A Distributed Feedback-based Framework for Nonlinear Aggregative Optimal Control

Published Version:
Sforni, L., Carnevale, G., Notarstefano, G. (2025). A Distributed Feedback-based Framework for Nonlinear Aggregative Optimal Control. IEEE TRANSACTIONS ON AUTOMATIC CONTROL, 70(6), 3784-3799 [10.1109/TAC.2024.3518256].

Availability:
This version is available at: https://hdl.handle.net/11585/999286 since: 2024-12-19

Published:
DOI: http://doi.org/10.1109/TAC.2024.3518256

A Distributed Feedback-based Framework for Nonlinear Aggregative Optimal Control

Lorenzo Sforni *Student Member, IEEE*, Guido Carnevale *Member, IEEE*, and Giuseppe Notarstefano *Member, IEEE*

***Abstract*— In this paper, we propose a distributed, first-order, feedback-based approach to solve nonlinear optimal control problems with aggregative cost functions over networks of cooperative multi-agent systems. Taking inspiration from a centralized, first-order optimal control framework, named GoPRONTO, we propose a distributed method exploiting a feedback scheme iteratively updated according to a distributed tracking mechanism. Due to the aggregative structure of the problem and the desired distributed paradigm, the centralized scheme would require global quantities that are not locally available. Thus, our distributed method concurrently updates a proxy of the centralized scheme with a set of local, auxiliary variables named trackers which suitably exploit inter-agent communication to reconstruct the global quantities. By relying on LaSalle-based arguments, we theoretically prove that our algorithm generates a sequence of trajectories converging to the set of trajectories satisfying the first-order necessary conditions for optimality. Finally, we corroborate the theoretical results with numerical simulations on a distributed optimal control application for a fleet of 50 quadrotors.**

***Index Terms*— Optimal control, distributed optimization, multi-agent systems, optimization algorithms.**

I. Introduction

Optimal control problems for multi-agent systems are emerging in a variety of automation and robotics applications, see the surveys [1], [2] for an overview. Due to the network dimension (and thus the problem one) and to robustness issues, "centralized" approaches where a single unity knows all data and controls all the agents are often undesirable or even not applicable. For these reasons, decentralized and distributed solutions are becoming more and more appealing. In our paper, we consider distributed methods meant as algorithms involving cooperation via local communication (over a network) among control nodes in order to accomplish a common task.

Literature Review: Existing centralized numerical methods for the resolution of optimal control problems are typically classified as *indirect* methods (see the recent works [3], [4]) and *direct* ones (see [5] for an overview). The distributed versions of these centralized strategies to the resolution of multi-agent optimal control problem is a current area of research. In [6], distributed optimal control problems are addressed in the context of multiscale dynamical systems, where microscopic control laws at the agent level are derived from the optimal macroscopic description of the system. Distributed control schemes based on an Alternating Direction Method of Multipliers (ADMM) approach are proposed in [7]–[11] with applications in multi-robot systems. An extension of Differential Dynamic Programming (DDP) to a multi-agent setting is proposed in [12], [13], where some game theoretic DDP-based methods are presented. Distributed optimal control algorithms represent a building block for *cooperative distributed* Model Predictive Control (MPC) to plan feasibile trajectories. The survey paper [14] provides an overview of distributed (economic) MPC approaches, however in several works a distributed algorithm for the solution of the optimal control problem is assumed. From an algorithmic perspective, in [15] an accelerated gradient method is proposed for distributed cooperative MPC, while in [16] an augmented Lagrangian method is proposed in the context of convex MPC. Augmented Lagrangian and ADMM-based distributed optimization and optimal control algorithms gained traction in both cost- and constraint-coupled scenarios [17], [18]. For constraint coupled problems, in [19], a distributed MPC scheme, GRAMPC, is developed based on the ADMM approach. A distributed augmented Lagrangian algorithm is proposed in [20], which may be extended to optimal control applications. Notice that, a coordination step is required to ensure consensus among the dual variables. In [21], a decentralized Sequential Quadratic Programming approach, based on ADMM methods, tailored for nonlinear (constraint-coupled) problems. The implementation of real-time distributed MPC schemes on robotic platforms is studied in [22], for constraint-coupled optimal control problems. While these works address distributed optimal control and optimization scenarios, they are not specifically tailored to address optimization problems with the aggregative structure considered in this paper. Indeed, algorithms specifically designed for aggregative optimization scenarios are particularly compelling because (i) they allow each agent to store a solution estimate and instrumental quantities not depending on the agents' number itself, (ii) consensus is enforced only on low-dimensional instrumental quantities, and (iii) for all iterations, the agents do not know each other's solution estimate. On the contrary, algorithms designed for

This research was partially supported by the projects OPT4SMART n. 638992 funded by the European Research Council and "Distributed Optimization for Cooperative Machine Learning in Complex Networks" n. PGR10067 funded by Ministero degli Affari Esteri e della Cooperazione Internazionale. L. Sforni, G. Carnevale, and G. Notarstefano are with the Department of Electrical, Electronic and Information Engineering, Alma Mater Studiorum - Università di Bologna, Bologna, Italy ({lorenzo.sforni, guido.carnevale, giuseppe.notarstefano}@unibo.it)

cost-coupled problems, while being capable of addressing optimization problems falling within the aggregative optimization framework, would need to store, share, and reach consensus on a decision variable consisting of the stack of all agents' solutions. These advantages make the aggregative setting very versatile, especially for robotic applications.

The proposed optimization setting takes inspiration from the recently emerged distributed aggregative optimization framework introduced in [23] for a static, unconstrained case.

In this context, differently from the aggregative games [24]–[28], the network agents cooperate (rather than compete with each other) to minimize the sum of local functions that depend both on a local decision variable and an aggregation of them. An online version of the problem is addressed in [29], [30], while in [31] a learning-based distributed algorithm is proposed to deal with partially unknown local objective functions. In [32], [33], the aggregative framework is investigated in a continuous-time setup. Distributed algorithms based on a Franke-Wolfe update [34] and on ADMM [35] are proposed for a static, constrained setup. In [36], the setup with finite bits for communication among the agents is addressed. We remark that, differently from the above works, our paper aims to extend the aggregative optimization framework to the optimal control scenario, hence addressing the task of generating a trajectory for a nonlinear dynamical system while optimizing a cost functional involving an aggregate variable.

Our distributed optimal control approach is inspired by the (continuous-time) centralized approach in [37], named *PRojection Operator Newton Method for Trajectory Optimization (PRONTO)*. The key idea of PRONTO is the exploitation of a feedback mechanism transforming the dynamically constrained optimization problem into an unconstrained one addressed through a Newton's update. PRONTO has been extended to constrained problems [38] and Lie groups [39], and it has been applied to several contexts as motion planning of single and multiple vehicles, see, e.g., [40] and references therein. In [41], a first-order, feedback-based framework, named GoPRONTO, has been proposed for nonlinear discrete-time optimal control.

Contributions: The main contribution of the work is the development of a distributed scheme to address a novel class of nonlinear optimal control problems over networks of multi-agent systems. Such a class of problems is formulated by adopting the problem structure introduced in the context of aggregative optimization. The proposed approach addresses nonlinear aggregative optimization problems within an optimal control framework according to a peer-to-peer paradigm that does not require any centralized unit accessing global data and making global decisions. Inspired by the centralized GoPRONTO strategy, our multi-agent method uses, in each agent, a local feedback mechanism updated according to a distributed optimization-oriented scheme. Such an optimization scheme, due to the aggregative terms of the local cost functions, would require local knowledge of unavailable global quantities. Thus, we resort to a set of local, auxiliary variables named trackers that asymptotically reconstruct these global quantities through suitable consensus-based dynamics relying on inter-agent communication. Interlacing this mechanism with the optimization-oriented part of the algorithm gives rise to a complex interconnected system. In order to analyze its evolution, we exploit the adopted feedback-based algorithmic structure which allows for reformulating the multi-agent optimal control problem as an (unconstrained) nonconvex optimization problem. Moreover, the feedback-based nature ensures stability and, thus, allows for analyzing the overall distributed strategy with tools from system theory. In detail, we exploit LaSalle-based arguments to prove that the sequence of generated trajectories converges to the set of state-input trajectories satisfying the first-order necessary conditions for optimality of the considered optimal control problem. Finally, we test the proposed algorithm on a distributed optimal control example using a team of 50 mobile aerial robots for a surveillance task.

Organization: This paper is organized as follows. In Section II, we state the problem setup and provide some preliminaries about the single-agent case. In Section III, we present our novel distributed algorithm and provide its convergence guarantees. In Section IV, we analyze the proposed methodology. Finally, in Section V, we provide some numerical simulations in a cooperative multi-robot framework.

Notation: Given N vectors $x_1, \dots, x_N$, their vertical stack is denoted by $\text{col}(x_1, \dots, x_N)$. Given a function $\ell : \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_d} \to \mathbb{R}$, we denote its (total) gradient at a given point $(\bar{x}_1, \dots, \bar{x}_d)$ as $\nabla \ell(\bar{x}_1, \dots, \bar{x}_d) := \text{col}(\nabla_1 \ell(\bar{x}_1, \dots, \bar{x}_d), \dots, \nabla_d \ell(\bar{x}_1, \dots, \bar{x}_d))$, where the partial gradient of ℓ with respect to its k-th argument, for all $k = 1, \dots, d$ is denoted as $\nabla_k \ell(\bar{x}_1, \dots, \bar{x}_d) \in \mathbb{R}^{n_k}$. Moreover, given a vector field $f : \mathbb{R}^n \to \mathbb{R}^m$, the gradient of f is $\nabla f(x) := [\nabla f_1(x) \dots \nabla f_m(x)] \in \mathbb{R}^{n \times m}$. For a given $T \in \mathbb{N}$, we denote the discrete-time horizon as $[0, T] := \{0, 1, 2, \dots, T\}$. Given $x \in \mathbb{R}^n$ and $X \subseteq \mathbb{R}^n$, we define $\text{dist}(x, X) := \inf_{y \in X} \|x - y\|$. We denote with $\text{diag}(v_1, \dots, v_n)$ the diagonal matrix whose i-th diagonal element is given by v_i. 1_N denotes the N-dimensional vector with 1 in each entry. We denote with $\text{blkdiag}(M_1, \dots, M_N)$ the block diagonal matrix whose i-th block is $M_i \in \mathbb{R}^{n_i \times n_i}$. The symbol $\otimes$ denotes the Kronecker product. A function $\ell : \mathbb{R}^n \to \mathbb{R}$ is said to be radially unbounded if $\forall c > 0$ $\exists r > 0$ such that, $\forall x \in \mathbb{R}^n$, $\|x\| > r \Rightarrow \ell(x) > c$.

II. Problem Setup and Preliminaries

In this paper, we introduce the aggregative optimization framework in the context of distributed optimal control.

We consider a network of N (possibly) heterogeneous agents evolving according to the nonlinear dynamics

$$x_{i,t+1} = f_i(x_{i,t}, u_{i,t}), \tag{1}$$

for all $i = 1, \dots, N$, where $f_i : \mathbb{R}^{n_i} \times \mathbb{R}^{m_i} \to \mathbb{R}^{m_i}$, $x_{i,t} \in \mathbb{R}^{n_i}$ is the state of agent i at time t, and $u_{i,t} \in \mathbb{R}^{m_i}$ is the control input applied to agent i at time t. The stacks of the states and inputs of all the agents at time t are denoted as

$$x_t := \text{col}(x_{1,t}, \dots, x_{N,t}) \in \mathbb{R}^n \quad u_t := \text{col}(u_{1,t}, \dots, u_{N,t}) \in \mathbb{R}^m,$$

where $n = \sum_{i=1}^N n_i$ and $m = \sum_{i=1}^N m_i$. Moreover, considering a time horizon of length $T \in \mathbb{N}$, for each agent $i = 1, \dots, N$, we define as $\mathbf{x}_i \in \mathbb{R}^{n_i T}$ and $\mathbf{u}_i \in \mathbb{R}^{m_i T}$ the stack of the states $x_{i,1},\dots,x_{i,T}$ and inputs $u_{i,0},\dots,u_{i,T-1}$, respectively. For notational convenience, we denote the stack of all $\mathbf{x}_i$, $\mathbf{u}_i$ as

$$\mathbf{x} := \text{col}(\mathbf{x}_1,\dots,\mathbf{x}_N) \in \mathbb{R}^{nT} \quad \mathbf{u} := \text{col}(\mathbf{u}_1,\dots,\mathbf{u}_N) \in \mathbb{R}^{mT},$$

Then, we introduce the *aggregative* variables $\sigma_t : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^q$, for all $t \in [0, T-1]$, and $\sigma_T : \mathbb{R}^n \to \mathbb{R}^q$ that couple the states and inputs of all the agents according to

$$\sigma_t(x_t, u_t) = \frac{1}{N}\sum_{i=1}^{N} \varphi_{i,t}(x_{i,t}, u_{i,t}) \tag{2a}$$

$$\sigma_T(x_T) = \frac{1}{N}\sum_{i=1}^{N} \varphi_{i,T}(x_{i,T}), \tag{2b}$$

where, for all and $i = 1,\dots,N$, the functions $\varphi_{i,t} : \mathbb{R}^{n_i} \times \mathbb{R}^{m_i} \to \mathbb{R}^q$, for all $t \in [0, T-1]$, and $\varphi_{i,T} : \mathbb{R}^{n_i} \to \mathbb{R}^q$ represent the i-th contribution to the aggregative variable. We will use the symbol $\boldsymbol{\sigma}(\mathbf{x},\mathbf{u}) \in \mathbb{R}^{q(T+1)}$ to denote the stack of all $\sigma_t(\cdot,\cdot)$, for all $t \in [0, T-1]$, and $\sigma_T(\cdot)$, i.e., we define

$$\boldsymbol{\sigma}(\mathbf{x},\mathbf{u}) := \begin{bmatrix} \sigma_0(x_0,u_0) \\ \vdots \\ \sigma_{T-1}(x_{T-1},u_{T-1}) \\ \sigma_T(x_T) \end{bmatrix} = \frac{1}{N}\sum_{i=1}^{N} \varphi_i(\mathbf{x}_i,\mathbf{u}_i),$$

where $\varphi_i(\mathbf{x}_i,\mathbf{u}_i) := \text{col}(\varphi_{i,0}(x_{i,0},u_{i,0}),\dots,\varphi_{i,T}(x_{i,T}))$.

The agents of the network aim to cooperatively minimize the overall cost $\ell : \mathbb{R}^{nT} \times \mathbb{R}^{mT} \times \mathbb{R}^{q(T+1)} \to \mathbb{R}$ given by

$$\ell(\mathbf{x},\mathbf{u},\boldsymbol{\sigma}(\mathbf{x},\mathbf{u})) := \sum_{i=1}^{N} \ell_i(\mathbf{x}_i,\mathbf{u}_i,\boldsymbol{\sigma}(\mathbf{x},\mathbf{u})), \tag{3}$$

in which, for all $i = 1,\dots,N$, the local function $\ell_i : \mathbb{R}^{nT} \times \mathbb{R}^{mT} \times \mathbb{R}^{q(T+1)} \to \mathbb{R}$ of agent i depends on both the local state and input and the aggregative variable $\boldsymbol{\sigma}$ coupling the states and the inputs of all the agents. Specifically, ℓ_i reads as

$$\begin{aligned}\ell_i(\mathbf{x}_i,\mathbf{u}_i,\boldsymbol{\sigma}(\mathbf{x},\mathbf{u})) := \sum_{t=0}^{T-1} &\ell_{i,t}(x_{i,t},u_{i,t},\sigma_t(x_t,u_t)) \\ &+ \ell_{i,T}(x_{i,T},\sigma_T(x_T)),\end{aligned} \tag{4}$$

where $\ell_{i,t} : \mathbb{R}^{n_i} \times \mathbb{R}^{m_i} \times \mathbb{R}^q \to \mathbb{R}$ represents the so-called stage cost, while $\ell_{i,T} : \mathbb{R}^{n_i} \times \mathbb{R}^q \to \mathbb{R}$ is the terminal cost.

Overall, our goal is to design a system trajectory $(\mathbf{x},\mathbf{u})$ such that the global performance index (3) is minimized. More formally, we aim at solving the optimal control problem

$$\begin{aligned}\min_{\substack{\mathbf{x}_1,\dots,\mathbf{x}_N \\ \mathbf{u}_1,\dots,\mathbf{u}_N}} \; \sum_{i=1}^{N}\sum_{t=0}^{T-1} &\ell_{i,t}(x_{i,t},u_{i,t},\sigma_t(x_t,u_t)) \\ &+ \ell_{i,T}(x_{i,T},\sigma_T(x_T))\end{aligned} \tag{5a}$$

$$\text{subj. to} \;\; x_{i,t+1} = f_i(x_{i,t},u_{i,t}) \quad t = 0,\dots,T-1 \tag{5b}$$

$$x_{i,0} = x_{i,\text{init}} \quad i = 1,\dots,N, \tag{5c}$$

where $x_{i,\text{init}} \in \mathbb{R}^{n_i}$ represents the initial condition of agent i, for all $i = 1,\dots,N$.

Remark 1: For the sake of exposition, the algorithm presented in the paper is tailored to unconstrained optimal control problems. However, state and input constraints can be addressed by properly initializing the algorithm and adopting smooth barrier function strategies, as in, e.g., [38]. □

Remark 2: The recent survey [42] offers an overview of popular tasks arising in cooperative robotics (e.g., surveillance and resource allocation) that can be cast in the framework of distributed aggregative optimization. Moreover, applications such as smart grids management, economic market analysis, electric vehicles charging, and network congestion control are modeled in the literature as aggregative games [43]–[45], i.e., a network of players that compete with each other to minimize individual costs depending on both local quantities and an aggregation of all of them. Therefore, problem (5) allows for reformulating these tasks in cooperative scenarios. Remark also that, differently from problem (5), references [42]–[45] deal with dynamic systems (e.g., robots or energy generators) without optimizing or considering their low-level control. □

In this work, we consider an instance of problem (5) in which the dynamics, the cost function, and the aggregation functions satisfy the following assumptions.

Assumption 1 (Dynamics and cost regularity): For all $i = 1,\dots,N$, the dynamics function $f_i(x_{i,t},u_{i,t})$ and the cost functions $\ell_{i,t}(\cdot,\cdot,\cdot)$, $\ell_{i,T}(\cdot,\cdot)$, and $\varphi_i(\cdot,\cdot)$ are $\mathcal{C}^2$. Moreover, the cost function $\ell(\cdot,\cdot,\boldsymbol{\sigma}(\cdot,\cdot))$ is radially unbounded. □

In this work, we propose a distributed approach for a network of N agents to solve problem (5) cooperatively. To this end, we assume the agents exchange information among each other according to a directed graph $\mathcal{G} = (\{1,\dots,N\},\mathcal{E})$, where $\mathcal{E} \subset \{1,\dots,N\} \times \{1,\dots,N\}$ is the so-called edge set. Agent i receives information from agent j only if the edge $(j,i) \in \mathcal{E}$. The set of in-neighbors of i is $N_i := \{j \in \{1,\dots,N\} \mid (j,i) \in \mathcal{E}\}$. We associate to the graph $\mathcal{G}$ a weighted adjacency matrix $\mathcal{W} \in \mathbb{R}^{N\times N}$ whose entries match the graph, i.e., $w_{ij} > 0$ whenever $(j,i) \in \mathcal{E}$ and $w_{ij} = 0$ otherwise. Let us introduce the following assumption about the considered time-invariant network topology.

Assumption 2 (Network): The directed graph $\mathcal{G}$ is strongly connected and the matrix $\mathcal{W}$ is doubly stochastic. □

The aggregative optimization setup with time-varying graphs is considered in [29], which could be inspiring also for the optimal control scenario.

Remark 3: We consider decoupled dynamics, since it typically arise in a variety of robotics applications, where each robot has its own dynamics. The recent work [46] considers the aggregative setting in the presence of (linear) coupling constraints and, thus, offers a starting point for the extension of our algorithm to scenarios involving coupled dynamics. □

A. Preliminaries on Single Agent Optimal Control

Consider the single agent optimal control problem

$$\min_{\mathbf{x},\mathbf{u}} \; \sum_{t=0}^{T-1} \ell_t(x_t,u_t) + \ell_T(x_T) \tag{6a}$$

$$\text{subj. to} \;\; x_{t+1} = f(x_t,u_t) \qquad t = 0,\dots,T-1 \tag{6b}$$

$$x_0 = x_{\text{init}}.$$

Our goal is to select the input sequence $\mathbf{u}$ in order to minimize the performance index in (6a). Following the strategy proposed in [41], we adopt a feedback-based approach inspired by [37]. The underlying idea is to design a feedback policy mapping generic elements $(\boldsymbol{\alpha}, \boldsymbol{\mu})$ of the space $\mathbb{R}^{nT} \times \mathbb{R}^{mT}$, the so-called *curves*, into the space of trajectories feasible for the dynamics (6b). Specifically, we define as trajectory any state-input sequence, denoted as $(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^{nT} \times \mathbb{R}^{mT}$, which satisfies the nonlinear dynamics (6b) with initial condition $x_0 = x_{\text{init}}$. Among the different possibilities, the feedback policy is implemented, for all $t = 1, \ldots, T-1$, via the nonlinear tracking system

$$u_t = \mu_t + \kappa_t(\alpha_t - x_t), \qquad x_{t+1} = f(x_t, u_t), \tag{7}$$

where $\kappa_t : \mathbb{R}^n \to \mathbb{R}^m$ is a suitable feedback policy exhibiting numerical stability properties. That is, it would be an asymptotically stabilizing feedback if extended to an infinite horizon. The closed-loop policy (7) can be embedded as a redundant constraint in problem (6), thus resulting into the following optimal control problem

$$\begin{aligned}
\min_{\mathbf{x},\mathbf{u},\boldsymbol{\alpha},\boldsymbol{\mu}} \; & \sum_{t=0}^{T-1} \ell_t(x_t, u_t) + \ell_T(x_T) && \text{(8a)}\\
\text{subj. to} \; & x_{t+1} = f(x_t, u_t) \qquad t = 0, \ldots, T-1 && \text{(8b)}\\
& u_t = \mu_t + \kappa_t(\alpha_t - x_t), && \text{(8c)}\\
& x_0 = x_{\text{init}}.
\end{aligned}$$

Any solution of (8) represents a unique solution of (6).

In [41], a novel feedback-based, gradient-related optimal control framework is proposed to iteratively solve (8) and, thus, (6). Let $k \in \mathbb{N}$ denote the iteration index and $(\mathbf{x}^k, \mathbf{u}^k)$ denote the estimate at iteration k of a solution to problem (6). Then, the proposed algorithmic scheme provides a convergent sequence of (feasible) trajectories $(\mathbf{x}^k, \mathbf{u}^k)$ where $\mathbf{x}^k := \text{col}(x_1^k, \ldots, x_T^k)$ and $\mathbf{u}^k := \text{col}(u_0^k, \ldots, u_{T-1}^k)$. This first-order strategy is summarized in Algorithm 1, where the following compact notation is adopted

$$\begin{aligned}
a_t^k &:= \nabla_1 \ell_t(x_t^k, u_t^k), & b_t^k &:= \nabla_2 \ell_t(x_t^k, u_t^k), && \text{(9a)}\\
A_t^k &:= \nabla_1 f(x_t^k, u_t^k)^\top, & B_t^k &:= \nabla_2 f(x_t^k, u_t^k)^\top, && \text{(9b)}\\
K_t^k &:= \nabla \kappa(e_t^k)^\top, & & \text{where } e_t^k := \alpha_t^k - x_t^k, && \text{(9c)}
\end{aligned}$$

for all $t \in [0, T-1]$ and $k \in \mathbb{N}$.

In [41], it is shown that Algorithm 1 implements a numerically robust gradient descent method to solve problem (8) in the space of the curves by exploiting the beneficial effect of the feedback operator. This algorithm, named GoPRONTO, leverages a gradient method to solve an unconstrained reformulation of problem (8) in the space of curves $(\boldsymbol{\alpha}, \boldsymbol{\mu})$. Specifically, at each iteration $k \in \mathbb{N}$, a descent direction is provided by computing $\Delta\alpha_t^k$ and $\Delta\mu_t^k$ through a backward integration of the closed-loop dynamics (10). This adjoint update can be seen as a closed-loop (robustified) version of the equation appearing in adjoint sensitivity methods. Then, in (11), such a direction is used to update the current curve. Finally, the new (updated) trajectory is obtained by applying the (feedback) policy onto the updated curve (12). It can be shown, then, that any limit point of Algorithm 1 is a stationary point of problem (6). Notice that, although GoPRONTO involves a forward-backward sweep typical of many Dynamic Programming-based approaches, it differs in the sense that (i) it does not seek to compute a value function and (ii) it retains the flexibility of the feedback gain design in (7). Moreover, the problem reformulation in (7) allows also for a perturbation of the state sequence $\boldsymbol{\alpha}$ to further improve the cost (cf. (11)).

Algorithm 1 GoPRONTO

Require: initial trajectory $(\mathbf{x}^0, \mathbf{u}^0)$ with $x_0^0 = x_{\text{init}}$
  for $k = 0, 1, 2 \ldots$ **do**
    set $\lambda_T^k = \nabla \ell_T^k(x_T^k)$
    for $t = T-1, \ldots, 0$ **do**
      Compute descent direction

$$\begin{aligned}
\lambda_t &= (A_t^k - B_t^k K_t^k)^\top \lambda_{t+1} + a_t^k - K_t^{k,\top} b_t^k && \text{(10a)}\\
\Delta\mu_t^k &= -b_t^k - B_t^{k,\top} \lambda_t && \text{(10b)}\\
\Delta\alpha_t^k &= K_t^{k,\top} \Delta\mu_t^k && \text{(10c)}
\end{aligned}$$

    end for
    for $t = 0, \ldots, T-1$ **do**
      Update the state-input curve

$$\begin{aligned}
\alpha_t^{k+1} &= \alpha_t^k + \gamma \Delta\alpha_t^k\\
\mu_t^{k+1} &= \mu_t^k + \gamma \Delta\mu_t^k
\end{aligned} \tag{11}$$

      Compute new (feasible) trajectory with $x_0^{k+1} = x_{\text{init}}$

$$\begin{aligned}
u_t^{k+1} &= \mu_t^{k+1} + \kappa_t(\alpha_t^{k+1} - x_t^{k+1})\\
x_{t+1}^{k+1} &= f(x_t^{k+1}, u_t^{k+1})
\end{aligned} \tag{12}$$

    end for
  end for

III. Distributed Aggregative Optimal Control

In this section, we present the main contribution of this paper, i.e., a distributed optimal control algorithm tailored for aggregative optimal control scenarios.

Inspired by the single agent case, we design, for each agent i, a (local) feedback policy mapping (local) state-input curves into trajectories for the i-th system. Operatively, such a policy is implemented via the nonlinear tracking system

$$u_{i,t} = \mu_{i,t} + \kappa_{i,t}(\alpha_{i,t} - x_{i,t}), \quad x_{i,t+1} = f_i(x_{i,t}, u_{i,t}), \tag{13}$$

where the feedback policy $\kappa_{i,t} : \mathbb{R}^{n_i} \to \mathbb{R}^{m_i}$ depends only on the i-th agent's dynamics and does not involve the state of other agents. A local approximation of the policy about a given trajectory $(\mathbf{x}_i, \mathbf{u}_i)$ can be designed, e.g., via the resolution of a linear quadratic regulator problem about such trajectory.

The introduction of the feedback operator in (13) leads to the closed-loop formulation of (14) described by

$$\begin{aligned}
\min_{\substack{\mathbf{x}_i, \mathbf{u}_i, \boldsymbol{\alpha}_i, \boldsymbol{\mu}_i\\ i=1,\ldots,N}} \; & \sum_{i=1}^{N} \sum_{t=0}^{T-1} \ell_{i,t}(x_{i,t}, u_{i,t}, \sigma_t(x_t, u_t)) \\
& \qquad + \ell_{i,T}(x_{i,T}, \sigma_T(x_T)) && \text{(14a)}\\
\text{subj. to} \; & x_{i,t+1} = f_i(x_{i,t}, u_{i,t}) \quad t = 0, \ldots, T-1 && \text{(14b)}\\
& u_{i,t} = \mu_{i,t} + \kappa_{i,t}(\alpha_{i,t} - x_{i,t}) && \text{(14c)}\\
& x_{i,0} = x_{i,\text{init}} \quad i = 1, \ldots, N. && \text{(14d)}
\end{aligned}$$

Notice that the cost functions of problem (14) have an aggregative structure, while the constraints are local to each agent.

Remark 4: The independent decision variables of problem (14) are the local curves $(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i)$. Indeed, once the curves $(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i)$ are computed, the variables $(\mathbf{x}_i, \mathbf{u}_i)$ are uniquely determined by means of the (projection) policy (13). □

A. Distributed Aggregative GoPRONTO

We design a distributed algorithm to solve (14) cooperatively via a protocol in which each agent employs only local data and exchanges information only with its neighbors.

At each iteration k, each agent i updates its local estimate $(\mathbf{x}_i^k, \mathbf{u}_i^k, \boldsymbol{\alpha}_i^k, \boldsymbol{\mu}_i^k)$ of the i-th component of an optimal solution to problem (14). Specifically, for each agent i, we first update the independent variables $(\boldsymbol{\alpha}_i^k, \boldsymbol{\mu}_i^k)$, i.e., the curve, and, then, via the feedback operator (13) projecting curves into trajectories, we recover the corresponding trajectory $(\mathbf{x}_i^k, \mathbf{u}_i^k)$. The rationale behind the update mechanism of the curve $(\boldsymbol{\alpha}_i^k, \boldsymbol{\mu}_i^k)$ (cf. (16)) is to approximate the centralized (gradient-related) dynamics by replacing the global information (not locally available) with local proxies $s_{i,t}$ and $y_{i,t}$ for all $t \in [0, T]$. That is, for each $t \in [0, T]$, agent i updates two auxiliary variables $s_{i,t}^k \in \mathbb{R}^q$ and $y_{i,t}^k \in \mathbb{R}^q$ in order to reconstruct the aggregative variable $\sigma_t(x_t^k, u_t^k)$ and the related gradient $\sum_{i=1}^N \nabla_3 \ell_i(x_{i,t}^k, u_{i,t}^k, \sigma_t(x^k, u^k))$, respectively. Specifically, for all $i = 1, \ldots, N$ and $t \in [0, T]$, each variable $s_{i,t}$ is tracking the quantity $\sigma_t(x_t^k, u_t^k)$ that varies along the iterations k. To this end, we update each $s_{i,t}$ by combining a consensus term $\sum_j w_{ij} s_{j,t}^k$ with the correction one $\varphi_{i,t}(x_{i,t}^{k+1}, u_{i,t}^{k+1}) - \varphi_{i,t}(x_{i,t}^k, u_{i,t}^k)$ giving rise to the so-called perturbed consensus dynamics widely adopted in the context of distributed average tracking, see, e.g., [47], [48]. The same arguments apply also for the set of trackers $y_{i,t}$.

Informally, the desired asymptotic behavior of $s_{i,t}^k$ and $y_{i,t}^k$ is

$$
\begin{aligned}
s_{i,t}^k &\xrightarrow{\text{tracks}} \sigma_t(x_t^k, u_t^k) = \frac{1}{N}\sum_{j=1}^N \varphi_{j,t}(x_{j,t}^k, u_{j,t}^k) \\
y_{i,t}^k &\xrightarrow{\text{tracks}} \frac{1}{N}\sum_{j=1}^N \nabla_3 \ell_{j,t}(x_{j,t}^k, u_{j,t}^k, \sigma_t(x_t^k, u_t^k)),
\end{aligned}
$$

for all $t \in [0, T-1]$. Similar behaviors are desired for $s_{i,T}^k$ and $y_{i,T}^k$ too. The overall strategy is presented in Algorithm 2 from the perspective of agent i, and by adopting the shorthands

$$
\begin{aligned}
A_{i,t}^k &:= \nabla_1 f_i(x_{i,t}^k, u_{i,t}^k)^\top, & B_{i,t}^k &:= \nabla_2 f_i(x_{i,t}^k, u_{i,t}^k)^\top, \\
K_{i,t}^k &:= \nabla \kappa_{i,t}(e_{i,t}^k)^\top, & &\text{where } e_{i,t}^k := \alpha_{i,t}^k - x_{i,t}^k,
\end{aligned}
$$

for all $i = 1, \ldots, N$ and $t \in [0, T-1]$.

Notice that, the computational burden for each agent is proportional to the *local* state, input, and aggregate variable dimensions and time length only. As for the communication network usage, steps (18)–(19) require agents to exchange T pairs of vectors in $\mathbb{R}^q$. Thereby, the data exchanged between agents consists of $2q(T+1)$ floating point numbers, with q is the dimension of the trackers in (18)–(19).

Remark 5: Coherently with the single-agent case (cf. (12)), the step (17) takes advantage of a feedback mechanism that confers, together with a sufficiently small stepsize, suitable numerical robustness properties to our distributed strategy. □

Algorithm 2 Distributed Aggregative GoPRONTO – Agent i

for $k = 0, 1, 2 \ldots$ **do**

set $\lambda_{i,T}^k = \nabla_1 \ell_{i,T}(x_{i,T}^k, s_{i,T}^k) + \nabla \varphi_{i,T}^k(x_{i,T}^k) y_{i,T}^k$

for $t = T-1, \ldots, 0$ **do**

Compute proxy variables based on $s_{i,t}^k$ and $y_{i,t}^k$

$$
\begin{aligned}
\hat{a}_{i,t}^k &= \nabla_1 \ell_{i,t}(x_{i,t}^k, u_{i,t}^k, s_{i,t}^k) + \nabla_1 \varphi(x_{i,t}^k, u_{i,t}^k)\; y_{i,t}^k \\
\hat{b}_{i,t}^k &= \nabla_2 \ell_{i,t}(x_{i,t}^k, u_{i,t}^k, s_{i,t}^k) + \nabla_2 \varphi(x_{i,t}^k, u_{i,t}^k)\; y_{i,t}^k
\end{aligned} \tag{15a}
$$

Compute local proxy descent direction

$$
\lambda_{i,t}^k = (A_{i,t}^k - B_{i,t}^k K_{i,t}^k)^\top \lambda_{i,t+1}^k + \hat{a}_{i,t}^k - K_{i,t}^{k,\top} \hat{b}_{i,t}^k \tag{15b}
$$

$$
\Delta\mu_{i,t}^k = -\hat{b}_{i,t}^k - B_{i,t}^{k\top} \lambda_{i,t+1}^k \tag{15c}
$$

$$
\Delta\alpha_{i,t}^k = K_{i,t}^{k,\top} \Delta\mu_{i,t}^k \tag{15d}
$$

end for

Update the local solution estimates and trackers

for $t = 0, \ldots, T-1$ **do**

Update curve

$$
\begin{aligned}
\alpha_{i,t}^{k+1} &= \alpha_{i,t}^k + \gamma \Delta\alpha_{i,t}^k \\
\mu_{i,t}^{k+1} &= \mu_{i,t}^k + \gamma \Delta\mu_{i,t}^k
\end{aligned} \tag{16}
$$

Compute new (feasible) trajectory with $x_{i,0}^{k+1} = x_{i,\text{init}}$

$$
u_{i,t}^{k+1} = \mu_{i,t}^{k+1} + \kappa_{i,t}(\alpha_{i,t}^{k+1} - x_{i,t}^{k+1}) \tag{17a}
$$

$$
x_{i,t+1}^{k+1} = f_i(x_{i,t}^{k+1}, u_{i,t}^{k+1}) \tag{17b}
$$

Update trackers

$$
\begin{aligned}
s_{i,t}^{k+1} &= \sum_{j \in \mathcal{N}_i} w_{ij} s_{j,t}^k + \varphi_{i,t}(x_{i,t}^{k+1}, u_{i,t}^{k+1}) - \varphi_{i,t}(x_{i,t}^k, u_{i,t}^k) \\
y_{i,t}^{k+1} &= \sum_{j \in \mathcal{N}_i} w_{ij} y_{j,t}^k + \nabla_3 \ell_{i,t}(x_{i,t}^{k+1}, u_{i,t}^{k+1}, s_{i,t}^{k+1}) \\
&\qquad - \nabla_3 \ell_{i,t}(x_{i,t}^k, u_{i,t}^k, s_{i,t}^k)
\end{aligned} \tag{18}
$$

end for

$$
\begin{aligned}
s_{i,T}^{k+1} &= \sum_{j \in \mathcal{N}_i} w_{ij} s_{j,T}^k + \varphi_{i,T}(x_{i,T}^{k+1}) - \varphi_{i,T}(x_{i,T}^k) \\
y_{i,T}^{k+1} &= \sum_{j \in \mathcal{N}_i} w_{ij} y_{j,T}^k + \nabla_2 \ell_{i,T}(x_{i,T}^{k+1}, s_{i,T}^{k+1}) \\
&\qquad - \nabla_2 \ell_{i,T}(x_{i,T}^k, s_{i,T}^k)
\end{aligned} \tag{19}
$$

end for

Remark 6: Notice that, although Algorithm 2 leverages on consensus-based mechanisms to reconstruct global, unavailable quantities (cf. (18) and (19)), consensus among the decision variables of problem (14) is not enforced. It is worth noting that one could implement an algorithm tailored for cost-coupled optimization, [49], and enforce consensus on a decision vector of dimension $(n+m)TN$. □

B. Algorithm Convergence Result

In this section, we provide the convergence result characterizing Algorithm 2.

For the sake of readability, let us introduce the stack vectors $\mathbf{s}_i^k \in \mathbb{R}^{q(T+1)}$, $\mathbf{y}_i^k \in \mathbb{R}^{q(T+1)}$, and $G_{i,3}(\mathbf{x}_i^k, \mathbf{u}_i^k, \boldsymbol{\sigma}(\mathbf{x}^k, \mathbf{u}^k)) \in$

$\mathbb{R}^{q(T+1)}$ defined as

$$\mathbf{s}_i^k := \text{col}(s_{i,0}^k, \ldots, s_{i,T}^k), \qquad \mathbf{y}_i^k := \text{col}(y_{i,0}^k, \ldots, y_{i,T}^k),$$

and

$$G_{i,3}(\mathbf{x}_i^k, \mathbf{u}_i^k, \mathbf{s}_i^k) := \text{col}\left(\nabla_3 \ell_{i,0}(x_{i,0}^k, u_{i,0}^k, s_{i,0}^k), \ldots, \nabla_3 \ell_{i,T-1}(x_{i,T-1}^k, u_{i,T-1}^k, s_{i,T-1}^k), \nabla_2 \ell_{i,T}(x_{i,T}^k, s_{i,T}^k)\right).$$

Theorem 1: Consider the distributed optimal control Algorithm 2. Let Assumptions 1 and 2 hold. For any initial condition $(\mathbf{x}_i^0, \mathbf{u}_i^0, \mathbf{s}_i^0, \mathbf{y}_i^0) \in \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T} \times \mathbb{R}^{q(T+1)} \times \mathbb{R}^{q(T+1)}$ such that $(\mathbf{x}_i^0, \mathbf{u}_i^0)$ is a trajectory and for all $i = 1, \ldots, N$,

$$\mathbf{s}_i^0 = \varphi_i(\mathbf{x}_i^0, \mathbf{u}_i^0), \qquad \mathbf{y}_i^0 = G_{i,3}(\mathbf{x}_i^0, \mathbf{u}_i^0, \mathbf{s}_i^0), \tag{20}$$

there exists $\bar{\gamma} > 0$ such that, for any $\gamma \in (0, \bar{\gamma})$, it holds

$$\lim_{k \to \infty} \text{dist}\left((\mathbf{x}^k, \mathbf{u}^k), \Xi^\star\right) = 0,$$

where the symbol $\Xi^\star$ denotes the set of trajectories $(\mathbf{x}^\star, \mathbf{u}^\star)$ satisfying the first-order necessary conditions for optimality of the original problem (5). □

The complete proof of Theorem 1 is provided in Section IV-D.

We underline that the result of Theorem 1 is semi-global restricted to the set defined by conditions (20). Notice that, as customary in nonconvex optimization (cf. Assumption 1), Theorem 1 ensures that Algorithm 2 asymptotically converges to a point satisfying the first-order necessary conditions of optimality for problem (5). Indeed, since the costs are nonconvex and the dynamics nonlinear, the solution of problem (5) is, in general, not unique. Namely, satisfying first-order necessary conditions of optimality does not imply either local or global optimality. However, as detailed extensively in Remark 7, in a neighborhood of an (isolated) local minimum of (5) satisfying second-order sufficient conditions of optimality, Algorithm 2 shows a linear rate of convergence.

IV. Algorithm Analysis

The analysis of the Distributed Aggregative GoPRONTO algorithm relies on the following cornerstone steps.

First, in Section IV-A, the local feedback-based controllers (17a) allow us to reformulate (14) as a reduced (unconstrained) optimization problem. Second, in Section IV-B, we show that our strategy reads as an approximation of the gradient descent method (cf. Lemma 1) interlaced with the dynamics of the proxies for the unavailable global quantities (cf. Lemma 2). Third, in Section IV-C, we rewrite Algorithm 2 via a change of variables (cf. Lemma 3) that (i) highlights the approximation error due to the proxies and (ii) removes the influence of their average components on the algorithm dynamics. Finally, in Section IV-D, we close the proof of Theorem 1 via LaSalle's based arguments.

A. Reduced Reformulation of the Optimal Control Problem

Given an initial condition $x_{i,0} \in \mathbb{R}^{n_i}$, the space of trajectories of agent i $\mathcal{T}_{i,\text{init}} \subseteq \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T}$ is defined as

$$\mathcal{T}_{i,\text{init}} := \Big\{(\mathbf{x}_i, \mathbf{u}_i) \in \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T} \mid x_{i,0} = x_{i,\text{init}}$$
$$x_{i,t+1} = f_i(x_{i,t}, u_{i,t}) \quad \forall t \in [0, T]\Big\},$$

for all $i = 1, \ldots, N$. Specifically, for each agent $i = 1, \ldots, N$, the local policy (13) implements a (local) feedback-projection operator $\mathcal{P}_i : \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T} \to \mathcal{T}_{i,\text{init}}$, which maps (local) state-input curves into trajectories for the i-th system. More formally, $\mathcal{P}_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i)$ is designed such that for any $(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i) \in \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T}$ and $(\mathbf{x}_i, \mathbf{u}_i) \in \mathcal{T}_{i,\text{init}}$, it holds

$$\begin{bmatrix} \boldsymbol{\alpha}_i \\ \boldsymbol{\mu}_i \end{bmatrix} \longmapsto \begin{bmatrix} \mathbf{x}_i \\ \mathbf{u}_i \end{bmatrix} := \mathcal{P}_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i) := \begin{bmatrix} \mathcal{P}_i^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i) \\ \mathcal{P}_i^{\boldsymbol{\mu}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i) \end{bmatrix}, \tag{21}$$

where $\mathcal{P}_i^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i)$ and $\mathcal{P}_i^{\boldsymbol{\mu}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i)$ are the state and input components of $\mathcal{P}_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i)$. Notice that it holds $\mathcal{P}_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i) \in \mathcal{T}_{i,\text{init}}$ and it satisfies the projection property $\mathcal{P}_i(\mathbf{x}_i, \mathbf{u}_i) = (\mathbf{x}_i, \mathbf{u}_i)$. For all $t \in [0, T-1]$ and $i = 1, \ldots, N$, we can also define the maps

$$x_{i,t} = \mathcal{P}_{i,t}^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \qquad u_{i,t} = \mathcal{P}_{i,t}^{\boldsymbol{\mu}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \tag{22}$$

where $\mathcal{P}_{i,t}^{\boldsymbol{\alpha}} : \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T} \to \mathbb{R}^{n_i}$ and $\mathcal{P}_{i,t}^{\boldsymbol{\mu}} : \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T} \to \mathbb{R}^{m_i}$. We denote as $\mathcal{P}_i^{\boldsymbol{\alpha}}(\cdot, \cdot)$ and $\mathcal{P}_i^{\boldsymbol{\mu}}(\cdot, \cdot)$ the stack of all $\mathcal{P}_{i,t}^{\boldsymbol{\alpha}}(\cdot, \cdot)$ and $\mathcal{P}_{i,t}^{\boldsymbol{\mu}}(\cdot, \cdot)$ for all $t = 0, \ldots, T-1$ and, similarly, as $\mathcal{P}^{\boldsymbol{\alpha}}(\cdot, \cdot)$, $\mathcal{P}^{\boldsymbol{\mu}}(\cdot, \cdot)$ the stack of the maps $\mathcal{P}_i^{\boldsymbol{\alpha}}(\cdot, \cdot)$, $\mathcal{P}_i^{\boldsymbol{\mu}}(\cdot, \cdot)$ for all $i = 1, \ldots, N$. The maps (22) are compositions of continuous functions and thus continuous. Therefore, we can reformulate problem (14) as

$$\min_{\substack{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N \\ \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N}} \underbrace{\sum_{i=1}^N J_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i, \boldsymbol{\varsigma}(\boldsymbol{\alpha}, \boldsymbol{\mu}))}_{J(\boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{\varsigma}(\boldsymbol{\alpha}, \boldsymbol{\mu}))}, \tag{23}$$

where

$$J_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i, \boldsymbol{\varsigma}(\boldsymbol{\alpha}, \boldsymbol{\mu})) := \ell_i(\mathcal{P}_i^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \mathcal{P}_i^{\boldsymbol{\mu}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \boldsymbol{\sigma}(\mathcal{P}^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}, \boldsymbol{\mu}), \mathcal{P}^{\boldsymbol{\mu}}(\boldsymbol{\alpha}, \boldsymbol{\mu}))),$$

in which $\boldsymbol{\varsigma}(\boldsymbol{\alpha}, \boldsymbol{\mu})$ is denoted as

$$\boldsymbol{\varsigma}(\boldsymbol{\alpha}, \boldsymbol{\mu}) := \frac{1}{N} \sum_{i=1}^N \bar{\boldsymbol{\varphi}}_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i),$$

with $\bar{\boldsymbol{\varphi}}_i : \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T} \to \mathbb{R}^{q(T+1)}$ is defined as

$$\bar{\boldsymbol{\varphi}}_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i) := \text{col}(\varphi_{i,0}(\mathcal{P}_{i,t}^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \mathcal{P}_{i,t}^{\boldsymbol{\mu}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i)), \ldots, \varphi_{i,T}(\mathcal{P}_{i,T}^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \mathcal{P}_{i,T}^{\boldsymbol{\mu}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i))).$$

Problem (23) is an unconstrained optimization problem in $(\boldsymbol{\alpha}, \boldsymbol{\mu})$. Thus, it can be addressed by a distributed implementation of the gradient descent method. In this context, each agent i iteratively updates its local estimate $(\boldsymbol{\alpha}_i^k, \boldsymbol{\mu}_i^k) \in \mathbb{R}^{n_i T} \times \mathbb{R}^{m_i T}$ about the i-th component of a solution to problem (23), with $k \in \mathbb{N}$ denoting the iteration index. The gradient method update applied to problem (23) reads as

$$\boldsymbol{\alpha}_i^{k+1} = \boldsymbol{\alpha}_i^k - \gamma D_{\boldsymbol{\alpha}_i}^k, \qquad \boldsymbol{\mu}_i^{k+1} = \boldsymbol{\mu}_i^k - \gamma D_{\boldsymbol{\mu}_i}^k, \tag{24}$$

for all $i = 1, \ldots, N$, with $\gamma > 0$ the so-called step-size and descent directions $D_{\boldsymbol{\alpha}_i}^k \in \mathbb{R}^{n_i T}$ and $D_{\boldsymbol{\mu}_i} \in \mathbb{R}^{m_i T}$ defined as

$$D_{\boldsymbol{\alpha}_i} := \nabla_1 J_i(\boldsymbol{\alpha}_i^k, \boldsymbol{\mu}_i^k, \boldsymbol{\varsigma}(\boldsymbol{\alpha}^k, \boldsymbol{\mu}^k)) \tag{25a}$$

$$+\tfrac{1}{N}\nabla_1\bar{\varphi}(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k)\sum_{j=1}^{N}\nabla_3 J(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k,\varsigma(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k))$$

$$D_{\boldsymbol{\mu}_i} := \nabla_2 J_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k,\varsigma(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k)) + \tfrac{1}{N}\nabla_2\bar{\varphi}(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k)\sum_{j=1}^{N}\nabla_3 J(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k,\varsigma(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k)). \tag{25b}$$

In order to implement (24), we need to overcome two main challenges. Firstly, being the cost function in (23) a composition of ℓ_i with maps (22), we need to calculate $\nabla\mathcal{P}_i^{\boldsymbol{\alpha}}$ and $\nabla\mathcal{P}_i^{\boldsymbol{\mu}}$, which, in general is non trivial. Secondly, the cost function (23) also exhibits a composition with the aggregative variable ς that couples all the variables $(\boldsymbol{\alpha},\boldsymbol{\mu})$ of the network. Consequently, the update (24) requires, for each agent, the knowledge of unavailable global information. This issue is tackled by properly exploiting a consensus-based strategy. The next section provides insights into these two aspects.

B. Descent Direction and Local Proxies

In this section, we formally define a useful numerical routine for the computation of the gradient of the reduced cost function $J(\boldsymbol{\alpha},\boldsymbol{\mu},\varsigma(\boldsymbol{\alpha},\boldsymbol{\mu}))$. Then, we show how to compensate for the unavailable global information required by this routine.

Lemma 1: Consider a state-input curve $(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k)$. For all $i=1,\dots,N$, the integration backward in time from $t=T-1,\dots,0$ of the closed-loop costate equation reads as

$$\lambda_{i,t}^k = (A_{i,t}^k - K_{i,t}^k B_{i,t}^k)^\top \lambda_{i,t+1}^k + a_{i,t}^k - K_{i,t}^{k,\top}\hat{b}_{i,t}^k,$$

where $\lambda_{i,t}\in\mathbb{R}^{n_i}$,

$$a_{i,t}^k := \nabla_2\ell_{i,t}(x_{i,t}^k,u_{i,t}^k,\sigma_t(x_t^k,u_t^k)) + \tfrac{1}{N}\nabla_2\varphi_{i,t}(x_{i,t}^k,u_{i,t}^k)\sum_{j=1}^{N}\nabla_3\ell_{j,t}(x_{j,t}^k,u_{j,t}^k,\sigma_t(x_t^k,u_t^k)), \tag{26a}$$

and the initial condition $\lambda_{i,T}\in\mathbb{R}^{n_i}$, is set as

$$\lambda_{i,T}^k = \nabla\ell_{i,T}(x_{i,T}^k,\sigma_T(x_T^k)) + \tfrac{1}{N}\nabla\varphi_{i,T}(x_{i,T}^k)\sum_{j=1}^{N}\nabla_2\ell_{j,T}(x_{j,T}^k,\sigma_t(x_T^k)). \tag{26b}$$

Then, for all $i=1,\dots,N$, the gradient of the reduced cost function (23) with respect to $\mu_{i,t}$ and $\alpha_{i,t}$ read as

$$[D_{\boldsymbol{\alpha}_i}^k]_t = -b_{i,t}^k - B_{i,t}^k\lambda_{i,t+1}^k, \quad [D_{\boldsymbol{\mu}_i}^k]_t = K_{i,t}^{k,\top}\Delta\mu_{i,t}^k, \tag{27}$$

where the operator $[\cdot]_t$ denotes the t-th block of (25) and, for all $t=0,\dots,T$,

$$b_{i,t}^k := \nabla_2\ell_{i,t}(x_{i,t}^k,u_{i,t}^k) + + \tfrac{1}{N}\nabla_2\varphi_{i,t}(x_{i,t}^k,u_{i,t}^k)\sum_{j=1}^{N}\nabla_3\ell_{j,t}(x_{j,t}^k,u_{j,t}^k,\sigma_t(x_t^k,u_t^k)). \ \square \tag{28}$$

The proof is provided in Appendix A.

Notice that, the terms in (26) and (28) require the local knowledge of the global quantities $\sigma_t(\cdot,\cdot)$, $\sum_{j=1}^{N}\nabla_3\ell_{j,t}(x_{j,t},u_{j,t},\sigma_t(\cdot,\cdot))$, $\sigma_T(\cdot)$, and $\sum_{j=1}^{N}\nabla_2\ell_{j,T}(x_{j,T},\sigma_T(\cdot))$. For this reason, in the computation of each $a_{i,t}^k$, $b_{i,t}^k$, and $\lambda_{i,T}^k$, these global quantities are replaced by the trackers $s_{i,t}$ and $y_{i,t}$. Concurrently, these trackers are updated according to perturbed consensus dynamics exploiting inter-agent communication (cf. (18) and (19)).

Hence, we formally show the relationship between the directions $\boldsymbol{\Delta\alpha}_i^k$ and $\boldsymbol{\Delta\mu}_i^k$ computed via Algorithm 2 and the derivative of the reduced costs J_i with respect to $\boldsymbol{\alpha}_i$ and $\boldsymbol{\mu}_i$.

Lemma 2 (Local proxy of the descent direction): Consider Algorithm 2. Then, it holds, for all $i=1,\dots,N$,

$$\boldsymbol{\Delta\alpha}_i^k = -\nabla_1 J_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k,\mathbf{s}_i^k) - \nabla_1\bar{\varphi}_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k)\,\mathbf{y}_i^k$$
$$\boldsymbol{\Delta\mu}_i^k = -\nabla_2 J_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k,\mathbf{s}_i^k) - \nabla_2\bar{\varphi}_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k)\,\mathbf{y}_i^k.$$

Proof: At each iteration k, consider the quantities $\mathbf{s}_i^k$ and $\mathbf{y}_i^k$ as the numerical proxies of the centralized quantities $\sigma_t(x_t^k,u_t^k)$ and $\frac{1}{N}\sum_{i=1}^{N}\nabla_3\ell_{i,t}(x_{i,t}^k,u_{i,t}^k,\sigma_t(x_t^k,u_t^k))$, respectively. Considering equations (26)-(27) of Lemma 1, one can see that the backward sweep (15) computes the approximated gradient of the reduced cost J at the current iteration. ■

C. Algorithm Reformulation: Average and Perpendicular Dynamics of the Trackers in Error Coordinates

Next, we provide a reformulation of Algorithm 2 used to prove Theorem 1. The initial condition $(\mathbf{x}_i^0,\mathbf{u}_i^0)$ is a trajectory, $(\mathbf{s}_i^0,\mathbf{y}_i^0)$ is such that (20) holds, and Assumptions 1 and 2 hold.

By exploiting Lemma 2, the local update describing Algorithm 2 can be summarized as

$$\boldsymbol{\alpha}_i^{k+1} = \boldsymbol{\alpha}_i^k - \gamma\big(\nabla_1 J_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k,\mathbf{s}_i^k) + \nabla_1\bar{\boldsymbol{\varphi}}_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k,\mathbf{s}_i^k)\mathbf{y}_i^k\big) \tag{29a}$$

$$\boldsymbol{\mu}_i^{k+1} = \boldsymbol{\mu}_i^k - \gamma\big(\nabla_2 J_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k,\mathbf{s}_i^k) + \nabla_2\bar{\boldsymbol{\varphi}}_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k,\mathbf{s}_i^k)\mathbf{y}_i^k\big) \tag{29b}$$

$$\mathbf{s}_i^{k+1} = \sum_{j\in\mathcal{N}_i} w_{ij}\mathbf{s}_j^k + \bar{\boldsymbol{\varphi}}_i(\boldsymbol{\alpha}_i^{k+1},\boldsymbol{\mu}_i^{k+1}) - \bar{\boldsymbol{\varphi}}_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k) \tag{29c}$$

$$\mathbf{y}_i^{k+1} = \sum_{j\in\mathcal{N}_i} w_{ij}\mathbf{y}_j^k + \nabla_3 J_i(\boldsymbol{\alpha}_i^{k+1},\boldsymbol{\mu}_i^{k+1},\mathbf{s}_i^{k+1}) - \nabla_3 J_i(\boldsymbol{\alpha}_i^k,\boldsymbol{\mu}_i^k,\mathbf{s}_i^k). \tag{29d}$$

Now, we set $W := \mathcal{W}\otimes I_{q(T+1)}$ and use (29) to compactly describe the update of all the variables of the network as

$$\boldsymbol{\alpha}^{k+1} = \boldsymbol{\alpha}^k - \gamma(\mathrm{D}_1(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k,\mathbf{s}^k) + \nabla_1\bar{\boldsymbol{\varphi}}(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k)\mathbf{y}^k) \tag{30a}$$
$$\boldsymbol{\mu}^{k+1} = \boldsymbol{\mu}^k - \gamma(\mathrm{D}_2(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k,\mathbf{s}^k) + \nabla_2\bar{\boldsymbol{\varphi}}(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k)\mathbf{y}^k) \tag{30b}$$
$$\mathbf{s}^{k+1} = W\mathbf{s}^k + \bar{\boldsymbol{\varphi}}(\boldsymbol{\alpha}^{k+1},\boldsymbol{\mu}^{k+1}) - \bar{\boldsymbol{\varphi}}(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k) \tag{30c}$$
$$\mathbf{y}^{k+1} = W\mathbf{y}^k + \mathrm{D}_3(\boldsymbol{\alpha}^{k+1},\boldsymbol{\mu}^{k+1},\mathbf{s}^{k+1}) - \mathrm{D}_3(\boldsymbol{\alpha}^k,\boldsymbol{\mu}^k,\mathbf{s}^k), \tag{30d}$$

where we introduced the operators $\mathrm{D}_1 : \mathbb{R}^{nT}\times\mathbb{R}^{mT}\times\mathbb{R}^{q(T+1)N}\to\mathbb{R}^{nT}$, $\mathrm{D}_2 : \mathbb{R}^{nT}\times\mathbb{R}^{mT}\times\mathbb{R}^{q(T+1)N}\to\mathbb{R}^{mT}$, and $\mathrm{D}_3 : \mathbb{R}^{nT}\times\mathbb{R}^{mT}\times\mathbb{R}^{q(T+1)N}\to\mathbb{R}^{q(T+1)N}$ defined as

$$\begin{aligned}
&\mathrm{D}_1(\boldsymbol{\alpha},\boldsymbol{\mu},\mathbf{s}) := \mathrm{col}(\nabla_1 J_i(\boldsymbol{\alpha}_i,\boldsymbol{\mu}_i,\mathbf{s}_i)),\\
&\mathrm{D}_2(\boldsymbol{\alpha},\boldsymbol{\mu},\mathbf{s}) := \mathrm{col}(\nabla_2 J_i(\boldsymbol{\alpha}_i,\boldsymbol{\mu}_i,\mathbf{s}_i)), \quad i=1,\dots,N\\
&\mathrm{D}_3(\boldsymbol{\alpha},\boldsymbol{\mu},\mathbf{s}) := \mathrm{col}(\nabla_3 J_i(\boldsymbol{\alpha}_i,\boldsymbol{\mu}_i,\mathbf{s}_i)),
\end{aligned} \tag{31}$$

with $\boldsymbol{\alpha} := \mathrm{col}(\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_N)$, $\boldsymbol{\mu} := \mathrm{col}(\boldsymbol{\mu}_1,\dots,\boldsymbol{\mu}_N)$, and $\mathbf{s} := \mathrm{col}(\mathbf{s}_1,\dots,\mathbf{s}_N)$, with $\boldsymbol{\alpha}_i\in\mathbb{R}^{n_iT}$, $\boldsymbol{\mu}_i\in\mathbb{R}^{m_iT}$, and $\mathbf{s}_i\in\mathbb{R}^{q(T+1)}$ for all $i=1,\dots,N$.

The next lemma introduces a change of coordinates which simplifies the algorithm analysis. Before providing its formal statement, we introduce $d = (N-1)q(T+1)$, $\mathbf{1} := 1_N \otimes I_{q(T+1)}$, and the operators $\mathrm{D} : \mathbb{R}^{(n+m)T} \times \mathbb{R}^{q(T+1)N} \to \mathbb{R}^{(n+m)T}$, $u_1 : \mathbb{R}^{(n+m)T} \times \mathbb{R}^{q(T+1)N} \to \mathbb{R}^{q(T+1)N}$, and $u_2 : \mathbb{R}^{(n+m)T} \times \mathbb{R}^{(n+m)T} \to \mathbb{R}^{2q(T+1)N}$ defined as

$$\mathrm{D}(\boldsymbol{\xi}, \mathbf{s}) := \begin{bmatrix} \mathrm{D}_1(\boldsymbol{\xi}, \mathbf{s}) \\ \mathrm{D}_2(\boldsymbol{\xi}, \mathbf{s}) \end{bmatrix} + \nabla \bar{\varphi}(\boldsymbol{\xi}) \frac{\mathbf{1}\mathbf{1}^\top}{N} \mathrm{D}_3(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi})) + + \nabla \bar{\varphi}(\boldsymbol{\xi}) u_1(\boldsymbol{\xi}, \mathbf{s}) \tag{32a}$$

$$u_1(\boldsymbol{\xi}, \mathbf{s}) := \mathrm{D}_3(\boldsymbol{\xi}, \mathbf{s}) - \mathrm{D}_3(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi})) \tag{32b}$$

$$u_2(\boldsymbol{\xi}, \boldsymbol{\xi}') := \begin{bmatrix} \bar{\varphi}(\boldsymbol{\xi}) - \bar{\varphi}(\boldsymbol{\xi}') \\ \mathrm{D}_3(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi})) - \mathrm{D}_3(\boldsymbol{\xi}', \mathbf{1}\varsigma(\boldsymbol{\xi}')) \end{bmatrix}, \tag{32c}$$

where, with a slight abuse of notation, we joined the first two arguments of the operators ς, $\bar{\varphi}$, D_1, D_2, and D_3 to adapt them for this notation using $\boldsymbol{\xi}$ instead of $(\boldsymbol{\alpha}, \boldsymbol{\mu})$.

Lemma 3 (Dynamics reformulation): Consider (30). Then, there exists a set of coordinates $(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \in \mathbb{R}^{(n+m)T} \times \mathbb{R}^{2d}$ such that (30) is equivalent to

$$\boldsymbol{\xi}^{k+1} = \boldsymbol{\xi}^k - \gamma \mathrm{D}(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + [R\ 0]\boldsymbol{\vartheta}^k) + - \gamma [0\ \nabla \bar{\varphi}(\boldsymbol{\xi}^k) R] \boldsymbol{\vartheta}^k \tag{33a}$$

$$\boldsymbol{\vartheta}^{k+1} = \mathcal{A}\boldsymbol{\vartheta}^k + \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + \begin{bmatrix} R & 0 \end{bmatrix} \boldsymbol{\vartheta}^k) + + \mathcal{B}_2 u_2(\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1}), \tag{33b}$$

where $\mathcal{A}$, $\mathcal{B}_1$, $\mathcal{B}_2$ are given matrices defined as

$$\begin{aligned} \mathcal{A} &:= \begin{bmatrix} R^\top W R & 0 \\ 0 & R^\top W R \end{bmatrix}, \\ \mathcal{B}_1 &:= \begin{bmatrix} 0 \\ R^\top (W - I) \end{bmatrix}, \quad \mathcal{B}_2 := \begin{bmatrix} R^\top & 0 \\ 0 & R^\top \end{bmatrix}, \end{aligned} \tag{34}$$

and $R \in \mathbb{R}^{N(q(T+1)) \times d}$ is such that $R^\top R = I$, $R^\top \mathbf{1} = 0$. □

The proof of Lemma 3 is reported in Appendix B.

The new variable $\boldsymbol{\vartheta}^k$ describes the evolution of the error of both $\mathbf{s}^k$ and $\mathbf{y}^k$ with respect to the global information $\varsigma(\boldsymbol{\mu}^k, \boldsymbol{\alpha}^k)$ and $\frac{1}{N}\sum_{i=1}^N \nabla_3 J_i(\boldsymbol{\mu}_i^k, \boldsymbol{\alpha}_i^k, \varsigma(\boldsymbol{\mu}^k, \boldsymbol{\alpha}^k))$ after a proper change of variables aimed at isolating the invariant (average) components of $\mathbf{s}^k$ and $\mathbf{y}^k$ which are set to zero by the initialization (20) (cf. proof of Lemma 3 in Appendix B).

We are now ready to give the proof of the main theorem.

D. Proof of Theorem 1

Leveraging on Lemma 3, we provide the proof of Theorem 1 by going through three main steps. In step (i), we show that ∇J, D_3, and $\bar{\varphi}$ are locally Lipschitz continuous and compute the related constants. In step (ii), we employ these results to perform a LaSalle-based analysis demonstrating that the sequence $\{\boldsymbol{\alpha}^k, \boldsymbol{\mu}^k\}_{k \in \mathbb{N}}$ generated by Algorithm 2 converges to the set of stationary points of the reduced cost function $J(\boldsymbol{\alpha}, \boldsymbol{\mu}, \varsigma(\boldsymbol{\alpha}, \boldsymbol{\mu}))$. In step (iii), we show that the state-input trajectory $(\mathbf{x}^\star, \mathbf{u}^\star)$ obtained by projecting $(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)$ satisfies the first-order necessary conditions for optimality in correspondence of the costate sequence $\boldsymbol{\lambda}^\star$.

***Step (i)**: Lipschitz continuity of ∇J, D_3, and $\bar{\varphi}$:* let us define $V : \mathbb{R}^{(n+m)T} \times \mathbb{R}^{2d} \to \mathbb{R}$ as

$$V(\boldsymbol{\xi}, \boldsymbol{\vartheta}) := J(\boldsymbol{\xi}, \varsigma(\boldsymbol{\xi})) + \boldsymbol{\vartheta}^\top P \boldsymbol{\vartheta}, \tag{35}$$

where $P \in \mathbb{R}^{2d \times 2d}$ has the diagonal structure $P := \begin{bmatrix} P_1 & 0 \\ 0 & pP_2 \end{bmatrix}$, with $P_1, P_2 \in \mathbb{R}^{d \times d}$ with $P_1 = P_1^\top > 0$ and $P_2 = P_2^\top > 0$, while $p \in (0, 1]$ is a parameter that we will fix later. In particular, we arbitrarily choose $Q_1, Q_2 \in \mathbb{R}^{d \times d}$ such that $Q_1 = Q_1^\top > 0$ and $Q_2 = Q_2^\top > 0$ and pick P_1 and P_2 as the solutions of the Lyapunov equations

$$P_1 - (R^\top W R)^\top P_1 (R^\top W R) = Q_1 \tag{36a}$$

$$P_2 - (R^\top W R)^\top P_2 (R^\top W R) = Q_2. \tag{36b}$$

Indeed, in light of Assumption 2, the matrix $R^\top W R$ is Schur and, thus, given $Q_1 = Q_1^\top > 0$ and $Q_2 = Q_2^\top > 0$, there always exist P_1 and P_2 solving (36). Now, let $(\boldsymbol{\xi}^0, \boldsymbol{\vartheta}^0) \in \mathbb{R}^{(n+m)T} \times \mathbb{R}^{2d}$ be the vectors corresponding to the initial conditions $\mathbf{x}_i^0$, $\mathbf{u}_i^0$, $\mathbf{s}_i^0$, and $\mathbf{y}_i^0$ assumed in Theorem 1 and, hence, satisfying (20). Let $\boldsymbol{\vartheta} := \mathrm{col}(\boldsymbol{\vartheta}_1, \boldsymbol{\vartheta}_2)$ with $\boldsymbol{\vartheta}_1, \boldsymbol{\vartheta}_2 \in \mathbb{R}^d$ and let $c_0 \in \mathbb{R}$ and $r_0 > 0$ be the smallest numbers such that

$$J(\boldsymbol{\xi}^0, \varsigma(\boldsymbol{\xi}^0)) + {\boldsymbol{\vartheta}_1^0}^\top P \boldsymbol{\vartheta}_1^0 \le c_0, \qquad \left\|\boldsymbol{\vartheta}_2^0\right\| \le r_0. \tag{37}$$

Then, for any $c \ge c_0$, the corresponding level set $\Omega_1^c \subset \mathbb{R}^{(n+m)T} \times \mathbb{R}^{(N-1)q(T+1)}$ is denoted as

$$\Omega_1^c := \{(\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \mid J(\boldsymbol{\xi}^0, \varsigma(\boldsymbol{\xi}^0)) + {\boldsymbol{\vartheta}_1^0}^\top P_1 \boldsymbol{\vartheta}_1^0 \le c\}. \tag{38}$$

We underline that, as before, with a slight abuse of notation, we joined the first two arguments of J to adapt it for the notation using $\boldsymbol{\xi}$ instead of $(\boldsymbol{\alpha}, \boldsymbol{\mu})$. Since $J(\boldsymbol{\xi}, \varsigma(\boldsymbol{\xi}))$ is the composition of $\mathcal{C}^2$ functions (see Section IV-A and Assumption 1), then it is $\mathcal{C}^2$. Specifically, on a compact set the function itself and its derivatives, up to the second order, are bounded. Moreover, being the cost function radially unbounded (cf. Assumption 1), also the composition $J(\cdot, \varsigma(\cdot))$ is radially unbounded. Hence, each level set of $J(\cdot, \varsigma(\cdot))$ is compact. The same holds for $J(\boldsymbol{\xi}, \varsigma(\boldsymbol{\xi})) + {\boldsymbol{\vartheta}_1}^\top P_1 \boldsymbol{\vartheta}_1$. Let us define

$$\begin{aligned} M_1 &:= \max\{\|\boldsymbol{\xi}\| \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega_1^c\} \\ M_2 &:= \max\{\|\boldsymbol{\vartheta}_1\| \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega_1^c\} \\ M_3 &:= \max\{\|\mathrm{D}(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi}) + R\boldsymbol{\vartheta}_1)\| \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega_1^c\} \\ M_4 &:= \max\{\|\nabla \bar{\varphi}(\boldsymbol{\xi})\| \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega_1^c\} \\ M_5 &:= \max\{\|u_1(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi}) + R\boldsymbol{\vartheta}_1)\| \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega_1^c\} \\ M_6 &:= \max\{\|\mathrm{D}_3(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi}) + R\boldsymbol{\vartheta}_1)\| \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega_1^c\}, \end{aligned} \tag{39}$$

whose existence is guaranteed since each continuous function on a compact set is bounded, J and V are $\mathcal{C}^2$ functions, and the set Ω_1^c is compact. Now, given $r > 0$, we define

$$M_7 := M_3 + M_4 \|R\| r,$$

and the related sets $B_{\boldsymbol{\xi}}^c \subset \mathbb{R}^{(n+m)T}$, $B_2^r \subset \mathbb{R}^d$ given by

$$B_{\boldsymbol{\xi}}^c := \{\boldsymbol{\xi} \in \mathbb{R}^{(n+m)T} \mid \|\boldsymbol{\xi}\| \le M_1 + M_7\}. \tag{40a}$$

$$B_2^r := \{\boldsymbol{\vartheta}_2 \in \mathbb{R}^d \mid \|\boldsymbol{\vartheta}_2\| \le r\}. \tag{40b}$$

Now, assume $(\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega_1^c$ and $\boldsymbol{\vartheta}_2 \in B_2^r$. Then, the update (33a) allows us to write the bound

$$\begin{aligned} \|\boldsymbol{\xi}^+\| &\le \|\boldsymbol{\xi}\| + \gamma \|\mathrm{D}(\boldsymbol{\xi}, \varsigma(\boldsymbol{\xi}) + R\boldsymbol{\vartheta}_1)\| + \gamma \|\nabla \bar{\varphi}(\boldsymbol{\xi}) R \boldsymbol{\vartheta}_2\| \\ &\overset{(a)}{\le} M_1 + \gamma(M_3 + M_4 \|R\| r) = M_1 + \gamma M_7 \end{aligned} \tag{41}$$

where in (a) we exploited the bounds expressed in (39).Thus, by combining the inequality (41) and the fact that $\gamma \leq 1$, it holds $\left\|\boldsymbol{\xi}^+\right\| \leq M_1 + M_7$, namely it holds $\boldsymbol{\xi}^+ \in B^c_{\boldsymbol{\xi}}$ (cf. (40a)). Now, let us define $B^c_1 \subseteq \Omega^c_1$ as

$$B^c_1 := \{(\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \mathbb{R}^{(n+m)T} \times \mathbb{R}^d \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega^c_1, \boldsymbol{\xi} \in B^c_{\boldsymbol{\xi}}\}.$$

From this definition, it holds $(\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in \Omega^c_1$ for any $(\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in B^c_1$. Moreover, such definition guarantees that, if $(\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in B^c_1$, then it holds $\boldsymbol{\xi} \in B^c_{\boldsymbol{\xi}}$. Now, let $\mathcal{H}_1 : \mathbb{R}^{(n+m)T} \times \mathbb{R}^{Nq(T+1)} \to \mathbb{R}^{(n+m)T \times (n+m)T}$ and $\mathcal{H}_2 : \mathbb{R}^{(n+m)T} \times \mathbb{R}^{Nq(T+1)} \to \mathbb{R}^{q(T+1) \times q(T+1)}$ be the operators defined as

$$\mathcal{H}_1(\boldsymbol{\xi}, \mathbf{s}) = \text{blkdiag}(\nabla^2_1 J_1(\boldsymbol{\xi}_1, \mathbf{s}_1), \ldots, \nabla^2_1 J_N(\boldsymbol{\xi}_N, \mathbf{s}_N))$$
$$\mathcal{H}_2(\boldsymbol{\xi}, \mathbf{s}) = \text{blkdiag}(\nabla^2_2 J_1(\boldsymbol{\xi}_1, \mathbf{s}_1), \ldots, \nabla^2_2 J_N(\boldsymbol{\xi}_N, \mathbf{s}_N)),$$

where $\boldsymbol{\xi} := \text{col}(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_N)$ and $\mathbf{s} := \text{col}(\mathbf{s}_1, \ldots, \mathbf{s}_N)$, with $\boldsymbol{\xi}_i \in \mathbb{R}^{(n_i+m_i)T}$ and $\mathbf{s}_i \in \mathbb{R}^{q(T+1)}$ for all $i = 1, \ldots, N$. Since ∇J, D_3, and $\bar{\varphi}$ are compositions of continuous and differentiable functions (cf. Assumption 1) and B_1 is a compact set, we can define the finite constants L, L_2 and L_3 as

$$L := \max\{\|\nabla^2 J(\boldsymbol{\xi}, \varsigma(\boldsymbol{\xi}))\| \mid \boldsymbol{\xi} \in B^c_{\boldsymbol{\xi}}\} \tag{42a}$$
$$L_1 := \max\{\|\mathcal{H}_1(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi}) + R\boldsymbol{\vartheta}_1)\| \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in B^c_1\} \tag{42b}$$
$$L_2 := \max\{\|\mathcal{H}_2(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi}) + R\boldsymbol{\vartheta}_1)\| \mid (\boldsymbol{\xi}, \boldsymbol{\vartheta}_1) \in B^c_1\} \tag{42c}$$
$$L_3 := \max\{\|\nabla\bar{\varphi}(\boldsymbol{\xi})\| \mid \boldsymbol{\xi} \in B^c_{\boldsymbol{\xi}}\}. \tag{42d}$$

Hence, $\nabla J(\cdot, \cdot)$, D_3, and $\bar{\varphi}$ are Lipschitz with constants L, L_2, and L_3 , respectively, over B^c_1.

***Step (ii)**: LaSalle-based analysis:* let us define

$$\mathbf{c} := c_0 + \|P_2\| \, r_0^2, \tag{43}$$

and assume that $(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k_1) \in \Omega^{\mathbf{c}}_1$ and $\boldsymbol{\vartheta}^k_2 \in B^{r_0}_2$ (see (38) for the definitions of these sets). Such an assumption will be verified later by a proper selection of γ. Thus, for the arguments above, it holds $\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1} \in B^{\mathbf{c}}_{\boldsymbol{\xi}}$. Thus, by using the Lipschitz continuity over $B^{\mathbf{c}}_1$ of the functions $\text{D}_3(\cdot, \cdot)$ and $\bar{\varphi}(\cdot)$, we can bound the norm of $u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1)$ (cf. (32)) as

$$\left\|u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1)\right\| \leq L_2 \|R\| \left\|\boldsymbol{\vartheta}^k_1\right\|. \tag{44}$$

With the same arguments, we can bound the norm of $u_2(\boldsymbol{\xi}^{k+1}, \boldsymbol{\xi}^k)$ (cf. (32)) as

$$\begin{aligned}
&\left\|u_2(\boldsymbol{\xi}^{k+1}, \boldsymbol{\xi}^k)\right\| \leq (L_3 + L_2 \left\|\mathbf{1}\mathbf{1}^\top/N\right\| L_3) \left\|\boldsymbol{\xi}^{k+1} - \boldsymbol{\xi}^k\right\| \\
&\overset{(a)}{\leq} \gamma c_1 \left\|\text{D}(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + \begin{bmatrix} R & 0 \end{bmatrix} \boldsymbol{\vartheta}^k)\right\| + \gamma c_1 L_3 \|R\| \left\|\boldsymbol{\vartheta}^k_2\right\| \\
&\overset{(b)}{\leq} \gamma c_1 \|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\| + \gamma c_2 \|\boldsymbol{\vartheta}^k_1\| + \gamma c_3 \|\boldsymbol{\vartheta}^k_2\|,
\end{aligned} \tag{45}$$

where in (a) we defined $c_1 := L_3 + L_2 \left\|\mathbf{1}\mathbf{1}^\top/N\right\| L_3$, used the update (33a), the triangle inequality, the Cauchy-Schwarz inequality, and the Lipschitz continuity of $\nabla\varphi$ over $\Omega^{\mathbf{c}}_1$, while in (b) we added and subtracted $\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))$ within the first norm, we applied the triangle inequality and the Lipschitz continuity of D over $\Omega^{\mathbf{c}}_1$, and defined $c_2 := (L_1 + L_2 L_3)\|R\|$ and $c_3 := c_1 L_3 \|R\|$. With these results at hand, we use the function $V(\cdot, \cdot)$ defined in (35) as a candidate Lyapunov function. By evaluating $\Delta V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) := V(\boldsymbol{\xi}^{k+1}, \boldsymbol{\vartheta}^{k+1}) - V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k)$ along the trajectories of system (33), we obtain

$$\begin{aligned}
\Delta V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) &= J(\boldsymbol{\xi}^{k+1}, \varsigma(\boldsymbol{\xi}^{k+1})) - J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) \\
&\quad - {\boldsymbol{\vartheta}^k}^\top (P - \mathcal{A}^\top P \mathcal{A}) \boldsymbol{\vartheta}^k && (46a) \\
&\quad + 2{\boldsymbol{\vartheta}^k}^\top \mathcal{A}^\top P \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) && (46b) \\
&\quad + 2{\boldsymbol{\vartheta}^k}^\top \mathcal{A}^\top P \mathcal{B}_2 u_2(\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1}) \\
&\quad + 2u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1)^\top \mathcal{B}_1^\top P \mathcal{B}_2 u_2(\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1}) \\
&\quad + u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1)^\top \mathcal{B}_1^\top P \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \\
&\quad + u_2(\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1})^\top \mathcal{B}_2^\top P \mathcal{B}_2 u_2(\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1}). && (46c)
\end{aligned}$$

Now, we consider the term ${\boldsymbol{\vartheta}^k}^\top \mathcal{A}^\top P \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k)$ and exploit the structure of $\mathcal{A}^\top P \mathcal{B}_1$ to write

$$\begin{aligned}
&{\boldsymbol{\vartheta}^k}^\top \mathcal{A}^\top P \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \\
&= p{\boldsymbol{\vartheta}^k_2}^\top (R^\top W R)^\top P_2 R^\top (W - I) u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \\
&\overset{(a)}{\leq} p \left\|(R^\top W R)^\top P_2 R^\top (W - I)\right\| L_2 \|R\| \, \|\boldsymbol{\vartheta}^k_2\| \|\boldsymbol{\vartheta}^k_1\| \\
&\overset{(b)}{\leq} p c_4 \|\boldsymbol{\vartheta}^k_2\| \|\boldsymbol{\vartheta}^k_1\|,
\end{aligned} \tag{47}$$

where in (a) we apply the Cauchy-Schwarz inequality and the bound (44), and in (b) we set $c_4 := \left\|(R^\top W R)^\top P_2 R^\top\right\| L_2 \left\|R^\top (W - I)\right\|$. Now we consider the term $u_1^\top(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \mathcal{B}_1^\top P \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1)$. By exploiting the structures of the matrices $\mathcal{B}_1$ (cf. (34)) and P and the bound (44), we have

$$\begin{aligned}
&u_1^\top(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \mathcal{B}_1^\top P \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \\
&\leq p q_3 \left\|\boldsymbol{\vartheta}^k_1\right\|^2,
\end{aligned} \tag{48}$$

where $q_3 := {L_2}^2 \|R\|^2 \left\|(W - I)^\top R P_2 R^\top (W - I)\right\|$. Now, via (36), (47), and (48) we can bound (46a) and (46b) as

$$\begin{aligned}
&{\boldsymbol{\vartheta}^k}^\top (P - \mathcal{A}^\top P \mathcal{A}) \boldsymbol{\vartheta}^k + 2{\boldsymbol{\vartheta}^k}^\top \mathcal{A}^\top P \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \\
&\qquad + u_1^\top(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \mathcal{B}_1^\top P \mathcal{B}_1 u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1) \\
&\quad \leq -{\boldsymbol{\vartheta}^k_1}^\top Q_1 \boldsymbol{\vartheta}^k_1 - p{\boldsymbol{\vartheta}^k_2}^\top Q_2 \boldsymbol{\vartheta}^k_2 + 2c_4 p \|\boldsymbol{\vartheta}^k_2\| \|\boldsymbol{\vartheta}^k_1\| \\
&\quad \overset{(a)}{\leq} -\begin{bmatrix} \|\boldsymbol{\vartheta}^k_1\| & \|\boldsymbol{\vartheta}^k_2\| \end{bmatrix} \begin{bmatrix} q_1 - p q_3 & -p c_4 \\ -p c_4 & p q_2 \end{bmatrix} \begin{bmatrix} \|\boldsymbol{\vartheta}^k_1\| \\ \|\boldsymbol{\vartheta}^k_2\| \end{bmatrix} \\
&\quad \overset{(b)}{\leq} -\tilde{q} \|\boldsymbol{\vartheta}^k\|^2,
\end{aligned}$$

where in (a) the terms are rearranged in matrix form by introducing the smallest (positive) eigenvalues q_1 and q_2 of the matrices Q_1 and Q_2, respectively, and, in (b), this positive-definite symmetric matrix is bounded by leveraging the Sylvester Criterion and choosing $\tilde{q} \in (0, q_1 q_2)$ and p such that $p < \min\{(q_1 q_2 - \tilde{q})/(c_4^2 + q_2 q_3), 1\}$. Thus, by leveraging this inequality, we can upper bound (46) as

$$\begin{aligned}
\Delta V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) &\leq J(\boldsymbol{\xi}^{k+1}, \varsigma(\boldsymbol{\xi}^{k+1})) - J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) \\
&\quad - \tilde{q} \|\boldsymbol{\vartheta}^k\|^2 + 2{\boldsymbol{\vartheta}^k}^\top \mathcal{A}^\top P \mathcal{B}_2 u_2(\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1}) \\
&\quad + 2u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}^k_1)^\top \mathcal{B}_1^\top P \mathcal{B}_2 u_2(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \\
&\quad + u_2(\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1})^\top \mathcal{B}_2^\top P \mathcal{B}_2 u_2(\boldsymbol{\xi}^k, \boldsymbol{\xi}^{k+1}) \\
&\overset{(a)}{\leq} J(\boldsymbol{\xi}^{k+1}, \varsigma(\boldsymbol{\xi}^{k+1})) - J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))
\end{aligned}$$

$$
\begin{aligned}
&- \tilde{q}\|\boldsymbol{\vartheta}^k\|^2 + \gamma 2c_5\|\boldsymbol{\vartheta}^k\| \left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\| \\
&+ \gamma c_6\|\boldsymbol{\vartheta}^k\|^2 + \gamma^2 c_7\|\boldsymbol{\vartheta}^k\|^2 + \gamma^2 c_8\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\|^2 \\
&+ \gamma^2 c_9\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\|\|\boldsymbol{\vartheta}^k\|, \qquad (49)
\end{aligned}
$$

where in (a) the triangle inequality, the Cauchy-Schwarz inequality, the trivial relations $\|\boldsymbol{\vartheta}_1^k\| \le \|\boldsymbol{\vartheta}^k\|$ and $\|\boldsymbol{\vartheta}_2^k\| \le \|\boldsymbol{\vartheta}^k\|$, and the results (44)-(45) have been exploited and we shortened the notation by introducing the constants

$$
\begin{aligned}
&c_5 := c_1 \left(\left\|\mathcal{A}^\top P\mathcal{B}_2\right\| + L_2\left\|R\right\|\left\|\mathcal{B}_1^\top P\mathcal{B}_2\right\|\right) \\
&c_6 := 2(c_2 + c_3)\left(\left\|\mathcal{A}^\top P\mathcal{B}_2\right\| + L_2\left\|R\right\|\left\|\mathcal{B}_1^\top P\mathcal{B}_2\right\|\right) \\
&c_7 := \left\|\mathcal{B}_1^\top P\mathcal{B}_1\right\| {L_2}^2\left\|R\right\|^2 + \left\|\mathcal{B}_2^\top P\mathcal{B}_2\right\|\left(c_2^2 + c_3^2 + 2c_2c_3\right) \\
&c_8 := \left\|\mathcal{B}_2^\top P\mathcal{B}_2\right\| c_1^2, \qquad c_9 := 2\left\|\mathcal{B}_2^\top P\mathcal{B}_2\right\| c_1(c_2 + c_3).
\end{aligned}
$$

Now we consider $J(\boldsymbol{\xi}^{k+1}, \varsigma(\boldsymbol{\xi}^{k+1})) - J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))$. By performing a Taylor expansion (see, e.g., [50, Theorem 2]) of $J(\boldsymbol{\xi}^{k+1}, \varsigma(\boldsymbol{\xi}^{k+1}))$ around $\boldsymbol{\xi}^k$ and by using (33a), we have

$$
\begin{aligned}
&J(\boldsymbol{\xi}^{k+1}, \varsigma(\boldsymbol{\xi}^{k+1})) - J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) \\
&= -\gamma\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\|^2 - \gamma\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))^\top u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \\
&\quad + L_{\boldsymbol{\xi}^k,2}(-\gamma(\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k))), \qquad (50)
\end{aligned}
$$

where we shortened the notation by using

$$
\begin{aligned}
u_3(\boldsymbol{\xi}^k,\boldsymbol{\vartheta}^k) := &\begin{bmatrix}\mathrm{D}_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}_1^k) \\ \mathrm{D}_2(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}_1^k)\end{bmatrix} - \begin{bmatrix}\mathrm{D}_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k)) \\ \mathrm{D}_2(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k))\end{bmatrix} \\
&+ \nabla\bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k)R\boldsymbol{\vartheta}_2^k + \nabla\bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k)u_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + R\boldsymbol{\vartheta}_1^k), \qquad (51)
\end{aligned}
$$

and the remainder term $L_{\boldsymbol{\xi}^k,2}(-\gamma(\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k)))$ given by

$$
\begin{aligned}
&L_{\boldsymbol{\xi}^k,2}(-\gamma(\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k))) := \\
&\quad \sum_{|\alpha|=2} \partial^\alpha J\left(\boldsymbol{\xi}^k - \tau\gamma(\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k))\right) \\
&\quad \frac{-\gamma(\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k))}{\alpha!},
\end{aligned}
$$

for some $\tau \in (0,1)$. By using the definition of u_3 (cf. (51)), the definitions in (42), and the bound (44), we note that

$$
\left\|u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k)\right\| \le (L_2 + L_3(1 + L_2))\left\|R\right\|\left\|\boldsymbol{\vartheta}^k\right\|. \qquad (52)
$$

By Lipschitz continuity of the gradients of J, we know that $\partial^\alpha J(\boldsymbol{\xi}, \sigma(\boldsymbol{\xi})) \le L$ for all $\boldsymbol{\xi} \in B_{\boldsymbol{\xi}}$ and $|\alpha| = 2$. Thus, by applying [50, Corollary 1], we can bound $L_{\boldsymbol{\xi}^k,2}(-\gamma(\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k)))$ as

$$
\begin{aligned}
&L_{\boldsymbol{\xi}^k,2}(-\gamma(\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k))) \\
&\quad \le \gamma^2\frac{L}{2}\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k)\right\|^2 \\
&\quad \overset{(a)}{\le} \gamma^2 c_{10}\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|^2 + \gamma^2 c_{11}\|\boldsymbol{\vartheta}^k\|^2 \\
&\quad\quad + \gamma^2 c_{12}\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|\|\boldsymbol{\vartheta}^k\|, \qquad (53)
\end{aligned}
$$

where in (a) we exploited the square, applied the bound (52), and introduced $c_{10} := \frac{L}{2}$, $c_{11} := (L_2 + L_3(1 + L_2))^2\left\|R\right\|^2$, $c_{12} := 2c_{10}\sqrt{c_{11}}$. Hence, we can use (53) to bound (50) as

$$
\begin{aligned}
&J(\boldsymbol{\xi}^{k+1}, \varsigma(\boldsymbol{\xi}^{k+1})) - J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) \\
&\le -\gamma\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|^2 - \gamma\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))^\top u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \\
&\quad + \gamma^2 c_{10}\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|^2 \\
&\quad + \gamma^2 c_{11}\|\boldsymbol{\vartheta}^k\|^2 + \gamma^2 c_{12}\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|\|\boldsymbol{\vartheta}^k\| \\
&\overset{(a)}{\le} -\gamma\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\|^2 + \gamma^2 c_{10}\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\|^2 \qquad (54) \\
&\quad + \gamma^2 c_{11}\|\boldsymbol{\vartheta}^k\|^2 + (\gamma c_{13} + \gamma^2 c_{12})\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\|\|\boldsymbol{\vartheta}^k\|,
\end{aligned}
$$

where in (a) we use the local Lipschitz continuity of $\bar{\boldsymbol{\varphi}}$, the Cauchy-Schwarz inequality, and the result (52) to write the bound $-\gamma\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))^\top u_3(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \le \gamma c_{13}\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\|\|\boldsymbol{\vartheta}^k\|$ with

$$
c_{13} := (L_2 + L_3(1 + L_2))\left\|R\right\|.
$$

Now, we can use (54) to upper bound (49) as

$$
\begin{aligned}
&\Delta V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \\
&\le -\gamma\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|^2 + \gamma^2 c_{10}\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|^2 \\
&\quad + \gamma^2 c_{11}\|\boldsymbol{\vartheta}^k\|^2 + (\gamma c_{13} + \gamma^2 c_{12})\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|\|\boldsymbol{\vartheta}^k\| \\
&\quad - \tilde{q}\|\boldsymbol{\vartheta}^k\|^2 + \gamma 2c_5\|\boldsymbol{\vartheta}^k\|\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\| + \gamma c_6\|\boldsymbol{\vartheta}^k\|^2 \\
&\quad + \gamma^2 c_7\|\boldsymbol{\vartheta}^k\|^2 + \gamma^2 c_8\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|^2 \\
&\quad + \gamma^2 c_9\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\|\|\boldsymbol{\vartheta}^k\| \\
&\overset{(a)}{\le} -\begin{bmatrix}\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\| \\ \|\boldsymbol{\vartheta}^k\|\end{bmatrix}^\top M(\gamma)\begin{bmatrix}\left\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\right\| \\ \|\boldsymbol{\vartheta}^k\|\end{bmatrix}, \qquad (55)
\end{aligned}
$$

where in (a) we rearranged the inequality via

$$
M(\gamma) := \begin{bmatrix}\gamma - \gamma^2 c_{14} & -\gamma c_{15} - \gamma^2 c_{16} \\ -\gamma c_{15} - \gamma^2 c_{16} & \tilde{q} - \gamma c_{17} - \gamma^2 c_{18}\end{bmatrix},
$$

where $c_{14} := c_{10} + c_8$, $c_{15} := c_5 + c_{13}$, $c_{16} := \frac{c_{12}+c_9}{2}$, $c_{17} := c_6$, and $c_{18} := c_{11} + c_7$. By Sylvester Criterion, we know that $M(\gamma)$ (cf. (55)) is positive definite if and only if

$$
\begin{cases}\gamma > & \gamma^2 c_{14} \\ \gamma\tilde{q} > & (\gamma c_{15} + \gamma^2 c_{16})^2 + \gamma^2(c_{17} + c_{14}\tilde{q}) \\ & \quad -\gamma^3(c_{18} + c_{14}c_{17}) - \gamma^4 c_{14}c_{18}.\end{cases} \qquad (56)
$$

Note that the terms on the left-hand sides of (56) are linear in γ, while the ones on the right have higher order terms. Then, there exists $\bar{\gamma} > 0$ such that for any $\gamma \in (0, \bar{\gamma})$ conditions (56) are satisfied. Hence, $M(\gamma)$ is positive definite and we bound (55) as

$$
\Delta V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \le -\mathrm{m}(\|\nabla J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k))\|^2 + \|\boldsymbol{\vartheta}^k\|^2), \qquad (57)
$$

where $\mathrm{m} > 0$ is the smallest eigenvalue of $M(\gamma)$. Inequality (57) implies that, for all $(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}_1^k, \boldsymbol{\vartheta}_2^k) \in \Omega_1^{\mathrm{c}} \times B_2^{r_0}$, it holds

$$
V(\boldsymbol{\xi}^{k+1}, \boldsymbol{\vartheta}^{k+1}) \le V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k). \qquad (58)
$$

By using (i) $p{\boldsymbol{\vartheta}_2^k}^\top P_2\boldsymbol{\vartheta}_2^k \ge 0$ for any $\boldsymbol{\vartheta}^k$ and (ii) $p \in (0,1)$, the result (58) leads to

$$
\begin{aligned}
&J(\boldsymbol{\xi}^{k+1}, \varsigma(\boldsymbol{\xi}^{k+1})) + {\boldsymbol{\vartheta}_1^{k+1}}^\top P_1\boldsymbol{\vartheta}_1^{k+1} \\
&\le J(\boldsymbol{\xi}^k, \varsigma(\boldsymbol{\xi}^k)) + {\boldsymbol{\vartheta}_1^k}^\top P_1\boldsymbol{\vartheta}_1^k + \|P_2\|\, r_0^2,
\end{aligned}
$$

which guarantees $(\boldsymbol{\xi}^{k+1}, \boldsymbol{\vartheta}_1^{k+1}) \in \Omega_1^{\mathrm{c}}$ (see (43)). This fact, combined with the variables' initialization (cf. (37)), guarantees that $(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}_1^k) \in \Omega_1^{\mathrm{c}}$ for all $k \in \mathbb{N}$, which, in turns, allows us to claim that (57) is verified for all $k \in \mathbb{N}0$. However, we point out that the inequality (57) does not implies the negative definiteness of $\Delta V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k)$ because the right-hand side of (57) is null on the subspace $E \subset \mathbb{R}^{(n+m)T} \times \mathbb{R}^{2d}$ defined as

$$E := \{\mathrm{col}(\boldsymbol{\xi}, \boldsymbol{\vartheta}) \in \mathbb{R}^{(n+m)T} \times \mathbb{R}^{2d} \mid \|\nabla J(\boldsymbol{\xi}, \varsigma(\boldsymbol{\xi}))\| = 0, \boldsymbol{\vartheta} = 0\}. \tag{59}$$

Studying the system (33) restricted to the subspace E it holds

$$\boldsymbol{\xi}^{k+1}\Big|_{(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \in E} = \boldsymbol{\xi}^k, \qquad \boldsymbol{\vartheta}^{k+1}\Big|_{(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \in E} = \boldsymbol{\vartheta}^k. \tag{60a}$$

By inspecting system (60), we can conclude that the E is invariant for system (33). Thus, we can conclude by LaSalle's Invariance Principle (cf. [Theorem 3.7] [51]) that

$$\lim_{k\to\infty} \mathrm{dist}\left(\begin{bmatrix}\boldsymbol{\xi}^k \\ \boldsymbol{\vartheta}^k\end{bmatrix}, E\right) = 0.$$

Turning out to the original coordinates $(\boldsymbol{\alpha}, \boldsymbol{\mu}, \mathbf{s}, \mathbf{y})$, it follows that the trajectories of the system (30), i.e., the sequence produced by Algorithm 2, converge within the subspace

$$\mathcal{X} := \Bigg\{(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star, \mathbf{s}^\star, \mathbf{y}^\star) \mid \nabla J(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star, \varsigma(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)) = 0, \\ \mathbf{s}^\star = \frac{\mathbf{1}\mathbf{1}^\top}{N}\bar{\varphi}(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star),\ \mathbf{y}^\star = \frac{\mathbf{1}\mathbf{1}^\top}{N}\mathrm{D}_3(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star, \mathbf{s}^\star)\Bigg\}. \tag{61}$$

Namely, the sequence $\{\boldsymbol{\alpha}^k, \boldsymbol{\mu}^k\}$ converges into the set of stationary points of $J(\boldsymbol{\alpha}, \boldsymbol{\mu}, \varsigma(\boldsymbol{\alpha}, \boldsymbol{\mu}))$.

***Step (iii)**: first-order necessary conditions for optimality:* let us consider any point $(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)$ such that $\nabla J(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star, \varsigma(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)) = 0$. Then, let us consider the corresponding state-input trajectories $(\mathbf{x}^\star, \mathbf{u}^\star)$ computed projecting $(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)$ and the costate vectors $\boldsymbol{\lambda}^\star \in \mathbb{R}^{nT}$. Specifically, for all $i = 1, \ldots, N$, we calculate $(\mathbf{x}_i^\star, \mathbf{u}_i^\star) = \mathcal{P}_i(\boldsymbol{\alpha}_i^\star, \boldsymbol{\mu}_i^\star)$ via (17). As for $\boldsymbol{\lambda}^\star \in \mathbb{R}^{nT}$, each $\boldsymbol{\lambda}_i^\star \in \mathbb{R}^{n_i T}$ is computed via (15b). Next, we show that the tuple $(\mathbf{x}^\star, \mathbf{u}^\star, \boldsymbol{\lambda}^\star)$ satisfies the first order necessary optimality conditions for problem (5). Hence, for all $t \in [0, T-1]$, let the Hamiltonian function of problem (5) given by

$$H_t(x_t, u_t, \lambda_{t+1}) = \sum_{i=1}^{N} H_{i,t}(x_{i,t}, u_{i,t}, \lambda_{i,t+1}), \tag{62}$$

where

$$H_{i,t}(x_{i,t}, u_{i,t}, \lambda_{i,t+1}) := \ell_{i,t}(x_{i,t}, u_{i,t}) + g_{i,t}(\sigma_t(x_t, u_t)) \\ + f_i(x_{i,t}, u_{i,t})^\top \lambda_{i,t+1}.$$

Next we show that, for all $i = 1, \ldots, N$,

$$\begin{bmatrix} \nabla_2 H_{1,t}(x_{1,t}^\star, u_{1,t}^\star, \lambda_{1,t+1}^\star) \\ \vdots \\ \nabla_2 H_{N,t}(x_{N,t}^\star, u_{N,t}^\star, \lambda_{N,t+1}^\star) \end{bmatrix} = 0,$$

and

$$\lambda_{i,t}{}^\star = \nabla_1 H_{i,t}(x_{i,t}^\star, u_{i,t}^\star, \lambda_{t+1}^\star), \tag{63}$$

with terminal condition $\lambda_{i,T}^\star = \nabla \ell_{i,T}(x_{i,T}^\star, \sigma_T(x_T^\star))$.

Leveraging projection step in (17), the pair $(\mathbf{x}_i^\star, \mathbf{u}_i^\star)$ satisfies the dynamics (5b) by construction, i.e., it is a trajectory.

Now, let us define the shorthand for the linearization of the cost and the dynamics about the trajectory $(\mathbf{x}_i^*, \mathbf{u}_i^*)$

$$A_{i,t}^\star := \nabla_1 f_i(x_{i,t}^\star, u_{i,t}^\star)^\top, \quad B_{i,t}^\star := \nabla_2 f_i(x_{i,t}^\star, u_{i,t}^\star)^\top, \\ K_{i,t}^\star := \nabla \kappa_{i,t}(e_{i,t}^\star)^\top, \qquad \text{where } e_{i,t}^\star := \alpha_{i,t}^\star - x_{i,t}^\star.$$

Notice that the sequence $\{\mathbf{s}^k, \mathbf{y}^k\}$ generated by Algorithm 2 converges to the subspace defined by (61), namely, for all $i = 1, \ldots, N$, the proxies in (15a) are equal to

$$a_{i,t}^\star = \nabla_1 \ell_{i,t}(x_{i,t}^\star, u_{i,t}^\star, \sigma_t(x_{i,t}^\star, u_{i,t}^\star)) \\ + \tfrac{1}{N}\nabla_1 \varphi_{i,t}(x_{i,t}^\star, u_{i,t}) \sum_{j=1}^{N} \nabla_3 \ell_{j,t}(x_{j,t}^\star, u_{j,t}^\star, \sigma_t(x_t^\star, u_t^\star)) \tag{64a}$$

$$b_{i,t}^\star = \nabla_2 \ell_{i,t}(x_{i,t}^\star, u_{i,t}^\star) \\ + \tfrac{1}{N}\nabla_2 \varphi_{i,t}(x_{i,t}^\star, u_{i,t}) \sum_{j=1}^{N} \nabla_3 \ell_{j,t}(x_{j,t}^\star, u_{j,t}^\star, \sigma_t(x_t^\star, u_t^\star)) \tag{64b}$$

and

$$a_{i,T}^\star = \nabla_1 \ell_{i,T}(x_{i,T}^\star, \sigma_T(x_T^\star)) \\ + \tfrac{1}{N}\nabla \varphi_{i,T}(x_{i,T}^\star) \sum_{j=1}^{N} \nabla_2 \ell_{j,T}(x_{j,T}^\star, \sigma_T(x_T^\star)). \tag{64c}$$

Then, we can define $\boldsymbol{\lambda}_i^\star$ as the stack of the costate vectors $\lambda_{i,t}^\star \in \mathbb{R}^n$, obtained from the adjoint equation (15b) evaluated at $(\boldsymbol{\alpha}_i^\star, \boldsymbol{\mu}_i^\star)$, i.e., for all $t \in [T-1, 0]$

$$\lambda_{i,t}^\star = \left(A_{i,t}^\star - B_{i,t}^\star K_{i,t}^\star\right)^\top \lambda_{i,t+1}^\star + a_{i,t}^\star - K_{i,t}^{\star,\top} b_{i,t}^\star, \tag{65}$$

with terminal condition $\lambda_{i,T}^\star = a_{i,T}^\star$.

Equation (65) corresponds to the gradient with respect to $x_{i,t}$ of the Hamiltonian evaluated along the trajectory $(\mathbf{x}_i^\star, \mathbf{u}_i^\star)$, i.e., the first order necessary condition for optimality (63) holds by construction. Now, we recall that

$$\nabla_2 H_{i,t}(x_{i,t}^\star, u_{i,t}^\star, \lambda_{i,t+1}^\star) = b_{i,t}^\star + B_{i,t}^{\star\top} \lambda_{i,t+1}^\star, \tag{66}$$

for all $i = 1, \ldots, N$ and $t \in [0, T-1]$. The right hand side of (66) corresponds to the derivative of $J(\cdot, \cdot, \varsigma(\cdot, \cdot))$ with respect to $\mu_{i,t}$ evaluated at $(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)$. Notice that, the costate $\boldsymbol{\lambda}_i^\star$, solution of (65), coincides with the Lagrange multipliers associated to the dynamics constraints at $(\mathbf{x}^\star, \mathbf{u}^\star)$. In light of the first part of the proof, the right hand side of (66) is equal to zero. Therefore, the first-order necessary conditions for optimality are satisfied by the trajectory $(\mathbf{x}_i^*, \mathbf{u}_i^*)$, for all i. Hence, for any point $(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)$ such that $\nabla J((\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star), \varsigma(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)) = 0$, it holds that the corresponding point $(\mathbf{x}^\star, \mathbf{u}^\star)$ is such that $(\mathbf{x}^\star, \mathbf{u}^\star) \in \Xi^\star$. This concludes the proof.

Notice that, any stationary state-input curve $(\boldsymbol{\alpha}^\star, \boldsymbol{\mu}^\star)$ corresponds to a trajectory $(\mathbf{x}^\star, \mathbf{u}^\star)$ satisfying the first-order optimality conditions for (5).

Remark 7: If the reduced cost $J(\cdot, \cdot, \varsigma(\cdot, \cdot))$ (cf. (23)) is strongly convex, then problem (23) has a unique minimizer

$\xi^\star \in \mathbb{R}^{(n+m)T}$. Thus, in this case, one can apply the well-known Polyak-Lojasiewicz (PL) inequality to further bound the right-hand side of (57) as

$$\Delta V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k) \leq -\tilde{\mathrm{m}} V(\boldsymbol{\xi}^k, \boldsymbol{\vartheta}^k), \tag{67}$$

for some $\tilde{\mathrm{m}} > 0$ depending on m and the strong convexity parameter. In turn, global exponential stability of $\mathrm{col}(\xi^\star, 0)$ holds for (33) and thus linear convergence of Algorithm 2 toward the (unique) solution to optimal control problem (5). Notice that PL inequality holds true in the neighborhood of local minima of (5) that satisfy the second order sufficient conditions for optimality. □

V. Numerical Simulations

In this section, we show simulations of the proposed algorithm in a multi-agent cooperative robotics setting. We consider a team of $N = 50$ quadrotors in a robotic surveillance scenario. Each robot has to patrol a pre-defined area trying to stay as close as possible to a given reference position while keeping the formation barycenter over a (moving) target. Each quadrotor i has a continuous-time dynamics described by

$$\ddot{p}_{x,i} = \tfrac{u_{i,1}+u_{i,2}}{M_i}\sin\theta_i, \qquad \ddot{p}_{y,i} = \tfrac{u_{i,1}+u_{i,2}}{M_i}\cos\theta_i - g$$
$$\ddot{\theta}_i = \tfrac{L_i}{2J_i}(u_{i,1} - u_{i,2}),$$

where $(p_{x,i}, p_{y,i})$ is the position of the robot in the $x-y$ plane, θ_i its orientation, $u_{i,1}$ and $u_{i,2}$ are the thrusts applied to the right and left motor respectively. Agents have heterogeneous dynamics. The mechanical parameters are randomly assigned to each agent, i.e., for all i, $M_i \in \{1, 5, 10\}$ [kg], $L_i \in \{0.5, 1.0\}$ [m], while $J_i = M_i L_i^2/12$ [$\mathrm{m}\cdot\mathrm{kg}^2$].

For each agent i, the discrete-time dynamics, with state $x_{i,t} \in \mathbb{R}^6$ with $x_{i,t} := (p_{x,i,t}, \dot{p}_{x,i,t}, p_{y,i,t}, \dot{p}_{y,i,t}, \theta_{i,t}, \dot{\theta}_{i,t})^\top \in \mathbb{R}^6$ and input $u_{i,t} := (u_{i,1,t}, u_{i,2,t}) \in \mathbb{R}^2$ is obtained via a Runge-Kutta integrator of order 4 with sampling period $\delta = 0.05$ seconds. Hence, we have $n_i = 6$, $m_i = 2$ and $n = 60$, $m = 20$. The time horizon is $t_f = 30$ $[s]$, resulting in $T = 600$ samples. The network is modeled as a randomly generated Erdós-Rényi graph with edge probability $p = 0.5$. The weighted adjacency matrix $\mathcal{W}$ is generated via the Metropolis-Hastings rule using DISROPT [52]. Each quadrotor i wants to track its reference signal $(\mathbf{x}_i^{\text{ref}}, \mathbf{u}_i^{\text{ref}})$. The team also aims to maintain the formation barycenter, modeled via $\sigma_t(x_t, u_t)$, as close as possible to the target located at $b_t^{\text{target}} \in \mathbb{R}^2$ at time t. Each stage cost is defined as $\ell_{i,t}(x_{i,t}, u_{i,t}, \sigma_t(x_t)) = \|x_{i,t} - x_{i,t}^{\text{des}}\|_{Q_{i,t}}^2 + \|u_{i,t} - u_{i,t}^{\text{des}}\|_{R_{i,t}}^2 + \|\sigma(x_t) - b_t\|_{Q_{i,t}^\sigma}^2$ where $Q_{i,t} \in \mathbb{R}^{n_i \times n_i}$, $R_{i,t} \in \mathbb{R}^{m_i \times m_i}$, $Q_{i,t}^\sigma \in \mathbb{R}^{q \times q}$, and $\|\cdot\|_M$ denotes the L2 norm weighted by matrix M. The terminal cost is defined by $\ell_{i,T}(\cdot) = \|x_{i,T} - x_{i,T}^{\text{des}}\|_{Q_{i,f}}^2 + \|\sigma(x_T) - b_T\|_{Q_{i,f}^\sigma}^2$ where $Q_{i,f} \in \mathbb{R}^{n_i \times n_i}$, and $Q_{i,f}^\sigma \in \mathbb{R}^{q \times q}$. The aggregate variable is defined as $\sigma(x_t) := \sum_{i=1}^N H_i x_{i,t}$ where $H_i = [I_2 \; 0_{2\times 4}]$ for all $i = 1, \ldots, N$. We set $Q_{i,t} = \mathrm{diag}(10, 1, 10, 1, 0.01, 0.01)$, $Q_{i,f} = Q_{i,t}$, $R_{i,t} = \mathrm{diag}(0.1, 0.1)$, $Q_{i,t}^\sigma = \mathrm{diag}(100, 100)$, $Q_{i,f}^\sigma = 10\, Q_{i,t}^\sigma$. The reference signal consists of a randomly generated fixed position in the $x-y$ plane, while the reference input is the equilibrium thrust. Formally, we define $(x_{i,t}^{\text{des}}, u_{i,t}^{\text{des}})$ as $p_{x,i,t}^{\text{des}} = \bar{p}_x$, $p_{y,i,t}^{\text{des}} = \bar{p}_y$, $u_{i,t}^{\text{des}} = \frac{1}{2} M_i g \mathbf{1}_{m_i}$, where $\bar{p}_x$ and $\bar{p}_y$ are drawn randomly from an uniform distribution between $\bar{p}_{x,\min} = -5$, $\bar{p}_{x,\max} = 5$, and $\bar{p}_{y,\min} = -5$, $\bar{p}_{y,\max} = 5$, respectively. All the other reference states are set identically equal to 0. The target position in $b_t \in \mathbb{R}^2$ is defined via a second-order polynomial between the checkpoints $(0,0)$, $(1,1)$ and $(1, 1.5)$ reached at the time $0, T/2$ and T respectively.

Algorithm 2 is run with a fixed stepsize $\gamma = 10^{-4}$. The feedback policy (13) is approximated, at each iteration, as a linear feedback policy where the sequence of gains $K_{i,t} = \nabla\kappa_{i,t}(\alpha_{i,t} - x_{i,t})^\top$ are chosen by solving an LQR problem involving the linearization of each agent's dynamics about the current trajectory $(\mathbf{x}_i^k, \mathbf{u}_i^k)$ with costs $Q_{i,t}^{\text{reg}} = Q_{i,t}$, $R_{i,t}^{\text{reg}} = R_{i,t}$, and a terminal cost $Q_{i,f}^{\text{reg}}$ obtained from the associated algebraic solution of the Riccati equation. The initial trajectory $(\mathbf{x}^0, \mathbf{u}^0)$ is chosen as the equilibrium trajectory at the local reference signal for each agent. The positions associated to the optimal trajectories $(\mathbf{x}_i^\star, \mathbf{u}_i^\star)$ of the 50 quadrotors are depicted in Fig. 1. In Fig. 2 and 3 we compare the cost and the input trajectories of the distributed algorithm with the ones computed by a centralized solver. Finally, from Fig. 4, we notice that the trackers converge to the desired value.

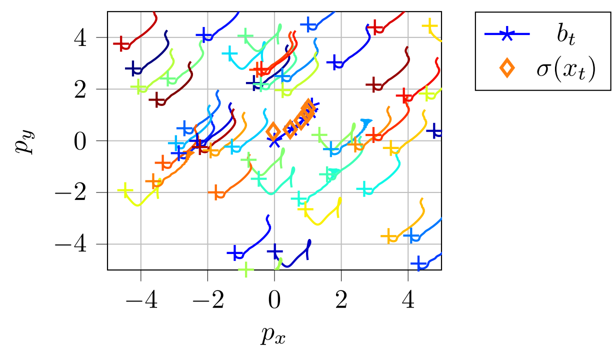

Fig. 1. Quadrotor positions in the x-y plane: optimal trajectories $(\mathbf{x}_i^\star, \mathbf{u}_i^\star)$ (solid), reference position $\mathbf{x}_i^{\text{des}}$ (crosses), target position $\boldsymbol{b_t}$ (starred blue), formation barycenter $\boldsymbol{\sigma_t(x_t^\star)}$ (orange diamonds).

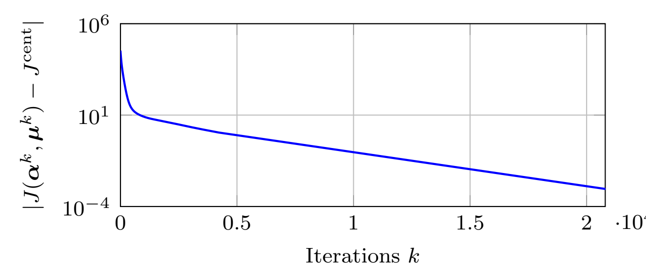

Fig. 2. Error between the cost $\boldsymbol{J(\alpha^k, \mu^k)}$ with respect to the cost $\boldsymbol{J^{\text{cent}}}$ computed by a centralized solver.

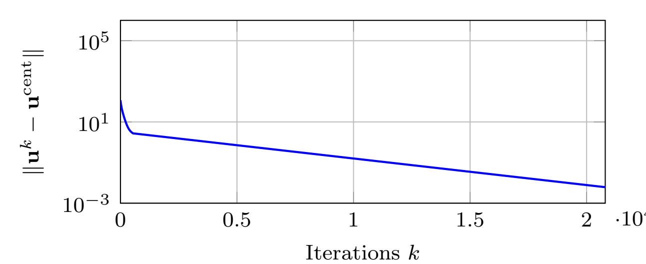

Fig. 3. Error between the input sequence $\mathbf{u}^k$ with respect to the input sequence $\mathbf{u}^{\text{cent}}$ computed by a centralized solver.

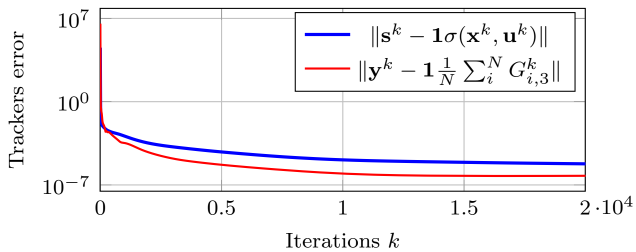

Fig. 4. Tracking error for the tracking variables $\mathbf{s}^k$, $\mathbf{y}^k$. We define as $G_{i,3}^k := G_{i,3}(\mathbf{x}_i^k, \mathbf{u}_i^k, \sigma(\mathbf{x}^k, \mathbf{u}^k))$.

VI. Conclusions

In this paper, we considered nonlinear optimal control problems over networks of multi-agent systems with aggregative structure. To address this framework, we designed a distributed method combining a feedback structure with a first-order optimization mechanism in each agent of the network. Moreover, we properly introduced a set of local trackers using consensus-based dynamics to locally reconstruct missing (global) information. The overall methodology has been analyzed by using tools from system theory. In detail, we proved that our distributed scheme generates a sequence of trajectories converging to the set whose belonging points satisfy the first-order necessary conditions for optimality. Finally, we presented the algorithm capabilities on a distributed optimal control example for a target surveillance task.

Appendix

A. Proof of Lemma 1

First of all, we rewrite the equality constraints represented by the nonlinear dynamics in (14b) and the closed-loop structure of the control input (14c) in its implicit form $h : \mathbb{R}^{n_iT} \times \mathbb{R}^{m_iT} \times \mathbb{R}^{n_iT} \times \mathbb{R}^{m_iT} \to \mathbb{R}^{n_iT+m_iT}$ defined as

$$h_i(\mathbf{x}_i, \mathbf{u}_i, \boldsymbol{\alpha}_i, \boldsymbol{\mu}_i) := \mathrm{col}(f_i(x_{i,0}, u_{i,0}) - x_{i,1}, \dots, f_i(x_{i,T-1}, u_{i,T-1}) - x_{i,T}, \mu_{i,0} + \kappa_{i,t}(\alpha_{i,0} - x_{i,0}) - u_{i,0}, \dots, \mu_{i,T-1} + \kappa_{i,t}(\alpha_{i,T-1} - x_{i,T-1}) - u_{i,T-1}).$$

We can now introduce an auxiliary function $\mathcal{L} : \mathbb{R}^{nT} \times \mathbb{R}^{mT} \times \mathbb{R}^{nT} \times \mathbb{R}^{mT} \times \mathbb{R}^{(n+m)T} \to \mathbb{R}$, defined as

$$\mathcal{L}(\mathbf{x}, \mathbf{u}, \boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{\lambda}) := \sum_{i=1}^N \ell_i(\mathbf{x}_i, \mathbf{u}_i, \boldsymbol{\sigma}(\mathbf{x}, \mathbf{u})) + h_i(\mathbf{x}_i, \mathbf{u}_i, \boldsymbol{\alpha}_i, \boldsymbol{\mu}_i)^\top \boldsymbol{\lambda}_i,$$

where, for all $i = 1, \dots, N$, the (multiplier) vector $\boldsymbol{\lambda}_i \in \mathbb{R}^{n_iT+m_iT}$ is arranged as $\boldsymbol{\lambda}_i := \mathrm{col}(\lambda_{i,1}, \dots, \lambda_{i,T}, \tilde{\lambda}_{i,1}, \dots, \tilde{\lambda}_{i,T})$, with $\lambda_{i,t} \in \mathbb{R}^n$ and $\tilde{\lambda}_{i,t} \in \mathbb{R}^m$, for all $t \in [1, T]$. By construction, for any $(\boldsymbol{\alpha}, \boldsymbol{\mu}) \in \mathbb{R}^{nT} \times \mathbb{R}^{mT}$, it holds $h_i(\mathcal{P}_i^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \mathcal{P}_i^{\boldsymbol{\mu}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \boldsymbol{\alpha}_i, \boldsymbol{\mu}_i) = 0$. Hence, the auxiliary function $\mathcal{L}(\cdot)$ enjoys the property

$$\begin{aligned}
&\mathcal{L}(\mathcal{P}^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}, \boldsymbol{\mu}), \mathcal{P}^{\boldsymbol{\mu}}(\boldsymbol{\alpha}, \boldsymbol{\mu}), \boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{\lambda}) \\
&= \sum_{i=1}^N \ell_i(\mathcal{P}_i^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i), \mathcal{P}_i^{\boldsymbol{\mu}}(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i, \boldsymbol{\sigma}(\mathcal{P}^{\boldsymbol{\alpha}}(\boldsymbol{\alpha}, \boldsymbol{\mu}), \mathcal{P}^{\boldsymbol{\mu}}(\boldsymbol{\alpha}, \boldsymbol{\mu})))) \\
&= \sum_{i=1}^N J_i(\boldsymbol{\alpha}_i, \boldsymbol{\mu}_i, \boldsymbol{\varsigma}(\boldsymbol{\alpha}, \boldsymbol{\mu})), \qquad (68)
\end{aligned}$$

for *any* $(\boldsymbol{\alpha}, \boldsymbol{\mu})$ and $\boldsymbol{\lambda} \in \mathbb{R}^{nT+mT}$. Building on (68), one can simplify the computation of the gradient of J evaluated at a given $(\boldsymbol{\alpha}^k, \boldsymbol{\mu}^k)$ with a proper choice of $\boldsymbol{\lambda}$ (see [41] for further details). The proof follows by noticing that this choice allows to compute the t-th components of the descent directions

$$[D_{\boldsymbol{\alpha}_i}^k]_t = -b_{i,t}^k - B_{i,t}^k \lambda_{i,t+1}^k, \quad [D_{\boldsymbol{\mu}_i}^k]_t = K_{i,t}^{k,\top} \Delta\mu_{i,t}^k, \qquad (69a)$$

via backward integration from $t = T-1, \dots, 0$,

$$\lambda_{i,t}^k = (A_{i,t}^k - K_{i,t}^k B_{i,t}^k)^\top \lambda_{i,t+1}^k + a_{i,t}^k - K_{i,t}^{k,\top} \hat{b}_{i,t}^k, \qquad (69b)$$

with initial condition $\lambda_{i,T}^k = a_T^k$, where

$$\begin{aligned}
a_{i,t}^k &= \nabla_1 \ell_{i,t}(x_{i,t}^k, u_{i,t}^k, \sigma_t(x_t^k, u_t^k)) + && (70a)\\
&\quad + \tfrac{1}{N} \nabla_1 \varphi_{i,t}(x_{i,t}^k, u_{i,t}^k) \sum_{j=1}^N \nabla_3 \ell_{j,t}(x_{j,t}^k, u_{j,t}^k, \sigma_t(x_t^k, u_t^k)) \\
b_{i,t}^k &= \nabla_2 \ell_{i,t}(x_{i,t}^k, u_{i,t}^k) + && (70b)\\
&\quad + \tfrac{1}{N} \nabla_2 \varphi_{i,t}(x_{i,t}^k, u_{i,t}^k) \sum_{j=1}^N \nabla_3 \ell_{j,t}(x_{j,t}^k, u_{j,t}^k, \sigma_t(x_t^k, u_t^k)). \\
a_{i,T}^k &= \nabla_1 \ell_{i,T}(x_{i,T}^k, \sigma_T(x_T^k)) + && (70c)\\
&\quad + \tfrac{1}{N} \nabla \varphi_{i,T}(x_{i,T}^k) \sum_{j=1}^N \nabla_2 \ell_{j,T}(x_{j,T}^k, \sigma_t(x_T^k)).
\end{aligned}$$

B. Proof of Lemma 3

Let $\boldsymbol{\xi}_i^k := \mathrm{col}(\boldsymbol{\alpha}_i^k, \boldsymbol{\mu}_i^k)$ for all i and $\boldsymbol{\xi}^k := \mathrm{col}(\boldsymbol{\xi}_1^k, \dots, \boldsymbol{\xi}_N^k)$. Then, by using the notation in (31), we rewrite (30) as

$$\begin{aligned}
\boldsymbol{\xi}^{k+1} &= \boldsymbol{\xi}^k - \gamma \begin{bmatrix} \mathrm{D}_1(\boldsymbol{\xi}^k, \mathbf{s}^k) \\ \mathrm{D}_2(\boldsymbol{\xi}^k, \mathbf{s}^k) \end{bmatrix} - \gamma \nabla \bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k) \mathbf{y}^k && (71a)\\
\mathbf{s}^{k+1} &= W\mathbf{s}^k + \bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^{k+1}) - \bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k) && (71b)\\
\mathbf{y}^{k+1} &= W\mathbf{y}^k + \mathrm{D}_3(\boldsymbol{\xi}^{k+1}, \mathbf{s}^{k+1}) - \mathrm{D}_3(\boldsymbol{\xi}^k, \mathbf{s}^k), && (71c)
\end{aligned}$$

where, with a slight abuse of notation, we joined the first two arguments of the operators D_1 and D_2. Let us introduce the new coordinates $\mathbf{w}^k, \mathbf{z}^k \in \mathbb{R}^{Nq(T+1)}$ defined as

$$\mathbf{w}^k := \mathbf{s}^k - \bar{\boldsymbol{\varphi}}(\mathbf{s}^k), \quad \mathbf{z}^k := \mathbf{y}^k - \mathrm{D}_3(\boldsymbol{\xi}^k, \mathbf{s}^k),$$

which allow us to rewrite (71) as

$$\begin{aligned}
\boldsymbol{\xi}^{k+1} &= \boldsymbol{\xi}^k - \gamma D_{\boldsymbol{\xi}}(\boldsymbol{\xi}^k, \bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k) + \mathbf{w}^k) - \gamma \nabla \bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k) \mathbf{z}^k && (72a)\\
\mathbf{w}^{k+1} &= W\mathbf{w}^k + (W - I)\bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k) && (72b)\\
\mathbf{z}^{k+1} &= W\mathbf{z}^k + (W - I)\mathrm{D}_3(\boldsymbol{\xi}^k, \bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k) + \mathbf{w}^k), && (72c)
\end{aligned}$$

where the operator $D_{\boldsymbol{\xi}}$ is defined as

$$D_{\boldsymbol{\xi}}(\boldsymbol{\xi}, \mathrm{s}) := \begin{bmatrix} \mathrm{D}_1(\boldsymbol{\xi}, \mathrm{s}) \\ \mathrm{D}_2(\boldsymbol{\xi}, \mathrm{s}) \end{bmatrix} + \nabla \bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}) \frac{\mathbf{1}\mathbf{1}^\top}{N} \mathrm{D}_3(\boldsymbol{\xi}, \mathrm{s}). \qquad (73)$$

Note that the initial conditions of Theorem 1 imply $\mathbf{w}^0 = \mathbf{z}^0 = 0$. Let $\mathcal{S} \subset \mathbb{R}^{(n+m)T} \times \mathbb{R}^{q(T+1)} \times \mathbb{R}^{q(T+1)}$ be defined as

$$\mathcal{S} := \{(\boldsymbol{\xi}, \mathbf{w}, \mathbf{z}) \mid \mathbf{1}^\top \mathbf{w}^0 = 0, \mathbf{1}^\top \mathbf{z}^0 = 0\}. \qquad (74)$$

Since $\mathcal{W}$ is doubly stochastic (cf. Assumption 2), it holds $\mathbf{1}^\top W = \mathbf{1}^\top$ and, thus, $\mathbf{1}^\top (W - I) = 0$. This property allows for verifying that $\mathcal{S}$ is invariant for (72). Moreover, W has all the eigenvalues strictly inside the unit circle except for the $q(T+1)$ equal to 1 with eigenvectors equal to columns of $\mathbf{1}$. We leverage these properties to isolate the invariant portion of (72). A suitable decomposition of W is encoded into the transformation matrix T defined as $\mathrm{T} := \begin{bmatrix}\frac{\mathbf{1}}{N} & R\end{bmatrix}^\top$, where $R \in \mathbb{R}^{N(q(T+1))\times(N-1)(q(T+1))}$ such that $R^\top R = I$, $R^\top \mathbf{1} = 0$. Moreover, the following useful relation holds true

$$RR^\top = I - \tfrac{1}{N}\mathbf{1}\mathbf{1}^\top. \tag{75}$$

Now, with T at hand, we perform the change coordinates

$$\begin{bmatrix}\boldsymbol{\xi}^k \\ \mathbf{w}^k \\ \mathbf{z}^k\end{bmatrix} \longmapsto \begin{bmatrix}\boldsymbol{\xi}^k \\ \mathbf{w}^k_{\text{avg}} \\ \boldsymbol{\eta}^k \\ \mathbf{z}^k_{\text{avg}} \\ \boldsymbol{\zeta}^k\end{bmatrix} := \begin{bmatrix}I & 0 & 0 \\ 0 & \mathrm{T} & 0 \\ 0 & 0 & \mathrm{T}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi}^k \\ \mathbf{w}^k \\ \mathbf{z}^k\end{bmatrix}, \tag{76}$$

with $\boldsymbol{\eta}^k, \boldsymbol{\zeta}^k \in \mathbb{R}^{(N-1)q(T+1)}$ and $\mathbf{w}^k_{\text{avg}}, \mathbf{z}^k_{\text{avg}} \in \mathbb{R}^{q(T+1)}$. System (72) in the new coordinates (76) reads as

$$\begin{aligned}
\boldsymbol{\xi}^{k+1} &= \boldsymbol{\xi}^k - \gamma D_{\boldsymbol{\xi}}(\boldsymbol{\xi}^k, \bar{\varphi}(\boldsymbol{\xi}^k) + R\boldsymbol{\eta}^k + \bar{\varphi}(\boldsymbol{\xi}^k)\mathbf{1}\mathbf{w}^k_{\text{avg}}) \\
&\quad - \gamma\nabla\bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k)R\boldsymbol{\zeta}^k - \gamma\nabla\bar{\boldsymbol{\varphi}}(\boldsymbol{\xi}^k)\mathbf{1}\mathbf{z}^k_{\text{avg}} && \text{(77a)}\\
\mathbf{w}^{k+1}_{\text{avg}} &= \mathbf{w}^k_{\text{avg}} && \text{(77b)}\\
\boldsymbol{\eta}^{k+1} &= R^\top W R\boldsymbol{\eta}^k + R^\top(W-I)\bar{\varphi}(\boldsymbol{\xi}^k) && \text{(77c)}\\
\mathbf{z}^{k+1}_{\text{avg}} &= \mathbf{z}^k_{\text{avg}} && \text{(77d)}\\
\boldsymbol{\zeta}^{k+1} &= R^\top W R\boldsymbol{\zeta}^k && \text{(77e)}\\
&\quad + R^\top(W-I)\mathrm{D}_3(\boldsymbol{\xi}^k, \bar{\varphi}(\boldsymbol{\xi}^k) + R\boldsymbol{\eta}^k + \mathbf{1}\mathbf{w}^k_{\text{avg}}).
\end{aligned}$$

We observe that $R^\top W R$ is Schur and that $\mathbf{w}^k_{\text{avg}} = \mathbf{w}^0_{\text{avg}}$ and $\mathbf{z}^k_{\text{avg}} = \mathbf{z}^0_{\text{avg}}$ for any $k \in \mathbb{N}$. Further, the choice of the initial conditions implies $\mathbf{w}^0 = \mathbf{z}^0 = 0$ and, thus, $\mathbf{w}^k_{\text{avg}} = 0$ and $\mathbf{z}^k_{\text{avg}} = 0$ for all $k \in \mathbb{N}$. This initialization allows us to ignore $\mathbf{w}^k_{\text{avg}}$ and $\mathbf{z}^k_{\text{avg}}$ rewriting (77) as the reduced, equivalent system

$$\begin{aligned}
\boldsymbol{\xi}^{k+1} &= \boldsymbol{\xi}^k - \gamma D_{\boldsymbol{\xi}}(\boldsymbol{\xi}^k, \bar{\varphi}(\boldsymbol{\xi}^k) + R\boldsymbol{\eta}^k) - \gamma\nabla\bar{\varphi}(\boldsymbol{\xi}^k)R\boldsymbol{\zeta}^k && \text{(78a)}\\
\boldsymbol{\eta}^{k+1} &= R^\top W R\boldsymbol{\eta}^k + R^\top(W-I)\bar{\varphi}(\boldsymbol{\xi}^k) && \text{(78b)}\\
\boldsymbol{\zeta}^{k+1} &= R^\top W R\boldsymbol{\zeta}^k + R^\top(W-I)\mathrm{D}_3(\boldsymbol{\xi}^k, \bar{\varphi}(\boldsymbol{\xi}^k) + R\boldsymbol{\eta}^k). && \text{(78c)}
\end{aligned}$$

Being $\mathcal{W}$ doubly stochastic (cf. Assumption 2) and using (75), it is possible to show that, for any $\boldsymbol{\xi}^k \in \mathbb{R}^{(n+m)T}$, the point

$$h(\boldsymbol{\xi}^k) := \begin{bmatrix}-R^\top\bar{\varphi}(\boldsymbol{\xi}^k) \\ -R^\top\mathrm{D}_3(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k))\end{bmatrix} \in \mathbb{R}^{2d},$$

represents an equilibrium of (78b) and (78c) parametrized in $\boldsymbol{\xi}^k$. Hence, let the error coordinates $\boldsymbol{\vartheta}^k \in \mathbb{R}^{2d}$ be defined as

$$\begin{bmatrix}\boldsymbol{\eta}^k \\ \boldsymbol{\zeta}^k\end{bmatrix} \longmapsto \boldsymbol{\vartheta}^k := \begin{bmatrix}\boldsymbol{\vartheta}^k_1 \\ \boldsymbol{\vartheta}^k_2\end{bmatrix} := \begin{bmatrix}\boldsymbol{\eta}^k \\ \boldsymbol{\zeta}^k\end{bmatrix} - h(\boldsymbol{\xi}^k), \tag{79}$$

which, being $\mathcal{W}$ doubly stochastic (cf. Assumption 2), the definition (73) and (75), allow us to rewrite system (78) as

$$\begin{aligned}
\boldsymbol{\xi}^{k+1} &= \boldsymbol{\xi}^k - \gamma\begin{bmatrix}\mathrm{D}_1(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + \begin{bmatrix}R & 0\end{bmatrix}\boldsymbol{\vartheta}^k) \\ \mathrm{D}_2(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + \begin{bmatrix}R & 0\end{bmatrix}\boldsymbol{\vartheta}^k)\end{bmatrix} \\
&\quad - \gamma\nabla\bar{\varphi}(\boldsymbol{\xi}^k)\frac{\mathbf{1}\mathbf{1}^\top}{N}\mathrm{D}_3(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k)) - \gamma[0\ \nabla\bar{\varphi}(\boldsymbol{\xi}^k)R]\boldsymbol{\vartheta}^k \\
&\quad - \gamma\nabla\bar{\varphi}(\boldsymbol{\xi}^k)\big[\mathrm{D}_3(\boldsymbol{\xi}, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + \begin{bmatrix}R & 0\end{bmatrix}\boldsymbol{\vartheta}^k) - \mathrm{D}_3(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k))\big] \\
\boldsymbol{\vartheta}^{k+1} &= \begin{bmatrix}R^\top W R & 0 \\ 0 & R^\top W R\end{bmatrix}\boldsymbol{\vartheta}^k \\
&\quad + \begin{bmatrix}0 \\ R^\top(W-I)\end{bmatrix}(\mathrm{D}_3(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k) + \begin{bmatrix}R & 0\end{bmatrix}\boldsymbol{\vartheta}^k) \\
&\quad - \begin{bmatrix}0 \\ R^\top(W-I)\end{bmatrix}\mathrm{D}_3(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k))) \\
&\quad + \begin{bmatrix}R^\top & 0 \\ 0 & R^\top\end{bmatrix}\begin{bmatrix}\bar{\varphi}(\boldsymbol{\xi}^{k+1}) - \bar{\varphi}(\boldsymbol{\xi}^k) \\ (\mathrm{D}_3(\boldsymbol{\xi}^{k+1}, \mathbf{1}\varsigma(\boldsymbol{\xi}^{k+1})) - \mathrm{D}_3(\boldsymbol{\xi}^k, \mathbf{1}\varsigma(\boldsymbol{\xi}^k)))\end{bmatrix}.
\end{aligned}$$

The proof follows by using the operators in (32) and using the definitions of $\mathcal{A}$, $\mathcal{B}_1$ and $\mathcal{B}_2$ in Lemma 3.

References

[1] F. Chen, W. Ren *et al.*, "On the control of multi-agent systems: A survey," *Foundations and Trends® in Systems and Control*, vol. 6, no. 4, pp. 339–499, 2019.

[2] R. Hult, M. Zanon, S. Gros, H. Wymeersch, and P. Falcone, "Optimisation-based coordination of connected, automated vehicles at intersections," *Vehicle Systems Dynamics*, vol. 58, no. 5, pp. 726–747, 2020.

[3] M. Sassano and A. Astolfi, "Combining pontryagin's principle and dynamic programming for linear and nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 65, no. 12, pp. 5312–5327, 2020.

[4] S. Park, D. Lee, H. J. Ahn, C. Tomlin, and S. Moura, "Optimal control of battery fast charging based-on Pontryagin's minimum principle," in *IEEE Conference on Decision and Control*, 2020, pp. 3506–3513.

[5] M. Diehl, H. J. Ferreau, and N. Haverbeke, "Efficient numerical methods for nonlinear MPC and moving horizon estimation," in *Nonlinear model predictive control*. Springer, 2009, pp. 391–417.

[6] G. Foderaro, S. Ferrari, and T. A. Wettergren, "Distributed optimal control for multi-agent trajectory optimization," *Automatica*, vol. 50, no. 1, pp. 149–154, 2014.

[7] R. Van Parys and G. Pipeleers, "Distributed MPC for multi-vehicle systems moving in formation," *Robotics and Autonomous Systems*, vol. 97, pp. 144–152, 2017.

[8] A. D. Saravanos, A. Tsolovikos, E. Bakolas, and E. A. Theodorou, "Distributed covariance steering with consensus ADMM for stochastic multi-agent systems." in *Robotics: Science and Systems*, 2021.

[9] F. Rey, Z. Pan, A. Hauswirth, and J. Lygeros, "Fully decentralized ADMM for coordination and collision avoidance," in *2018 European Control Conference*, 2018, pp. 825–830.

[10] O. Shorinwa, T. Halsted, and M. Schwager, "Scalable distributed optimization with separable variables in multi-agent networks," in *American Control Conference*, 2020, pp. 3619–3626.

[11] T. Halsted, O. Shorinwa, J. Yu, and M. Schwager, "A survey of distributed optimization methods for multi-robot systems," *preprint arXiv:2103.12840*, 2021.

[12] T. Kavuncu, A. Yaraneri, and N. Mehr, "Potential iLQR: A potential-minimizing controller for planning multi-agent interactive trajectories," *arXiv preprint arXiv:2107.04926*, 2021.

[13] O. So, K. Stachowicz, and E. A. Theodorou, "Multimodal maximum entropy dynamic games," *preprint arXiv:2201.12925*, 2022.

[14] M. A. Müller and F. Allgöwer, "Economic and distributed model predictive control: Recent developments in optimization-based control," *SICE Journal of Control, Measurement, and System Integration*, vol. 10, no. 2, pp. 39–52, 2017.

[15] P. Giselsson, M. D. Doan, T. Keviczky, B. De Schutter, and A. Rantzer, "Accelerated gradient methods and dual decomposition in distributed model predictive control," *Automatica*, vol. 49, no. 3, pp. 829–833, 2013.

[16] B. Houska and J. Shi, "Distributed MPC with ALADIN — A tutorial," in *American Control Conference*, 2022, pp. 358–363.

[17] N. Bastianello, R. Carli, L. Schenato, and M. Todescato, "Asynchronous distributed optimization over lossy networks via relaxed ADMM: Stability and linear convergence," *IEEE Transactions on Automatic Control*, vol. 66, no. 6, pp. 2620–2635, 2020.

[18] A. Falsone, I. Notarnicola, G. Notarstefano, and M. Prandini, "Tracking-ADMM for distributed constraint-coupled optimization," *Automatica*, vol. 117, p. 108962, 2020.

[19] D. Burk, A. Völz, and K. Graichen, "A modular framework for distributed model predictive control of nonlinear continuous-time systems (GRAMPC-D)," *Optimization and Engineering*, pp. 1–25, 2021.

[20] A. Engelmann, Y. Jiang, B. Houska, and T. Faulwasser, "Decomposition of nonconvex optimization via bi-level distributed ALADIN," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 4, pp. 1848–1858, 2020.

[21] G. Stomberg, A. Engelmann, and T. Faulwasser, "Decentralized non-convex optimization via bi-level SQP and ADMM," in *IEEE Conference on Decision and Control*, 2022.

[22] G. Stomberg, H. Ebel, T. Faulwasser, and P. Eberhard, "Cooperative distributed MPC via decentralized real-time optimization: Implementation results for robot formations," *Control Engineering Practice*, vol. 138, p. 105579, 2023.

[23] X. Li, L. Xie, and Y. Hong, "Distributed aggregative optimization over multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 67, no. 6, pp. 3165–3171, 2021.

[24] F. Parise, S. Grammatico, B. Gentile, and J. Lygeros, "Distributed convergence to Nash equilibria in network and average aggregative games," *Automatica*, vol. 117, p. 108959, 2020.

[25] G. Belgioioso, A. Nedić, and S. Grammatico, "Distributed generalized Nash equilibrium seeking in aggregative games on time-varying networks," *IEEE Transactions on Automatic Control*, vol. 66, no. 5, pp. 2061–2075, 2020.

[26] D. Gadjov and L. Pavel, "Single-timescale distributed gne seeking for aggregative games over networks via forward–backward operator splitting," *IEEE Transactions on Automatic Control*, vol. 66, no. 7, pp. 3259–3266, 2020.

[27] F. Fabiani, M. A. Tajeddini, H. Kebriaei, and S. Grammatico, "Local Stackelberg equilibrium seeking in generalized aggregative games," *IEEE Transactions on Automatic Control*, vol. 67, no. 2, pp. 965–970, 2021.

[28] G. Carnevale, F. Fabiani, F. Fele, K. Margellos, and G. Notarstefano, "Tracking-based distributed equilibrium seeking for aggregative games," *IEEE Transactions on Automatic Control*, 2024.

[29] X. Li, X. Yi, and L. Xie, "Distributed online convex optimization with an aggregative variable," *IEEE Transactions on Control of Network Systems*, vol. 9, no. 1, pp. 438–449, 2021.

[30] G. Carnevale, A. Camisa, and G. Notarstefano, "Distributed online aggregative optimization for dynamic multi-robot coordination," *IEEE Transactions on Automatic Control*, 2022.

[31] G. Carnevale and G. Notarstefano, "A learning-based distributed algorithm for personalized aggregative optimization," in *IEEE Conference on Decision and Control*, 2022, pp. 1576–1581.

[32] G. Carnevale, N. Mimmo, and G. Notarstefano, "Aggregative feedback optimization for distributed cooperative robotics," *IFAC-PapersOnLine*, vol. 55, no. 13, pp. 7–12, 2022.

[33] M. Chen, D. Wang, X. Wang, Z.-G. Wu, and W. Wang, "Distributed aggregative optimization via finite-time dynamic average consensus," *IEEE Transactions on Network Science and Engineering*, 2023.

[34] T. Wang and P. Yi, "Distributed projection-free algorithm for constrained aggregative optimization," *International Journal of Robust and Nonlinear Control*, vol. 33, no. 10, pp. 5273–5288, 2023.

[35] P. D. Grontas, M. W. Fisher, and F. Dörfler, "Distributed and constrained H_2 control design via system level synthesis and dual consensus admm," in *IEEE Conference on Decision and Control*, 2022, pp. 301–307.

[36] Z. Chen and S. Liang, "Distributed aggregative optimization with quantized communication," *Kybernetica*, vol. 58, no. 1, pp. 123–144, 2022.

[37] J. Hauser, "A projection operator approach to the optimization of trajectory functionals," *IFAC Proceedings Volumes*, vol. 35, no. 1, pp. 377–382, 2002.

[38] J. Hauser and A. Saccon, "A barrier function method for the optimization of trajectory functionals with constraints," in *IEEE Conference on Decision and Control*, 2006, pp. 864–869.

[39] A. Saccon, J. Hauser, and A. P. Aguiar, "Optimal control on lie groups: The projection operator approach," *IEEE Transactions on Automatic Control*, vol. 58, no. 9, pp. 2230–2245, 2013.

[40] A. P. Aguiar, F. A. Bayer, J. Hauser, A. J. Häusler, G. Notarstefano, A. M. Pascoal, A. Rucco, and A. Saccon, "Constrained optimal motion planning for autonomous vehicles using PRONTO," in *Sensing and control for autonomous vehicles*. Springer, 2017, pp. 207–226.

[41] L. Sforni, S. Spedicato, I. Notarnicola, and G. Notarstefano, "Gopronto: a feedback-based framework for nonlinear optimal control," *preprint arXiv:2108.13308*, 2021.

[42] A. Testa, G. Carnevale, and G. Notarstefano, "A tutorial on distributed optimization for cooperative robotics: from setups and algorithms to toolboxes and research directions," *preprint arXiv:2309.04257*, 2023.

[43] M. K. Jensen, "Aggregative games and best-reply potentials," *Economic theory*, vol. 43, no. 1, pp. 45–66, 2010.

[44] F. Parise and A. Ozdaglar, "Analysis and interventions in large network games," *Annual Reviews of Control, Robotics, and Autonomous Systems*, vol. 4, pp. 455–486, 2021.

[45] G. Belgioioso, P. Yi, S. Grammatico, and L. Pavel, "Distributed generalized Nash equilibrium seeking: An operator-theoretic perspective," *IEEE Control Systems Magazine*, vol. 42, no. 4, pp. 87–102, 2022.

[46] K. Du and M. Meng, "Linear convergence of distributed aggregative optimization with coupled inequality constraints," *preprint arXiv:2306.06700*, 2023.

[47] M. Zhu and S. Martínez, "Discrete-time dynamic average consensus," *Automatica*, vol. 46, no. 2, pp. 322–329, 2010.

[48] S. S. Kia, B. Van Scoy, J. Cortes, R. A. Freeman, K. M. Lynch, and S. Martinez, "Tutorial on dynamic average consensus: The problem, its applications, and the algorithms," *IEEE Control Systems Magazine*, vol. 39, no. 3, pp. 40–72, 2019.

[49] G. Notarstefano, I. Notarnicola, and A. Camisa, "Distributed optimization for smart cyber-physical networks," *Foundations and Trends® in Systems and Control*, vol. 7, no. 3, pp. 253–383, 2019.

[50] G. Folland, "Higher-order derivatives and Taylor's formula in several variables." [Online]. Available: http://sites.math.washington.edu/~folland/Math425/taylor2.pdf

[51] T. Ge, W. Lin, and J. Feng, "Invariance principles allowing of non-lyapunov functions for estimating attractor of discrete dynamical systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 2, pp. 500–505, 2011.

[52] 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.

Lorenzo Sforni (S'21) received the Ph.D. degree in "Biomedical, Electrical, and Systems Engineering" from the Alma Mater Studiorum Università di Bologna in 2024. From the same institution, he was awarded a Master's degree with honours in 2020. He has been visiting researcher at the California Institute of Technology in 2023. His research interests include nonlinear optimal control, optimization algorithms and reinforcement learning.

Guido Carnevale (M'23) 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 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 and optimal control.

Giuseppe Notarstefano (M'11) is a Professor in the Department of Electrical, Electronic, and Information Engineering G. Marconi at Alma Mater Studiorum Università di Bologna. He was Associate Professor (June '16 - June '18) and previously Assistant Professor, Ricercatore, (from Feb '07) 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 Ph.D. 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 part of the Conference Editorial Board of IEEE Control Systems Society and EUCA. He was a recipient of an ERC Starting Grant 2014.