (cf. Assumptions 1), in (b), we use the basic algebraic property $\sum_{i=1}^N \|\theta_i\| \leq \sqrt{N}\|\theta\|$ for a generic vector $\theta \triangleq \text{col}(\theta_1, \dots, \theta_N)$, and in (c), we add and subtract the term $\mathbf{1}\bar{x}^t$ and apply the triangle inequality. The proof follows by combining the bounds (25) and (26).

B. Proof of Lemma 3

By using (3c) and (4c), one has

$$\begin{aligned} \|\mathbf{d}^{t+1} - \mathbf{1}\bar{d}^{t+1}\| &= \left\| \left(I - \frac{1}{N} \mathbf{1}\mathbf{1}^\top \right) (\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{m}^{t+1} \right\| \\ &\stackrel{(a)}{\leq} \left\| (\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \right\| \|\mathbf{m}^{t+1}\| \stackrel{(b)}{\leq} \frac{1}{\sqrt{\epsilon}} \|\mathbf{m}^{t+1}\| \\ &\stackrel{(c)}{\leq} \frac{1}{\sqrt{\epsilon}} \|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\| + \frac{\sqrt{N}}{\sqrt{\epsilon}} \|\bar{m}^{t+1}\| \end{aligned} \quad (27)$$

where, in (a), we apply the Cauchy–Schwarz inequality combined with $\|I - \frac{1}{N} \mathbf{1}\mathbf{1}^\top\| \leq 1$, in (b), we use the bound $\|(\mathbf{V}^{t+1} + \epsilon I)^{-1/2}\| \leq \frac{1}{\sqrt{\epsilon}}$ (justified by the fact that $\mathbf{v}^{t+1} \geq 0$ for all $t \geq 0$), in (c), we add and subtract within the norm $\mathbf{1}\bar{m}^{t+1}$ and apply the triangle inequality and an algebraic property.

The proof follows by using Lemmas 1 and 2 in (27).

C. Proof of Lemma 4

By combining (3e) and (4e), one has

$$\begin{aligned} \|\mathbf{s}^{t+1} - \mathbf{1}\bar{s}^{t+1}\| &= \left\| W\mathbf{s}^t + \mathbf{g}^{t+1} - \mathbf{g}^t - \mathbf{1} \left(\bar{s}^t + \frac{1}{N} \sum_{i=1}^N (g_i^{t+1} - g_i^t) \right) \right\| \\ &\stackrel{(a)}{\leq} \left\| \left(W - \frac{1}{N} \mathbf{1}\mathbf{1}^\top \right) (\mathbf{s}^t - \mathbf{1}\bar{s}^t) \right\| \\ &\quad + \left\| \left(I - \frac{1}{N} \mathbf{1}\mathbf{1}^\top \right) (\mathbf{g}^{t+1} - \mathbf{g}^t) \right\| \\ &\stackrel{(b)}{=} \sigma_W \|\mathbf{s}^t - \mathbf{1}\bar{s}^t\| + \|\mathbf{g}^{t+1} - \mathbf{g}^t\| \end{aligned} \quad (28)$$

where (a) uses $\mathbf{1} \in \ker(W - \frac{1}{N} \mathbf{1}\mathbf{1}^\top)$ and the triangle inequality and (b) combines the Cauchy–Schwarz inequality with the bounds $\|W - \frac{1}{N} \mathbf{1}\mathbf{1}^\top\| \leq \sigma_W$ and $\|I - \frac{1}{N} \mathbf{1}\mathbf{1}^\top\| \leq 1$.

Let $\tilde{\mathbf{g}}^t \triangleq \text{col}(\nabla f_1^{t+1}(x_1^t), \dots, \nabla f_N^{t+1}(x_N^t))$ and manipulate the term $\|\mathbf{g}^{t+1} - \mathbf{g}^t\|$ in (28) as

$$\begin{aligned} \|\mathbf{g}^{t+1} - \mathbf{g}^t\| &\leq \|\mathbf{g}^{t+1} - \tilde{\mathbf{g}}^t\| + \|\tilde{\mathbf{g}}^t - \mathbf{g}^t\| \\ &\stackrel{(a)}{\leq} L \|\mathbf{x}^{t+1} - \mathbf{x}^t\| + \|\tilde{\mathbf{g}}^t - \mathbf{g}^t\| \stackrel{(b)}{\leq} L \|\mathbf{x}^{t+1} - \mathbf{x}^t\| + \sqrt{N}\eta^t \end{aligned}$$

$$\stackrel{(c)}{=} L \|W\mathbf{x}^t - \alpha\mathbf{d}^{t+1} - \mathbf{x}^t\| + \sqrt{N}\eta^t \quad (29)$$

where, in (a), we use the Lipschitz continuity of the gradients of the cost functions (cf. Assumption 1), (b) uses the variable η^t [cf (6)], and (c) uses the update (3d) of \mathbf{x}^{t+1} . Let us manipulate the first term on the right-hand side of (29)

$$\begin{aligned} \|W\mathbf{x}^t - \alpha\mathbf{d}^{t+1} - \mathbf{x}^t\| &\stackrel{(a)}{=} \|(W - I)(\mathbf{x}^t - \mathbf{1}\bar{x}^t) - \alpha\mathbf{d}^{t+1}\| \\ &\stackrel{(b)}{\leq} \|W - I\| \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + \alpha \|\mathbf{d}^{t+1} - \mathbf{1}\bar{d}^{t+1}\| + \alpha \|\mathbf{1}\bar{d}^{t+1}\| \end{aligned} \quad (30)$$

where (a) uses the fact that $\ker(W - I) = \text{span}(\mathbf{1})$ and, in (b), we add and subtract the term $\mathbf{1}\bar{d}^{t+1}$ within the norm and we apply the triangle inequality and the Cauchy–Schwarz inequality. Regarding $\|\mathbf{1}\bar{d}^{t+1}\|$, we use (3c) and (4c) to write

$$\begin{aligned} \|\mathbf{1}\bar{d}^{t+1}\| &= \left\| \frac{1}{N} \mathbf{1}\mathbf{1}^\top \mathbf{d}^{t+1} \right\| = \left\| \frac{1}{N} \mathbf{1}\mathbf{1}^\top (\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{m}^{t+1} \right\| \\ &\stackrel{(a)}{\leq} \frac{1}{\sqrt{\epsilon}} \|\mathbf{m}^{t+1}\| \stackrel{(b)}{\leq} \frac{1}{\sqrt{\epsilon}} \|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\| + \frac{\sqrt{N}}{\sqrt{\epsilon}} \|\bar{m}^{t+1}\| \end{aligned} \quad (31)$$

where, in (a), we apply the Cauchy–Schwarz inequality and the bounds $\|\frac{1}{N} \mathbf{1}\mathbf{1}^\top\| \leq 1$ and $\|(\mathbf{V}^{t+1} + \epsilon)^{-1/2}\| \leq \frac{1}{\sqrt{\epsilon}}$, in (b), we add and subtract within the norm the term $\mathbf{1}\bar{m}^{t+1}$, apply the triangle inequality, and use an algebraic property. By combining (30) and (31), we bound (29) as

$$\begin{aligned} \|\mathbf{g}^{t+1} - \mathbf{g}^t\| &\leq L \|W - I\| \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + \alpha L \|\mathbf{d}^{t+1} - \mathbf{1}\bar{d}^{t+1}\| \\ &\quad + \alpha \frac{L}{\sqrt{\epsilon}} \|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\| + \alpha \frac{L\sqrt{N}}{\sqrt{\epsilon}} \|\bar{m}^{t+1}\| + \sqrt{N}\eta^t. \end{aligned} \quad (32)$$

Now, by using the bound (32) within (30), we get

$$\begin{aligned} \|\mathbf{s}^{t+1} - \mathbf{1}\bar{s}^{t+1}\| &\leq \sigma_W \|\mathbf{s}^t - \mathbf{1}\bar{s}^t\| \\ &\quad + L \|W - I\| \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + \alpha L \|\mathbf{d}^{t+1} - \mathbf{1}\bar{d}^{t+1}\| \\ &\quad + \alpha \frac{L}{\sqrt{\epsilon}} \|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\| + \alpha \frac{L\sqrt{N}}{\sqrt{\epsilon}} \|\bar{m}^{t+1}\| + \sqrt{N}\eta^t. \end{aligned} \quad (33)$$

The proof follows by using Lemmas 1, 2, and 3 to bound $\|\bar{m}^{t+1}\|$, $\|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\|$, and $\|\mathbf{d}^{t+1} - \mathbf{1}\bar{d}^{t+1}\|$.

D. Proof of Lemma 5

By combining (3d) and (4d), we have

$$\begin{aligned} \|\mathbf{x}^{t+1} - \mathbf{1}\bar{x}^{t+1}\| &= \|W\mathbf{x}^t - \alpha\mathbf{d}^{t+1} - \mathbf{1}\bar{x}^t + \alpha\mathbf{1}\bar{d}^{t+1}\| \\ &\stackrel{(a)}{\leq} \|W\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + \alpha \|\mathbf{d}^{t+1} - \mathbf{1}\bar{d}^{t+1}\| \\ &\stackrel{(b)}{\leq} \sigma_W \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + \alpha \|\mathbf{d}^{t+1} - \mathbf{1}\bar{d}^{t+1}\| \end{aligned}$$

where, in (a), we apply the triangle inequality and (b) follows by $\|W - \frac{1}{N} \mathbf{1}\mathbf{1}^\top\| \leq \sigma_W$. The proof follows by Lemma 3.

E. Proof of Lemma 6

From the updates of \bar{s}^{t+1} and \bar{m}^{t+1} [cf. (4e) and (4a)], we get

$$\begin{aligned} \|\bar{s}^{t+1} - \bar{m}^{t+1}\| &= \left\| \bar{s}^t + \frac{1}{N} \sum_{i=1}^N \nabla f_i^{t+1}(x_i^{t+1}) - \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) \right. \\ &\quad \left. - \beta_1 \bar{m}^t - (1 - \beta_1) \bar{s}^t \right\| \end{aligned}$$

$$\stackrel{(a)}{\leq} \beta_1 \|\bar{s}^t - \bar{m}^t\| + \left\| \frac{1}{N} \sum_{i=1}^N \nabla f_i^{t+1}(x_i^{t+1}) - \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) \right\|$$

where (a) uses the triangle inequality. By adding and subtracting within the second norm $\frac{1}{N} \sum_{i=1}^N \nabla f_i^{t+1}(\bar{x}^{t+1})$ and $\frac{1}{N} \sum_{i=1}^N \nabla f_i^{t+1}(x_i^t)$, we use the triangle inequality to obtain

$$\begin{aligned} \|\bar{s}^{t+1} - \bar{m}^{t+1}\| &\leq \beta_1 \|\bar{s}^t - \bar{m}^t\| \\ &+ \left\| \frac{1}{N} \sum_{i=1}^N \nabla f_i^{t+1}(x_i^{t+1}) - \frac{1}{N} \sum_{i=1}^N \nabla f_i^{t+1}(\bar{x}^{t+1}) \right\| \\ &+ \left\| \frac{1}{N} \sum_{i=1}^N \nabla f_i^{t+1}(x_i^t) - \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) \right\| \\ &+ \left\| \frac{1}{N} \sum_{i=1}^N \nabla f_i^{t+1}(\bar{x}^{t+1}) - \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) \right\| \\ &\stackrel{(a)}{\leq} \beta_1 \|\bar{s}^t - \bar{m}^t\| + \frac{L}{\sqrt{N}} \|\mathbf{x}^{t+1} - \mathbf{1}\bar{x}^{t+1}\| + \frac{1}{\sqrt{N}} \eta^t \\ &+ \frac{L}{\sqrt{N}} \|\mathbf{x}^t - \mathbf{1}\bar{x}^{t+1}\| \end{aligned} \quad (34)$$

where, in (a), we use the Lipschitz continuity of the gradients of the cost functions (cf. Assumptions 1) for the second and the third norms, and we use η^t [cf. (6)]. Now, we replace \bar{x}^{t+1} with its update (4d) within the last term of (34) obtaining

$$\begin{aligned} \|\bar{s}^{t+1} - \bar{m}^{t+1}\| &\leq \beta_1 \|\bar{s}^t - \bar{m}^t\| + \frac{L}{\sqrt{N}} \|\mathbf{x}^{t+1} - \mathbf{1}\bar{x}^{t+1}\| \\ &+ \frac{L}{\sqrt{N}} \eta^t + \frac{L}{\sqrt{N}} \|\mathbf{1}\bar{x}^t - \alpha \mathbf{1}\bar{d}^{t+1} - \mathbf{x}^t\| \\ &\stackrel{(a)}{\leq} \beta_1 \|\bar{s}^t - \bar{m}^t\| + \frac{L}{\sqrt{N}} \|\mathbf{x}^{t+1} - \mathbf{1}\bar{x}^{t+1}\| + \frac{L}{\sqrt{N}} \eta^t \\ &+ \frac{L}{\sqrt{N}} \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + \alpha \frac{L}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\| \\ &+ \alpha \frac{L}{\sqrt{\epsilon}} \|\bar{m}^{t+1}\| \end{aligned} \quad (35)$$

where, in (a), we use (31) to bound $\|\mathbf{1}\bar{d}^{t+1}\|$. The proof follows by using Lemmas 5, 1, and 2 to bound $\|\mathbf{x}^{t+1} - \mathbf{1}\bar{x}^{t+1}\|$, $\|\bar{m}^{t+1}\|$, and $\|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\|$, respectively.

F. Proof of Lemma 7

By using (4d), one has

$$\begin{aligned} \|\bar{x}^{t+1} - x_\star^{t+1}\| &= \|\bar{x}^t - \alpha \bar{d}^{t+1} - x_\star^{t+1}\| \\ &\stackrel{(a)}{\leq} \|\bar{x}^t - \alpha \bar{d}^{t+1} - x_\star^t\| + \zeta^t \end{aligned}$$

where, in (a), we add and subtract within the norm x_\star^t , use the triangle inequality, and use ζ^t [cf. (6)]. Now, we add and subtract within the norm $\alpha \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N^2} \nabla f^t(\bar{x}^t)$ and we use the triangle inequality to write

$$\begin{aligned} \|\bar{x}^{t+1} - x_\star^{t+1}\| &\leq \left\| \bar{x}^t - \alpha \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N^2} \nabla f^t(\bar{x}^t) - x_\star^t \right\| \\ &+ \alpha \left\| \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N^2} \nabla f^t(\bar{x}^t) - \bar{d}^{t+1} \right\| + \zeta^t. \end{aligned} \quad (36)$$

Consider the second term of (36) and use (4c) to write

$$\begin{aligned} &\alpha \left\| \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N} \frac{\nabla f^t(\bar{x}^t)}{N} - \bar{d}^{t+1} \right\| \\ &= \alpha \left\| \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N} \frac{\nabla f^t(\bar{x}^t)}{N} - \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2}}{N} \mathbf{m}^{t+1} \right\| \\ &\stackrel{(a)}{\leq} \alpha \left\| \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N} \left(\frac{\nabla f^t(\bar{x}^t)}{N} - \bar{m}^{t+1} \right) \right\| \\ &+ \alpha \left\| \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2}}{N} (\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}) \right\| \\ &\stackrel{(b)}{\leq} \frac{\alpha}{\sqrt{\epsilon}} \left\| \frac{\nabla f^t(\bar{x}^t)}{N} - \bar{m}^{t+1} \right\| + \frac{\alpha}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\| \end{aligned} \quad (37)$$

where, in (a), we add and subtract within the norm the term $\frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N} \bar{m}^{t+1}$ and we apply the triangle inequality, in (b), we apply the Cauchy–Schwarz inequality combined with the bounds $\left\| \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N} \right\| \leq \frac{1}{\sqrt{\epsilon}}$ and $\left\| \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2}}{N} \right\| \leq \frac{1}{\sqrt{\epsilon}\sqrt{N}}$.

Now, we add and subtract the term $\frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t)$ and then we use the triangle inequality to rewrite the first term of the second member of (37) as

$$\begin{aligned} \alpha \frac{1}{\sqrt{\epsilon}} \left\| \frac{\nabla f^t(\bar{x}^t)}{N} - \bar{m}^{t+1} \right\| &\leq \alpha \frac{1}{\sqrt{\epsilon}} \left\| \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) - \bar{m}^{t+1} \right\| \\ &+ \alpha \frac{1}{\sqrt{\epsilon}} \left\| \frac{\nabla f^t(\bar{x}^t)}{N} - \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) \right\| \\ &\stackrel{(a)}{=} \alpha \frac{1}{\sqrt{\epsilon}} \left\| \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) - \beta_1 \bar{m}^t - (1 - \beta_1) \bar{s}^t \right\| \\ &+ \alpha \frac{1}{\sqrt{\epsilon}} \left\| \frac{\nabla f^t(\bar{x}^t)}{N} - \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t) \right\| \\ &\stackrel{(b)}{\leq} \alpha \frac{\beta_1}{\sqrt{\epsilon}} \|\bar{s}^t - \bar{m}^t\| + \alpha \frac{L}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| \end{aligned} \quad (38)$$

where, in (a), we use (4a), (b) uses the relation $\bar{s}^t = \frac{1}{N} \sum_{i=1}^N \nabla f_i^t(x_i^t)$, and the Lipschitz continuity of the gradients of the cost functions (cf. Assumption 1). Next, in order to bound the right-hand side of (36), first notice that $\frac{1}{\sqrt{G+\epsilon}} < \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N} < \frac{1}{\sqrt{\epsilon}}$. Moreover, being f^t μ -strongly convex for all $t \geq 0$ (cf. Assumption 2) and having L -Lipschitz continuous gradients (cf. Assumption 1), we apply Lemma 8 (in the Appendix) to write

$$\left\| \bar{x}^t - \alpha \frac{\mathbf{1}^\top(\mathbf{V}^{t+1} + \epsilon I)^{-1/2} \mathbf{1}}{N^2} \nabla f^t(\bar{x}^t) - x_\star^t \right\| \leq \phi \|\bar{x}^t - x_\star^t\| \quad (39)$$

where $\phi \triangleq \max\{1 - \frac{\alpha}{\sqrt{\epsilon+G}}\mu, |1 - \frac{\alpha}{\sqrt{\epsilon}}L|\}$. If we take $\alpha < \min\{\frac{\sqrt{\epsilon+G}}{\mu}, \frac{\sqrt{\epsilon}}{L}\}$, then it holds $\phi = 1 - \alpha\delta$, where δ is defined in the statement of Theorem 1. By combining the latter with (38) and (39), it is possible to upper bound (36) as

$$\begin{aligned} \|\bar{x}^{t+1} - x_\star^{t+1}\| &\leq (1 - \alpha\delta) \|\bar{x}^t - x_\star^t\| + \alpha \frac{\beta_1}{\sqrt{\epsilon}} \|\bar{s}^t - \bar{m}^t\| \\ &+ \frac{\alpha}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{m}^{t+1} - \mathbf{1}\bar{m}^{t+1}\| + \frac{\alpha L}{\sqrt{\epsilon}\sqrt{N}} \|\mathbf{x}^t - \mathbf{1}\bar{x}^t\| + \zeta^t. \end{aligned} \quad (40)$$

The proof follows by invoking Lemma 2 to bound $\|\mathbf{m}^{t+1} - \bar{\mathbf{m}}^{t+1}\|$ within (40).

G. Proof of Corollary 1

We add and subtract $f^t(\bar{x}^t)$ to $f^t(x_i^t) - f^t(x_\star^t)$, obtaining

$$\begin{aligned} f^t(x_i^t) - f^t(x_\star^t) &= f^t(x_i^t) - f^t(\bar{x}^t) + f^t(\bar{x}^t) - f^t(x_\star^t) \\ &\stackrel{(a)}{\leq} f^t(x_i^t) - f^t(\bar{x}^t) + \frac{L}{2} \|\bar{x}^t - x_\star^t\|^2 \\ &\stackrel{(b)}{\leq} \nabla f^t(\bar{x}^t)^\top (x_i^t - \bar{x}^t) + \frac{L}{2} \|x_i^t - \bar{x}^t\|^2 + \frac{L}{2} \|\bar{x}^t - x_\star^t\|^2 \end{aligned} \quad (41)$$

where, in (a), we apply (15) and in (b) we use the Lipschitz continuity of the gradients of the cost functions (cf. Assumption 1). Being $\nabla f^t(x_\star^t) = 0$, we rewrite (41) as

$$\begin{aligned} f^t(x_i^t) - f^t(x_\star^t) &\leq (\nabla f^t(\bar{x}^t) - \nabla f^t(x_\star^t))^\top (x_i^t - \bar{x}^t) \\ &\quad + \frac{L}{2} \|x_i^t - \bar{x}^t\|^2 + \frac{L}{2} \|\bar{x}^t - x_\star^t\|^2 \\ &\stackrel{(a)}{\leq} L \|\bar{x}^t - x_\star^t\| \|x_i^t - \bar{x}^t\| + \frac{L}{2} \|x_i^t - \bar{x}^t\|^2 + \frac{L}{2} \|\bar{x}^t - x_\star^t\|^2 \end{aligned} \quad (42)$$

where, in (a), we use the Cauchy–Schwarz inequality and the Lipschitz continuity of the gradients of the cost functions (cf. Assumption 1). Now, we notice that both $\|\bar{x}^t - x_\star^t\|$ and $\|x_i^t - \bar{x}^t\|$ represent a component of the vector y^t defined in (5), and thus, can be both upper bounded by $\|y^t\|$. Hence, the inequality (42) can be elaborated as

$$f^t(x_i^t) - f^t(x_\star^t) \leq 2L \|y^t\|^2. \quad (43)$$

By summing over t the inequality in (43), we bound $R_{T,i}$ as

$$R_{T,i} \leq 2L \sum_{t=1}^T \|y^t\|^2 \stackrel{(a)}{\leq} 2L\lambda_1^2 \sum_{t=1}^T \|y^t\|_\gamma^2 \quad (44)$$

where, in (a), we apply (16a). As done above to prove (7), the proof follows by combining (44), (14), and (16b).

H. Proof of Corollary 2

Using the same arguments of Theorem 1, we start from (14). Differently from the dynamic case, in the static set-up, we have $\nabla f_i^t(x) = \nabla f_i(x)$ for all t and i , leading to $x_\star^t = x_\star$ for all t . Thus, we can combine (14) with $q^t \equiv 0$, the Lipschitz continuity of the gradient of the cost function (cf. Assumption 1) and (16a), to write $f(\bar{x}^t) - f(x_\star) \leq \tilde{\rho}^{2t} \frac{L\lambda_1^2}{2} \|y^0\|_\gamma^2 \leq \tilde{\rho}^{2t} \frac{L\lambda_1^2\lambda_2^2}{2} \|y^0\|^2$, in which we use (16b). The proof follows by using the definition (18) of λ .

REFERENCES

- [1] X. Li, L. Xie, and N. Li, “A survey of decentralized online learning,” 2022, *arXiv:2205.00473*.
- [2] A. Simonetto, A. Mokhtari, A. Koppel, G. Leus, and A. Ribeiro, “A class of prediction-correction methods for time-varying convex optimization,” *IEEE Trans. Signal Process.*, vol. 64, no. 17, pp. 4576–4591, Sep. 2016.
- [3] M. Fazlyab, S. Paternain, V. M. Preciado, and A. Ribeiro, “Prediction-correction interior-point method for time-varying convex optimization,” *IEEE Trans. Autom. Control*, vol. 63, no. 7, pp. 1973–1986, Jul. 2018.
- [4] R. L. Cavalcante and S. Stanczak, “A distributed subgradient method for dynamic convex optimization problems under noisy information exchange,” *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 2, pp. 243–256, Apr. 2013.
- [5] Z. J. Towfic and A. H. Sayed, “Adaptive penalty-based distributed stochastic convex optimization,” *IEEE Trans. Signal Process.*, vol. 62, no. 15, pp. 3924–3938, Aug. 2014.
- [6] D. Mateos-Núñez and J. Cortés, “Distributed online convex optimization over jointly connected digraphs,” *IEEE Trans. Netw. Sci. Eng.*, vol. 1, no. 1, pp. 23–37, Jan. 2014.
- [7] M. Akbari, B. Ghahesifard, and T. Linder, “Distributed online convex optimization on time-varying directed graphs,” *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 3, pp. 417–428, Sep. 2015.
- [8] S. Hosseini, A. Chapman, and M. Mesbahi, “Online distributed convex optimization on dynamic networks,” *IEEE Trans. Autom. Control*, vol. 61, no. 11, pp. 3545–3550, Nov. 2016.
- [9] S. Shahrampour and A. Jadbabaie, “Distributed online optimization in dynamic environments using mirror descent,” *IEEE Trans. Autom. Control*, vol. 63, no. 3, pp. 714–725, Mar. 2017.
- [10] M. Akbari, B. Ghahesifard, and T. Linder, “Individual regret bounds for the distributed online alternating direction method of multipliers,” *IEEE Trans. Autom. Control*, vol. 64, no. 4, pp. 1746–1752, Apr. 2019.
- [11] X. Yi, X. Li, L. Xie, and K. H. Johansson, “Distributed online convex optimization with time-varying coupled inequality constraints,” *IEEE Trans. Signal Process.*, vol. 68, pp. 731–746, 2020.
- [12] S. S. Ram, A. Nedić, and V. V. Veeravalli, “Distributed stochastic subgradient projection algorithms for convex optimization,” *J. Optim. Theory Appl.*, vol. 147, no. 3, pp. 516–545, 2010.
- [13] F. Farina and G. Notarstefano, “Randomized block proximal methods for distributed stochastic big-data optimization,” *IEEE Trans. Autom. Control*, vol. 66, no. 9, pp. 4000–4014, Sep. 2021.
- [14] S. Pu and A. Nedić, “Distributed stochastic gradient tracking methods,” *Math. Program.*, vol. 187, pp. 409–457, 2020.
- [15] W. Shi, Q. Ling, G. Wu, and W. Yin, “Extra: An exact first-order algorithm for decentralized consensus optimization,” *SIAM J. Optim.*, vol. 25, no. 2, pp. 944–966, 2015.
- [16] D. Varagnolo, F. Zanella, A. Cenedese, G. Pillonetto, and L. Schenato, “Newton-Raphson consensus for distributed convex optimization,” *IEEE Trans. Autom. Control*, vol. 61, no. 4, pp. 994–1009, Apr. 2016.
- [17] P. Di Lorenzo and G. Scutari, “NEXT: In-network nonconvex optimization,” *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 2, pp. 120–136, Jun. 2016.
- [18] A. Nedić, A. Olshevsky, and W. Shi, “Achieving geometric convergence for distributed optimization over time-varying graphs,” *SIAM J. Optim.*, vol. 27, no. 4, pp. 2597–2633, 2017.
- [19] G. Qu and N. Li, “Harnessing smoothness to accelerate distributed optimization,” *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1245–1260, Sep. 2018.
- [20] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, “Convergence of asynchronous distributed gradient methods over stochastic networks,” *IEEE Trans. Autom. Control*, vol. 63, no. 2, pp. 434–448, Feb. 2018.
- [21] R. Xin and U. A. Khan, “A linear algorithm for optimization over directed graphs with geometric convergence,” *IEEE Control Syst. Lett.*, vol. 2, no. 3, pp. 315–320, Jul. 2018.
- [22] G. Scutari and Y. Sun, “Distributed nonconvex constrained optimization over time-varying digraphs,” *Math. Program.*, vol. 176, no. 1/2, pp. 497–544, 2019.
- [23] M. Zhu and S. Martínez, “Discrete-time dynamic average consensus,” *Automatica*, vol. 46, no. 2, pp. 322–329, 2010.
- [24] S. S. Kia, B. Van Scoy, J. Cortes, R. A. Freeman, K. M. Lynch, and S. Martínez, “Tutorial on dynamic average consensus: The problem, its applications, and the algorithms,” *IEEE Control Syst. Mag.*, vol. 39, no. 3, pp. 40–72, Jun. 2019.
- [25] Y. Zhang, R. J. Ravier, M. M. Zavlanos, and V. Tarokh, “A distributed online convex optimization algorithm with improved dynamic regret,” in *Proc. IEEE Conf. Decis. Control*, 2019, pp. 2449–2454.
- [26] I. Notarnicola, A. Simonetto, F. Farina, and G. Notarstefano, “Distributed personalized gradient tracking with convex parametric models,” *IEEE Trans. Autom. Control*, to be published, doi: [10.1109/TAC.2022.3147007](https://doi.org/10.1109/TAC.2022.3147007).
- [27] A. Simonetto, E. Dall’Anese, J. Monteil, and A. Bernstein, “Personalized optimization with user’s feedback,” *Automatica*, vol. 131, 2021, Art. no. 109767.
- [28] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, *arXiv:1412.6980*.
- [29] S. Bock, J. Goppold, and M. Weiß, “An improvement of the convergence proof of the adam-optimizer,” 2018, *arXiv:1804.10587*.
- [30] X. Chen, S. Liu, R. Sun, and M. Hong, “On the convergence of a class of adam-type algorithms for non-convex optimization,” in *Proc. Int. Conf. Learn. Representations*, 2018.
- [31] S. J. Reddi, S. Kale, and S. Kumar, “On the convergence of adam and beyond,” in *Proc. Int. Conf. Learn. Representations*, 2018.

- [32] Z. Zhou, Q. Zhang, G. Lu, H. Wang, W. Zhang, and Y. Yu, "AdaShift: Decorrelation and convergence of adaptive learning rate methods," in *Proc. Int. Conf. Learn. Representations*, 2018.
- [33] P. Nazari, D. A. Tarzanagh, and G. Michailidis, "DADAM: A consensus-based distributed adaptive gradient method for online optimization," 2019, *arXiv:1901.09109*.
- [34] A. Mokhtari, S. Shahrapour, A. Jadbabaie, and A. Ribeiro, "Online optimization in dynamic environments: Improved regret rates for strongly convex problems," in *Proc. IEEE Conf. Decis. Control*, 2016, pp. 7195–7201.
- [35] E. Dall'Anese, A. Simonetto, S. Becker, and L. Madden, "Optimization and learning with information streams: Time-varying algorithms and applications," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 71–83, May 2020.
- [36] Y. Li, G. Qu, and N. Li, "Online optimization with predictions and switching costs: Fast algorithms and the fundamental limit," *IEEE Trans. Autom. Control*, vol. 66, no. 10, pp. 4761–4768, Oct. 2021.
- [37] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
- [38] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
- [39] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proc. Int. Symp. Inf. Process. Sensor Netw.*, 2004, pp. 20–27.
- [40] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.



Guido Carnevale (Graduate Student Member, IEEE) received the M.Sc. degree "summa cum laude" in automation engineering in 2019 from the University of Bologna, Bologna, Italy, where he is currently working toward the Ph.D. degree within the Ph.D. Programme "Biomedical, Electrical, and Systems Engineering" with the Department of Electrical, Electronic and Information Engineering.

He was a Visiting Scholar with the University of Oxford in 2022. His research interests include distributed optimization and games over networks.



Francesco Farina received the M.Sc. degree in management engineering and the Ph.D. degree in information engineering and science from the University of Siena, Siena, Italy, in 2015 and 2019, respectively.

From 2018 to 2020, he was a Research Fellow with the Department of Electrical, Electronic and Information Engineering G. Marconi, Alma Mater Studiorum—Università di Bologna, and a Research Associate with the University of Siena. His current research interests include optimization, machine learning, and artificial intelligence.



Ivano Notarnicola (Member, IEEE) received the M.Sc. degree in computer engineering and the Ph.D. degree in engineering of complex systems from the Università del Salento, Lecce, Italy, in 2014 and 2018, respectively.

He is currently a Junior Assistant Professor with the Department of Electrical, Electronic, and Information Engineering G. Marconi, Alma Mater Studiorum Università di Bologna, Bologna, Italy. He was a Visiting Student with the University of Stuttgart, Stuttgart, Germany, in 2014 and a Visiting Scholar with Purdue University in 2017. His research interests include distributed methods for optimization, control, and machine learning.



Giuseppe Notarstefano (Member, IEEE) received the Laurea degree *summa cum laude* in electronics engineering from the Università di Pisa, Pisa, Italy, in 2003, and the Ph.D. degree in automation and operation research from the Università di Padova, Padua, Italy, in 2007.

He is currently a Professor with the Department of Electrical, Electronic, and Information Engineering G. Marconi, Alma Mater Studiorum Università di Bologna, Bologna, Italy. He was an Associate Professor (from 2016 to 2018) and has been an Assistant Professor, Ricercatore (since 2007), with the Università del Salento, Lecce, Italy. He is also a Visiting Scholar with the University of Stuttgart, Stuttgart, Germany, University of California, Santa Barbara, Santa Barbara, CA, USA, and the University of Colorado Boulder, Boulder, CO, USA. His research interests include distributed optimization, cooperative control in complex networks, applied nonlinear optimal control, and trajectory optimization and maneuvering of aerial and car vehicles.

Dr. Notarstefano is an Associate Editor for IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, and IEEE CONTROL SYSTEMS LETTERS. He is also part of the Conference Editorial Board of the IEEE Control Systems Society and EUCA. He is a recipient of an ERC Starting Grant.