

Extremum Seeking Tracking for Derivative-free Distributed Optimization

Nicola Mimmo *Member, IEEE*, Guido Carnevale, *Member, IEEE*, Andrea Testa, *Member, IEEE* and Giuseppe Notarstefano, *Member, IEEE*

Abstract— In this paper, we deal with a network of agents that want to cooperatively minimize the sum of local cost functions depending on a common decision variable. We consider the challenging scenario in which objective functions are unknown and agents have only access to local measurements of their local functions. We propose a novel distributed algorithm that combines a recent gradient tracking policy with an extremum seeking technique to estimate the global descent direction. The joint use of these two techniques results in a distributed optimization scheme that provides arbitrarily accurate solution estimates through the combination of Lyapunov and averaging analysis approaches with consensus theory. We perform numerical simulations in a personalized optimization framework to corroborate the theoretical results.

I. INTRODUCTION

In recent years, distributed optimization over networks has become a key research topic, see, e.g., [1]–[3] for an overview, also in the setting with partially unknown cost function [4]. Examples include data analytics in machine learning [5] as well as automatic controller tuning [6], [7].

We organize the literature review in two blocks: *collaborative/distributed extremum seeking* and *distributed zeroth-order/derivative-free optimization*. Early references on distributed extremum seeking for the so-called *consensus optimization* framework are [8] for a discrete-time setting and [9] for a continuous-time one. A proportional-integral extremum seeking design technique is proposed in [10], while, in [11], the gradient is approximated through a real-time protocol. In [12], authors propose the use of the sliding mode to generate the dither signal at the base of the extremum seeking. More recently, in [13], a distributed stochastic extremum seeking scheme is proposed for a source localization problem, while in [14] a distributed scheme approximating a Newton method is proposed. As for constraint-coupled distributed optimization, in [15], a distributed extremum seeking control, based on evolutionary game theory, is designed for real-time resource allocation. In [16], instead, a distributed, continuous-time scheme based on sign-based consensus is designed. In [17], a Lie bracket technique and extremum seeking are used for

problems with linear constraints. In [18], resource allocation problems are addressed by an extremum seeking scheme.

As for *distributed zeroth-order/derivative-free optimization*, authors in [19] develop a zeroth-order scheme based on a 1-point estimator and a gradient tracking policy. The work [20] instead proposes a zeroth-order algorithm based on a two-point estimator with a distributed gradient descent strategy and another one based on an n -point estimator with a gradient tracking policy. Authors in [21] propose a continuous-time gradient-free approach emulating a distributed gradient algorithm for which optimal asymptotic convergence is guaranteed. In [22], a sampled version of [21] is proposed. In [23], a continuous-time distributed algorithm based on random gradient-free oracles is proposed for convex optimization problems. In [24], the randomized gradient-free oracles introduced in [23] are used to build a gradient-free distributed algorithm in directed networks. Randomized gradient-free algorithms are used also in [25], where sequential Gaussian smoothing is used for non-smooth distributed convex constrained optimization. In [26], gradient-free optimization is addressed with the additional constraint that each agent can only transmit quantized information. The work in [27] instead develops a “directed-distributed projected pseudo-gradient” descent method for directed graphs. Paper [28] combines the gradient-free strategy of [23] with a saddle-point algorithm. Authors in [29] address an online constrained optimization problem by relying on the Kiefer-Wolfowitz algorithm to approximate the gradients, and [30] combines the estimation of the gradient via a “simultaneous perturbation stochastic approximation” technique with the so-called matrix exponential learning optimization method. In [31], a randomized gradient-free method is combined with a state-of-the-art distributed gradient descent approach for directed networks. In [32], an overview of zeroth-order methods based on a Frank-Wolfe framework is provided. Authors in [33] propose a distributed random gradient-free protocol to solve constrained optimization problems by using projection techniques. All the cited papers but [30] prove stability of the proposed schemes by using arguments not based on Lyapunov theory. We also note that [30] proposes an algorithm neither based on gradient tracking nor using extremum seeking.

This paper proposes a novel algorithm to solve a distributed optimization problem in which network agents can only evaluate their local cost function at a given point, but not its gradient. The proposed solution consists of a distributed protocol in which the (unavailable) local gradients are approximated through an extremum seeking scheme. The approximations of

This work was supported in part by the Italian Ministry of Foreign Affairs and International Cooperation”, grant number BR22GR01. The authors are with the Department of Electrical and Information Engineering, “Guglielmo Marconi”, University of Bologna, 40126 Bologna, Italy, e-mail: {nicola.mimmo2, guido.carnevale, a.testa, giuseppe.notarstefano}@unibo.it.

the gradients are used to feed a suitable tracking mechanism which, in turn, allows for steering the local solution estimates along approximations of the global descent direction. The distributed algorithm uses a further consensus action on the local solution estimates. The convergence of our scheme is proved through Lyapunov and averaging theories for discrete-time systems. It is worth mentioning that our scheme, together with the ones in [34], [35], is the only distributed extremum seeking scheme proposed in discrete-time with the following distinctive features. The work [34] (i) does not address a consensus optimization problem and (ii) relies on consensus dynamics estimating the global cost, while ours estimates the global gradient. Instead, in [35], (i) the addressed consensus optimization problems have scalar decision variables and (ii) an extremum seeking technique is combined with a distributed gradient algorithm, i.e., without a tracking mechanism. As for the literature on distributed zeroth-order/derivative-free methods, the closest work is [19] in which extremum seeking is not used and the global gradient is approximated via a randomized 1-point policy. Consistently with the comparison table provided in [19], we highlight that there are no other distributed algorithms in the literature using 1-point gradient estimators. Indeed, although the distributed scheme in [36] uses 1-point gradient estimators, we remark that it is tailored for partition-based optimization, i.e., a simplified setup in which each local function depends only on the neighbors' decision variables, see [3]. Also, our Lyapunov-based tools combined with averaging theory for discrete-time systems represent a distinctive feature in the algorithm analysis.

The paper unfolds as follows. Section II introduces the problem and the proposed algorithm. The main result is provided in Section III and proved in Section IV. In Section V, we test the proposed algorithm via numerical simulations.

Notation: Given N vectors $x_1, \dots, x_N \in \mathbb{R}^n$, we denote by $\text{col}(x_1, \dots, x_N)$ their column stacking. Given N scalars d_1, \dots, d_N , we denote by $\text{diag}(d_1, \dots, d_N)$ the diagonal matrix with i -th entry d_i . The Kronecker product is denoted by \otimes . The identity matrix in $\mathbb{R}^{n \times n}$ is I_n . The column vector of N ones is denoted by $\mathbf{1}_N$ and we define $\mathbf{1} := \mathbf{1}_N \otimes I_n$. Dimensions are omitted whenever they are clear from the context. Finally, for $r > 0$, we let $\mathcal{B}_r = \{x \in \mathbb{R}^n : \|x\| \leq r\}$.

II. PROBLEM FORMULATION

We consider a network of N agents communicating according to an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, where $\mathcal{V} = \{1, \dots, N\}$ is the set of agents, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges, and $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$ is the weighted adjacency matrix. Agent i and j can exchange information only if $(i, j) \in \mathcal{E}$. Accordingly, it holds $a_{ij} \geq 0$ if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. We denote as $\mathcal{N}_i = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}\}$ the set of neighbors of agent i . Moreover, we also associate to the graph the Laplacian matrix $\mathcal{L} := D - \mathcal{A} \in \mathbb{R}^{N \times N}$, where $D := \text{diag}(\text{deg}_1, \dots, \text{deg}_N) \in \mathbb{R}^{N \times N}$ is the so-called degree matrix in which $\text{deg}_i := \sum_{j \in \mathcal{N}_i} a_{ij}$ is the degree of agent i . The next assumption specifies the class of considered graphs.

Assumption 1: The graph \mathcal{G} is connected and the adjacency matrix $\mathcal{A} \in \mathbb{R}^{N \times N}$ is symmetric. \square

In the proposed distributed setup, each agent i is equipped with a sensor only providing measurements of the local cost function $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ and aims at solving the problem

$$\min_{w \in \mathbb{R}^n} \sum_{i=1}^N f_i(w). \quad (1)$$

We enforce the following assumptions about the problem.

Assumption 2: For all $i \in \{1, \dots, N\}$, the function f_i is \underline{L} -strongly convex for some $\underline{L} > 0$. \square

Assumption 3: Each cost f_i is \mathcal{C}^3 and has \bar{L}_i -Lipschitz continuous gradients. We denote $\bar{L} := \max\{\bar{L}_1, \dots, \bar{L}_N\}$. \square

Assumption 2 ensures the existence of a unique solution $x^* \in \mathbb{R}^n$ to problem (1). Our aim is to iteratively find it via a distributed algorithm. Namely, given the iteration index $t \in \mathbb{N}$ and by denoting with $w_i^t \in \mathbb{R}^n$ the i -th agent's estimate, at iteration t , of the solution to problem (1), our goal is to design a distributed protocol able to steer all these estimates to the minimizer x^* . The peculiar challenge of this paper is that each agent i cannot access either the gradients $\nabla f_i(w_i^t)$ (as in standard gradient-based methods) or the cost functions in arbitrary points (as in standard zeroth-order methods). More in detail, we assume that agent i can only use the single measurement $f_i(w_i^t)$ properly combined with the so-called dither signal $d_i^t \in \mathbb{R}^n$ and the amplitude parameter $\delta > 0$. The role of d_i^t and δ will become clearer in the next section.

III. EXTREMUM SEEKING TRACKING: ALGORITHM INTRODUCTION AND CONVERGENCE

Being the gradients ∇f_i not available, we replace them with an estimation based on a proper elaboration of the local costs excited via suitable dithering signals d_i^t . The arising distributed method, termed Extremum Seeking Tracking, is described in Algorithm 1, from agent i perspective. In Algorithm 1, $\gamma > 0$ represents the step size while the parameter $\delta > 0$ represents the amplitude of the dither signal d_i^t defined as

$$d_i^t = \text{col} \left(\sin \left(\frac{2\pi t}{\tau_{i_1}} + \phi_{i_1} \right), \dots, \sin \left(\frac{2\pi t}{\tau_{i_n}} + \phi_{i_n} \right) \right), \quad (2)$$

where $\tau_{i_p} \in \mathbb{N}$ and $\phi_{i_p} \in \mathbb{R}$ such that, given $p, q, r \in \{1, \dots, n\}$, $p \neq q$, $q \neq r$, $p \neq r$, it holds

$$\sum_{t=0}^{\tau_{\text{per}}-1} \sin \left(\frac{2\pi t}{\tau_{i_p}} + \phi_{i_p} \right) = 0 \quad (3a)$$

$$\sum_{t=0}^{\tau_{\text{per}}-1} \sin \left(\frac{2\pi t}{\tau_{i_p}} + \phi_{i_p} \right) \sin \left(\frac{2\pi t}{\tau_{i_q}} + \phi_{i_q} \right) = \frac{\tau_{\text{per}}}{2} \quad (3b)$$

$$\sum_{t=0}^{\tau_{\text{per}}-1} \sin \left(\frac{2\pi t}{\tau_{i_p}} + \phi_{i_p} \right) \sin \left(\frac{2\pi t}{\tau_{i_q}} + \phi_{i_q} \right) \sin \left(\frac{2\pi t}{\tau_{i_r}} + \phi_{i_r} \right) = 0, \quad (3c)$$

for all $i \in \{1, \dots, N\}$. Here, $\tau_{\text{per}} \in \mathbb{N}$ is the least common multiple of all periods τ_{i_p} . Sinusoidal dither functions are useful in practical applications to guarantee smooth inputs to the plant. However, other possibilities, e.g., as square or triangular waves, are possible [37].

The local gradient estimates generated by $\frac{f_i(w_i^t)d_i^t}{2\delta}$ are suitably interlaced with (i) the term $\sum_{j \in \mathcal{N}_i} \ell_{ij}(w_j^t - \delta d_j^t)$ to force consensus among the quantities $w_i^t - \delta d_i^t$ and (ii) a tracking

Algorithm 1 Extremum Seeking Tracking (agent i)

initialization: $x_i^0 \in \mathbb{R}^n$ and $z_i^0 = 0$

for $t = 0, 1, \dots$ **do**

$$w_i^{t+1} = w_i^t - \gamma \sum_{j \in \mathcal{N}_i} \ell_{ij} (w_j^t - \delta d_j^t) - \gamma s_i^t + \delta (d_i^{t+1} - d_i^t) \quad (4a)$$

$$s_i^{t+1} = s_i^t - \gamma \sum_{j \in \mathcal{N}_i} \ell_{ij} s_j^t + \frac{2}{\delta} (f_i(w_i^{t+1}) d_i^{t+1} - f_i(w_i^t) d_i^t) \quad (4b)$$

end for

mechanism to reconstruct the (estimated) global gradient. In detail, the consensus step is performed by using the entries ℓ_{ij} the (i, j) -entry of the Laplacian matrix \mathcal{L} associated to the graph \mathcal{G} , while the tracking mechanism is implemented by equipping each agent i with an auxiliary variable $s_i^t \in \mathbb{R}^n$. In this algorithm, agents exchange with their neighbors the information $\text{col}(w_i^t - \delta d_i^t, s_i^t)$ involving $2n$ components.

Remark 1: The main distinctive features of our method are as follows. First, errors in our method are given by third-order residuals as opposed to second-order ones in finite-difference methods. Second, gradient estimation is based on a single-function query per agent. This could be advantageous in scenarios in which multiple queries per agent are expensive or even not allowed. Third and final, the estimation policy updates, and so the convergence guarantees, are purely deterministic. \square

The next theorem formalizes the convergence properties of Extremum Seeking Tracking. To this end, for all $i \in \{1, \dots, N\}$ and given $r > 0$, we define the set $\mathcal{D}_{i,r} \subset \mathbb{R}^{2n}$ as $\mathcal{D}_{i,r} := \{\text{col}(w_i, s_i) \in \mathbb{R}^{2n} \mid \|w_i - x^*\| \leq r, s_i = 2f_i(w_i)/\delta\}$.

Theorem 1: Consider (4) and let Assumptions 1, 2, and 3 hold. Then, for any $r, \bar{\rho} > 0$, there exist $\gamma^*, \delta^*, k_1 > 0$, $\epsilon \in (0, \bar{\rho}/2)$, and $k_2 \geq (\bar{\rho}/2 - \epsilon)$ such that, for any $\gamma \in (0, \gamma^*)$, $\delta \in (0, \delta^*)$, and $\text{col}(w_i^0, s_i^0) \in \mathcal{D}_{i,r}$ for all $i \in \{1, \dots, N\}$, the trajectories of (4) are bounded and satisfy

$$\|w_i^t - x^*\| \leq \bar{\rho}, \quad (5)$$

for all $i \in \{1, \dots, N\}$ and $t \geq t^* := -\frac{1}{\gamma k_1} \ln((\bar{\rho}/2 - \epsilon)/k_2)$, i.e., the convergence to the set $\{w_i \in \mathbb{R}^n \mid \|w_i - x^*\| \leq \bar{\rho}\}$ is linear. \square

The proof of Theorem 1 is provided in Section IV-C. Theorem 1 provides a semi-global, practical exponential-stability result restricted to the set $\mathcal{D}_r := \mathcal{D}_{1,r} \times \dots \times \mathcal{D}_{N,r}$. Indeed, it is semi-global because the parameters γ^* and δ^* depend on the initial radius r and it is practical because they also depend on the arbitrary small final radius $\bar{\rho} > 0$.

IV. EXTREMUM SEEKING TRACKING: STABILITY ANALYSIS

In this section, we analyze Extremum Seeking Tracking. Assumptions 1, 2, and 3 hold throughout the whole section.

First, let the coordinates $x_i^t, z_i^t \in \mathbb{R}^n$ be defined as

$$x_i^t := w_i^t - \delta d_i^t, \quad z_i^t := s_i^t - \frac{2f_i(w_i^t)d_i^t}{\delta}, \quad (6)$$

for all $i \in \{1, \dots, N\}$, which allow us to rewrite (4) as

$$x_i^{t+1} = x_i^t - \gamma \left(\sum_{j \in \mathcal{N}_i} \ell_{ij} x_j^t + z_i^t + \frac{2f_i(x_i^t + \delta d_i^t)d_i^t}{\delta} \right) \quad (7a)$$

$$z_i^{t+1} = z_i^t - \gamma \sum_{j \in \mathcal{N}_i} \ell_{ij} \left(z_j^t + \frac{2f_j(x_j^t + \delta d_j^t)d_j^t}{\delta} \right). \quad (7b)$$

Remark 2: The new variables x_i^t and z_i^t allow us to interpret Extremum Seeking Tracking as an approximated discrete-time version of the continuous-time gradient tracking method proposed in [38]. Indeed, if the case gradients were available, the distributed algorithm (7) would become

$$x_i^{t+1} = x_i^t - \gamma \sum_{j \in \mathcal{N}_i} \ell_{ij} x_j^t - \gamma (z_i^t + \nabla f_i(x_i^t)) \quad (8a)$$

$$z_i^{t+1} = z_i^t - \gamma \sum_{j \in \mathcal{N}_i} \ell_{ij} (z_j^t + \nabla f_j(x_j^t)). \quad (8b)$$

The analysis required to prove Theorem 1 will also require the investigation of the convergence properties of (8) (see Lemma 3 and Remark 3). \square

Then, we aggregate the local updates in (7) obtaining the compact algorithm description

$$x^{t+1} = x^t - \gamma (Lx^t + z^t + f_d(t, x^t)) \quad (9a)$$

$$z^{t+1} = z^t - \gamma (Lz^t + Lf_d(t, x^t)), \quad (9b)$$

where we introduced $L := \mathcal{L} \otimes I_n$, $x^t := \text{col}(x_1^t, \dots, x_N^t)$, $z^t := \text{col}(z_1^t, \dots, z_N^t)$, $d^t := \text{col}(d_1^t, \dots, d_N^t)$, and the function $f_d : \mathbb{N} \times \mathbb{R}^{Nn} \rightarrow \mathbb{R}^{Nn}$ defined as

$$f_d(t, x) := \begin{bmatrix} 2f_1(x_1 + \delta d_1^t)d_1^t/\delta \\ \vdots \\ 2f_N(x_N + \delta d_N^t)d_N^t/\delta \end{bmatrix}, \quad (10)$$

where we decomposed x according to $x := \text{col}(x_1, \dots, x_N)$ with $x_i \in \mathbb{R}^n$ for all $i \in \{1, \dots, N\}$. We point out that system (9) can be conceived as an extremum seeking scheme with output map $f(x + \delta d^t)$, see also Fig. 1.

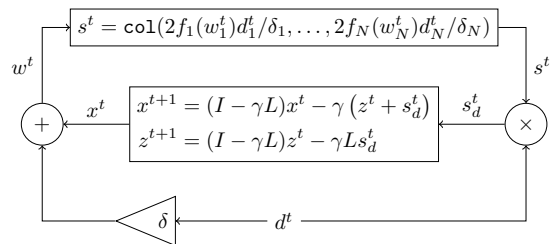


Fig. 1. Block scheme of the proposed Extremum Seeking Tracking algorithm in the (x, z) coordinates.

We now give an overview of the main steps of the stability analysis carried out to prove Theorem 1:

- (i) We perform two change of variables to describe the dynamics (9) in terms of the mean value (over the agents)

\tilde{z}_{avg} of z and the orthogonal part \tilde{z}_{\perp} (both in error coordinates). Then, by relying on averaging theory (see [39] for discrete-time systems or [40, Ch.10] and [41] for continuous-time ones), we introduce a suitable auxiliary system named *averaged system*. The latter is obtained by averaging the original algorithm dynamics over a common period. The averaged system is shown to be driven by the cost gradients with additive estimation errors.

- (ii) When neglecting these errors, the averaged system corresponds to an equivalent form of (8). Based on this observation, we rely on existing stability properties of the continuous gradient tracking to demonstrate that the trajectories of the averaged system exponentially converge to an arbitrarily small neighborhood of $\text{col}(\mathbf{1}x^*, \tilde{z}_{\perp}^{\text{eq}})$ for some $\tilde{z}_{\perp}^{\text{eq}}$ arising from the analysis.
- (iii) Finally, we prove Theorem 1 by exploiting the steps above and by using averaging theory to show the closeness between the trajectories of (9) and those of its averaged system.

Step (i) is performed in Section IV-A, step (ii) is carried out in Section IV-B, while Section IV-C is devoted to step (iii).

A. Coordinate changes and averaged system

We start by introducing a change of coordinates to highlight the error dynamics of (9) with respect to $\text{col}(x^*, -G(\mathbf{1}x^*))$, i.e., the point in which each x_i^t coincides with the optimal problem solution x^* and perfect tracking is achieved via z_i^t (see [38]). To this end, let $G : \mathbb{R}^{Nn} \rightarrow \mathbb{R}^{Nn}$ be defined as

$$G(x) := \text{col}(\nabla f_1(x_1), \dots, \nabla f_N(x_N)). \quad (11)$$

Then, let the error coordinates $\tilde{x}, \tilde{z} \in \mathbb{R}^{Nn}$ be defined as

$$\tilde{x} := x - \mathbf{1}x^*, \quad \tilde{z} = z + G(\mathbf{1}x^*), \quad (12)$$

and let us introduce $\phi_{xz} : \mathbb{N} \times \mathbb{R}^{Nn} \times \mathbb{R}^{Nn} \rightarrow \mathbb{R}^{2Nn}$ as

$$\phi_{xz}(t, \tilde{x}, \tilde{z}) := \begin{bmatrix} -L\tilde{x} - \tilde{z} - f_d(t, \tilde{x} + \mathbf{1}x^*) + G(\mathbf{1}x^*) \\ -L\tilde{z} - L(f_d(t, \tilde{x} + \mathbf{1}x^*) - G(\mathbf{1}x^*)) \end{bmatrix}.$$

Then, by using the new coordinates, we rewrite (9) as

$$\begin{bmatrix} \tilde{x}^{t+1} \\ \tilde{z}^{t+1} \end{bmatrix} = \begin{bmatrix} \tilde{x}^t \\ \tilde{z}^t \end{bmatrix} + \gamma \phi_{xz}(t, \tilde{x}^t, \tilde{z}^t), \quad (13)$$

where we have used the property $L\mathbf{1} = 0$. As in [38], we take advantage of the initialization $z_i^0 = 0$ for all $i \in \{1, \dots, N\}$. To this end, we introduce the novel coordinates $\tilde{z}_m \in \mathbb{R}^n$ and $\tilde{z}_{\perp} \in \mathbb{R}^{(N-1)n}$ representing the average of the variables \tilde{z}_i^t (over the agents) and the orthogonal counterpart, namely

$$\begin{bmatrix} \tilde{z}_m \\ \tilde{z}_{\perp} \end{bmatrix} := \begin{bmatrix} \mathbf{1}^{\top} \\ R^{\top} \end{bmatrix} \tilde{z}, \quad (14)$$

where we introduced the matrix $R \in \mathbb{R}^{Nn \times (N-1)n}$ such that $R^{\top}\mathbf{1} = 0$, $R^{\top}R = I$. Further, given $n_{\xi} := (2N-1)n$, let us also introduce $\xi \in \mathbb{R}^{n_{\xi}}$ defined as

$$\xi := \begin{bmatrix} \tilde{x}^{\top} & \tilde{z}_{\perp}^{\top} \end{bmatrix}^{\top}. \quad (15)$$

Since $\mathbf{1}^{\top}L = 0$ in light of Assumption 1 and $\mathbf{1}^{\top}G(\mathbf{1}x^*) = \sum_{i=1}^N \nabla f_i(x^*) = 0$, we use (14) and (15) to rewrite (13) as

$$\xi^{t+1} = \xi^t + \gamma \phi_{\xi}(t, \text{col}(\tilde{x}^t, \mathbf{1}\tilde{z}_m^t + R\tilde{z}_{\perp}^t)) \quad (16a)$$

$$\tilde{z}_m^{t+1} = \tilde{z}_m^t, \quad (16b)$$

where we introduced $\phi_{\xi} : \mathbb{N} \times \mathbb{R}^{2Nn} \rightarrow \mathbb{R}^{n_{\xi}}$ defined as

$$\phi_{\xi}(t, \text{col}(\tilde{x}, \tilde{z})) := \begin{bmatrix} I & 0 \\ 0 & R \end{bmatrix} \phi_{xz}(t, \tilde{x}, \tilde{z}). \quad (17)$$

The equation (16b) implies that $\tilde{z}_m^t = \tilde{z}_m^0$ for all $t \in \mathbb{N}$. Then, since $z_i^0 = 0$ for all $i \in \{1, \dots, N\}$ and $\mathbf{1}^{\top}G(\mathbf{1}x^*) = 0$, it holds $\tilde{z}_m^0 = 0$ which allows us to ignore (16b) and rewrite (16) according to the equivalent, reduced system

$$\xi^{t+1} = \xi^t + \gamma \phi(t, \xi^t), \quad (18)$$

where $\phi : \mathbb{N} \times \mathbb{R}^{n_{\xi}} \rightarrow \mathbb{R}^{n_{\xi}}$ is introduced to compactly describe system (16a) with $\tilde{z}_m^t = 0$ for all $t \in \mathbb{N}$, i.e., ϕ is defined as

$$\begin{aligned} \phi(t, \xi) &:= \phi_{\xi}(t, \text{col}(\tilde{x}, R\tilde{z}_{\perp})) \quad (19) \\ &\stackrel{(a)}{=} \begin{bmatrix} I & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} -L\tilde{x} - R\tilde{z}_{\perp} - f_d(t, \tilde{x} + \mathbf{1}x^*) + G(\mathbf{1}x^*) \\ -LR\tilde{z}_{\perp} - L(f_d(t, \tilde{x} + \mathbf{1}x^*) - G(\mathbf{1}x^*)) \end{bmatrix}, \end{aligned}$$

where in (a) we used the definition of ϕ_{ξ} (cf. (17)).

We resort to the averaging theory [39] to analyze the time-varying system (18). As customary when this tool is employed, we introduce an auxiliary scheme typically named *averaged system*, obtained by averaging the time-varying vector field $\phi(t, \xi)$ over τ_{per} samples by *freezing* the system state. In detail, the averaged system associated with (18) is

$$\xi_{\text{avg}}^{t+1} = \xi_{\text{avg}}^t + \gamma \phi_{\text{avg}}(\xi_{\text{avg}}^t), \quad (\text{averaged system}) \quad (20)$$

with $\xi_{\text{avg}}^0 = \xi_0$, where $\phi_{\text{avg}} : \mathbb{R}^{n_{\xi}} \rightarrow \mathbb{R}^{n_{\xi}}$ is defined as

$$\phi_{\text{avg}}(\xi) := \frac{1}{\tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} \phi(k, \xi), \quad \text{for any } t \geq 0. \quad (21)$$

Notice that, since $\phi(k, \xi)$ is periodic with period τ_{per} on the first argument, the function $\phi_{\text{avg}}(\xi)$ (cf. (21)) is time-invariant.

The next lemma shows that the gradient of each f_i can be approximated by averaging each block of f_d (cf. (10)) over the period τ_{per} on the first argument. For this reason, the next lemma will be useful to explicitly write ϕ_{avg} (cf. (21)) in terms of $G(x)$, (i.e., the stack of the gradients $\nabla f_i(x_i)$, see (11)).

Lemma 1 (Gradient estimation): For all $i \in \{1, \dots, N\}$, there exists $\ell_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$ such that, for any given $x_i \in \mathbb{R}^n$ and all $t \in \mathbb{N}$, it holds

$$\frac{2}{\delta \tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} f_i(x_i + \delta d_i^k) d_i^k = \nabla f_i(x_i) + \delta^2 \ell_i(x_i). \quad (22)$$

Moreover, given any compact set $\mathcal{S}_i \subset \mathbb{R}^n$, if $\delta \in (0, 1]$, there exists $L_{i, \mathcal{S}_i} > 0$ such that

$$\|\ell_i(x_i)\| \leq L_{i, \mathcal{S}_i}, \quad (23)$$

for all $x_i \in \mathcal{S}_i$ and $i \in \{1, \dots, N\}$. \square

The proof of Lemma 1 is in Appendix A.

We note that the quantity on the right-hand side of (22) is time-invariant because d_i^k is periodic, with period τ_{per} , and the

left-hand side of (22) is averaged over the period τ_{per} . Now, let us introduce the function $\ell : \mathbb{R}^{Nn} \rightarrow \mathbb{R}^{Nn}$ stacking all the approximation errors $\ell_i(x_i)$ used in (22), namely

$$\ell(x) := \text{col}(\ell_1(x_1), \dots, \ell_N(x_N)). \quad (24)$$

Then, by using (22), the definitions of f_d in (10), $G(x)$ as the stack of the gradients $\nabla f_i(x_i)$ in (11), and $\ell(x)$ as the stack of the approximation errors $\ell_i(x_i)$ in (24), it holds

$$\frac{1}{\tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} f_d(k, x) = G(x) + \ell(x), \quad (25)$$

for all $x \in \mathbb{R}^{Nn}$ and $t \in \mathbb{N}$. Hence, by introducing $\xi_{\text{avg}} := \text{col}(\tilde{x}_{\text{avg}}, \tilde{z}_{\perp, \text{avg}})$ and by combining the definition of ϕ_{avg} in (21), ϕ in (19), and (25), we obtain

$$\begin{aligned} \phi_{\text{avg}}(\xi_{\text{avg}}) &= \begin{bmatrix} -L\tilde{x}_{\text{avg}} - R\tilde{z}_{\perp, \text{avg}} - G(\tilde{x}_{\text{avg}} + \mathbf{1}x^*) + G(\mathbf{1}x^*) \\ -R^\top LR\tilde{z}_{\perp, \text{avg}} - R^\top L(G(\tilde{x}_{\text{avg}} + \mathbf{1}x^*) - G(\mathbf{1}x^*)) \end{bmatrix} \\ &+ \begin{bmatrix} -\ell(\tilde{x}_{\text{avg}} + \mathbf{1}x^*) \\ -R^\top L\ell(\tilde{x}_{\text{avg}} + \mathbf{1}x^*) \end{bmatrix}. \end{aligned} \quad (26)$$

We define

$$\begin{aligned} \phi_{GT}(\xi_{\text{avg}}) &:= \begin{bmatrix} -L\tilde{x}_{\text{avg}} - R\tilde{z}_{\perp, \text{avg}} - G(\tilde{x}_{\text{avg}} + \mathbf{1}x^*) + G(\mathbf{1}x^*) \\ -R^\top LR\tilde{z}_{\perp, \text{avg}} - R^\top L(G(\tilde{x}_{\text{avg}} + \mathbf{1}x^*) - G(\mathbf{1}x^*)) \end{bmatrix} \\ B &:= \begin{bmatrix} -I \\ -R^\top L \end{bmatrix}, \quad u(\xi_{\text{avg}}) := \ell(\tilde{x}_{\text{avg}} + \mathbf{1}x^*) \end{aligned}$$

to short (26) as

$$\phi_{\text{avg}}(\xi_{\text{avg}}) = \phi_{GT}(\xi_{\text{avg}}) + \delta^2 Bu(\xi_{\text{avg}}),$$

which, in turn, allows us to rewrite (20) as

$$\xi_{\text{avg}}^{t+1} = \xi_{\text{avg}}^t + \gamma \phi_{GT}(\xi_{\text{avg}}^t) + \gamma \delta^2 Bu(\xi_{\text{avg}}^t) \quad (\text{averaged system}). \quad (27)$$

B. Averaged System Analysis

In this subsection, we analyze the averaged system (27). To this end, we first consider an additional nominal system in which the term $\gamma \delta^2 Bu(\xi_{\text{avg}}^t)$ (i.e., the term describing the gradients' estimation error) is neglected. Then, by using such a nominal system analysis as a building block, we provide the result concerning system (27). Therefore, we start by studying

$$\xi_{\text{avg}}^{t+1} = \xi_{\text{avg}}^t + \gamma \phi_{GT}(\xi_{\text{avg}}^t), \quad (28)$$

which corresponds to system (27) in the case of $\gamma \delta^2 Bu(\xi_{\text{avg}}^t) = 0$. The next lemma proves the global exponential stability of the origin for (28).

Lemma 2: There exist $P = P^\top \in \mathbb{R}^{n_\xi \times n_\xi}$ and $a_1, a_2, c_1, \gamma_0 > 0$ such that, for any $\gamma \in (0, \gamma_0)$, along the trajectories of (28) it holds

$$a_1 I \leq P \leq a_2 I \quad (29a)$$

$$\xi_{\text{avg}}^{t+1 \top} P \xi_{\text{avg}}^{t+1} - \xi_{\text{avg}}^{t \top} P \xi_{\text{avg}}^t \leq -\gamma c_1 \|\xi_{\text{avg}}^t\|^2, \quad (29b)$$

for all $\xi_{\text{avg}}^t \in \mathbb{R}^{n_\xi}$. \square

The proof of Lemma 2 is in Appendix B.

Remark 3: Notice that Lemma 2 proves that algorithm (8) linearly converges to the minimizer of (1), since (28) is an equivalent formulation of (8). \square

With this result at hand, we analyze the impact of $u(\cdot)$ thus obtaining the stability properties of the averaged system (27).

Lemma 3: Consider the averaged system (27). Then, for any $r_\xi > 0$ and $\rho \in (0, r_\xi)$, there exist $c_3 \in (0, c_1)$ and $\delta_1^* \in (0, 1]$ such that, for any $\gamma \in \min\{\gamma_0, 1\}$, $\delta \in (0, \delta_1^*)$, and $\|\xi_{\text{avg}}^0\| \leq r_\xi$, it holds

- (i) $\xi_{\text{avg}}^t \in \mathcal{B}_{\sqrt{a_2/a_1} r_\xi}$ for all $t \in \mathbb{N}$,
- (ii)

$$\|\xi_{\text{avg}}^t\| \leq \sqrt{a_2/a_1} \exp(-t\gamma c_3) \|\xi_{\text{avg}}^0\|, \quad (30)$$

for all $\|\xi_{\text{avg}}^t\| \geq \rho$. \square

The proof of Lemma 3 is in Appendix C.

Remark 4: The result of Lemma 3 only involves the averaged system (27). In the next section, such a result will be used as a building block to study the original dynamics, i.e., system (18) and, thus, to conclude the proof of Theorem 1. However, it is a per se result that can be used to show robust stability for the distributed algorithm (8). \square

C. Proof of Theorem 1

Despite averaging tools for discrete-time systems are already present in the literature, see [39] for example, we got the inspiration from continuous-time averaging [40, Ch. 10] and [41] for elaborating the proof of Theorem 1 to make clear how γ affects the closeness of the trajectories of (18) and (27). Since Assumptions 1, 2, and 3 hold, we apply Lemma 2 to claim that there exist $P = P^\top \in \mathbb{R}^{n_\xi \times n_\xi}$ and $a_1, a_2, c_1, \gamma_0 > 0$ such that, if $\gamma \in (0, \gamma_0)$, the conditions (29) are satisfied. Then, we evaluate the norm of the initial conditions of system (18) and (27), i.e., $\|\xi^0\| = \|\xi_{\text{avg}}^0\|$. By using the definition of ξ (cf. (15)), the changes of variables (12) and (14), and the triangle inequality, we get

$$\begin{aligned} \|\xi^0\| &\leq \|x^0 - \mathbf{1}x^*\| + \|R^\top(z^0 + G(\mathbf{1}x^*))\| \\ &+ \left\| \frac{\mathbf{1}^\top}{N}(z^0 + G(\mathbf{1}x^*)) \right\| \stackrel{(a)}{\leq} r\sqrt{N} + \|R\| \|G(\mathbf{1}x^*)\|, \end{aligned}$$

where in (a) we combine the initialization $\|x_i^0 - x^*\| \leq r$ and $z_i^0 = 0$ for all $i \in \{1, \dots, N\}$ with the fact that $\mathbf{1}^\top G(\mathbf{1}x^*) = \sum_{i=1}^N f_i(x^*) = 0$. Hence, by defining $r_\xi := r\sqrt{N} + \|R\| \|G(\mathbf{1}x^*)\|$, we claim that

$$\|\xi^0\| = \|\xi_{\text{avg}}^0\| \leq r_\xi.$$

Once the initial distance from the origin has been evaluated, we choose any $\bar{\rho} > 0$, set $c_2 := \sqrt{a_2/a_1}$, and choose any $\epsilon \in (0, \bar{\rho}(1 + c_2)/2)$. Then, we pick $\rho \in (0, (\bar{\rho}/2 - (1 + c_2)\epsilon)/c_2)$, $c'_1 \in (0, c_1)$, and use the matrix P satisfying (29) to apply Lemma 3. Specifically, we claim that there exist $c_3 > 0$, $\delta_1^* \in (0, 1]$, and $\gamma_0 > 0$ such that, for any $\delta \in (0, \delta_1^*)$ and $\gamma \in (0, \min\{\gamma_0, 1\})$, it holds $\xi_{\text{avg}}^t \in \mathcal{B}_{c_2 r_\xi}$ for all $t \in \mathbb{N}$ and

$$\|\xi_{\text{avg}}^t\| \leq c_2 \exp(-t\gamma c_3) \|\xi_{\text{avg}}^0\|, \quad (31)$$

for all ξ_{avg}^t such that $\|\xi_{\text{avg}}^t\| \geq \rho$. Now, we proceed by finding a bound $\gamma_1 > 0$ such that, for any $\gamma \in (0, \gamma_1)$, we guarantee the ϵ -closeness between the state ξ_{avg}^t of the averaged system (20)

and the one of the original system (18), i.e., that $\|\xi^t - \xi_{\text{avg}}^t\| \leq \epsilon$ holds true for all $t \in \mathbb{N}$. To this end, let us introduce

$$v(t, \xi_{\text{avg}}) := \sum_{k=0}^{t-1} (\phi(k, \xi_{\text{avg}}) - \phi_{\text{avg}}(\xi_{\text{avg}})).$$

By using this definition and the one of ϕ_{avg} (cf. (21)), it holds

$$\begin{aligned} v(t+1, \xi_{\text{avg}}^{t+1}) - v(t, \xi_{\text{avg}}^t) \\ = \phi(t, \xi_{\text{avg}}^{t+1}) - \phi_{\text{avg}}(\xi_{\text{avg}}^{t+1}) + v(t, \xi_{\text{avg}}^{t+1}) - v(t, \xi_{\text{avg}}^t). \end{aligned} \quad (32)$$

Then, let $r'_\xi := c_2 r_\xi$ and define $\Delta := \delta \sqrt{Nn}$. Under the assumption of $\xi^t \in \mathcal{B}_{r'_\xi + \epsilon}$ for all $t \in \mathbb{N}$ (later verified by a proper selection of γ), we claim that the arguments of the functions f_i and their derivatives (embedded into the definitions of $\phi(t, \cdot)$ and $\phi_{\text{avg}}(\cdot)$ and their derivatives) lie into the compact set $\mathcal{B}_{r'_\xi + \epsilon + \Delta}$. Thus, since the functions f_i and its derivatives are continuous (cf. Assumption 3) and the functions $\phi(\cdot, \cdot)$ and $v(\cdot, \cdot)$ are periodic in the first argument, we define

$$L_\phi := \sup_{\substack{\xi \in \mathcal{B}_{r'_\xi + \epsilon} \\ t \in [0, \tau_{\text{per}}]}} \left\{ \|\phi(t, \xi)\|, \|\phi_{\text{avg}}(\xi)\|, \left\| \frac{\partial \phi(t, \xi)}{\partial \xi} \right\|, \left\| \frac{\partial \phi_{\text{avg}}(\xi)}{\partial \xi} \right\|, \left\| \frac{\partial v(t, \xi)}{\partial \xi} \right\| \right\}. \quad (33)$$

Consequently, for all $\xi, \xi' \in \mathcal{B}_{r'_\xi + \epsilon}$ and $t \in \mathbb{N}$, it holds

$$\|v(t, \xi)\| \leq 2L_\phi \tau_{\text{per}} \quad (34a)$$

$$\|\phi(t, \xi) - \phi(t, \xi')\| \leq L_\phi \|\xi - \xi'\| \quad (34b)$$

$$\|\phi_{\text{avg}}(\xi) - \phi_{\text{avg}}(\xi')\| \leq L_\phi \|\xi - \xi'\| \quad (34c)$$

$$\|v(t, \xi) - v(k, \xi')\| \leq 2L_\phi \tau_{\text{per}} \|\xi - \xi'\| \quad (34d)$$

$$\|\phi_{\text{avg}}(\xi)\| \leq L_\phi. \quad (34e)$$

Let us introduce $\zeta^t \in \mathbb{R}^{n_\xi}$ defined as

$$\zeta^t := \xi_{\text{avg}}^t + \gamma v(t, \xi_{\text{avg}}^t). \quad (35)$$

Then, it holds

$$\xi^t - \zeta^t = \sum_{k=0}^{t-1} (\xi^{k+1} - \xi^k) - (\zeta^{k+1} - \zeta^k),$$

add $\pm \gamma \sum_{k=0}^{t-1} (\phi(k, \zeta^k) + \phi(k, \xi_{\text{avg}}^k))$, and use (32) to get

$$\begin{aligned} \xi^t - \zeta^t &= \gamma \sum_{k=0}^{t-1} (\phi(k, \xi^k) - \phi(k, \zeta^k)) \\ &+ \gamma \sum_{k=0}^{t-1} (\phi(k, \zeta^k) - \phi(k, \xi_{\text{avg}}^k)) \\ &- \gamma \sum_{k=0}^{t-1} (\phi(k, \xi_{\text{avg}}^{k+1}) - \phi(k, \xi_{\text{avg}}^k)) \\ &+ \gamma \sum_{k=0}^{t-1} (\phi_{\text{avg}}(\xi_{\text{avg}}^{k+1}) - \phi_{\text{avg}}(\xi_{\text{avg}}^k)) \\ &- \gamma \sum_{k=0}^{t-1} (v(k, \xi_{\text{avg}}^{k+1}) - v(k, \xi_{\text{avg}}^k)). \end{aligned}$$

Use (18), (27), and (34) to bound

$$\|\xi^t - \zeta^t\| \leq \gamma L_\phi \sum_{k=0}^{t-1} \|\xi^k - \zeta^k\| + \gamma^2 L_\phi^2 2(1 + 2\tau_{\text{per}})t. \quad (36)$$

Apply the discrete Gronwall inequality (see [42], [43]) and

$$\sum_{k=0}^{t-1} \gamma L_\phi k \exp(-\gamma L_\phi k) \leq \sum_{k=0}^{\infty} \gamma L_\phi k \exp(-\gamma L_\phi k) = 1$$

to further bound (36) as

$$\begin{aligned} \|\xi^t - \zeta^t\| &\leq \gamma^2 L_\phi^2 2(1 + 2\tau_{\text{per}})t \\ &+ \gamma L_\phi 2(1 + 2\tau_{\text{per}}) \exp(\gamma L_\phi t). \end{aligned} \quad (37)$$

The definition of ζ^t (35) and the triangle inequality lead to

$$\begin{aligned} \|\xi^t - \xi_{\text{avg}}^t\| &\leq \|\xi^t - \zeta^t\| + \gamma \|v(t, \xi_{\text{avg}}^t)\| \\ &\stackrel{(a)}{\leq} \gamma^2 L_\phi^2 2(1 + 2\tau_{\text{per}})t + \gamma 2L_\phi \tau_{\text{per}} \\ &+ \gamma L_\phi 2(1 + 2\tau_{\text{per}}) \exp(\gamma L_\phi t). \end{aligned} \quad (38)$$

where (a) uses (37) to bound the first term and (34a) to bound the second one. Then, set $\theta^* \in \mathbb{R}$ such that

$$\theta^* \geq -\frac{1}{c_3} \ln \left(\frac{(\bar{\rho}/2 - \epsilon)/c_2}{c_2 r_\xi} \right). \quad (39)$$

Let $\gamma_2 := \frac{\epsilon/(3c_2)}{L_\phi^2 2(1 + 2\tau_{\text{per}})\theta^*}$, $\gamma_3 := \frac{\epsilon/(3c_2)}{2L_\phi(1 + 2\tau_{\text{per}})\exp(L_\phi \theta^*)}$, $\gamma_4 := \frac{\epsilon/(3c_2)}{2L_\phi \tau_{\text{per}}}$, $\gamma_1 := \min\{\gamma_0, \gamma_2, \gamma_3, \gamma_4, 1\}$. Pick $\gamma \in (0, \gamma_1)$ such that $t^* := \frac{\theta^*}{\gamma} \in \mathbb{N}$. This can be done without loss of generality since θ^* is a design parameter. Then, we bound (38) as

$$\|\xi^t - \xi_{\text{avg}}^t\| \leq \frac{\epsilon}{c_2}, \quad (40)$$

for all $t \in \{0, \dots, t^*\}$. As a consequence, since $\xi_{\text{avg}}^t \in \mathcal{B}_{r'_\xi}$ for all $t \in \mathbb{N}$, it holds $\xi^t \in \mathcal{B}_{r'_\xi + \epsilon}$ for all $t \in \{0, \dots, t^*\}$, i.e., we have verified that the bounds (34) can be used into the interval $\{0, \dots, t^*\}$. Moreover, the exponential law (31) and the expression of θ^* (cf. (39)) ensure that it holds

$$\|\xi_{\text{avg}}^t\| \leq (\bar{\rho}/2 - \epsilon)/c_2, \quad (41)$$

for all $t \geq t^*$. Now, by using the triangle inequality, we write

$$\|\xi^{t^*}\| \leq \|\xi^{t^*} - \xi_{\text{avg}}^{t^*}\| + \|\xi_{\text{avg}}^{t^*}\| \stackrel{(a)}{\leq} \frac{\bar{\rho}}{2c_2}, \quad (42)$$

where in (a) we combined (40) and (41). The inequality (42) guarantees that $\xi^{t^*} \in \mathcal{B}_{\frac{\bar{\rho}}{2c_2}}$, hence we proved that the trajectories of (27) enters into $\mathcal{B}_{\frac{\bar{\rho}}{2c_2}}$ with linear rate. Next, in order to show that $\xi^t \in \mathcal{B}_{\bar{\rho}/2}$ for all $t \geq t^*$, we divide the set of natural numbers in intervals as $\mathbb{N} = \{0, \dots, t^*\} \cup \{t^*, \dots, 2t^*\} \cup \dots$. Define $\psi_{\text{avg}}(k + t^*, \xi^{t^*})$ as the solution to (27) for $\xi_{\text{avg}}^0 = \xi^{t^*}$ and $k \in \{0, \dots, t^*\}$. Thus, at the beginning of each interval $\{t^*, \dots, 2t^*\}$, the initial condition of (27) coincides with the one of $\psi_{\text{avg}}(k + t^*, \xi^{t^*})$ and lies into $\mathcal{B}_{\bar{\rho}/2} \subseteq \mathcal{B}_{r_\xi}$. Thus, we apply the same arguments above to guarantee that

$$\begin{aligned} \|\xi^{k+t^*} - \psi_{\text{avg}}(k + t^*, \xi^{t^*})\| &\leq \epsilon \\ \psi_{\text{avg}}(2t^*, \xi^{t^*}) &\in \mathcal{B}_{(\bar{\rho}/2 - \epsilon)/c_2}, \end{aligned}$$

for all $\gamma \in (0, \gamma^*)$ and $k \in \{0, \dots, t^*\}$. By using Lemma 3, we guarantee that system (27) cannot escape from $\mathcal{B}_{\bar{\rho}/2 - \epsilon}$, namely

$\xi_{\text{avg}}^t \in \mathcal{B}_{\bar{\rho}/2-\epsilon}$ for all $t \geq t^*$. Thus, we get $\xi^t \in \mathcal{B}_{\bar{\rho}/2}$ for all $t \in \{t^*, \dots, 2t^*\}$. By recursively applying the same arguments above for each interval $\{jt^*, \dots, (j+1)t^*\}$ with $j = 2, 3, \dots$ and using $\|x_i^t - x^*\| \leq \|\xi^t\|$ for all $i \in \{1, \dots, N\}$ and $t \in \mathbb{N}$, we get

$$\|x_i^t - x^*\| \leq \bar{\rho}/2, \quad (43)$$

for all $i \in \{1, \dots, N\}$ and $t \in \mathbb{N}$. The change of coordinates (6), (43), and the triangle inequality lead to

$$\|w_i^t - x^*\| \leq \frac{\bar{\rho}}{2} + \delta \|d_i^t\| \stackrel{(a)}{\leq} \frac{\bar{\rho}}{2} + \delta \sqrt{n}, \quad (44)$$

where in (a) we use the boundedness of the dither signals. The proof follows from (44) by setting $\delta^* := \min \left\{ \delta_1^*, \frac{\bar{\rho}}{2\sqrt{n}} \right\}$.

V. NUMERICAL COMPUTATIONS ON DISTRIBUTED PERSONALIZED OPTIMIZATION

To corroborate the theoretical analysis, in this section, we provide numerical computations for the proposed distributed algorithm on a personalized optimization framework.

In several engineering applications, a problem of interest consists of optimizing a performance metric while keeping in account user discomfort terms is usually not known in advance but can be only accessed by measurements. Specifically, we associate to each agent $i \in \{1, \dots, N\}$ a cost function in the form $f_i(w) = w^\top Q_i w + r_i^\top w + \log(\sum_{\ell=1}^n a_{i\ell} e^{b_{i\ell} w_\ell})$ with $Q_i = Q_i^\top \in \mathbb{R}^{n \times n}$, $r_i \in \mathbb{R}^n$ and $a_{i\ell}, b_{i\ell} > 0$, for all $\ell \in \{1, \dots, n\}$. In the following, we provide different sets of simulations to study in detail the features of Extremum Seeking Tracking. In particular, each set consists of Monte Carlo simulations over 20 randomly generated scenarios in which each one differs from the other in terms of cost functions, communication graphs, and algorithmic variables' initialization. In particular, unless differently stated, in each trial the agents communicate according to Erdős-Rényi random graphs (see, e.g., [45]) with edge probabilities equal to 0.2. As for the problem parameters, for each trial and all $i \in \{1, \dots, N\}$, we generate each matrix Q_i by pre- and post-multiplying a diagonal matrix (whose diagonal elements are randomly extracted from the interval $[10^{-3}, 5 \cdot 10^{-3}]$ with uniform probability) with an orthonormal matrix (randomly generated by extracting its elements from the interval $[0, 1]$ with uniform probability) and its transpose, respectively. Further, for all $i \in \{1, \dots, N\}$ and $\ell \in \{1, \dots, n\}$, we randomly extract the components of r_i within the interval $[-10^{-2}, 3 \cdot 10^{-2}]$ and the parameters $a_{i\ell}, b_{i\ell}$ within the interval $[0, 10^{-3}]$ with a uniform probability. In each set of simulations, we choose the parameters τ_{i_p} and ϕ_{i_p} according to the following procedure. For all $i \in \{1, \dots, N\}$, we take $\phi_{i_p} = \frac{\pi}{4} (1 + (-1)^p)$ for all $p = 1, \dots, n$, while τ_{i_p} have been chosen as the first $\lfloor (n+1)/2 \rfloor$ elements of odd numbers greater than 3 since it is possible to show that such a set of frequencies satisfy (3). Roughly speaking, $\phi_{i,p} = 0$ and $\phi_{i,p} = \pi/2$ are used to create orthogonal functions with the same frequency. It is important to note that this selection is not unique. As for the algorithm parameters γ and δ , due to the complexity of the dependencies of the bounds γ^* and

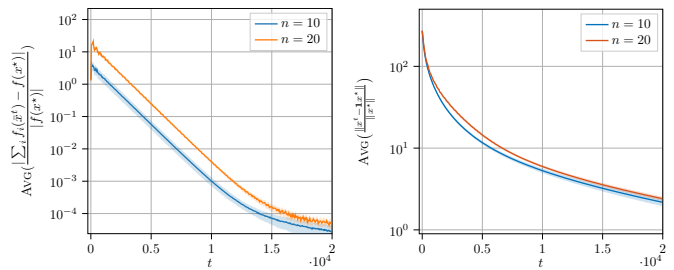


Fig. 2. Monte Carlo simulations for large scale problems: mean and 1-standard deviation band of cost error (left) and variable error (right).

δ^* provided in Theorem 1, we choose them via a trial-and-error procedure. In each set of simulations, the performance is evaluated by providing graphical results involving the relative errors $|\sum_i f_i(\bar{x}^t) - f(x^*)|/|f(x^*)|$ and $\|\bar{x}^t - x^*\|/\|x^*\|$ achieved along the trials of the Monte Carlo simulations, where $\bar{x}^t := \frac{1}{N} \sum_{i=1}^N x_i^t$. Simulations are performed using DISROPT [46], a Python package based on MPI to encode and simulate distributed optimization algorithms.

A. Monte Carlo simulations for large-scale problems

First, we perform numerical simulations over large-scale problems with networks made of $N = 250$ agents. We consider different optimization variable sizes, namely $n = 10, 20$. Part of these simulations has been run on the Marconi100 HPC Cluster of the Italian Cineca. We used 10 nodes of the cluster and, for each node, we used 25 cores and 4 GPUs. The code has been adapted in order to perform part of the computation directly on GPUs. The results are shown in Fig. 2. In detail, as the number of agents increases, the Lipschitz constant of the system to be averaged increases too. Moreover, a larger domain of initial conditions also implies a potentially larger L_ϕ constant (cf. (33)). This implies smaller γ^* , which, fixed the other parameters, makes the convergence slower. The decision variable dimension instead impacts the selection of the dither signal. A larger number of states implies a larger number of frequencies. This, in turn, means a longer time to estimate the gradient (cf. Lemma 1). Notice that, however, the accuracy of the final estimate is guaranteed by design. Indeed, since δ and γ are designed on $\bar{\rho}$, the trajectories of (5) converge to a ball of radius $\bar{\rho}$ independently of the problem size.

B. Monte Carlo simulations varying number of agents

Second, we test Extremum Seeking Tracking in a framework with $n = 10$ and different number of agents, i.e., $N = 5, 10, 20, 30$. For each value of N , we generate communication graphs with a diameter d such that the ratio N/d is constant while varying N . The results depicted in Fig. 3 show that the algorithm slows down as the number of agents increases.

C. Monte Carlo simulations varying the size n

Then, we test the algorithm features with a fixed number of agents $N = 10$ and different decision variable dimensions $n = 1, 10, 100$. The achieved results are reported in Fig. 4 and show that Algorithm 1 slows down as n increases.

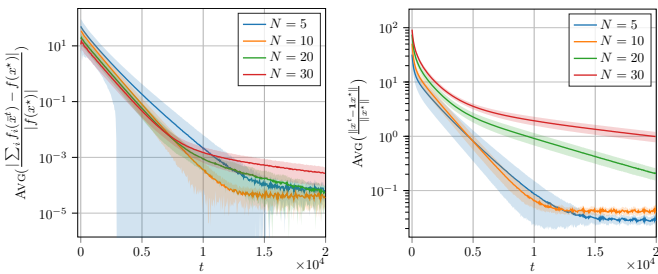


Fig. 3. Monte Carlo simulations for a varying number of agents: mean and 1-standard deviation band of cost error (left) and variable error (right).

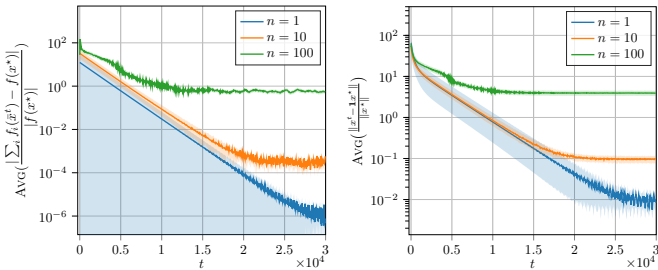


Fig. 4. Monte Carlo simulations varying n : mean and 1-standard deviation band of cost error (left) and variable error (right).

We conclude this part by providing in Fig. 5 the results of a single trial in the case with $n = 1$ to show the evolution of the solution estimate of each agent in error coordinate with respect to the optimal solution to the problem.

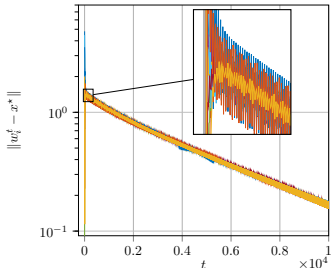


Fig. 5. Evolution of the agents' estimates w_i^t in coordinate error with respect to the optimal solution x^* .

D. Monte Carlo simulations varying the parameter δ

Third, we provide numerical simulations in which we vary δ in the case in which $N = n = 10$. As one may expect from Theorem 1, we provide Fig. 6 to show that the final accuracy of the algorithm increases with smaller values of δ .

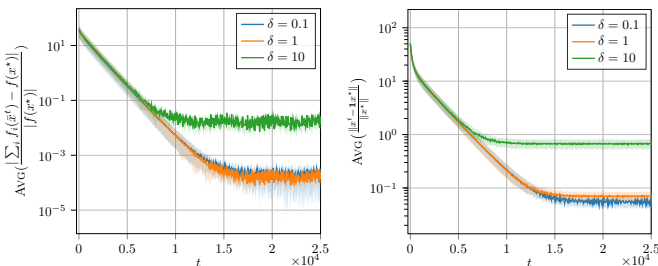


Fig. 6. Monte Carlo simulations for varying amplitude δ : mean and 1-standard deviation band of cost error (left) and variable error (right).

E. Monte Carlo simulations for comparisons

We now perform simulations with $n = 30$ and $N = 10$ to compare our method with the 1-Point Distributed Stochastic Gradient-Tracking Method (1P-DSGT) by [19]. We remark that, similarly to our scheme, when running 1P-DSGT each agent estimates the local gradient with one query of the objective function at each iteration. These results are depicted in Fig. 7 and have been obtained by running the considered distributed schemes with the same objective functions, communication graphs, and initial conditions of the solution estimates.

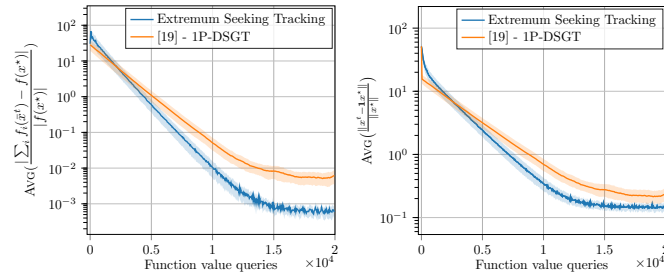


Fig. 7. Monte Carlo simulations for comparison between Extremum Seeking Tracking and 1P-DSGT in [19]: mean and 1-standard deviation band of cost error (left) and variable error (right).

In detail, Fig. 7 shows that, although the algorithm by [19] exhibits a faster convergence in the beginning phase of the simulations, our scheme has a better convergence rate and final accuracy. Both 1P-DSGT and Extremum Seeking Tracking employ a single function query per agent to approximate the gradient. These methods rely on the concept that averaging across iterations around the “quasi-static” local solution estimate x_i^t provides an accurate gradient approximation. However, in the context of their theoretical frameworks, 1P-DSGT attains this approximation using an infinite number of samples (mean of an ergodic process), whereas Extremum Seeking Tracking only needs τ_{per} samples. This makes the convergence rate of our algorithm faster.

F. Monte Carlo simulations in stochastic scenarios

Finally, we compare the considered distributed algorithms in a stochastic scenario in which each agent receives cost measurements affected by noise, i.e., in the case of $f_i(w_i^t) + \eta_i^t$ in place of $f_i(w_i^t)$ in Algorithm 1, where each component $\eta_i^t \in \mathbb{R}^n$ is randomly generated according to the Gaussian distribution with expected value 0 and standard deviation 0.1. Fig. 8 compares the behavior of Extremum Seeking Tracking and 1P-DSGT by [19] in the case in which $N = 30$ and $n = 10$. These plots confirm that Extremum Seeking Tracking exhibits faster convergence and greater final accuracy with respect to the considered scheme also in the considered stochastic setup. To interpret these results, we also remark that Theorem 1 ensures (semi-global, practical) stability properties for Extremum Seeking Tracking. Coherently, Fig. 8 shows the typical behavior exhibited by the trajectories of perturbed systems in the neighborhood of (practically) stable equilibria.

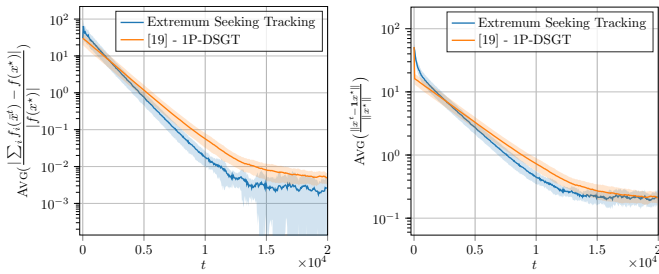


Fig. 8. Monte Carlo simulations in stochastic scenarios: mean and 1-standard deviation band of cost error (left) and variable error (right).

VI. CONCLUSIONS

In this paper, we addressed a distributed optimization problem in which the cost function is unknown and agents have only access to local measurements. Taking inspiration from a continuous gradient tracking algorithm, we proposed a novel gradient-free distributed optimization algorithm in which gradients are estimated via extremum seeking. We analyzed the convergence properties of the proposed algorithm by using Lyapunov and averaging tools from system theory. We corroborated the theoretical analysis through Monte Carlo simulations on personalized optimization problems.

APPENDIX

A. Proof of Lemma 1

Given $\alpha = \text{col}(\alpha_1, \dots, \alpha_n) \in \mathbb{N}^n$, $y = \text{col}(y_1, \dots, y_n) \in \mathbb{R}^n$, and a smooth function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, we define

$$\begin{aligned} \alpha! &:= \alpha_1! \dots \alpha_n!, & y^\alpha &:= y_1^{\alpha_1} \dots y_n^{\alpha_n}, \\ \partial^\alpha f(y) &:= \frac{\partial^{\alpha_1}}{\partial y_1^{\alpha_1}} \dots \frac{\partial^{\alpha_n}}{\partial y_n^{\alpha_n}} f(y), & |\alpha| &:= \alpha_1 + \dots + \alpha_n. \end{aligned}$$

Being each function f_i smooth (cf. Assumption 3), we can apply Taylor's expansion (cf. [47, Theorem 2]) and write

$$\begin{aligned} f_i(x_i + \delta d_i^t) &= f_i(x_i) + \delta d_i^{t\top} \nabla f_i(x_i) + \frac{\delta^2}{2} d_i^{t\top} \nabla^2 f_i(x_i) d_i^t \\ &\quad + \delta^3 R_{i,2}(x_i, \delta d_i^t), \end{aligned} \quad (45)$$

where the remainder $R_{i,2}(x_i, \delta d_i^t)$ is given by

$$R_{i,2}(x_i, \delta d_i^t) = \sum_{|\alpha|=3} \frac{\partial^\alpha f_i(x_i + c\delta d_i^t)}{\alpha!} (\delta d_i^t)^\alpha, \quad (46)$$

for some $c \in (0, 1)$. Then, we can use (45) to write

$$\begin{aligned} &\frac{2}{\delta \tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} d_i^k f_i(x_i + \delta d_i^k) \\ &= \frac{2f_i(x_i)}{\delta \tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} d_i^k + \left[\frac{2}{\tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} \left(d_i^k d_i^{k\top} \right) \right] \nabla f_i(x_i) \\ &\quad + \frac{\delta}{\tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} \left(d_i^k d_i^{k\top} \right) \nabla^2 f_i(x_i) d_i^k \\ &\quad + \frac{2}{\delta \tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} d_i^k R_{i,2}(x_i, \delta d_i^k). \end{aligned} \quad (47)$$

By combining (3) and (47), we get

$$\begin{aligned} \sum_{k=t+1}^{t+\tau_{\text{per}}} d_i^k &= 0, & \frac{2}{\tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} \left(d_i^k d_i^{k\top} \right) &= I_n \\ \sum_{k=t+1}^{t+\tau_{\text{per}}} \left(d_i^k d_i^{k\top} \right) \nabla^2 f_i(x_i) d_i^k &= 0, \end{aligned}$$

which combined with (47), allow us to write

$$f_i(x_i + \delta d_i^t) = \nabla f_i(x_i) + \frac{2}{\delta \tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} d_i^k R_{i,2}(x_i, \delta d_i^k).$$

The proof follows by setting $\ell_i(x_i) = \frac{2}{\tau_{\text{per}}} \sum_{k=t+1}^{t+\tau_{\text{per}}} d_i^k R_{i,2}(x_i, \delta d_i^k) / \delta^3$. Finally, given a compact set $\mathcal{S}_i \subset \mathbb{R}^n$, let us bound $\|\ell_i(x_i)\|$ for all $x_i \in \mathcal{S}_i$. Note that $\|\delta d_i^k\| \leq \delta \sqrt{n}$ for all $t \in \mathbb{N}$ and let $\mathcal{S}'_i \subset \mathbb{R}^n$ be a compact set such that (i) $\mathcal{S}_i \subseteq \mathcal{S}'_i \subset \mathbb{R}^n$, and (ii) $x_i + \delta d_i^k \in \mathcal{S}'_i$ for all $x_i \in \mathcal{S}_i$, $\delta \in (0, 1]$, and $t \in \mathbb{N}$. Thus, we can write

$$\begin{aligned} &\sup_{x_i \in \mathcal{S}'_i} \left\| \frac{R_{i,2}(x_i, \delta d_i^k)}{\delta^3} \right\| \\ &\stackrel{(a)}{\leq} \sup_{x_i \in \mathcal{S}'_i} \left\| \frac{1}{\delta^3} \sum_{|\alpha|=3} \frac{\partial^\alpha f_i(x_i)}{\alpha!} (\delta d_i^k)^\alpha \right\| \\ &\stackrel{(b)}{\leq} \sup_{x_i \in \mathcal{S}'_i} \left\| \sum_{|\alpha|=3} \frac{\partial^\alpha f_i(x_i)}{\alpha!} (d_i^k)^\alpha \right\| =: L'_{i,\mathcal{S}'_i}, \end{aligned} \quad (48)$$

where in (a) we use the expression (46) of $R_{i,2}(x_i, \delta d_i^k)$, the definition of \mathcal{S}'_i , and the fact that $\delta \in (0, 1]$, while in (b) we drop out the term δ^3 from $(\delta d_i^k)^\alpha$. We underline that, since the set \mathcal{S}'_i is compact and f_i is smooth, L'_{i,\mathcal{S}'_i} exists and is finite. The bound of $\ell_i(x_i)$ follows by defining $L_{i,\mathcal{S}_i} := 2L'_{i,\mathcal{S}'_i} \sqrt{n}$ and combining the result (48) with the bound about the norm of the dither signal, i.e., $\|d_i^k\| \leq \sqrt{n}$ for all $t \in \mathbb{N}$.

B. Proof of Lemma 2

In [38], it is provided a Lyapunov function proving that, under the Assumptions 1, 2, and 3, the point $\xi^* = (\mathbf{1}x^*, \bar{z}_\perp^{\text{eq}})$, with $\bar{z}_\perp^{\text{eq}} := -R^\top G(\mathbf{1}x^*)$, is a globally exponentially stable equilibrium for the continuous-time system

$$\dot{\xi}(t) = \phi_{\text{GT}}(\xi(t)).$$

In detail, [38, Th. 3.1] introduces a full-rank matrix $\bar{T} \in \mathbb{R}^{n_\xi \times n_\xi}$ to define $\bar{\xi} := \bar{T}\xi$, and a matrix $\bar{P} = \bar{P}^\top \in \mathbb{R}^{n_\xi \times n_\xi}$ such that

$$\bar{a}_1 I \leq \bar{P} \leq \bar{a}_2 I \quad (49a)$$

$$2\bar{\xi}_{\text{avg}}^\top \bar{P} \bar{T} \phi_{\text{GT}}(\bar{T}^{-1} \bar{\xi}_{\text{avg}}) \leq -\bar{a}_3 \|\bar{\xi}_{\text{avg}}\|^2, \quad (49b)$$

for all $\xi_{\text{avg}} \in \mathbb{R}^{n_\xi}$ and some $\bar{a}_1, \bar{a}_2, \bar{a}_3 > 0$. Let $P := \bar{T}^\top \bar{P} \bar{T}$. Then, there exist $a_1, a_2, a_3 > 0$ such that

$$a_1 I \leq P \leq a_2 I \quad (50a)$$

$$2\xi_{\text{avg}}^\top P \phi_{\text{GT}}(\xi_{\text{avg}}) \leq -a_3 \|\xi_{\text{avg}}\|^2, \quad (50b)$$

for all $\xi_{\text{avg}} \in \mathbb{R}^{n_\xi}$. Hence, we define the candidate Lyapunov function $V(\xi_{\text{avg}}) := \xi_{\text{avg}}^\top P \xi_{\text{avg}}$ and bound $\Delta V(\xi_{\text{avg}}^t) := V(\xi_{\text{avg}}^{t+1}) - V(\xi_{\text{avg}}^t)$ along the trajectories of (28) as

$$\Delta V(\xi_{\text{avg}}^t) \leq -\gamma a_3 \|\xi_{\text{avg}}^t\|^2 + \gamma^2 \phi_{\text{GT}}(\xi_{\text{avg}}^t)^\top P \phi_{\text{GT}}(\xi_{\text{avg}}^t). \quad (51)$$

Moreover, by using the Lipschitz continuity of the gradients of the objective functions (cf. Assumption 3) and the definition of ϕ_{GT} , there exists $a_4 > 0$ such that

$$\|\phi_{\text{GT}}(\xi_{\text{avg}}^t)\| \leq a_4 \|\xi_{\text{avg}}^t\|. \quad (52)$$

Finally, for any $c_1 \in (0, a_3)$, let $\gamma_0 := (a_3 - c_1)/(a_2 a_4^2)$ and the proof follows by using (51) and (52).

C. Proof of Lemma 3

The proof relies on (i) the matrix P satisfying (29), and (ii) the fact that the norm of the perturbation term $\gamma \delta B u(\xi_{\text{avg}}^t)$ can be arbitrarily reduced through the parameter δ as long as ξ_{avg}^t lies into a compact set. First of all, without loss of generality, we assume $\rho \leq r_\xi$. Indeed, we will use the parameter r_ξ to define a (compact) ball and arbitrarily bound the norm of the perturbation term $\gamma \delta B u(\xi_{\text{avg}}^t)$ through the parameters δ as long as ξ_{avg}^t lies into this ball. Hence, we can always use the more conservative condition. In detail, we introduce the candidate Lyapunov function $V(\xi_{\text{avg}}) := \xi_{\text{avg}}^\top P \xi_{\text{avg}}$ and the set $\Omega_{r_\xi} := \{\xi_{\text{avg}} \in \mathbb{R}^{n_\xi} \mid V(\xi_{\text{avg}}) \leq a_2 r_\xi^2\} \subset \mathbb{R}^{n_\xi}$. Then, from (29a), we derive $\mathcal{B}_{r_\xi} \subseteq \Omega_{r_\xi} \subseteq \mathcal{B}_{r'_{\xi_{\text{avg}}}}$, where $r'_{\xi_{\text{avg}}} := \sqrt{a_2/a_1} r_\xi$. Now, under the assumption $\xi_{\text{avg}}^t \in \mathcal{B}_{r_\xi}$ (later verified by a proper selection of the algorithm parameters), it holds $\xi_{\text{avg}}^t \in \Omega_{r_\xi}$. By using this property and since $\gamma \in (0, \gamma_0)$, we use (29b), the Cauchy-Schwarz inequality, and (52) to bound $\Delta V(\xi_{\text{avg}}^t) := V(\xi_{\text{avg}}^{t+1}) - V(\xi_{\text{avg}}^t)$ along the trajectories of (27) as

$$\begin{aligned} \Delta V(\xi_{\text{avg}}^t) &\leq -\gamma c_1 \|\xi_{\text{avg}}^t\|^2 + \gamma \delta^2 2 \|PB\| \|\xi_{\text{avg}}^t\| \|u(\xi_{\text{avg}}^t)\| \\ &\quad + \delta^2 \gamma^2 2 a_4 \|PB\| \|\xi_{\text{avg}}^t\| \|u(\xi_{\text{avg}}^t)\| \\ &\quad + \delta^4 \gamma^2 \|B^\top PB\| \|u(\xi_{\text{avg}}^t)\|^2. \end{aligned} \quad (53)$$

Now, for all $i \in \{1, \dots, N\}$, let us introduce the $\mathcal{S}_i := \{x_i \in \mathbb{R}^n \mid \|x_i - x^*\| \leq r'_{\xi_{\text{avg}}}\}$. Hence, we note that $\xi_{\text{avg}} := (\tilde{x}_{\text{avg}}, \tilde{z}_{\perp, \text{avg}}) \in \Omega_{r_\xi} \implies \tilde{x} \in \mathcal{S} \subset \mathbb{R}^{Nn}$, where $\mathcal{S} := \mathcal{S}_1 \times \dots \times \mathcal{S}_N$. Then, we apply result (23) to claim that, for all $i \in \{1, \dots, N\}$, it holds $\ell_i(x_i) \leq L_{i, \mathcal{S}_i}$ for all $x_i \in \mathcal{S}_i$. Thus, by defining $L_{\mathcal{S}} := \max_i \{L_{1, \mathcal{S}_1}, \dots, L_{N, \mathcal{S}_N}\}$ and using the definition $u(\xi_{\text{avg}}^t) = \ell(\tilde{x}^t + \mathbf{1}x^*)$, we get

$$\|u(\xi_{\text{avg}}^t)\| \leq \sqrt{N} L_{\mathcal{S}}. \quad (54)$$

Since $\gamma \in (0, 1]$ and $\delta \in (0, 1]$, we bound (53) as

$$\Delta V(\xi_{\text{avg}}^t) \leq -\gamma c_1 \|\xi_{\text{avg}}^t\|^2 + \gamma \delta^2 (b_1 \|\xi_{\text{avg}}^t\| + b_2), \quad (55)$$

where we introduced

$$\begin{aligned} b_1 &:= 2 \|PB\| \sqrt{N} L_{\Omega_{r_\xi}} + 2 a_4 \|PB\| \sqrt{N} L_{\mathcal{S}} \\ b_2 &:= \|B^\top PB\| N L_{\mathcal{S}}^2. \end{aligned}$$

Therefore, for any $\rho \in (0, r_\xi)$ and $c'_1 \in (0, c_1)$, we define

$$\delta_1^* := \min \left\{ \sqrt{(c_1 - c'_1) \rho^2 / (b_1 r'_{\xi_{\text{avg}}} + b_2)}, 1 \right\}. \quad (56)$$

Hence, by combining (55) and (56), if $\delta \in (0, \delta_1^*)$, then, for all $\xi_{\text{avg}}^t \in \Omega_{r'_{\xi_{\text{avg}}}}$ such that $\|\xi_{\text{avg}}^t\| \geq \rho$, it holds

$$\Delta V(\xi_{\text{avg}}^t) \leq -\gamma c'_1 \|\xi_{\text{avg}}^t\|^2. \quad (57)$$

Thus, the inequality (57) ensures that the set Ω_{r_ξ} is forward-invariant for system (27). Hence, if we pick $\xi_{\text{avg}}^0 \in \mathcal{B}_{r_\xi}$, we prove that $\xi_{\text{avg}}^t \in \Omega_{r_\xi}$ for all $t \in \mathbb{N}$. Consequently, the bound (54) holds for all $t \in \mathbb{N}$ and, in turn, also the inequality (57) is verified for all $t \in \mathbb{N}$, namely we proved that the trajectories of system (27) enter the ball \mathcal{B}_ρ exponentially fast. The result (30) follows from the inequality (57) and (29a) by setting $c_3 := c'_1 / (2a_2)$.

REFERENCES

- [1] A. Nedić and J. Liu, "Distributed optimization for control," *Ann. Rev. of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 77–103, 2018.
- [2] T. Yang, X. Yi, J. Wu, Y. Yuan, D. Wu, Z. Meng, Y. Hong, H. Wang, Z. Lin, and K. H. Johansson, "A survey of distributed optimization," *Annual Reviews in Control*, vol. 47, pp. 278–305, 2019.
- [3] G. Notarstefano, I. Notarnicola, A. Camisa, et al., "Distributed optimization for smart cyber-physical networks," *Foundations and Trends® in Systems and Control*, vol. 7, no. 3, pp. 253–383, 2019.
- [4] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to derivative-free optimization*. SIAM, 2009.
- [5] S. Liu, P.-Y. Chen, B. Kaikhura, G. Zhang, A. O. Hero III, and P. K. Varshney, "A primer on zeroth-order optimization in signal processing and machine learning: Principals, recent advances, and applications," *IEEE Signal Processing Magazine*, vol. 37, no. 5, pp. 43–54, 2020.
- [6] D. S. Zalkind, E. Dall'Anese, and L. Y. Pao, "Automatic controller tuning using a zeroth-order optimization algorithm," *Wind Energy Science Discussions*, vol. 2020, pp. 1–32, 2020.
- [7] A. I. Maass, C. Manzie, I. Shames, and H. Nakada, "Zeroth-order optimization on subsets of symmetric matrices with application to mpc tuning," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 4, pp. 1654–1667, 2021.
- [8] A. Menon and J. S. Baras, "Collaborative extremum seeking for welfare optimization," in *IEEE Conf. on Decision and Contr.*, pp. 346–351, 2014.
- [9] M. Ye and G. Hu, "Distributed extremum seeking for constrained networked optimization and its application to energy consumption control in smart grid," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 6, pp. 2048–2058, 2016.
- [10] M. Guay, I. Vandermeulen, S. Dougherty, and P. J. McLellan, "Distributed extremum-seeking control over networks of dynamically coupled unstable dynamic agents," *Automatica*, vol. 93, pp. 498–509, 2018.
- [11] S. Dougherty and M. Guay, "An extremum-seeking controller for distributed optimization over sensor networks," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 928–933, 2017.
- [12] Y. B. Salamah, L. Fiorentini, and U. Ozguner, "Cooperative extremum seeking control via sliding mode for distributed optimization," in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 1281–1286, 2018.
- [13] Z. Li, K. You, and S. Song, "Cooperative source seeking via networked multi-vehicle systems," *Automatica*, vol. 115, p. 108853, 2020.
- [14] M. Guay, "Distributed newton seeking," *Computers & Chemical Engineering*, vol. 146, p. 107206, 2021.
- [15] J. Poveda and N. Quijano, "Distributed extremum seeking for real-time resource allocation," in *American Control Conf.*, pp. 2772–2777, 2013.
- [16] J. Poveda, M. Benosman, and A. Teel, "Distributed extremum seeking in multi-agent systems with arbitrary switching graphs," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 735–740, 2017.
- [17] S. Michalowsky, B. Ghahesifard, and C. Ebenbauer, "Distributed extremum seeking over directed graphs," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 2095–2101, 2017.
- [18] D. Wang, M. Chen, and W. Wang, "Distributed extremum seeking for optimal resource allocation and its application to economic dispatch in smart grids," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 10, pp. 3161–3171, 2019.
- [19] E. Mhanna and M. Assaad, "Single point-based distributed zeroth-order optimization with a non-convex stochastic objective function," in *International Conference on Machine Learning*, pp. 24701–24719, PMLR, 2023.

[20] Y. Tang, J. Zhang, and N. Li, "Distributed zero-order algorithms for nonconvex multiagent optimization," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 1, pp. 269–281, 2020.

[21] J. Lu and C. Y. Tang, "Zero-gradient-sum algorithms for distributed convex optimization: The continuous-time case," *IEEE Transactions on Automatic Control*, vol. 57, no. 9, pp. 2348–2354, 2012.

[22] J. Liu and W. Chen, "Sample-based zero-gradient-sum distributed consensus optimization of multi-agent systems," in *Proc. of the 11th World Congress on Intelligent Control and Automation*, pp. 215–219, 2014.

[23] D. Yuan and D. W. C. Ho, "Randomized gradient-free method for multiagent optimization over time-varying networks," *IEEE Trans. on Neural Net. and Learning Systems*, vol. 26, no. 6, pp. 1342–1347, 2015.

[24] Y. Pang and G. Hu, "Randomized gradient-free distributed optimization methods for a multiagent system with unknown cost function," *IEEE Transactions on Automatic Control*, vol. 65, no. 1, pp. 333–340, 2019.

[25] X. Chen, C. Gao, M. Zhang, and Y. Qin, "Randomized gradient-free distributed algorithms through sequential gaussian smoothing," in *2017 36th Chinese Control Conference (CCC)*, pp. 8407–8412, 2017.

[26] J. Ding, D. Yuan, G. Jiang, and Y. Zhou, "Distributed quantized gradient-free algorithm for multi-agent convex optimization," in *29th Chinese Control And Decision Conference (CCDC)*, pp. 6431–6435, 2017.

[27] Y. Pang and G. Hu, "Exact convergence of gradient-free distributed optimization method in a multi-agent system," in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 5728–5733, 2018.

[28] C. Wang and X. Xie, "Design and analysis of distributed multi-agent saddle point algorithm based on gradient-free oracle," in *2018 Australian New Zealand Control Conference (ANZCC)*, pp. 362–365, 2018.

[29] L. Wang, Y. Wang, and Y. Hong, "Distributed online optimization with gradient-free design," in *Chinese Control Conf.*, pp. 5677–5682, 2019.

[30] O. Bilenne, P. Mertikopoulos, and E. V. Belmega, "Fast optimization with zeroth-order feedback in distributed, multi-user mimo systems," *IEEE Transactions on Signal Processing*, vol. 68, pp. 6085–6100, 2020.

[31] Y. Pang and G. Hu, "Randomized gradient-free distributed optimization methods for a multiagent system with unknown cost function," *IEEE Transactions on Automatic Control*, vol. 65, no. 1, pp. 333–340, 2020.

[32] A. K. Sahu and S. Kar, "Decentralized zeroth-order constrained stochastic optimization algorithms: Frank–wolfe and variants with applications to black-box adversarial attacks," *Proceedings of the IEEE*, vol. 108, no. 11, pp. 1890–1905, 2020.

[33] D. Wang, J. Zhou, Z. Wang, and W. Wang, "Random gradient-free optimization for multiagent systems with communication noises under a time-varying weight balanced digraph," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 1, pp. 281–289, 2020.

[34] I. Vandermeulen, M. Guay, and P. J. McLellan, "Discrete-time distributed extremum-seeking control over networks with unstable dynamics," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1182–1192, 2018.

[35] K. Kvaternik and L. Pavel, "An analytic framework for decentralized extremum seeking control," in *2012 American Control Conference (ACC)*, pp. 3371–3376, IEEE, 2012.

[36] W. Li and M. Assaad, "Distributed stochastic optimization in networks with low informational exchange," *IEEE Transactions on Information Theory*, vol. 67, no. 5, pp. 2989–3008, 2021.

[37] Y. Tan, D. Nešić, and I. Mareels, "On the choice of dither in extremum seeking systems: A case study," *Automatica*, vol. 44, no. 5, pp. 1446–1450, 2008.

[38] G. Carnevale, I. Notarnicola, L. Marconi, and G. Notarstefano, "Triggered gradient tracking for asynchronous distributed optimization," *Automatica*, vol. 147, p. 110726, 2023.

[39] E.-W. Bai, L.-C. Fu, and S. S. Sastry, "Averaging analysis for discrete time and sampled data adaptive systems," *IEEE Transactions on Circuits and Systems*, vol. 35, no. 2, pp. 137–148, 1988.

[40] H. K. Khalil and J. W. Grizzle, *Nonlinear systems*, vol. 3. Prentice hall Upper Saddle River, NJ, 2002.

[41] J. A. Sanders, F. Verhulst, and J. Murdock, *Averaging methods in nonlinear dynamical systems*, vol. 59. Springer, 2007.

[42] J. Popenda, "On the discrete analogy of Gronwall lemma," *Demonstratio Mathematica*, vol. 16, no. 1, pp. 11–26, 1983.

[43] J. M. Holte, "Discrete Gronwall lemma and applications," in *MAA-NCS meeting at the University of North Dakota*, vol. 24, pp. 1–7, 2009.

[44] A. M. Ospina, A. Simonetto, and E. Dall'Anese, "Time-varying optimization of networked systems with human preferences," *IEEE Trans. on Control of Network Systems*, vol. 10, no. 1, pp. 503–515, 2022.

[45] P. Erdős, A. Rényi, et al., "On the evolution of random graphs," *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*, vol. 5, no. 1, pp. 17–60, 1960.

[46] F. Farina, A. Camisa, A. Testa, I. Notarnicola, and G. Notarstefano, "Disropt: a python framework for distributed optimization," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 2666–2671, 2020.

[47] G. Folland, "Higher-order derivatives and Taylor's formula in several variables," *Preprint*, pp. 1–4, 2005.



Nicola Mimmo received the Ph.D. in automation from University of Bologna, in 2015. Since 2010, he has been working with the Department of Electrical, Electronic, and Information Engineering "Guglielmo Marconi", University of Bologna, on national and European projects for the development of unmanned aerial vehicles for civil applications with focus on search and rescue scenarios. He worked with the major national and European aircraft companies for the development of both flight systems and flight control laws that are currently protected by international patents. He is co-author of tens of scientific papers. His research interests range from fault-tolerant controls to nonlinear control systems in aerospace.



Guido Carnevale is a Junior Assistant Professor in the Department of Electrical, Electronic, and Information Engineering G. Marconi at Alma Mater Studiorum Università di Bologna. He received the M.Sc. degree "summa cum laude" in Automation Engineering from the University of Bologna, in 2019, and the Ph.D. degree "summa cum laude" in "Biomedical, Electrical, and Systems Engineering" from the same university. He was a visiting scholar at the University of Oxford in 2022. His research interests include distributed optimization and games over networks, robotics, and optimal control.



Andrea Testa received the Laurea degree "summa cum laude" in Computer Engineering from the Università del Salento, Lecce, Italy in 2016 and the Ph.D. degree in Engineering of Complex Systems from the same university in 2020.

He is a Junior Assistant Professor in the Department of Electrical, Electronic, and Information Engineering G. Marconi at Alma Mater Studiorum Università di Bologna, Bologna, Italy. He was a visiting scholar at LAAS-CNRS, Toulouse, (July to September 2015 and February 2016) and at Alma Mater Studiorum Università di Bologna (October 2018 to June 2019). His research interests include control of UAVs and distributed optimization. Dr. Andrea Testa serves as an Associate Editor for the *IEEE International Conference on Automation Science and Engineering (CASE)*.



Giuseppe Notarstefano is a Professor in the Department of Electrical, Electronic, and Information Engineering at University of Bologna. Previously, he was Associate Professor at the Università del Salento, Lecce, Italy. He received the Laurea degree "summa cum laude" in Electronics Engineering from the Università di Pisa in 2003 and the PhD degree in Automation and Operation Research from the Università di Padova in 2007. He has been visiting scholar at the University of Stuttgart, University of California Santa

Barbara, and University of Colorado Boulder. 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. He has served as an Associate Editor for *IEEE Transactions on Automatic Control*, *IEEE Transactions on Control Systems Technology*, and *IEEE Control Systems Letters*. He has been also part of the Conference Editorial Board of IEEE Control Systems Society and EUCA. He was recipient of an ERC Starting Grant.