

Toward Step-Size-Free Distributed Optimization: A Port-Hamiltonian Approach

Rodrigo Aldana-López , *Member, IEEE*, Alessandro Macchelli , *Senior Member, IEEE*,
Giuseppe Notarstefano , *Senior Member, IEEE*, Rosario Aragüés , *Senior Member, IEEE*,
and Carlos Sagüés , *Senior Member, IEEE*

Abstract—This article presents a novel distributed optimization technique that, under suitable assumptions on problem and communication graph, is able to operate with any step-size parameter, a capability not evident in previous methods. Traditionally, selecting the step size in distributed optimization often leads to a conservative performance, resulting in slow convergence or even divergence if the step size is incorrectly chosen. In this work, we propose a systems theory approach based on the port-Hamiltonian formalism to develop algorithms for consensus optimization problems. Starting from a continuous-time flow, we propose a distributed algorithm based on a discretized port-Hamiltonian system that works regardless of the step size. In addition, we introduce mixed implicit discretization, which lowers communication overhead while maintaining, under suitable conditions, the ability to converge with any step size. This feature is formally verified in specific scenarios and demonstrated experimentally in general cases with large step sizes. These results illustrate that the proposed method can utilize step sizes several orders of magnitude larger than those in previous work, where conventional methods fail, ultimately enhancing convergence speed.

Index Terms—Distributed optimization, port-Hamiltonian systems (PHSs), step size.

I. INTRODUCTION

CURRENT technological trends lean toward the extended use of machine learning algorithms with increasing complexity and computing power requirements. Fortunately, computation and communication capabilities are readily available

in modern devices. For this reason, distributed implementations have emerged as scalable and robust alternatives to deploy these technologies. Since many tasks of importance, e.g., the training of neural networks, can be posed as optimization programs, distributed optimization has remained an active area of research in the last years (see [1] and [2]). The problem of consensus optimization is particularly important, where the unconstrained cost is comprised of the sum of local costs that can be accessed at each device in the network separately [1].

In the following, we review the literature on algorithms developed for distributed optimization problems. First, we examine algorithms formulated in continuous time, leveraging tools from systems theory to guarantee stability. Subsequently, we discuss methodologies that operate in discrete time, reflecting more practical communication scenarios. Finally, we mention works that use the port-Hamiltonian system (PHS) formalism in multiagent contexts and step-size robustness in discretization.

Many algorithms for addressing distributed optimization problems through continuous-time systems exist in the literature. For instance, a method for consensus optimization over weighted digraphs was introduced in [3]. A different approach, employing a proportional–integral protocol to tackle constrained distributed optimization, was proposed in [4]. The strategy in [5] aims to resolve constrained distributed convex optimization problems via nonsmooth analysis. Meanwhile, Lin et al. [6] presented a protocol with finite-time convergence in solving constrained distributed optimization issues. The use of adaptive control techniques for unconstrained distributed optimization was explored in [7]. A passivity-based methodology for designing and analyzing distributed protocol algorithms in the unconstrained scenario was used in [8], with a similar approach adopted for balanced digraphs in [9]. Furthermore, a distributed protocol leveraging Newton–Raphson iterations was detailed in [10]. These contributions underscore the benefits of modeling distributed optimization algorithms in continuous time, allowing for a more straightforward application of systems theory tools, such as Lyapunov theory or passivity. However, given that communication among agents in digital platforms occurs in discrete time, research has pivoted toward considering discrete-time interactions. A leading methodology for consensus optimization in discrete time, known as the gradient tracking method, was introduced in [11], incorporating a dynamic average consensus component to track the gradient of the objective function. This method has spawned various adaptations, including those

Received 28 February 2025; revised 3 December 2025; accepted 6 April 2026. Date of publication 23 April 2026; date of current version 29 September 2026. This work was supported in part by project REMAIN-S1/1.1/E0111 (Interreg Sudoe Programme, ERDF), by projects PID2021-124137OB-I00 and PID2024-159279OB-I00 funded by MICIU/AEI/10.13039/501100011033 and by ERDF/EU, in part by the Gobierno de Aragón under Project DGA T45-23R, and in part by MICIU/AEI/10.13039/501100011033 under Grant BG24/00121. Recommended by Associate Editor I. Necoara. (Corresponding author: Rodrigo Aldana-López.)

Rodrigo Aldana-López, Rosario Aragüés, and Carlos Sagüés are with the Departamento de Informática e Ingeniería de Sistemas and Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, 50018 Zaragoza, Spain (e-mail: raldana@unizar.es; raragues@unizar.es; csagues@unizar.es).

Alessandro Macchelli and Giuseppe Notarstefano are with the Department of Electrical, Electronic and Information Engineering, University of Bologna, 40136 Bologna, Italy (e-mail: alessandro.macchelli@unibo.it; giuseppe.notarstefano@unibo.it).

Digital Object Identifier 10.1109/TAC.2026.3687481

tailored for nonconvex programs [12] and protocols integrating Newton–Raphson iterations [13]. Extensions to accommodate time-varying and stochastic networks were proposed in [11] and [14], respectively. Efforts to enhance convergence speed are used in [15] and [16]. Techniques, such as push-sum and push–pull protocols for message passing, aimed at addressing digraphs, were applied in [17] and [18]. An adaptive momentum term to boost convergence rates was introduced in [19]. Using a systems structure perspective, most of these variations can be understood using a general structural framework as in [20]. Despite their generally effective performance, these algorithms constrain the learning rate, often taking the role of a step-size parameter. Tuning the step size is critical, especially in multiagent settings where the constraints depend on global information, such as the network topology. For instance, a large step size may result in instability, and a small step size may lead to slow convergence. A notable contribution is [21], which removes the dependence of step-size selection on network properties. However, the method still imposes an upper bound on the step size, beyond which the algorithm diverges. While this bound is network-independent, it still depends on the objective function properties, which may be unknown to some or all agents.

The previous discussion highlights the need to develop distributed optimization methods that either operate independently of the step size or allow significantly larger step sizes before divergence occurs. We refer to [22] and the references therein for methods that propose optimization algorithms without tunable parameters. However, these approaches are centralized. An important breakthrough in this context was presented in [23], where the PHS formalism was used to design a continuous-time gradient flow for unconstrained optimization in a centralized setting. This formalism allowed the authors to propose a symplectic discretization that works with any step size. Generally, the PHS is capable of modeling a wide range of application frameworks through power-conserving interconnections between subsystems [24]. This formalism has been proved useful to describe and analyze multiagent algorithms for consensus [25] and distributed control [26]. Moreover, geometric integration using the discrete gradient has been shown to preserve the PHS structure [27], [28], with a precise parallelism between continuous time and discrete time. In addition, under suitable conditions, this type of discretization can ensure convergence regardless of the step-size parameter. For instance, the discrete gradient method was used in [29] to develop an optimization algorithm robust to large step sizes. However, despite the advantages of PHS and geometric integration methods, these tools have not been proposed to solve distributed optimization problems.

The contribution of this work is twofold. First, we propose, for the first time, a systems theory framework based on the port-Hamiltonian formalism to design and analyze distributed optimization algorithms. Specifically, we apply this approach to solve an unconstrained consensus optimization problem, with convergence proofs relying on tools from the port-Hamiltonian framework. Second, using this framework, we design a discretization method that ensures convergence regardless of the step size while preserving the port-Hamiltonian structure. Building on this, we introduce the mixed implicit discretization

(MID), a novel approach that reduces communication overhead. We formally show that MID converges with arbitrarily large step sizes in specific scenarios and experimentally demonstrate that it performs effectively in more general cases with step sizes orders of magnitude larger than those supported by prior methods. In addition, we show that treating the step size as a single free design parameter for all agents can improve convergence speed.

The rest of this article is organized as follows. First, in Section II, we provide a problem statement and some preliminaries on the PHS. Section III presents our PHS-based framework for consensus optimization in continuous time, including some concrete examples. Section IV discusses PHS preserving discretization with convergence capabilities under arbitrarily big step sizes. In Section V, we provide several numerical examples, comparing the proposal’s performance against other methods in the literature. Finally, Section VI concludes this article, and auxiliary results and proofs are placed in the appendixes.

Notations: For a set S , let $|S|$ represent its cardinality. Given vectors $\{\mathbf{x}_i\}_{i \in \mathcal{I}}$ with index set $\mathcal{I} = \{1, \dots, N\}$, we denote with $[\mathbf{x}_i]_{i \in \mathcal{I}} = [\mathbf{x}_1^\top, \dots, \mathbf{x}_N^\top]^\top$. Denote with ∇ the gradient operator. Let $\mathbf{1}_n$, $\mathbf{0}_{n \times m}$, $\mathbf{I}_n$ denote the column vector of n ones, the zero $n \times m$ matrix, and the $n \times n$ identity matrix, respectively. We write $\mathbf{1}$, $\mathbf{0}$, $\mathbf{I}$ when there is no ambiguity on the dimensions, according to the context. The symbol $\otimes$ represents the Kronecker product. For discrete-time sequences $\{\mathbf{x}[k]\}_{k \in \mathbb{N}}$, we often write $\mathbf{x}$ to represent the value of the sequence at an arbitrary time step; $\mathbf{x}^+$ represents its value at the next time step. Let $\|\mathbf{x}\|$ represent the Euclidean norm of a vector $\mathbf{x}$, and $\|\mathbf{X}\|$ represents the induced Euclidean norm of a matrix $\mathbf{X}$.

II. PROBLEM STATEMENT AND PRELIMINARIES

A. Problem Statement

Consider a group of $N \in \mathbb{N}$ agents labeled using an index set $\mathcal{I} = \{1, \dots, N\}$, whose goal is to solve a consensus optimization program of the form

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^m} \sum_{i=1}^N f_i(\boldsymbol{\theta}) \quad (1)$$

where each $f_i : \mathbb{R}^m \rightarrow \mathbb{R}$ is a function that can only be accessed by agent $i \in \mathcal{I}$.

Assumption 1: The program in (1) has a unique optimal solution $\boldsymbol{\theta}^* \in \mathbb{R}^m$.

To obtain such an optimal solution, agents cooperate by exchanging information with their neighbors. The communication network is modeled by a fixed undirected connected graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$. Here, $\mathcal{I}$ is the vertex set, where each $i \in \mathcal{I}$ represents a concrete agent, and $\mathcal{E} \subseteq \mathcal{I} \times \mathcal{I}$ is the edge set where $(i, j) \in \mathcal{E}$ when agents $i, j \in \mathcal{I}$ can communicate. Moreover, we denote with $\mathcal{N}_i \subseteq \mathcal{I}$ the neighbor set for agent $i \in \mathcal{I}$.

This work proposes a solution to address the program (1) using the port-Hamiltonian structure. Furthermore, we demonstrate that this framework can generate algorithms capable of operating with arbitrarily large step sizes in some scenarios.

B. Port-Hamiltonian Systems

A PHS is a particular class of dynamical system that can be described through the interaction of ports, which are input–output pairs, with power-conserving features. In particular, it has the form

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x})\nabla H(\mathbf{x}) + \mathbf{B}_I(\mathbf{x})\mathbf{u}_I + \mathbf{B}_C(\mathbf{x})\mathbf{u}_C \quad (2)$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the state variable. The system in (2) has four classes of ports: first, a storage port $(-\dot{\mathbf{x}}, \nabla H(\mathbf{x}))$ with scalar storage function $H : \mathbb{R}^n \rightarrow \mathbb{R}$ also called the Hamiltonian energy; second, a resistive structure associated with the symmetric part of $\mathbf{F}$, which satisfies $\mathbf{F}(\mathbf{x}) + \mathbf{F}(\mathbf{x})^\top \preceq \mathbf{0}$ leading to a port $(\nabla H(\mathbf{x}), \mathbf{F}(\mathbf{x})\nabla H(\mathbf{x}))$; third, an interaction port with $(\mathbf{u}_I, \mathbf{y}_I)$, which usually represents the interaction of (2) with other systems; and fourth, a control port $(\mathbf{u}_C, \mathbf{y}_C)$. This port is used for feedback control. By defining outputs, it can be verified that

$$\begin{aligned} \mathbf{y}_I &= \mathbf{B}_I(\mathbf{x})^\top \nabla H(\mathbf{x}) \\ \mathbf{y}_C &= \mathbf{B}_C(\mathbf{x})^\top \nabla H(\mathbf{x}). \end{aligned}$$

Then, (2) is passive in the sense that

$$\dot{H}(\mathbf{x}) \leq \mathbf{y}_I^\top \mathbf{u}_I + \mathbf{y}_C^\top \mathbf{u}_C.$$

As will become evident in subsequent sections, the passivity property helps to design control feedback laws and to obtain Lyapunov functions based on H . We refer the reader to [24] for more details on the PHS.

III. PHS-BASED OPTIMIZATION IN CONTINUOUS TIME

A. System Proposal

We start by considering a continuous-time formulation based on a PHS. Assume that each agent $i \in \mathcal{I}$ stores a state variable $\mathbf{x}_i(t) \in \mathbb{R}^n$. The proposed dynamics for $\mathbf{x}_i(t)$ are given as

$$\begin{aligned} \dot{\mathbf{x}}_i &= \mathbf{F}_i \nabla H_i(\mathbf{x}_i) + \phi_i(\nabla H_i(\mathbf{x}_i)) \\ &+ \sum_{j \in \mathcal{N}_i} \mathbf{M}(\nabla H_i(\mathbf{x}_i) - \nabla H_j(\mathbf{x}_j)) \end{aligned} \quad (3)$$

where $\mathbf{F}_i$ and $\mathbf{M}$ are constant matrices satisfying $\mathbf{F}_i + \mathbf{F}_i^\top \preceq \mathbf{0}$ and $\mathbf{M} + \mathbf{M}^\top \preceq \mathbf{0}$, respectively. Moreover, $H_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is the storage function at agent i , and $\phi_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a nonlinear vector field. Time dependence for the state was omitted in (3) for readability. For the same reason, we will also omit state dependence on H_i when there is no ambiguity.

To see that (3) is a PHS, we can rewrite it as

$$\begin{aligned} \dot{\mathbf{x}}_i &= (\mathbf{F}_i + |\mathcal{N}_i| \mathbf{M}) \nabla H_i(\mathbf{x}_i) \\ &- \sum_{j \in \mathcal{N}_i} \mathbf{M} \nabla H_j(\mathbf{x}_j) + \phi_i(\nabla H_i(\mathbf{x}_i)) \end{aligned}$$

where the symmetric part of $(\mathbf{F}_i + |\mathcal{N}_i| \mathbf{M})$ is negative semidefinite, and this plays the role of $\mathbf{F}(\mathbf{x})$ in (2). Moreover, it includes $|\mathcal{N}_i|$ interaction terms $-\mathbf{M} \nabla H_j(\mathbf{x}_j)$, each connected to one of the neighbors of the agent with input $\nabla H_j(\mathbf{x}_j)$ and input matrix $-\mathbf{M}$, playing the role of the term $\mathbf{B}_I(\mathbf{x})\mathbf{u}_I$ in (2). In addition, it

features a local feedback control port with input $\phi_i(\nabla H_i(\mathbf{x}_i))$ and input matrix $\mathbf{I}_n$, playing the role of $\mathbf{B}_C(\mathbf{x})\mathbf{u}_C$ in (2). Note that the terms of the form $\nabla H_i(\mathbf{x}_i) - \nabla H_j(\mathbf{x}_j)$ in (3) define an agreement error between neighboring agents and promote consensus.

The rest of this section aims to show that the PHS structure in (3) naturally ensures convergence toward equilibrium. Hence, given that stability is provided, the remaining degrees of freedom, namely, $\mathbf{F}_i, \mathbf{M}, \phi_i$, and H_i , can be used to design equilibrium to coincide with the solution of an optimization program of interest. To formalize these ideas, we make the following assumption.

Assumption 2 (Well-posedness): Let $H_i, \mathbf{F}_i, \mathbf{M}$, and ϕ_i be such that:

- 1) the PHS in (3) has a solution for all $t \geq 0$, $\mathbf{x}_i(0) \in \mathbb{R}^n$, $i \in \mathcal{I}$;
- 2) there exist unique constant vectors $\{\mathbf{x}_i^*\}_{i \in \mathcal{I}}$ such that

$$\begin{aligned} \mathbf{0} &= \mathbf{F}_i \nabla H_i(\mathbf{x}_i^*) + \phi_i(\nabla H_i(\mathbf{x}_i^*)) \\ &+ \sum_{j \in \mathcal{N}_i} \mathbf{M}(\nabla H_i(\mathbf{x}_i^*) - \nabla H_j(\mathbf{x}_j^*)). \end{aligned} \quad (4)$$

Remark 1: Assumption 2 requires only the existence of solutions for (3). This is sufficient for all convergence results in this work, since the Lyapunov arguments apply along every admissible trajectory. Nevertheless, in the examples presented later, uniqueness of solutions follows automatically from the Cauchy–Lipschitz theorem when ∇H_i and ϕ_i are globally Lipschitz.

To show convergence toward the unique equilibrium point satisfying (4), we write the joint dynamics of all agents incorporating (3) for all $i \in \mathcal{I}$. Consider the joint storage function

$$H(\mathbf{x}) = \sum_{i \in \mathcal{I}} H_i(\mathbf{x}_i),$$

the joint output $\mathbf{y} = [\nabla H_i(\mathbf{x}_i)]_{i \in \mathcal{I}}$, and the joint input $\phi(\mathbf{y}) = [\phi_i(\nabla H_i(\mathbf{x}_i))]_{i \in \mathcal{I}}$ with joint state $\mathbf{x} = [\mathbf{x}_i]_{i \in \mathcal{I}}$. Moreover, let $\mathbf{F} = \text{blockdiag}(\mathbf{F}_1, \dots, \mathbf{F}_N)$. Hence, (3) can be written in compact form as follows:

$$\dot{\mathbf{x}} = (\mathbf{F} + \mathbf{L} \otimes \mathbf{M}) \nabla H(\mathbf{x}) + \phi(\nabla H(\mathbf{x})) \quad (5)$$

where $\mathbf{L} = \text{diag}(\mathbf{A}\mathbf{1}) - \mathbf{A}$ is the standard Laplacian matrix of the graph $\mathcal{G}$ associated with the 0–1 adjacency matrix $\mathbf{A}$.

One of the most important features of (5) inherited from the PHS structure is its passivity property concerning the input–output pair $(\phi(\mathbf{y}), \mathbf{y})$.

Proposition 1: The trajectories of (5) satisfy

$$\dot{H}(\mathbf{x}) \leq \mathbf{y}^\top \phi(\mathbf{y}).$$

As discussed in Section II-B, passivity can be used to construct a Lyapunov function, ensuring convergence to the equilibrium. This idea is formalized in the following.

Corollary 1: Let Assumption 2 hold and consider (5). Moreover, assume that $H(\mathbf{x})$ is strongly convex with constant $\mu > 0$. Assume that each ϕ_i satisfies the incremental passivity property

$$(\mathbf{u} - \mathbf{v})^\top (\phi_i(\mathbf{u}) - \phi_i(\mathbf{v})) \leq 0 \quad (6)$$

for arbitrary $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, with equality only when $\mathbf{u} = \mathbf{v}$. Then, the unique equilibrium of (5) is globally asymptotically stable. In addition, if (6) is replaced by the strong incremental passivity property

$$(\mathbf{u} - \mathbf{v})^\top (\phi_i(\mathbf{u}) - \phi_i(\mathbf{v})) \leq -\alpha \|\mathbf{u} - \mathbf{v}\|^2 \quad \forall \mathbf{u} \neq \mathbf{v} \quad (7)$$

for some $\alpha > 0$ and $\nabla H(\mathbf{x})$ is Lipschitz continuous with constant L , then the unique equilibrium of (5) is globally exponentially stable with rate $2\alpha\mu^2/L$.

B. Matching the Equilibrium With the Solution of an Optimization Program

Given that convergence of the PHS in (3) is ensured by Proposition 1, it only remains to match the equilibrium equations (4) with the solution of the consensus optimization program in (1). This can be performed by an appropriate design of the degrees of freedom we have in $\mathbf{F}_i, \mathbf{M}, \phi_i$, and H_i . To provide intuition, we start with the centralized setting.

Example 1 (Centralized optimization): Consider (1) with $N = 1$ and convex and differentiable $f := f_1$, $\mathbf{x} := \mathbf{x}_1$. For an equilibrium $\mathbf{x}^*$ of (5) to coincide with θ^* , the first-order optimality constraints must be met, namely, $\mathbf{0} = \nabla f(\mathbf{x}^*)$. Compared with the equilibrium conditions (4), this condition can be achieved with either one of the following choices:

- a) $H_1(\mathbf{x}) = \mathbf{x}^\top \mathbf{x}/2$, $\mathbf{F}_1 = \mathbf{M} = \mathbf{0}$, $\phi_1(\mathbf{x}) = -\nabla f(\mathbf{x})$.
- b) $H_1(\mathbf{x}) = f(\mathbf{x})$, $\mathbf{F}_1 = -\mathbf{I}$, $\mathbf{M} = \mathbf{0}$, $\phi_1(\mathbf{x}) = \mathbf{0}$.

In both cases, the same system results from (3), which coincides with the standard gradient flow

$$\dot{\mathbf{x}} = -\nabla f(\mathbf{x}). \quad (8)$$

Note that convexity of f implies that the incremental passivity property (6) is satisfied for $\phi_1(\mathbf{x}) = -\nabla f(\mathbf{x})$. Thus, the convergence of trajectories of the standard gradient flow (8) toward $\mathbf{x}^*$ follows from Corollary 1. Moreover, for case a), if f is strongly convex with globally Lipschitz gradients, it can be concluded that $\nabla H_1(\mathbf{x}) = \mathbf{x}$ is Lipschitz, and (7) is satisfied with $\alpha = \mu$ as a result of Proposition 2 in Appendix I. Hence, Assumption 2 is satisfied, and using Corollary 1, the stronger result of exponential stability of $\mathbf{x}^*$ is obtained.

Now, we provide an example aiming to solve (1).

Example 2 (Unconstrained distributed optimization): Consider (1) with arbitrary $N \in \mathbb{N}$, connected $\mathcal{G}$ and convex f_i . Similarly as in the previous example, we assume globally Lipschitz continuous gradients of f_i to ensure that Assumption 2 is satisfied. The first-order optimality conditions for the global optimum of the program (1) are $\mathbf{0} = \sum_{i=1}^N \nabla f_i(\theta^*)$. However, this condition can be expressed in a more appropriate distributed fashion as

$$\mathbf{0} = \sum_{i=1}^N \nabla f_i(\mathbf{q}_i^*), \quad \mathbf{q}_1^* = \dots = \mathbf{q}_N^* \quad (9)$$

where we introduce auxiliary variables $\mathbf{q}_i^* = \theta^* \in \mathbb{R}^m$. These include a condition for the gradients and a condition for consensus. Setting $\mathbf{q}^* = [\mathbf{q}_i^*]_{i \in \mathcal{I}}$, the conditions in (9) at the equilibrium can be written as

$$\mathbf{0} = (\mathbf{1}_N^\top \otimes \mathbf{I}_m) \nabla f(\mathbf{q}^*), \quad \mathbf{0} = (\mathbf{L} \otimes \mathbf{I}_m) \mathbf{q}^* \quad (10)$$

which implies $\mathbf{q}^* = (\mathbf{1}_N \otimes \theta^*)$ and with

$$f(\mathbf{q}^*) = \sum_{i=1}^N f_i(\mathbf{q}_i^*). \quad (11)$$

Having two sets of constraints motivates the addition of a second set of auxiliary variables $\mathbf{p}_i \in \mathbb{R}^m$ and partition $\mathbf{x}_i = [\mathbf{q}_i^\top, \mathbf{p}_i^\top]^\top \in \mathbb{R}^n$ with $n = 2m$. The PHS equilibrium equations in (4) for each partition will be used to match either one of the two conditions (10). This can be performed by setting

$$\begin{aligned} \mathbf{F}_i &= \mathbf{0}_{n \times n}, \quad \mathbf{H}_i(\mathbf{x}_i) = \frac{1}{2} \mathbf{x}_i^\top \mathbf{x}_i \\ \phi_i(\mathbf{x}_i) &= \begin{bmatrix} -\nabla f_i(\mathbf{q}_i) \\ \mathbf{0}_m \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} -1 & -1 \\ 1 & 0 \end{bmatrix} \otimes \mathbf{I}_m. \end{aligned} \quad (12)$$

Henceforth, from (4), the equilibrium for the partitioned states can be written as

$$\begin{aligned} \mathbf{0} &= -(\mathbf{L} \otimes \mathbf{I}_m) \mathbf{q}^* - (\mathbf{L} \otimes \mathbf{I}_m) \mathbf{p}^* - \nabla f(\mathbf{q}^*) \\ \mathbf{0} &= (\mathbf{L} \otimes \mathbf{I}_m) \mathbf{q}^* \end{aligned} \quad (13)$$

from which the optimality conditions in (10) follow by multiplying (13) with $(\mathbf{1}_N^\top \otimes \mathbf{I}_m)$ from the left and using $\mathbf{1}_N^\top \mathbf{L} = \mathbf{0}$. The resulting system at each agent from (3) is

$$\begin{aligned} \dot{\mathbf{q}}_i &= - \sum_{j \in \mathcal{N}_i} (\mathbf{q}_i - \mathbf{q}_j) - \sum_{j \in \mathcal{N}_i} (\mathbf{p}_i - \mathbf{p}_j) - \nabla f_i(\mathbf{q}_i) \\ \dot{\mathbf{p}}_i &= \sum_{j \in \mathcal{N}_i} (\mathbf{q}_i - \mathbf{q}_j). \end{aligned} \quad (14)$$

Finally, we show that this reasoning can be used to extend these ideas to include linear constraints in (1).

Example 3 (Handling linear constraints): Consider the following program:

$$\begin{aligned} \theta^* &= \arg \min_{\theta \in \mathbb{R}^m} \sum_{i=1}^N f_i(\theta) \\ \text{s.t. } \mathbf{A}\theta &= \mathbf{b} \end{aligned} \quad (15)$$

where $\mathbf{A} \in \mathbb{R}^{\ell \times m}$, $\mathbf{b} \in \mathbb{R}^\ell$ are constant. The first-order conditions for optimality are obtained by constructing a Lagrangian $\mathcal{L}(\theta, \lambda) = \sum_{i=1}^N f_i(\theta) + \lambda^\top (\mathbf{b} - \mathbf{A}\theta)$ and then taking $\nabla \mathcal{L}(\theta^*, \lambda^*) = \mathbf{0}$. Hence, optimality for (15) can be expressed using three separate conditions

$$\mathbf{0} = \sum_{i=1}^N \nabla f_i(\mathbf{q}^*) - \mathbf{A}^\top \lambda^*, \quad \mathbf{q}_1^* = \dots = \mathbf{q}_N^*, \quad \mathbf{A} \mathbf{q}_i^* = \mathbf{b} \quad (16)$$

for some λ^* . This motivates to partition $\mathbf{x}_i = [\mathbf{q}_i^\top, \mathbf{p}_i^\top, \mathbf{g}_i^\top]^\top$ with $\mathbf{q}_i, \mathbf{p}_i \in \mathbb{R}^m$ and $\mathbf{g}_i \in \mathbb{R}^\ell$, $n = 2m + \ell$. The equilibrium for each partition will be used to match the three optimality conditions. This can be performed by choosing

$$\mathbf{F}_i = \begin{bmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{A}^\top \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} -\mathbf{I}_m & -\mathbf{I}_m & \mathbf{0} \\ \mathbf{I}_m & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

as well as $\phi_i(\mathbf{x}_i) = [-\nabla f_i(\mathbf{q}_i)^\top, \mathbf{0}^\top, -\mathbf{b}^\top]^\top$ and $H(\mathbf{x}_i) = \mathbf{x}_i^\top \mathbf{x}_i/2$. From (4), the equilibrium points satisfy

$$\begin{aligned} \mathbf{0} &= -(\mathbf{I}_N \otimes \mathbf{A}^\top) \mathbf{g}^* - (\mathbf{L} \otimes \mathbf{I}_m) \mathbf{q}^* - (\mathbf{L} \otimes \mathbf{I}_m) \mathbf{p}^* - \nabla f(\mathbf{q}^*) \\ \mathbf{0} &= (\mathbf{L} \otimes \mathbf{I}_m) \mathbf{q}^* \\ \mathbf{0} &= (\mathbf{I}_N \otimes \mathbf{A}) \mathbf{q}^* - (\mathbf{I}_N \otimes \mathbf{b}) \end{aligned} \quad (17)$$

with $f(\mathbf{q}^*) = \sum_{i \in \mathcal{I}} f_i(\mathbf{q}_i^*)$. Each of the three equations in (17) coincides with one of the conditions in (16). For the last two conditions, this observation is trivial. For the first condition on the gradients, this can be observed by multiplying the first equation of (17) by $-(\mathbf{1}_N^\top \otimes \mathbf{I}_m)$ from the left to obtain

$$\mathbf{0} = (\mathbf{1}_N^\top \otimes \mathbf{I}_m) \nabla f(\mathbf{q}^*) - \mathbf{A}^\top \boldsymbol{\lambda}^*$$

with $\boldsymbol{\lambda}^* = -(\mathbf{1}_N^\top \otimes \mathbf{I}_m) \mathbf{g}^*$, which coincide with the first condition of (16). The resulting update at each agent from (3) is

$$\begin{aligned} \dot{\mathbf{q}}_i &= -\mathbf{A}^\top \mathbf{g}_i - \sum_{j \in \mathcal{N}_i} (\mathbf{q}_i - \mathbf{q}_j) - \sum_{j \in \mathcal{N}_i} (\mathbf{p}_i - \mathbf{p}_j) - \nabla f_i(\mathbf{q}_i) \\ \dot{\mathbf{p}}_i &= \sum_{j \in \mathcal{N}_i} (\mathbf{q}_i - \mathbf{q}_j) \\ \dot{\mathbf{g}}_i &= \mathbf{A} \mathbf{q}_i - \mathbf{b}. \end{aligned} \quad (18)$$

Note that in all previous examples, global convergence is ensured by Corollary 1.

IV. PHS-BASED OPTIMIZATION IN DISCRETE TIME

A. Discrete Gradient-Based PHS Discretization

The continuous time system in (3) can be implemented in discrete time by taking advantage of the PHS structure one more time. For instance, we make the following changes:

$$\begin{aligned} i) \quad \dot{\mathbf{x}}_i &\rightarrow \frac{\mathbf{x}_i^+ - \mathbf{x}_i}{\tau} \\ ii) \quad \nabla H_i(\mathbf{x}_i) &\rightarrow \bar{\nabla} H_i(\mathbf{x}_i, \mathbf{x}_i^+) \end{aligned} \quad (19)$$

where we denote with $\mathbf{x}_i = \mathbf{x}_i[k]$ at an arbitrary time step $k \in \mathbb{N}$ and $\mathbf{x}_i^+ = \mathbf{x}_i[k+1]$ the updated state at the next time step, and the object $\bar{\nabla} H_i$ is defined later in Definition 1. Note that the change (19)-i) corresponds to the finite difference approximation of the derivative $\dot{\mathbf{x}}_i$ between two time instants $t = k\tau$ and $t = (k+1)\tau$, where $\tau > 0$ represents a step-size parameter. If this was the only change to be applied, the discretization corresponds to the standard forward Euler method. For instance, applying this change to (8) leads to the standard gradient descent method

$$\mathbf{x}^+ = \mathbf{x} - \tau \nabla f(\mathbf{x})$$

from which $\tau > 0$ can be interpreted as a learning rate or step-size parameter. In general, upper bound constraints need to be applied to τ to ensure sufficiently a small τ , which leads to convergence.

To circumvent any constraint to τ , we introduce (19)-ii), such that our discretization no longer corresponds to a forward Euler step, where the symbol $\bar{\nabla}$ denotes the discrete gradient defined as follows.

Definition 1 (See [29]): Let $g : \mathbb{R}^n \rightarrow \mathbb{R}$ be a scalar differentiable function. Hence, $\bar{\nabla} g : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ is defined by the properties

$$\begin{aligned} \bar{\nabla} g(\mathbf{u}, \mathbf{v})^\top (\mathbf{v} - \mathbf{u}) &= g(\mathbf{v}) - g(\mathbf{u}) \\ \lim_{\mathbf{v} \rightarrow \mathbf{u}} \bar{\nabla} g(\mathbf{u}, \mathbf{v}) &= \nabla g(\mathbf{u}). \end{aligned} \quad (20)$$

While there are many possible ways to construct a discrete gradient, in this work, we consider

$$\bar{\nabla} g(\mathbf{u}, \mathbf{v}) = \int_0^1 \nabla g((1-s)\mathbf{u} + s\mathbf{v}) ds$$

which, for the case of quadratic functions $H_i(\mathbf{x}_i) = \mathbf{x}_i^\top \mathbf{x}_i/2$, results in the average

$$\bar{\nabla} H_i(\mathbf{x}_i, \mathbf{x}_i^+) = \frac{\mathbf{x}_i + \mathbf{x}_i^+}{2}.$$

With these changes, the discrete-time equivalent of the PHS (3) corresponds to

$$\begin{aligned} \frac{\mathbf{x}_i^+ - \mathbf{x}_i}{\tau} &= \mathbf{F}_i \bar{\nabla} H_i(\mathbf{x}_i, \mathbf{x}_i^+) + \phi_i(\bar{\nabla} H_i(\mathbf{x}_i, \mathbf{x}_i^+)) \\ &\quad + \sum_{j \in \mathcal{N}_i} \mathbf{M} (\bar{\nabla} H_i(\mathbf{x}_i, \mathbf{x}_i^+) - \bar{\nabla} H_j(\mathbf{x}_j, \mathbf{x}_j^+)). \end{aligned} \quad (21)$$

Note that the discrete-time system in (21) imposes an implicit equation of $\mathbf{x}_i^+$ at each time step. Hence, we require an additional regularity condition, analogous to the existence and uniqueness of solutions in the second part of Assumption 2, which, similar to [27] and [29], can be fulfilled for $\phi_i, \bar{\nabla} H_i$ Lipschitz.

Assumption 3: Given $\{\mathbf{x}_i\}_{i \in \mathcal{I}}$, there exists a sequence $\{\mathbf{x}_i^+\}_{i \in \mathcal{I}}$ complying (21).

The advantage of (21) is twofold. First, note that since

$$\bar{\nabla} H_i(\mathbf{x}_i^*, \mathbf{x}_i^*) = \nabla H_i(\mathbf{x}_i^*)$$

by construction, then the equilibrium of (21) is the same as the one from (3), namely, the unique point $\mathbf{x}_i^*$ complying (4). To verify this, evaluate (21) at $\mathbf{x}_i^+ = \mathbf{x}_i = \mathbf{x}_i^*$ for equilibrium $\mathbf{x}_i^*$ leading to

$$\begin{aligned} \mathbf{0} &= \mathbf{F}_i \bar{\nabla} H_i(\mathbf{x}_i^*, \mathbf{x}_i^*) + \phi_i(\bar{\nabla} H_i(\mathbf{x}_i^*, \mathbf{x}_i^*)) \\ &\quad + \sum_{j \in \mathcal{N}_i} \mathbf{M} (\bar{\nabla} H_i(\mathbf{x}_i^*, \mathbf{x}_i^*) - \bar{\nabla} H_j(\mathbf{x}_j^*, \mathbf{x}_j^*)) \\ &= \mathbf{F}_i \nabla H_i(\mathbf{x}_i^*) + \phi_i(\nabla H_i(\mathbf{x}_i^*)) \\ &\quad + \sum_{j \in \mathcal{N}_i} \mathbf{M} (\nabla H_i(\mathbf{x}_i^*) - \nabla H_j(\mathbf{x}_j^*)) \end{aligned}$$

which has the same unique solution as in continuous time due to Assumption 2.

Second, the asymptotic stability proof for (21) is practically the same as (3) regardless of $\tau > 0$. This is formalized in the following, where convergence guarantees are given for several τ regimes.

Theorem 1: Let Assumptions 2-2) and 3 hold, and consider (21). Moreover, assume that H_i and ϕ_i satisfy the conditions described in Corollary 1. Then, the unique equilibrium of (21) is globally asymptotically stable for any $\tau > 0$. If ϕ_i is strongly incrementally passive with constant $\alpha > 0$, Lipschitz with constant $L > 0$, and $H_i(\mathbf{x}_i) = \frac{1}{2} \mathbf{x}_i^\top \mathbf{x}_i$, then the unique equilibrium of (21) is globally exponentially stable, with rate $\lambda > 0$ from (35) (see Appendix II-C), for all $\tau \in (0, 2/(\|\mathbf{S}\| + L))$, where $\mathbf{S} = \mathbf{F} + \mathbf{L} \otimes \mathbf{M}$ and is locally exponentially stable for all $\tau > 0$.

The formulation presented so far from (21) is already explicitly useful for step-size-free optimization in a nondistributed setting, as evidenced in the following example.

Example 4: Consider Example 1 and apply the two design choices of $\mathbf{F}_1, \mathbf{M}, H_1$, and ϕ_1 on (21). Hence, two possible discrete-time systems for $\mathbf{x} := \mathbf{x}_1$ are obtained

$$\frac{\mathbf{x}^+ - \mathbf{x}}{\tau} = -\nabla f\left(\frac{\mathbf{x}^+ + \mathbf{x}}{2}\right) \quad (22)$$

$$\frac{\mathbf{x}^+ - \mathbf{x}}{\tau} = -\bar{\nabla} f(\mathbf{x}, \mathbf{x}^+). \quad (23)$$

Both systems are ensured to converge to the correct equilibrium regardless of $\tau > 0$. While (23) was discussed in [29], requiring computing the discrete gradient of the cost function, which can be expensive, (22) uses the usual gradient, which is more standard and readily available.

From the point of view of the PHS, the intuition behind this approach is that the PHS properties are preserved after discretization. In particular, (21) has a similar port structure as its continuous-time counterpart. Therefore, evaluating the rate of change of the combined storage function, analogous to $\dot{H}(\mathbf{x})$ in continuous time, leads to

$$\frac{H(\mathbf{x}^+) - H(\mathbf{x})}{\tau} = \bar{\nabla} H(\mathbf{x}, \mathbf{x}^+)^\top \left(\frac{\mathbf{x}^+ - \mathbf{x}}{\tau} \right) \leq \mathbf{y}^\top \phi(\mathbf{y})$$

with output $\mathbf{y} = \bar{\nabla} H(\mathbf{x}, \mathbf{x}^+)$, where we applied the defining property of the discrete gradient (20) and followed a procedure analogous to that in the proof of Proposition 1. Hence, (21) is passive, regardless of the parameter τ , since properties in (20) do not depend on it. As a result, the convergence proof in Theorem 1 is very similar to that of Corollary 1 for the continuous time case.

Moreover, it is important to note that Theorem 1 already establishes the result of an algorithm capable of solving (1), even with arbitrarily large step sizes. Note that to implement the discrete-time system in (21), each agent $i \in \mathcal{I}$ needs to solve its implicit equations for $\mathbf{x}_i^+$, as shown in Algorithm 1. While this is feasible, it can be expensive from the point of view of communication overhead since the implicit equations also depend on the neighbor values $\mathbf{x}_j^+$. Hence, distributed equation solvers can be used at each time step. Therefore, we propose an additional novel modification to the discretization methodology to remove the dependence on the neighbors $\{\mathbf{x}_i^+\}_{i \in \mathcal{I}}$ in the implicit equations at agent $i \in \mathcal{I}$.

B. MID Method: Fully Distributed PHS in Discrete Time

In the following, we show a simple modification of (21), which results in a fully distributed algorithm that does not require additional distributed equation solvers. Aiming to obtain

Algorithm 1: Discrete Gradient-Based PHS at Agent i .

Require: Local Hamiltonian function $H_i : \mathbb{R}^m \rightarrow \mathbb{R}$ and its discrete gradient.

Require: Local mapping functions $\mathbf{F}_i$ and ϕ_i .

Require: Coupling matrix $\mathbf{M}$.

Require: Arbitrary initial state $\mathbf{x}_i(0)$.

Require: Arbitrary step size $\tau > 0$.

Require: Termination tolerance $\epsilon > 0$.

- 1: Initialize $\mathbf{x}_i$.
 - 2: **while** not converged **do**
 - 3: Exchange $\mathbf{x}_i$ with neighbors $j \in \mathcal{N}_i$.
 - 4: Use a distributed equation solver to obtain a solution to the implicit equation for $\mathbf{x}_i^+$

$$\frac{\mathbf{x}_i^+ - \mathbf{x}_i}{\tau} = \mathbf{F}_i \bar{\nabla} H_i(\mathbf{x}_i, \mathbf{x}_i^+) + \phi_i(\bar{\nabla} H_i(\mathbf{x}_i, \mathbf{x}_i^+))$$

$$+ \sum_{j \in \mathcal{N}_i} \mathbf{M} (\bar{\nabla} H_i(\mathbf{x}_i, \mathbf{x}_i^+) - \bar{\nabla} H_j(\mathbf{x}_j, \mathbf{x}_j^+)).$$
 - 5: **if** $\|\mathbf{x}_i^+ - \mathbf{x}_i\| \leq \epsilon$ **then**
 - 6: Terminate loop.
 - 7: **end if**
 - 8: Update state: $\mathbf{x}_i \leftarrow \mathbf{x}_i^+$.
 - 9: **end while**
 - 10: **return** Local solution $\mathbf{x}_i$.
-

a solution of the program in (1), we set (21) with the choice $H_i(\mathbf{x}_i) = \mathbf{x}_i^\top \mathbf{x}_i/2$, $\mathbf{F}_i = \mathbf{0}$, which coincide with the design in (14). These choices result in

$$\frac{\mathbf{x}_i^+ - \mathbf{x}_i}{\tau} = \sum_{j \in \mathcal{N}_i} \mathbf{M} \left(\frac{\mathbf{x}_i + \mathbf{x}_i^+}{2} - \frac{\mathbf{x}_j + \mathbf{x}_j^+}{2} \right) + \phi_i \left(\frac{\mathbf{x}_i + \mathbf{x}_i^+}{2} \right). \quad (24)$$

We remove the occurrence of $\mathbf{x}_j^+$ in the previous equation by making the following modification, which we refer to as the MID:

$$\frac{\mathbf{x}_i^+ - \mathbf{x}_i}{\tau} = \sum_{j \in \mathcal{N}_i} \mathbf{M} (\mathbf{x}_i^+ - \mathbf{x}_j) + \phi_i \left(\frac{\mathbf{x}_i + \mathbf{x}_i^+}{2} \right) \quad (25)$$

where the disagreement difference is computed with the future state for agent i and the present state for agent j . This modification allows each agent to locally solve (25) without requiring further communication between its neighbors besides sharing $\mathbf{x}_i$ once at each time step. It is important to note that this modification has the same equilibrium point as (24). This can be verified by replacing $\mathbf{x}_i = \mathbf{x}_i^+ = \mathbf{x}_i^*$ in both (24) and (25) resulting in the same condition for both cases

$$\mathbf{0} = \sum_{j \in \mathcal{N}_i} \mathbf{M} (\mathbf{x}_i^* - \mathbf{x}_j^*) + \phi_i(\mathbf{x}_i^*).$$

Henceforth, MID applied to the continuous time PHS in (14) results in

$$\frac{\mathbf{q}_i^+ - \mathbf{q}_i}{\tau} = - \sum_{j \in \mathcal{N}_i} (\mathbf{q}_i^+ - \mathbf{q}_j + \mathbf{p}_i^+ - \mathbf{p}_j) - \nabla f_i \left(\frac{\mathbf{q}_i^+ + \mathbf{q}_i}{2} \right)$$

Algorithm 2: MID-Based Algorithm at Agent i .**Require:** Local cost function $f_i : \mathbb{R}^m \rightarrow \mathbb{R}$ and its gradient.**Require:** Initial states $\mathbf{q}_i[0], \mathbf{p}_i[0]$.**Require:** step size $\tau > 0$.**Require:** Termination tolerance $\epsilon > 0$.

- 1: Initialize $\mathbf{q}_i$ and $\mathbf{p}_i$.
- 2: **while** not converged **do**
- 3: Exchange $\mathbf{q}_i$ and $\mathbf{p}_i$ with neighbors $j \in \mathcal{N}_i$.
- 4: Solve the implicit equations for $\mathbf{q}_i^+, \mathbf{p}_i^+$

$$\frac{\mathbf{q}_i^+ - \mathbf{q}_i}{\tau} = - \sum_{j \in \mathcal{N}_i} (\mathbf{q}_i^+ - \mathbf{q}_j + \mathbf{p}_i^+ - \mathbf{p}_j) - \nabla f_i \left(\frac{\mathbf{q}_i^+ + \mathbf{q}_i}{2} \right)$$

$$\frac{\mathbf{p}_i^+ - \mathbf{p}_i}{\tau} = \sum_{j \in \mathcal{N}_i} (\mathbf{q}_i^+ - \mathbf{q}_j).$$

- 5: **if** $\|\mathbf{q}_i^+ - \mathbf{q}_i\| \leq \epsilon$ and $\|\mathbf{p}_i^+ - \mathbf{p}_i\| \leq \epsilon$ **then**
- 6: Terminate loop.
- 7: **end if**
- 8: Update states: $\mathbf{q}_i \leftarrow \mathbf{q}_i^+, \mathbf{p}_i \leftarrow \mathbf{p}_i^+$.
- 9: **end while**
- 10: **return** Local solution $\mathbf{q}_i, \mathbf{p}_i$.

$$\frac{\mathbf{p}_i^+ - \mathbf{p}_i}{\tau} = \sum_{j \in \mathcal{N}_i} (\mathbf{q}_i^+ - \mathbf{q}_j). \quad (26)$$

Algorithm 2 describes how (26) can be executed in practice.

Now, we study several nontrivial properties of (26), providing convergence guarantees for several regimes of τ . All these properties can be derived around a linear matrix inequality (LMI) formulation as in the following.

Theorem 2: Let Assumptions 1 and 2-2) hold and consider (26). Let f in (11) be strongly convex with constant $\mu > 0$ and have Lipschitz continuous gradient with constant $L > 0$. Let the matrix definitions for $\Lambda(\tau), \mathbf{S}_r(\tau)$ in (39) and (46) be placed in Appendix II for readability. Then, given $\tau > 0$, the unique equilibrium of (26) is globally asymptotically stable if the following LMI is satisfied:

$$\mathbf{P}(\tau) := \begin{bmatrix} \Lambda(\tau) & \mathbf{P}_{12} \\ \mathbf{P}_{12}^\top & \mathbf{P}_{22} \end{bmatrix} \succ \mathbf{0} \quad (27a)$$

$$\begin{bmatrix} \mathbf{U} & \mathbf{P}_{12}^\top \\ \mathbf{P}_{12} & \mathbf{I} \end{bmatrix} \succeq \mathbf{0}, \quad u > 0 \quad (27b)$$

$$\mathbf{P}(\tau)\mathbf{S}_r(\tau) + \mathbf{S}_r(\tau)^\top \mathbf{P}(\tau) + \mathbf{S}_\gamma(\tau, \varepsilon) \preceq -u \begin{bmatrix} \mathbf{I}, \mathbf{0} \\ \mathbf{0}, \mathbf{0} \end{bmatrix} \quad (27c)$$

for some $\varepsilon > 0$ and decision variables $\mathbf{U}, \mathbf{P}_{12}, \mathbf{P}_{22} \in \mathbb{R}^{Nm \times Nm}$, $u \in \mathbb{R}$.

The proof for Theorem 2 is given in Appendix II. Note that Theorem 2 does not rely on an assumption similar to Assumption

3. Instead, we show the existence of a unique solution to the implicit equation in (26) directly in the proofs.

Remark 2: The number of variables in the LMI is on the order of $(mN)^2$. Since the cost of a single Newton step in standard interior-point methods grows cubically with the number of variables, the overall computational complexity is $O((mN)^6)$, where $O(\bullet)$ denotes the standard big- O notation [30]. Similarly, the memory requirements scale as $O((mN)^4)$, as storing and factorizing the Hessians requires comparable resources. However, these estimates depend strongly on the specific algorithm and implementation used. Moreover, the matrices appearing in the LMI, detailed in Appendix II-D, involve nontrivial products and powers of the Laplacian $\mathbf{L}$ and the signless Laplacian-like matrix $\mathbf{Q}$. Exploiting sparsity in these structures is not straightforward and may require further analysis to effectively reduce complexity. Nevertheless, the following results provide convergence guarantees that do not require solving the LMI explicitly, thereby establishing properties that hold regardless of whether the LMI is computed in practice.

Corollary 2: Let Assumptions 1 and 2-2) hold, and consider (26) with $\mathbf{x}^* = [\mathbf{x}_i]_{i \in \mathcal{I}}$ its unique equilibrium. Let f in (11) be strongly convex and have Lipschitz continuous gradient. Let $\mathbf{A}$ be the 0 – 1 adjacency matrix of $\mathcal{G}$ and $\mathbf{D} = \text{diag}(\mathbf{A}\mathbf{1})$ denote the degree matrix. Then, if

$$\mathbf{D}^2 - \mathbf{A}^2 \succeq \mathbf{0} \quad (28)$$

it follows that $\mathbf{x}^*$ is globally asymptotically stable for all $\tau > 0$. In particular, (28) holds if $\mathcal{G}$ is a cycle or a fully connected graph of arbitrary size.

In the following, we show that even if the graph does not comply with (28), one can find a guaranteed upper bound for the time step $\tau > 0$, which holds for any connected undirected graphs.

Corollary 3: Let Assumptions 1 and 2-2) hold, and consider (26) with $\mathbf{x}^* = [\mathbf{x}_i]_{i \in \mathcal{I}}$ its unique equilibrium. Let f in (11) be strongly convex with constant $\mu > 0$ and have Lipschitz continuous gradient. Moreover, let

$$\tau < \frac{\mu}{\|\mathbf{D}^2 - \mathbf{A}^2\|}.$$

Then, $\mathbf{x}^*$ is globally asymptotically stable for all connected undirected graphs $\mathcal{G}$.

Conversely, the following result shows that, for all connected undirected graphs, there exists a strictly positive threshold τ^* such that any step size $\tau \geq \tau^*$ guarantees convergence for quadratic problems in (1). This property is distinctive, as it implies that increasing the step size beyond a certain bound preserves stability, contrary to the behavior commonly observed in existing methods.

Corollary 4: Let Assumptions 1 and 2-2) hold, and consider (26) with $\mathbf{x}^* = [\mathbf{x}_i]_{i \in \mathcal{I}}$ its unique equilibrium. Let $f_i(\mathbf{x}_i) = \mathbf{x}_i^\top \mathbf{H}_i \mathbf{x}_i / 2 + \mathbf{b}_i^\top \mathbf{x}_i$, with $\mathbf{H}_i \succ \mathbf{0}$. Then, there exists $\tau^* > 0$ such that, for all $\tau \geq \tau^*$, the equilibrium point $\mathbf{x}^*$ is globally asymptotically stable for every connected undirected graph $\mathcal{G}$.

Remark 3: It is important to clarify the computational and communication costs of the proposed methods relative to existing approaches, as their balance depends on the specific platform and application. In many distributed optimization settings, message exchange between agents dominates local computation time, although this is not universal. Typical end-to-end communication latencies in modern wired deterministic networks (e.g., IEEE TSN or DetNet) range from a few microseconds to below 1 ms, while wireless technologies, such as Wi-Fi or 5-G exhibit latencies, between 1 and 10 ms [31]. In contrast, in our numerical experiments, solving the implicit equations locally required runtimes on the order of 10^{-4} s on average (across all agents), confirming that communication is the dominant cost in our setup. This makes it preferable, in practice, to rely on local computation rather than additional communication, as local processing reduces network usage and congestion. Hence, it is reasonable to measure performance primarily in terms of communication rounds. Consequently, although (21) introduces additional communication per iteration, the MID system in (25) performs its implicit computations locally and, therefore, does not increase communication overhead.

V. NUMERICAL EXAMPLES

In this section, we test the MID-based algorithm in (26) in several scenarios and compare it with other works in the literature. For the implementation, we used the function `fsolve` in MATLAB so that each agent can solve its local implicit equation at each time step. We used the solution of each `fsolve` instance as the initial condition for the next one to save computing time. This resulted in an average of 7.76 iterations per `fsolve` to reach a relative error below 10^{-10} , accounting for about 250×10^{-4} seconds per call on a standard laptop. Moreover, in this section, discrete-time will be measured in terms of communication rounds.

We start by testing the proposal's performance in a practical data analytics problem, similar to that in [32, Sect. 6]. Each agent $i \in \mathcal{I}$ is equipped with $d_i \in \mathbb{N}$ points $\mathbf{m}_{i,1}, \dots, \mathbf{m}_{i,d_i} \in \mathbb{R}^{m-1}$ with binary labels $l_{i,1}, \dots, l_{i,d_i} \in \{-1, 1\}$. The cost function is then of the form (1) with

$$f_i(\boldsymbol{\theta}) = \sum_{\mu=1}^{d_i} \log(1 + \exp(-l_{i,\mu}(\tilde{\boldsymbol{\theta}}^\top \mathbf{m}_{i,\mu} + \theta))) + \frac{C\|\boldsymbol{\theta}\|^2}{2N}$$

where $\theta \in \mathbb{R}$ and $\boldsymbol{\theta} = [\tilde{\boldsymbol{\theta}}^\top, \theta]^\top$ is the optimization variable, and $C > 0$ is a regularization parameter, making $f_i(\boldsymbol{\theta})$ strongly convex. For this example, we picked $N = 10$ agents, connected in an Erdős-Rényi graph $\mathcal{G}$ with parameter 0.4 [32]. Moreover, we set $m = 3$, $d_i = 10$ for all agents and $C = 0.1$, with randomly chosen $\{\mathbf{m}_{i,\mu}\}_{\mu=1}^{d_i}$. In addition, initial conditions for the state at each agent were chosen randomly. For the following examples, we compare $\|\mathbf{q}[k] - (\mathbf{1} \otimes \mathbf{I}_m)\boldsymbol{\theta}^*\|$, which measures the error between all $\mathbf{q}_i[k]$ with respect to the optimal solution $\boldsymbol{\theta}^*$.

We start by comparing the convergence speed between MID and explicit forward Euler discretization by performing several simulations with 200 values of τ uniformly distributed in the interval $[0, 10]$. The figure of merit for the convergence speed

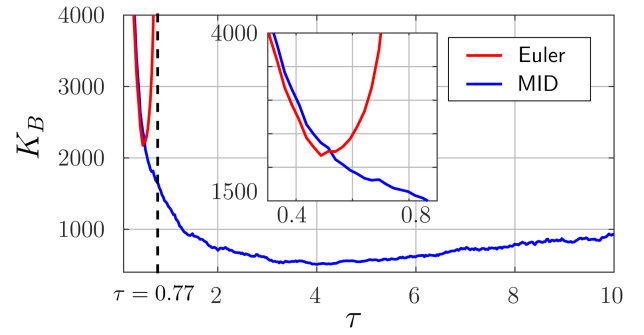


Fig. 1. Comparison of the number of iterations K_B required to reach $\|\mathbf{q}[k] - (\mathbf{1} \otimes \mathbf{I}_m)\boldsymbol{\theta}^*\| \leq B$ with $B = 10^{-6}$. The proposed MID allows the reduction of the number of iterations by increasing τ , with an optimal value around $\tau = 4$. In contrast, increasing τ for Euler discretization does not feature such great reduction and leads to divergence at around $\tau = 0.77$.

is computed as the number of iterations $K_B \in \mathbb{N}$ required such that the condition $\|\mathbf{q}[k] - (\mathbf{1} \otimes \mathbf{I}_m)\boldsymbol{\theta}^*\| \leq B$ is satisfied from that point forward. Here, we choose $B = 10^{-6}$ as an accuracy bound representing when the error between the system output and true optimal is sufficiently close for illustrative purposes. The results are shown in Fig. 1, where it is observed that both methods achieve similar convergence speed for small τ . However, near $\tau \approx 0.77$, Euler discretization becomes unstable, while the sampling step can be increased further for MID. This allows us to decrease K_B for our method, reaching a minimum near $\tau \approx 4$. From this point, K_B increases with τ . This shows that allowing bigger values of τ can increase convergence speed compared to classical discretization methods, for which increasing τ leads to divergence. Choosing the optimal τ for maximum speed is not straightforward, as it may depend on the network structure and objective function properties. However, if the network is static, the best τ can be determined offline through simulations. Developing a distributed method to compute the optimal τ remains an open problem. For the same experiment, we analyze the effect of increasing the network size N . To maintain a consistent baseline, we preserve the original network structure and progressively add agents in a cyclic manner. The behavior of K_B as a function of N for different values of τ is shown in Fig. 2. As expected, the required number of iterations K_B increases with N . Moreover, the choice $\tau = 4$ consistently yields faster convergence, remaining below the curves corresponding to other step-size values even for networks as large as $N = 120$.

Moreover, we compare with other methods in the literature for consensus distributed optimization. In particular, for discrete time, we compare with the discrete-time Gradient Tracking (DTGT) [1], the DIGing method [11], network independent step sizes (NIDS) [21], and the decentralized exact first-order algorithm (EXTRA) [33]. In these cases, a step-size parameter τ is used, which needs to be sufficiently small. Moreover, we compare with Euler discretization of continuous-time approaches, such as the Continuous Time Gradient Tracking (CTGT) [32], and the coordination algorithm (COOR) from [34]. In both cases, we assume communication at each time step for fairness in all methods. In addition, the time step for these methods is assigned

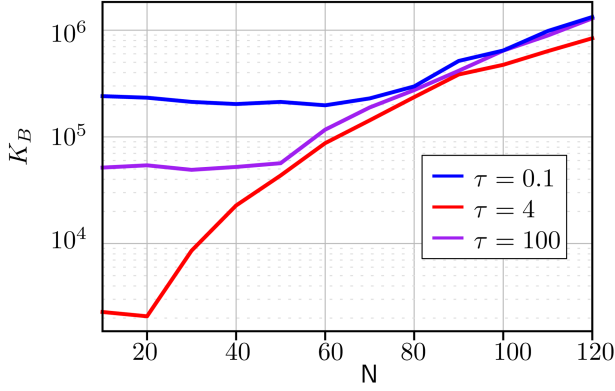


Fig. 2. Comparison of the number of iterations K_B required to reach $\|q[k] - (\mathbf{I} \otimes \mathbf{I}_m)\theta^*\| \leq B$ with $B = 10^{-6}$ for the proposed MID as the number of agents N in the network is increased, for different values of τ .

to τ . Finally, we also compare with Euler discretization of the PHS in (3).

For comparison, we used quadratic cost functions, which allows us to verify a less conservative LMI for any graph, as stated in the following result.

Corollary 5: Let Assumptions 1 and 2-2) hold, and consider (26) with $\mathbf{x}^* = [\mathbf{x}_i]_{i \in \mathcal{I}}$ its unique equilibrium. Given $\tau > 0$, $\mathcal{G}$, as well as $f_i(\mathbf{x}_i) = \mathbf{x}_i^\top \mathbf{H}_i \mathbf{x}_i / 2 + \mathbf{b}_i^\top \mathbf{x}_i$, $\mathbf{x}^*$ is globally asymptotically stable if the LMI comprised by (27a), $u > 0$ and

$$\mathbf{P}(\tau)\mathbf{S}_r(\tau) + \mathbf{S}_r(\tau)^\top \mathbf{P}(\tau) + \mathbf{S}_h(\tau) \preceq -u \begin{bmatrix} \mathbf{I}, 0 \\ 0, 0 \end{bmatrix} \quad (29)$$

is satisfied, with $\mathbf{S}_h(\tau)$ defined in (50).

Moreover, for this comparison, we looked for the best value of $\tau > 0$, leading to the fastest behavior in each method tested. These values of τ were obtained through an empirical sweep over admissible step sizes, identifying the largest values within the stability region and selecting those that provided the fastest convergence. For MID, Euler discretization, DTGT, CTGT, DIGing, and COOR, the values of τ resulted in approximately 3.78, 0.71, 0.58, 0.65, 0.05, 0.078, 1.24, and 0.55, respectively. Trajectories for each case are shown in Fig. 3, where all methods converge to the global optimum, with MID having a clear advantage in speed, due to being able to use a larger step size. All methods compared here have similar per-step complexity, as none require solving implicit equations. Consequently, on average, our method incurs approximately 14.7 times higher local computation time. However, as discussed in Remark 3, this overhead may be acceptable depending on the application context.

Fig. 4 shows the same scenario as before, but now with $\tau = 10$, showing that all other methods diverge, while our proposal manages to converge. For both $\tau = 3.78, 10$, we verified the feasibility of the LMI in (29), which guarantees convergence.

To further stress our proposal, we performed simulations with $\tau = 1, 10, 100, 1000$. In all cases, the LMI in (29) was feasible. The convergence results are shown in Fig. 5, where

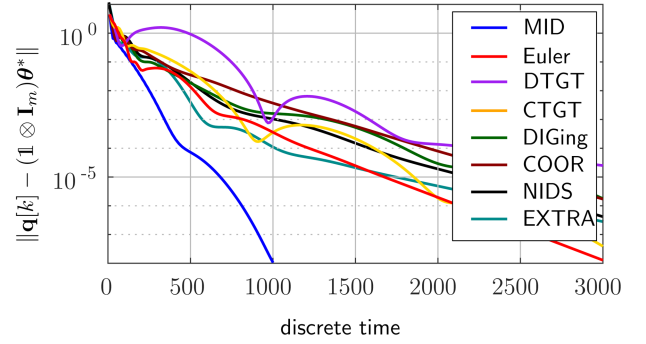


Fig. 3. Comparison of MID, Euler discretization, as well as DTGT [1], CTGT [32], DIGing [11], COOR [34], NIDS [21], and EXTRA [33] for a quadratic optimization problem. The fastest values of τ were found to be approximately 3.78, 0.71, 0.58, 0.65, 0.05, 0.078, 1.24, and 0.55. The proposal MID has a clear advantage in terms of speed of convergence.

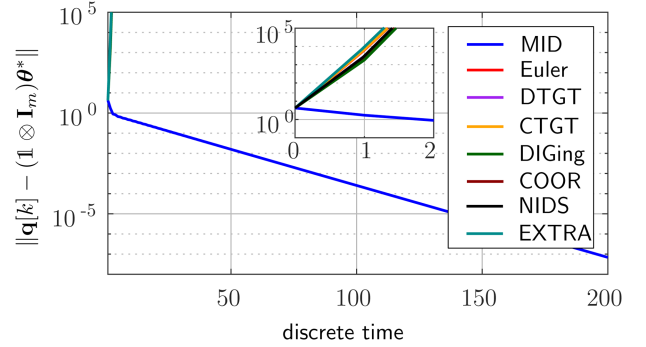


Fig. 4. Comparison of MID, Euler discretization, as well as DTGT [1], CTGT [32], DIGing [11], COOR [34], NIDS [21], and EXTRA [33] for a quadratic optimization problem with $\tau = 10$.

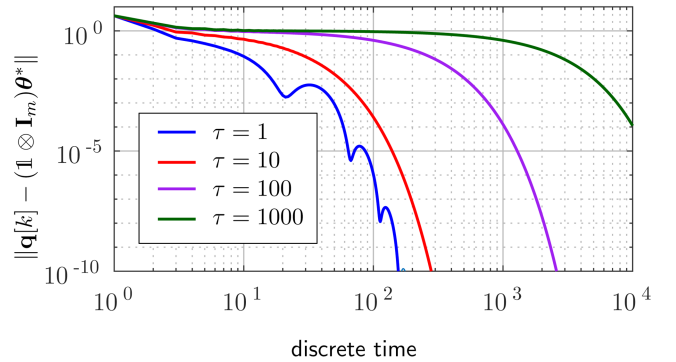


Fig. 5. Comparison of MID for a quadratic optimization problem with $\tau = 1, 10, 100, 1000$.

it is observed that even in the extreme case of $\tau = 1000$; our proposal converges despite a slower convergence rate.

VI. CONCLUSION

In this work, we introduced a novel distributed optimization approach based on the PHS framework. By leveraging the MID

of a continuous-time PHS, we demonstrated how this system-based strategy enhances convergence speeds while ensuring stability for arbitrarily large step sizes in specific scenarios. This capability represents an unprecedented feature in the distributed optimization literature. Our method used the inherent structure of the PHS to ensure convergence by aligning the system's equilibrium with the solution of the optimization problem, even when transitioning from continuous time to discrete time. The MID scheme enabled agents in the network to resolve nonlinear implicit equations locally, reducing communication overhead while maintaining convergence guarantees in formalized scenarios. Furthermore, we experimentally demonstrated that our approach performs effectively in general cases with large step sizes, significantly outperforming traditional methods that are constrained by step-size sensitivity. Numerical experiments highlighted the robustness and efficiency of our method, especially in scenarios where conventional approaches fail due to limitations related to step-size selection, thereby achieving faster convergence and advancing the state of the art in distributed optimization. As future work, we plan to investigate whether the proposed approach remains valid under less conservative conditions on the graph and the step size, as well as to study its robustness to communication delays and network imperfections.

APPENDIX I CONVEX FUNCTIONS

This section provides some useful facts used in the rest of this article. We refer the reader to [30] for standard definitions of convex functions.

Proposition 2 (See [30, Sect. 9.1.2]): Let $g : \mathbb{R}^n \rightarrow \mathbb{R}$ be a strongly convex differentiable function. Then,

$$g(\mathbf{v}) \geq g(\mathbf{u}) + \nabla g(\mathbf{u})^\top (\mathbf{v} - \mathbf{u}) + \frac{\mu}{2} \|\mathbf{v} - \mathbf{u}\|^2$$

for arbitrary $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ and some $\mu > 0$ referred to as the strong convexity constant. As a consequence,

$$(\mathbf{v} - \mathbf{u})^\top (\nabla g(\mathbf{v}) - \nabla g(\mathbf{u})) \geq \mu \|\mathbf{v} - \mathbf{u}\|^2.$$

Lemma 1: Let $H : \mathbb{R}^n \rightarrow \mathbb{R}$ be a differentiable function. Moreover, let some fixed $\mathbf{x}^* \in \mathbb{R}^n$ and $V_{\mathbf{x}^*} : \mathbb{R}^n \rightarrow \mathbb{R}$ be defined as

$$V_{\mathbf{x}^*}(\mathbf{x}) = H(\mathbf{x}) - H(\mathbf{x}^*) - \nabla H(\mathbf{x}^*)^\top (\mathbf{x} - \mathbf{x}^*).$$

Then, if H is strongly convex with constant $\mu > 0$, $V_{\mathbf{x}^*}(\mathbf{x})$ satisfies the radially unboundedness condition given by

$$V_{\mathbf{x}^*}(\mathbf{x}) \geq \frac{\mu}{2} \|\mathbf{x} - \mathbf{x}^*\|^2.$$

In addition, if $\nabla H(\mathbf{x})$ is Lipschitz with constant $L > 0$, $V_{\mathbf{x}^*}(\mathbf{x})$ satisfies

$$V_{\mathbf{x}^*}(\mathbf{x}) \leq \frac{L}{2} \|\mathbf{x} - \mathbf{x}^*\|^2.$$

Proof: For the first part of the lemma, use Proposition 2 with $g = H$, $\mathbf{v} = \mathbf{x}$, and $\mathbf{u} = \mathbf{x}^*$

$$V_{\mathbf{x}^*}(\mathbf{x}) = H(\mathbf{x}) - H(\mathbf{x}^*) - \nabla H(\mathbf{x}^*)^\top (\mathbf{x} - \mathbf{x}^*)$$

$$\begin{aligned} &\geq \left(H(\mathbf{x}^*) + \nabla H(\mathbf{x}^*)^\top (\mathbf{x} - \mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x} - \mathbf{x}^*\|^2 \right) \\ &\quad - H(\mathbf{x}^*) - \nabla H(\mathbf{x}^*)^\top (\mathbf{x} - \mathbf{x}^*) = \frac{\mu}{2} \|\mathbf{x} - \mathbf{x}^*\|^2. \end{aligned}$$

For the second part, from a Taylor expansion of $V_{\mathbf{x}^*}(\mathbf{x})$ around $\mathbf{x}^*$ and the mean value theorem, it follows that there exists $\mathbf{z} \in \mathbb{R}^n$ such that

$$\begin{aligned} V_{\mathbf{x}^*}(\mathbf{x}) &= V_{\mathbf{x}^*}(\mathbf{x}^*) + \nabla V_{\mathbf{x}^*}(\mathbf{x}^*)^\top (\mathbf{x} - \mathbf{x}^*) \\ &\quad + \frac{1}{2} (\mathbf{x} - \mathbf{x}^*)^\top \nabla^2 V_{\mathbf{x}^*}(\mathbf{z}) (\mathbf{x} - \mathbf{x}^*) \\ &= \frac{1}{2} (\mathbf{x} - \mathbf{x}^*)^\top \nabla^2 H(\mathbf{z}) (\mathbf{x} - \mathbf{x}^*) \leq \frac{L}{2} \|\mathbf{x} - \mathbf{x}^*\|^2. \end{aligned}$$

□

APPENDIX II PROOFS

A. Proof of Proposition 1

Take arbitrary $\mathbf{v} \in \mathbb{R}^{Nm}$ with $\mathbf{v} \neq \mathbf{0}$. Recall $\mathbf{M} + \mathbf{M}^\top \preceq \mathbf{0}$ and $\mathbf{F} + \mathbf{F}^\top \preceq \mathbf{0}$ by assumption. Define $\mathbf{w} = (\mathbf{E}^\top \otimes \mathbf{I}_m) \mathbf{v}$, where $\mathbf{E}$ is an incidence matrix of $\mathcal{G}$ satisfying $\mathbf{L} = \mathbf{E}\mathbf{E}^\top$. Hence,

$$\begin{aligned} \mathbf{v}^\top (\mathbf{F} + \mathbf{L} \otimes \mathbf{M}) \mathbf{v} &= \mathbf{v}^\top \mathbf{F} \mathbf{v} + \mathbf{v}^\top (\mathbf{E}\mathbf{E}^\top \otimes \mathbf{M}) \mathbf{v} \\ &\leq \mathbf{v}^\top (\mathbf{E} \otimes \mathbf{I}_m) (\mathbf{I} \otimes \mathbf{M}) (\mathbf{E}^\top \otimes \mathbf{I}_m) \mathbf{v} \\ &\leq \mathbf{w}^\top (\mathbf{I} \otimes \mathbf{M}) \mathbf{w} \leq 0. \end{aligned}$$

Therefore, $\mathbf{F} + \mathbf{L} \otimes \mathbf{M}$ has symmetric part negative semidefinite. Hence, the proposition follows by taking $\dot{H} = \nabla H^\top \dot{\mathbf{x}}$ along (5).

B. Proof of Corollary 1

Let $\mathbf{x}^*$ denote the equilibrium of (5). To simplify the notation, let $\nabla H^* = \nabla H(\mathbf{x}^*)$. Set

$$V_{\mathbf{x}^*}(\mathbf{x}) = H(\mathbf{x}) - H(\mathbf{x}^*) - \nabla H(\mathbf{x}^*)^\top (\mathbf{x} - \mathbf{x}^*) \quad (30)$$

as a Lyapunov function candidate. Note that $V_{\mathbf{x}^*}(\mathbf{x})$ satisfies the required conditions for this purpose. First, $V_{\mathbf{x}^*}(\mathbf{x}^*) = 0$. Second, since H is strongly convex, Lemma 1 in Appendix A is used to conclude that $V_{\mathbf{x}^*}(\mathbf{x})$ is always positive and radially unbounded. Consider $\mathbf{S} = (\mathbf{F} + \mathbf{L} \otimes \mathbf{M})$. Then,

$$\begin{aligned} \dot{V}_{\mathbf{x}^*} &= (\nabla H - \nabla H^*)^\top \dot{\mathbf{x}} \\ &= (\nabla H - \nabla H^*)^\top (\dot{\mathbf{x}} - \mathbf{S} \nabla H^* - \phi(\nabla H^*)) \\ &= (\nabla H - \nabla H^*)^\top (\mathbf{S} \nabla H + \phi(\nabla H) - \mathbf{S} \nabla H^* - \phi(\nabla H^*)) \\ &= (\nabla H - \nabla H^*)^\top \mathbf{S} (\nabla H - \nabla H^*) \\ &\quad + (\nabla H - \nabla H^*)^\top (\phi(\nabla H) - \phi(\nabla H^*)) \leq 0 \end{aligned}$$

where we used the fact that $\mathbf{S} + \mathbf{S}^\top \preceq \mathbf{0}$ from the proof of Proposition 1. Moreover, we used the incremental passivity property from (6) in the last step. Furthermore, note that $\dot{V}_{\mathbf{x}^*} = 0$ only when $\nabla H(\mathbf{x}) = \nabla H(\mathbf{x}^*)$, which occurs only at the equilibrium

point. Henceforth, $\dot{V}_{\mathbf{x}^*} < 0$ for all $\mathbf{x} \neq \mathbf{x}^*$. Therefore, $\mathbf{x}^*$ is a globally asymptotically stable equilibrium.

Now, if (7) is satisfied then, following from the previous case:

$$\begin{aligned} \dot{V}_{\mathbf{x}^*} &\leq (\nabla H - \nabla H^*)^\top (\phi(\nabla H) - \phi(\nabla H^*)) \\ &\leq -\alpha \|\nabla H - \nabla H^*\|^2 = -\alpha \frac{\|\mathbf{x} - \mathbf{x}^*\|^2}{\|\mathbf{x} - \mathbf{x}^*\|^2} \|\nabla H - \nabla H^*\|^2 \\ &\leq \frac{-\alpha}{\|\mathbf{x} - \mathbf{x}^*\|^2} ((\mathbf{x} - \mathbf{x}^*)^\top (\nabla H - \nabla H^*))^2 \\ &\leq \frac{-\alpha \mu^2}{\|\mathbf{x} - \mathbf{x}^*\|^2} (\|\mathbf{x} - \mathbf{x}^*\|^2)^2 \\ &= -\alpha \mu^2 \|\mathbf{x} - \mathbf{x}^*\|^2 \leq -\frac{2\alpha \mu^2}{L} V_{\mathbf{x}^*} \end{aligned}$$

where we used strong convexity of H with constant μ and Lipschitzness of H with constant L as in Lemma 1 in Appendix A. Henceforth, exponential stability of $\mathbf{x}^*$ follows.

C. Proof of Theorem 1

First, write the discrete-time system (21) in compact form, similarly as done before with the continuous-time PHS from (5)

$$\frac{\mathbf{x}^+ - \mathbf{x}}{\tau} = (\mathbf{F} + \mathbf{L} \otimes \mathbf{M}) \bar{\nabla} H(\mathbf{x}, \mathbf{x}^+) + \phi(\bar{\nabla} H(\mathbf{x}, \mathbf{x}^+)) \quad (31)$$

with $\mathbf{x} = [\mathbf{x}_i]_{i \in \mathcal{I}}$, $\mathbf{x}^+ = [\mathbf{x}_i^+]_{i \in \mathcal{I}}$. Moreover, denote $\mathbf{S} = \mathbf{F} + \mathbf{L} \otimes \mathbf{M}$. Similarly, as in the proof Corollary 1, set $V_{\mathbf{x}^*}(\mathbf{x})$ defined in (30) as a Lyapunov function candidate. Then, due to the definition of the discrete gradient, it follows that

$$\begin{aligned} V_{\mathbf{x}^*}(\mathbf{x}^+) - V_{\mathbf{x}^*}(\mathbf{x}) &= \bar{\nabla} V_{\mathbf{x}^*}(\mathbf{x}, \mathbf{x}^+)^\top (\mathbf{x}^+ - \mathbf{x}) \\ &= \tau \bar{\nabla} V_{\mathbf{x}^*}(\mathbf{x}, \mathbf{x}^+)^\top (\mathbf{S} \bar{\nabla} H(\mathbf{x}, \mathbf{x}^+) + \phi(\bar{\nabla} H(\mathbf{x}, \mathbf{x}^+))). \end{aligned}$$

Then, use

$$\bar{\nabla} V_{\mathbf{x}^*}(\mathbf{x}, \mathbf{x}^+) = \bar{\nabla} H(\mathbf{x}, \mathbf{x}^+) - \bar{\nabla} H(\mathbf{x}^*, \mathbf{x}^*)$$

since $\bar{\nabla} H(\mathbf{x}^*, \mathbf{x}^*) = \nabla H(\mathbf{x}^*)$. From this point, the proof follows the same steps as those in the proof of Corollary 1 with the discrete gradient of H in place of the gradient to show that $V_{\mathbf{x}^*}(\mathbf{x}^+) - V_{\mathbf{x}^*}(\mathbf{x}) \leq 0$, with equality only when $\mathbf{x} = \mathbf{x}^*$, concluding the proof. For the globally exponentially stable result, from previous reasoning, we obtain

$$\begin{aligned} V_{\mathbf{x}^*}(\mathbf{x}^+) - V_{\mathbf{x}^*}(\mathbf{x}) &\leq \tau (\bar{\nabla} H - \nabla H^*)^\top (\phi(\bar{\nabla} H) - \phi(\nabla H^*)) \\ &= \tau (\bar{\mathbf{x}} - \mathbf{x}^*)^\top (\phi(\bar{\mathbf{x}}) - \phi(\mathbf{x}^*)) \leq -\tau \alpha \|\bar{\mathbf{x}} - \mathbf{x}^*\|^2 \end{aligned} \quad (32)$$

taking $H_i(\mathbf{x}_i) = \frac{1}{2} \mathbf{x}_i^\top \mathbf{x}_i$, so that $\bar{\nabla} H(\mathbf{x}, \mathbf{x}^+) = \bar{\mathbf{x}}$ with $\bar{\mathbf{x}} = (\mathbf{x}^+ + \mathbf{x})/2$. Note that

$$\begin{aligned} \|\bar{\mathbf{x}} - \mathbf{x}^*\|^2 &= \left\| \frac{(\mathbf{x}^+ - \mathbf{x}^*) + (\mathbf{x} - \mathbf{x}^*)}{2} \right\|^2 \\ &= \frac{1}{2} \left(\|\mathbf{x}^+ - \mathbf{x}^*\|^2 + \|\mathbf{x} - \mathbf{x}^*\|^2 - \frac{1}{2} \|\mathbf{x}^+ - \mathbf{x}\|^2 \right) \end{aligned} \quad (33)$$

and that

$$\begin{aligned} \|\mathbf{x}^+ - \mathbf{x}\| &= \tau \|\mathbf{S} \bar{\mathbf{x}} + \phi(\bar{\mathbf{x}}) - \mathbf{S} \mathbf{x}^* - \phi(\mathbf{x}^*)\| \\ &\leq \tau (\|\mathbf{S}\| + L) \|\bar{\mathbf{x}} - \mathbf{x}^*\| \end{aligned}$$

implying that

$$\|\mathbf{x}^+ - \mathbf{x}\|^2 \leq \frac{\tau^2}{2} (\|\mathbf{S}\| + L)^2 (\|\mathbf{x}^+ - \mathbf{x}^*\|^2 + \|\mathbf{x} - \mathbf{x}^*\|^2) \quad (34)$$

where we used (33). Combining (34) and (33) to replace it in (32), we obtain

$$V_{\mathbf{x}^*}(\mathbf{x}^+) - V_{\mathbf{x}^*}(\mathbf{x}) \leq -\varepsilon (\|\mathbf{x}^+ - \mathbf{x}^*\|^2 + \|\mathbf{x} - \mathbf{x}^*\|^2)$$

with

$$\varepsilon = \frac{\alpha \tau}{2} \left(1 - \left[\frac{\tau (\|\mathbf{S}\| + L)}{2} \right]^2 \right) > 0$$

since $\tau < 2/(\|\mathbf{S}\| + L)$. Moreover, we know that $-\|\mathbf{x} - \mathbf{x}^*\| \leq -2V_{\mathbf{x}^*}(\mathbf{x})$ from Lemma 1 in Appendix A. Therefore, we have

$$V_{\mathbf{x}^*}(\mathbf{x}^+) - V_{\mathbf{x}^*}(\mathbf{x}) \leq -2\varepsilon (V_{\mathbf{x}^*}(\mathbf{x}^+) + V_{\mathbf{x}^*}(\mathbf{x}))$$

equivalently, $V_{\mathbf{x}^*}(\mathbf{x}^+) \leq \lambda V_{\mathbf{x}^*}(\mathbf{x})$ with

$$\lambda = \frac{1 - 2\varepsilon}{1 + 2\varepsilon} \quad (35)$$

which satisfies $|\lambda| < 1$. Henceforth, global exponential convergence with rate λ follows in this case.

For the locally exponential stability result, recall (31) can be written as

$$\frac{\mathbf{x}^+ - \mathbf{x}}{\tau} = \mathbf{S} \bar{\mathbf{x}} + \phi(\bar{\mathbf{x}}).$$

Consider the corresponding continuous-time system

$$\dot{\mathbf{x}} = \mathbf{S} \mathbf{x} + \phi(\mathbf{x}).$$

Under the strong incremental passivity and Lipschitz assumptions on ϕ_i , Corollary 1 guarantees that this continuous-time system is globally exponentially stable. Hence, its Jacobian at the equilibrium $\mathbf{x}^*$

$$\mathbf{J} := \frac{\partial}{\partial \mathbf{x}} (\mathbf{S} \mathbf{x} + \phi(\mathbf{x})) \Big|_{\mathbf{x}=\mathbf{x}^*}$$

is Hurwitz. Linearizing (31) around $\mathbf{x}^*$, and introducing the error $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}^*$, yields the discrete-time linear system

$$\frac{\tilde{\mathbf{x}}^+ - \tilde{\mathbf{x}}}{\tau} = \mathbf{J} \left(\frac{\tilde{\mathbf{x}}^+ + \tilde{\mathbf{x}}}{2} \right)$$

so that

$$\tilde{\mathbf{x}}^+ = \mathbf{A}_d(\tau) \tilde{\mathbf{x}}, \quad \mathbf{A}_d(\tau) := (\mathbf{I} - \frac{\tau}{2} \mathbf{J})^{-1} (\mathbf{I} + \frac{\tau}{2} \mathbf{J}). \quad (36)$$

Since $\mathbf{J}$ is Hurwitz, all its eigenvalues λ satisfy that its real part is strictly negative. The eigenvalues of $\mathbf{A}_d(\tau)$ are then

$$\lambda_\tau = \frac{1 + \tau \lambda / 2}{1 - \tau \lambda / 2} = \frac{1 - \tau |\lambda| / 2}{1 + \tau |\lambda| / 2}$$

and a direct computation shows that $|\lambda_\tau| < 1$ for all $\tau > 0$. Thus, the origin of the linearized discrete-time system (36) is

exponentially stable for all $\tau > 0$. Finally, since the right-hand side of (31) is continuously differentiable, local exponential stability of $\mathbf{x}^*$ for the nonlinear discrete-time system follows using the Lyapunov indirect method arguments from [35, Thm. 4.7], completing the proof.

D. Proof of Theorem 2

Through this section, for any variable $\mathbf{x}$, the notations $\bar{\bullet}, \tilde{\bullet}$ act as $\bar{\mathbf{x}} = (\mathbf{x}^+ + \mathbf{x})/2$, $\tilde{\mathbf{x}} = \mathbf{x}^+ - \mathbf{x}$. We start by showing the well-posedness of (26).

Lemma 2: Consider the conditions of Theorem 2. Then, for any $\{\mathbf{x}_i\}_{i \in \mathcal{I}}$, there always exists a unique $\{\mathbf{x}_i^+\}_{i \in \mathcal{I}}$ satisfying (26).

Proof: First, recall the partition $\mathbf{x}_i = [\mathbf{q}_i^\top, \mathbf{p}_i^\top]^\top$. In this context, and with the choices of $\mathbf{M}, \phi_i$, one can solve for $\mathbf{p}_i^+$ as

$$\mathbf{p}_i^+ = \mathbf{p}_i + \tau \left(|\mathcal{N}_i| \mathbf{q}_i^+ - \sum_{j \in \mathcal{N}_i} \mathbf{q}_j \right). \quad (37)$$

Substituting (37) into the update equation for $\mathbf{q}_i$ in (26) and rearranging results in

$$\mathbf{G} \mathbf{q}_i^+ + \nabla f_i \left(\frac{\mathbf{q}_i^+ + \mathbf{q}_i}{2} \right) + \mathbf{c} = 0 \quad (38)$$

with

$$\mathbf{G} = \left(\frac{1}{\tau} + |\mathcal{N}_i| + \tau |\mathcal{N}_i|^2 \right) \mathbf{I}_m$$

$$\mathbf{c}_i = \sum_{j \in \mathcal{N}_i} \left(\mathbf{p}_i - \mathbf{p}_j - \mathbf{q}_j - \sum_{\ell \in \mathcal{N}_i} \mathbf{q}_\ell \right) - \frac{\mathbf{q}_i}{\tau}.$$

Note that (38) is equivalent to $\nabla g(\mathbf{q}_i^+) = \mathbf{0}$, where ∇ is taken with respect to $\mathbf{q}_i^+$ and with

$$g(\mathbf{q}_i^+) = \frac{1}{2} (\mathbf{q}_i^+)^T \mathbf{G} \mathbf{q}_i^+ + 2f_i \left(\frac{\mathbf{q}_i^+ + \mathbf{q}_i}{2} \right) + \mathbf{c}^T \mathbf{q}_i^+.$$

Moreover, $\mathbf{G} \succ \mathbf{0}$ and f_i is a convex function. Hence, g is convex as well. Henceforth, there exists a unique solution to $\nabla g(\mathbf{q}_i^+) = \mathbf{0}$, which coincides with the solution of the convex optimization program $\min_{\mathbf{q}_i^+ \in \mathbb{R}^m} g(\mathbf{q}_i^+)$. This implies the existence of a unique $\mathbf{p}_i^+$ as well from (37). $\square$

Now, we write (26) in a more convenient representation.

Lemma 3: Consider the conditions of Theorem 2, and let $\mathbf{q} = [\mathbf{q}_i]_{i \in \mathcal{I}}$ and $\mathbf{p} = [\mathbf{p}_i]_{i \in \mathcal{I}}$. Moreover, let $\mathbf{A}, \mathbf{D} = \text{diag}(\mathbf{A}\mathbf{1})$ denote the adjacency and degree matrices of $\mathcal{G}$, and define

$$\mathbf{L}_m = (\mathbf{D} - \mathbf{A}) \otimes \mathbf{I}_m, \quad \mathbf{Q} = (\mathbf{D} + \mathbf{A}) \otimes \mathbf{I}_m/2$$

$$\Lambda(\tau) = \frac{1}{\tau^2} \mathbf{I} + \frac{1}{\tau} \mathbf{Q} + \mathbf{Q}^2$$

$$\mathbf{S}_p(\tau) = \begin{bmatrix} -\Lambda(\tau)^{-1} \left(\frac{1}{\tau} \mathbf{I} + \mathbf{Q} \right) \mathbf{L}_m & -\frac{1}{\tau} \Lambda(\tau)^{-1} \mathbf{L}_m \\ \frac{1}{\tau} \Lambda(\tau)^{-1} \mathbf{L}_m & -\Lambda(\tau)^{-1} \mathbf{Q} \mathbf{L}_m \end{bmatrix}$$

$$\mathbf{S}_r(\tau) = \begin{bmatrix} -\Lambda(\tau)^{-1} \left(\frac{1}{\tau} \mathbf{L}_m + \mathbf{Q} \mathbf{L}_m + \mathbf{L}_m \mathbf{Q} \right) & -\Lambda(\tau)^{-1} \mathbf{L}_m \\ \mathbf{L}_m & \mathbf{0} \end{bmatrix}. \quad (39)$$

Then, $\Lambda(\tau)$ is invertible. Moreover, using $\mathbf{r} = \mathbf{p}/\tau - \mathbf{Q}\mathbf{q}$, trajectories of (26) satisfy

$$\begin{bmatrix} \tilde{\mathbf{q}} \\ \tilde{\mathbf{r}} \end{bmatrix} = \mathbf{S}_r(\tau) \begin{bmatrix} \bar{\mathbf{q}} \\ \bar{\mathbf{r}} \end{bmatrix} - \begin{bmatrix} \Lambda(\tau)^{-1}/\tau \\ \mathbf{0} \end{bmatrix} \nabla f(\bar{\mathbf{q}}). \quad (40)$$

Proof: First, write (25) as

$$\tilde{\mathbf{x}}/\tau = (\mathbf{D} \otimes \mathbf{M}) \mathbf{x}^+ - (\mathbf{A} \otimes \mathbf{M}) \mathbf{x} + \phi(\bar{\mathbf{x}}). \quad (41)$$

Now, note that the following relations hold:

$$\mathbf{x}^+ = \bar{\mathbf{x}} + \tilde{\mathbf{x}}/2$$

$$\mathbf{x} = \bar{\mathbf{x}} - \tilde{\mathbf{x}}/2.$$

Replacing these expressions in (41) and rearranging results in

$$(\mathbf{I}/\tau - (\mathbf{A} + \mathbf{D}) \otimes \mathbf{M}/2) \tilde{\mathbf{x}} = (\mathbf{L} \otimes \mathbf{M}) \bar{\mathbf{x}} + \phi(\bar{\mathbf{x}}).$$

In terms of the partitions of $\mathbf{x}$ and the expression for $\mathbf{M}$ as in (12), we have

$$\begin{bmatrix} \mathbf{I}/\tau + \mathbf{Q} & \mathbf{Q} \\ -\mathbf{Q} & \mathbf{I}/\tau \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{q}} \\ \tilde{\mathbf{p}} \end{bmatrix} = \begin{bmatrix} -\mathbf{L}_m & -\mathbf{L}_m \\ \mathbf{L}_m & \mathbf{0} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{q}} \\ \bar{\mathbf{p}} \end{bmatrix} - \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \nabla f(\bar{\mathbf{q}}). \quad (42)$$

Note that

$$\begin{bmatrix} \mathbf{I}/\tau + \mathbf{Q} & \mathbf{Q} \\ -\mathbf{Q} & \mathbf{I}/\tau \end{bmatrix}^{-1} = (\Lambda(\tau)^{-1} \otimes \mathbf{I}_2) \begin{bmatrix} \mathbf{I}/\tau & -\mathbf{Q} \\ \mathbf{Q} & \mathbf{I}/\tau + \mathbf{Q} \end{bmatrix}.$$

Invertibility of $\Lambda(\tau)$ comes from the fact that $\mathbf{Q}$ is diagonally dominant and thus is positive semidefinite [36, Sect. 6.1]. Hence, $\mathbf{Q}^2$ is positive semidefinite, being $\Lambda(\tau)$ positive definite and thus invertible. Therefore, from (42), it follows that

$$\begin{bmatrix} \tilde{\mathbf{q}} \\ \tilde{\mathbf{p}} \end{bmatrix} = \mathbf{S}_p(\tau) \begin{bmatrix} \bar{\mathbf{q}} \\ \bar{\mathbf{p}} \end{bmatrix} - \begin{bmatrix} \Lambda(\tau)^{-1}/\tau \\ \Lambda(\tau)^{-1} \mathbf{Q} \end{bmatrix} \nabla f(\bar{\mathbf{q}}). \quad (43)$$

The rest of the lemma follows from the change of variables:

$$\begin{bmatrix} \mathbf{q} \\ \mathbf{r} \end{bmatrix} = \mathbf{T} \begin{bmatrix} \mathbf{q} \\ \mathbf{p} \end{bmatrix}$$

with

$$\mathbf{T} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{Q} & \mathbf{I}/\tau \end{bmatrix}, \quad \mathbf{T}^{-1} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \tau \mathbf{Q} & \tau \mathbf{I} \end{bmatrix}.$$

Note that $\Lambda(\tau)$ and $\mathbf{Q}$ commute so that $\mathbf{Q}$ also commutes with $\Lambda(\tau)^{-1}$. From this, a straightforward calculation can verify that

$$\mathbf{T} \begin{bmatrix} \Lambda(\tau)^{-1}/\tau \\ \Lambda(\tau)^{-1} \mathbf{Q} \end{bmatrix} = \begin{bmatrix} \Lambda(\tau)^{-1}/\tau \\ \mathbf{0} \end{bmatrix}$$

$$\mathbf{S}_r(\tau) = \mathbf{T} \mathbf{S}_p(\tau) \mathbf{T}^{-1}$$

from which (40) is obtained, completing the proof. $\square$

Given these results, we can show Theorem 2.

Proof: (Of Theorem 2) First, use Lemma 3 to write (26) as (40). Denote with $\mathbf{y} = [\mathbf{q}^\top, \mathbf{r}^\top]^\top$. Let $\mathbf{y}^*$ the equilibrium of (40) and set $\mathbf{e} = \mathbf{y} - \mathbf{y}^*$. Hence, it follows from (40) that:

$$\begin{aligned}\dot{\mathbf{e}} &= \mathbf{S}_r(\tau)\bar{\mathbf{y}} - \mathbf{B}(\tau)\nabla f(\bar{\mathbf{q}}) \\ &= \mathbf{S}_r(\tau)\bar{\mathbf{e}} - \mathbf{B}(\tau)\psi(\bar{\mathbf{q}})\end{aligned}$$

with $\mathbf{B}(\tau) = [\Lambda(\tau)^{-1}/\tau, \mathbf{0}]^\top$ and $\psi(\bar{\mathbf{q}}) = \nabla f(\bar{\mathbf{q}}) - \nabla f(\mathbf{q}^*)$.

Now, assume that (27) holds, and set a Lyapunov function candidate

$$V(\mathbf{e}) = \mathbf{e}^\top \mathbf{P}(\tau) \mathbf{e} / 2$$

with $\mathbf{P}(\tau)$ in (27a). Hence,

$$\begin{aligned}V(\mathbf{e}^+) - V(\mathbf{e}) &= \bar{\mathbf{e}}^\top \mathbf{P}(\tau) \bar{\mathbf{e}} \\ &= \bar{\mathbf{e}}^\top (\mathbf{P}(\tau)\mathbf{S}_r(\tau) + \mathbf{S}_r(\tau)^\top \mathbf{P}(\tau)) \bar{\mathbf{e}} - \bar{\mathbf{e}}^\top \mathbf{P}(\tau) \mathbf{B}(\tau) \psi(\bar{\mathbf{q}}).\end{aligned}$$

Now, note that $\mathbf{P}(\tau)\mathbf{B}(\tau) = [\mathbf{I}/\tau, \mathbf{P}_{12}\Lambda(\tau)^{-1}/\tau]^\top$ such that

$$\begin{aligned}-\bar{\mathbf{e}}^\top \mathbf{P}(\tau) \mathbf{B}(\tau) \psi(\bar{\mathbf{q}}) &= -(\bar{\mathbf{q}} - \mathbf{q}^*)(\nabla f(\bar{\mathbf{q}}) - \nabla f(\mathbf{q}^*)) / \tau \\ &\quad - (\bar{\mathbf{r}} - \mathbf{r}^*)^\top \mathbf{P}_{12}^\top \Lambda(\tau)^{-1} \psi(\bar{\mathbf{q}}) / \tau \\ &\leq -(\mu/\tau) \|\bar{\mathbf{q}} - \mathbf{q}^*\|^2 + \gamma(\tau) \|\mathbf{P}_{12}(\bar{\mathbf{r}} - \mathbf{r}^*)\| \|\bar{\mathbf{q}} - \mathbf{q}^*\| \quad (44)\end{aligned}$$

where we used Proposition 2 due to strong convexity of f , used

$$\|(\nabla f(\bar{\mathbf{q}}) - \nabla f(\mathbf{q}^*))\| \leq L \|\bar{\mathbf{q}} - \mathbf{q}^*\|$$

due to the Lipschitz property of ∇f and setting

$$\gamma(\tau) = (L/\tau) \|\Lambda(\tau)^{-1}\|.$$

Moreover, one can use Young's inequality to get

$$\begin{aligned}& -(\mu/\tau) \|\bar{\mathbf{q}} - \mathbf{q}^*\|^2 + \gamma(\tau) \|\mathbf{P}_{12}(\bar{\mathbf{r}} - \mathbf{r}^*)\| \|\bar{\mathbf{q}} - \mathbf{q}^*\| \\ & \leq \left(-\frac{\mu}{\tau} + \frac{\varepsilon\gamma(\tau)}{2} \right) \|\bar{\mathbf{q}} - \mathbf{q}^*\|^2 + \frac{\gamma(\tau)}{2\varepsilon} \|\mathbf{P}_{12}(\bar{\mathbf{r}} - \mathbf{r}^*)\|^2 \\ & \leq \bar{\mathbf{e}}^\top \mathbf{S}_\gamma(\tau, \varepsilon) \bar{\mathbf{e}} \quad (45)\end{aligned}$$

with

$$\mathbf{S}_\gamma(\tau, \varepsilon) = \begin{bmatrix} (\gamma(\tau)\varepsilon/2 - \mu/\tau)\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \gamma(\tau)\mathbf{U}/(2\varepsilon) \end{bmatrix} \quad (46)$$

where $\mathbf{P}_{12}^\top \mathbf{P}_{12} \preceq \mathbf{U}$, which follows from (27b) due to a Schur complement. Henceforth

$$\begin{aligned}V(\mathbf{e}^+) - V(\mathbf{e}) &\leq \bar{\mathbf{e}}^\top (\mathbf{P}(\tau)\mathbf{S}_r(\tau) + \mathbf{S}_r(\tau)^\top \mathbf{P}(\tau) + \mathbf{S}_\gamma(\tau, \varepsilon)) \bar{\mathbf{e}} \\ &\leq -u \|\bar{\mathbf{q}} - \mathbf{q}^*\|^2 \quad (47)\end{aligned}$$

due to (27c). Hence, $V(\mathbf{e})$ decreases unless $\bar{\mathbf{q}} = \mathbf{q}^*$. However, $\bar{\mathbf{q}} = \mathbf{q}^*$ implies $\bar{\mathbf{q}} = (\mathbf{1} \otimes \mathbf{I}_m)\theta^*$ (see Example 2) such that $\bar{\mathbf{r}} = \mathbf{L}_m \bar{\mathbf{q}} = \mathbf{0}$ from (40), meaning that equilibrium was reached for $\bar{\mathbf{r}}$ as well. Henceforth, $\mathbf{e}[k]$ converges asymptotically to the origin, completing the proof. $\square$

E. Proof of Corollary 2

Set $\mathbf{P}_{12} = \mathbf{U} = \mathbf{0}$, $\mathbf{P}_{22} = \mathbf{I}$ and $u = \mu, \varepsilon = 0$. Note that $\varepsilon = 0$ is possible since $\mathbf{P}_{12} = \mathbf{0}$ and Young inequality is no longer

required in (45). Then, we can define

$$\mathbf{S}_\gamma(\tau, 0) := \begin{bmatrix} -\mu\mathbf{I}/\tau & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

With these choices, (27a) is satisfied since $\Lambda(\tau)$ is positive definite, and (27b) is satisfied trivially. Moreover, $\mathbf{W}(\tau) = \mathbf{P}(\tau)\mathbf{S}_r(\tau) + \mathbf{S}_r(\tau)^\top \mathbf{P}(\tau) + \mathbf{S}_\gamma(\tau, 0)$ for these choices which result in

$$\mathbf{W}(\tau) = \begin{bmatrix} -(\frac{1}{\tau}\mathbf{L}_m + \mathbf{Q}\mathbf{L}_m + \mathbf{L}_m\mathbf{Q}) - \mu\mathbf{I}/\tau & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Since $\mathbf{L}_m$ is positive semidefinite and

$$\mathbf{Q}\mathbf{L}_m + \mathbf{L}_m\mathbf{Q} = (\mathbf{D}^2 - \mathbf{A}^2) \otimes \mathbf{I}_m / 2 \succeq \mathbf{0}$$

due to (28), (27c) is satisfied for these choices for all $\tau > 0$. For the last part of the result, if $\mathcal{G}$ is a cycle, then $\mathbf{D} = 2\mathbf{I}$ and $\mathbf{D}^2 - \mathbf{A}^2 = 4\mathbf{I} - (2\mathbf{I} + \mathbf{A}_2)$ where $\mathbf{A}_2$ is the adjacency matrix of the graph $\mathcal{G}_2$ with edges connecting agents two steps away in the original graph $\mathcal{G}$. This means that $\mathbf{A}_2$ is full of zeros, except for two 1's per row. Henceforth

$$\mathbf{D}^2 - \mathbf{A}^2 = 2\mathbf{I}_N - \mathbf{A}_2$$

is diagonally dominant and thus positive semidefinite [36, Sect 6.1]. Similarly, if $\mathcal{G}$ is fully connected, then $\mathbf{D}^2 - \mathbf{A}^2 = N\mathbf{I}_N - \mathbf{A}^2$ and $\mathbf{A} = \mathbf{1}\mathbf{1}^\top - \mathbf{I}_N$ so that

$$\mathbf{D}^2 - \mathbf{A}^2 = (N-1)\mathbf{I}_N - (N-2)\mathbf{1}\mathbf{1}^\top.$$

Similarly as before, $\mathbf{D}^2 - \mathbf{A}^2$ is diagonally dominant and thus positive semidefinite.

F. Proof of Corollary 3

Make the same choices as in the proof of Corollary 2, but let $\mathcal{G}$ be an arbitrary connected undirected graph and arbitrary $u > 0$. Henceforth

$$\begin{aligned}& -\left(\frac{1}{\tau}\mathbf{L}_m + \mathbf{Q}\mathbf{L}_m + \mathbf{L}_m\mathbf{Q}\right) - \frac{\mu}{\tau}\mathbf{I} \\ & \preceq \left(\|\mathbf{Q}\mathbf{L}_m + \mathbf{L}_m\mathbf{Q}\| - \frac{\mu}{\tau}\right)\mathbf{I} \preceq -u\mathbf{I}\end{aligned}$$

by the bound on τ for some $u > 0$. Hence, (27c) is satisfied due to $\mathbf{Q}\mathbf{L}_m + \mathbf{L}_m\mathbf{Q} = (\mathbf{D}^2 - \mathbf{A}^2) \otimes \mathbf{I}_m$.

G. Proof of Corollary 4

As a preliminary tool for the proof, we analyze the complex generalized eigenvalues λ and the corresponding eigenvectors $[\mathbf{u}^\top, \mathbf{v}^\top]^\top$ of the pencil

$$\lambda \begin{bmatrix} a\mathbf{I} + \mathbf{Q} & \mathbf{Q} \\ -\mathbf{Q} & a\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} -\mathbf{L}_m - \mathbf{H} & -\mathbf{L}_m \\ \mathbf{L}_m & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}$$

with $\mathbf{H} = \text{blockdiag}(\mathbf{H}_1, \dots, \mathbf{H}_N)$. First, consider the limiting case $a = 0$. This leads to the relation $-\lambda\mathbf{Q}\mathbf{u} = \mathbf{L}_m\mathbf{u}$, where pre-multiplying by the adjoint $\mathbf{u}^\dagger$ gives $\lambda = -\mathbf{u}^\dagger \mathbf{L}_m \mathbf{u} / (\mathbf{u}^\dagger \mathbf{Q} \mathbf{u}) \leq 0$ whenever $\mathbf{u}$ is not in the kernel of $\mathbf{Q}$. In this case, λ is real and nonpositive since both $\mathbf{L}_m$ and $\mathbf{Q}$ are symmetric. If $\mathbf{u}$ lies in the kernel of $\mathbf{Q}$, then $\mathbf{0} = \mathbf{L}_m \mathbf{u}$, implying that $\mathbf{u}$ also belongs to the

kernel of $\mathbf{L}_m$. However, the kernel of $\mathbf{L}_m$ is the span of vectors $\mathbf{1}_N \otimes \mathbf{w}$ with $\mathbf{w} \in \mathbb{R}^m$. Hence

$$\mathbf{Q}(\mathbf{1}_N \otimes \mathbf{w}) = ((\mathbf{D} + \mathbf{A})\mathbf{1}_N) \otimes \mathbf{w}/2 = \mathbf{D} \otimes \mathbf{1}_m \neq \mathbf{0}.$$

Therefore, the intersection of the kernels of $\mathbf{Q}$ and $\mathbf{L}_m$ is only at $\mathbf{u} = \mathbf{0}$. Substituting this result into the first row of the pencil yields $\mathbf{Q}\mathbf{v} = -\mathbf{L}_m\mathbf{v}$. The same reasoning as before applies to conclude $\lambda \leq 0$ in all cases, with $\lambda = 0$ only when $\mathbf{v}$ is in the span of vectors $\mathbf{1}_N \otimes \mathbf{w}$ and $\mathbf{u} = \mathbf{0}$ for some $\alpha \in \mathbb{R}$, which corresponds to the equilibrium. For the remaining eigenvalues, we have $\lambda < 0$. As $a > 0$ varies, the eigenvalue locus evolves continuously almost everywhere, except at points where branches meet. By continuity, there exists a sufficiently small $a^* > 0$ such that the nonzero eigenvalues remain strictly negative. Set $\tau^* = 1/a^*$. Therefore, the matrix

$$\mathbf{W}(\tau) = \begin{bmatrix} \mathbf{I}/\tau + \mathbf{Q} & \mathbf{Q} \\ -\mathbf{Q} & \mathbf{I}/\tau \end{bmatrix}^{-1} \begin{bmatrix} -\mathbf{L}_m - \mathbf{H} & -\mathbf{L}_m \\ \mathbf{L}_m & \mathbf{0} \end{bmatrix}$$

has real eigenvalues $\lambda \leq 0$ for all $\tau \geq \tau^*$. Now, consider (42) from Lemma 3, which reduces to the following for the quadratic choice of f :

$$\begin{bmatrix} a\mathbf{I} + \mathbf{Q} & \mathbf{Q} \\ -\mathbf{Q} & a\mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{q}} \\ \tilde{\mathbf{p}} \end{bmatrix} = \begin{bmatrix} -\mathbf{L}_m - \mathbf{H} & -\mathbf{L}_m \\ \mathbf{L}_m & \mathbf{0} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{q}} - \mathbf{q}^* \\ \bar{\mathbf{p}} - \mathbf{p}^* \end{bmatrix}. \quad (48)$$

Equivalently, $\tilde{\mathbf{x}} = \mathbf{W}(\tau)(\bar{\mathbf{x}} - \mathbf{x}^*)$, which can also be written as $\mathbf{e}^+ = \mathbf{W}'(\tau)\mathbf{e}$, where $\mathbf{e} = \mathbf{x} - \mathbf{x}^*$ and

$$\mathbf{W}'(\tau) = (\mathbf{I} - \mathbf{W}(\tau))^{-1}(\mathbf{I} + \mathbf{W}(\tau)).$$

The eigenvalues of $\mathbf{W}'(\tau)$ are given by

$$\lambda' = \frac{1 + \lambda}{1 - \lambda} = \frac{1 - |\lambda|}{1 + |\lambda|} \in [-1, 1]$$

for all $\tau \geq \tau^*$, with $|\lambda'| = 1$ only when $\lambda = 0$, corresponding to the equilibrium. Hence, all remaining modes of $\mathbf{W}'(\tau)$ have eigenvalues within the unit circle, and the iteration for $\mathbf{e}$ is asymptotically stable.

H. Proof of Corollary 5

First, note that

$$\psi(\bar{\mathbf{q}}) = \nabla f(\bar{\mathbf{q}}) - \nabla f(\mathbf{q}^*) = \mathbf{H}(\bar{\mathbf{q}} - \mathbf{q}^*)$$

with $\mathbf{H} = \text{blockdiag}(\mathbf{H}_1, \dots, \mathbf{H}_N)$. Now, we follow all steps from the proof of Theorem 2, but instead of (44), do

$$-\bar{\mathbf{e}}^\top \mathbf{P}(\tau) \mathbf{B}(\tau) \psi(\bar{\mathbf{q}}) = \bar{\mathbf{e}}^\top \mathbf{S}_h(\tau) \bar{\mathbf{e}} \quad (49)$$

with

$$\mathbf{S}_h(\tau) = \begin{bmatrix} \mathbf{H}/\tau & \mathbf{0} \\ \mathbf{P}_{12}^\top \mathbf{A}(\tau) \mathbf{H}/\tau & \mathbf{0} \end{bmatrix}. \quad (50)$$

Therefore, similar to (47), we have

$$\begin{aligned} V(\mathbf{e}^+) - V(\mathbf{e}) &\leq \bar{\mathbf{e}}^\top (\mathbf{P}(\tau) \mathbf{S}_r(\tau) + \mathbf{S}_r(\tau)^\top \mathbf{P}(\tau) + \mathbf{S}_h(\tau)) \bar{\mathbf{e}} \\ &\leq -u \|\bar{\mathbf{q}} - \mathbf{q}^*\|^2 \end{aligned} \quad (51)$$

from which convergence follows using the same arguments at the end of the proof of Theorem 2.

REFERENCES

- [1] G. Notarstefano, I. Notarnicola, and A. Camisa, "Distributed optimization for smart cyber-physical networks," *Found. Trends Syst. Control*, vol. 7, no. 3, pp. 253–283, 2019.
- [2] T. Yang et al., "A survey of distributed optimization," *Annu. Rev. Control*, vol. 47, pp. 278–305, 2019.
- [3] B. Ghareisifard and J. Cortés, "Distributed continuous-time convex optimization on weight-balanced digraphs," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 781–786, Mar. 2014.
- [4] S. Yang, Q. Liu, and J. Wang, "A multi-agent system with a proportional-integral protocol for distributed constrained optimization," *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3461–3467, Jul. 2017.
- [5] X. Zeng, P. Yi, and Y. Hong, "Distributed continuous-time algorithm for constrained convex optimizations via nonsmooth analysis approach," *IEEE Trans. Autom. Control*, vol. 62, no. 10, pp. 5227–5233, Oct. 2017.
- [6] P. Lin, W. Ren, and J. A. Farrell, "Distributed continuous-time optimization: Nonuniform gradient gains, finite-time convergence, and convex constraint set," *IEEE Trans. Autom. Control*, vol. 62, no. 5, pp. 2239–2253, May 2017.
- [7] Z. Li, Z. Ding, J. Sun, and Z. Li, "Distributed adaptive convex optimization on directed graphs via continuous-time algorithms," *IEEE Trans. Autom. Control*, vol. 63, no. 5, pp. 1434–1441, May 2018.
- [8] T. Hatanaka, N. Chopra, T. Ishizaki, and N. Li, "Passivity-based distributed optimization with communication delays using PI consensus algorithm," *IEEE Trans. Autom. Control*, vol. 63, no. 12, pp. 4421–4428, Dec. 2018.
- [9] M. Li, G. Chesi, and Y. Hong, "Input-feedforward-passivity-based distributed optimization over jointly connected balanced digraphs," *IEEE Trans. Autom. Control*, vol. 66, no. 9, pp. 4117–4131, Sep. 2021.
- [10] H. Moradian and S. S. Kia, "A distributed continuous-time modified Newton–Raphson algorithm," *Automatica*, vol. 136, 2022, Art. no. 109886.
- [11] A. Nedić, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM J. Optim.*, vol. 27, no. 4, pp. 2597–2633, 2017.
- [12] P. Di Lorenzo and G. Scutari, "Next: In-network nonconvex optimization," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 2, pp. 120–136, Jun. 2016.
- [13] D. Varagnolo, F. Zanella, A. Cenedese, G. Pillonetto, and L. Schenato, "Newton–Raphson consensus for distributed convex optimization," *IEEE Trans. Autom. Control*, vol. 61, no. 4, pp. 994–1009, Apr. 2016.
- [14] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "Convergence of asynchronous distributed gradient methods over stochastic networks," *IEEE Trans. Autom. Control*, vol. 63, no. 2, pp. 434–448, Feb. 2018.
- [15] C. Xi, R. Xin, and U. A. Khan, "ADD-OPT: Accelerated distributed directed optimization," *IEEE Trans. Autom. Control*, vol. 63, no. 5, pp. 1329–1339, May 2018.
- [16] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1245–1260, Sep. 2018.
- [17] G. Scutari and Y. Sun, "Distributed nonconvex constrained optimization over time-varying digraphs," *Math. Program.*, vol. 176, no. 1–2, pp. 497–544, 2019.
- [18] S. Pu, W. Shi, J. Xu, and A. Nedić, "Push-pull gradient methods for distributed optimization in networks," *IEEE Trans. Autom. Control*, vol. 66, no. 1, pp. 1–16, Jan. 2021.
- [19] G. Carnevale, F. Farina, I. Notarnicola, and G. Notarstefano, "GTAdam: Gradient tracking with adaptive momentum for distributed online optimization," *IEEE Trans. Control Netw. Syst.*, vol. 10, no. 3, pp. 1436–1448, Sep. 2023.
- [20] B. Van Scoy and L. Lessard, "A universal decomposition for distributed optimization algorithms," *IEEE Contr. Syst. Lett.*, vol. 6, pp. 3044–3049, 2022.
- [21] Z. Li, W. Shi, and M. Yan, "A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates," *IEEE Trans. Signal Process.*, vol. 67, no. 17, pp. 4494–4506, Sep. 2019.
- [22] A. Khaled, K. Mishchenko, and C. Jin, "DoWG unleashed: An efficient universal parameter-free gradient descent method," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2023, pp. 6748–6769.
- [23] M. Poli, S. Massaroli, A. Yamashita, H. Asama, and J. Park, "Port-Hamiltonian gradient flows," in *Proc. Workshop Integr. Deep Neural Models Differ. Equ.*, 2019. [Online]. Available: <https://openreview.net/forum?id=G6qPBpSfyN>
- [24] A. van der Schaft, *L2-Gain and Passivity Techniques in Nonlinear Control* (ser. Communications and Control Engineering), 3rd ed. Cham, Switzerland: Springer, 2016.

- [25] J. Wei, A. R. Everts, M. K. Camlibel, and A. J. van der Schaft, "Consensus dynamics with arbitrary sign-preserving nonlinearities," *Automatica*, vol. 83, pp. 226–233, 2017.
- [26] A. van der Schaft and B. Maschke, "Port-Hamiltonian dynamics on graphs: Consensus and coordination control algorithms," *IFAC Proc. Vol.*, vol. 43, no. 19, pp. 175–178, 2010.
- [27] A. Macchelli, "Control design for a class of discrete-time port-Hamiltonian systems," *IEEE Trans. Autom. Control*, vol. 68, no. 12, pp. 8224–8231, Dec. 2023.
- [28] A. Macchelli, "Trajectory tracking for discrete-time port-Hamiltonian systems," *IEEE Contr. Syst. Lett.*, vol. 6, pp. 3146–3151, 2022.
- [29] E. S. Riis, M. J. Ehrhardt, G. R. W. Quispel, and C.-B. Schönlieb, "A geometric integration approach to nonsmooth, nonconvex optimisation," *Found. Comput. Math.*, vol. 22, no. 5, pp. 1351–1394, Oct. 2022.
- [30] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [31] A. Nasrallah et al., "Ultra-low latency (ULL) networks: The IEEE TSN and IETF DetNet standards and related 5G ULL research," *IEEE Commun. Surveys Tut.*, vol. 21, no. 1, pp. 88–145, First Quarter 2019.
- [32] G. Carnevale, I. Notarnicola, L. Marconi, and G. Notarstefano, "Triggered gradient tracking for asynchronous distributed optimization," *Automatica*, vol. 147, 2023, Art. no. 110726.
- [33] W. Shi, Q. Ling, G. Wu, and W. Yin, "Extra: An exact first-order algorithm for decentralized consensus optimization," *SIAM J. Optim.*, vol. 25, no. 2, pp. 944–966, 2015.
- [34] S. S. Kia, J. Cortés, and S. Martínez, "Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication," *Automatica*, vol. 55, pp. 254–264, 2015.
- [35] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.
- [36] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. New York, NY, USA: Cambridge Univ. Press, 2012.



Rodrigo Aldana-López (Member, IEEE) received the B.Sc. degree in electrical engineering from the Instituto Tecnológico y de Estudios Superiores de Occidente, Guadalajara, Mexico, in 2016, and the Ph.D. degree in computer science and systems engineering from the University of Zaragoza, Zaragoza, Spain, in 2024.

He is currently a Distinguished Professor with the University of Zaragoza, under a Beatriz Galindo Fellowship. He has held multiple research stays at Intel Labs, Hillsboro, CA, USA.

His research interests include nonlinear and hybrid control, distributed estimation and optimization, and multiagent systems under communication constraints, delays, and uncertainty.

Dr. Aldana-Lopez is an Associate Editor for *Journal of the Franklin Institute*.



Alessandro Macchelli (Senior Member, IEEE) received the Graduate and Ph.D. degrees in automation and operational research from the University of Bologna, Bologna, Italy, in 2000 and 2003, respectively.

In 2003 and 2004, he held Postdoctoral positions with the University of Twente, Enschede, The Netherlands, and the University of Bologna, respectively. In 2005, he joined the Department of Electronics, Computer Science and Systems (now Department of Electrical, Electronic and Information Engineering), University of Bologna, as an Assistant Professor, where he is currently an Associate Professor. He held visiting positions at various academic/research international institutions. He is the author of more than 80 journal articles and conference papers on these topics. His research interests include port-Hamiltonian systems, with a particular emphasis on modeling, simulation, and control of distributed parameter systems, and applications of the port-Hamiltonian framework to robotics and mechatronics.

Dr. Macchelli is an Associate Editor for IEEE TRANSACTIONS ON AUTOMATIC CONTROL and IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY.



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

He is currently a Professor with the Department of Electrical, Electronic, and Information Engineering G. Marconi, Alma Mater Studiorum Università di Bologna, Bologna, Italy. From 2016

to 2018, he was an Associate Professor, and since 2007, has been an Assistant Professor, Ricercatore, with the Università del Salento, Lecce, Italy. He is also a Visiting Scholar with the University of Stuttgart, Stuttgart, Germany; University of California, Santa Barbara, CA, USA; and the University of Colorado Boulder, Boulder, CO, USA. His research interests include distributed optimization, cooperative control in complex networks, applied nonlinear optimal control, and trajectory optimization and maneuvering of aerial and car vehicles.

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



Rosario Aragües (Senior Member, IEEE) received the M.S. and Ph.D. degrees in system engineering and computer science from the University of Zaragoza, Zaragoza, Spain, in 2008 and 2012, respectively.

From 2012 to 2013, she was a Postdoctoral Researcher with the Institut Pascal, CNRS, UMR 6602, Clermont-Ferrand, France. She is currently an Associate Professor with the Department of Computer Science and Systems Engineering, University of Zaragoza. Her research interests include multirobot perception and control using distributed and decentralized strategies.

search interests include multirobot perception and control using distributed and decentralized strategies.



Carlos Sagüés (Senior Member, IEEE) received the M.S. degree in computer science and systems engineering and the Ph.D. degree in industrial engineering from the University of Zaragoza, Zaragoza, Spain, in 1989 and 1992, respectively.

In 1994, he joined, as an Associate Professor, the Departamento de Informatica e Ingenieria de Sistemas, University of Zaragoza, where he became a Full Professor in 2009 and was also the Head Teacher. He was involved in research

on force and infrared sensors for robots. His current research interests include control systems and industry applications, computer vision, visual control, and multivehicle cooperative control.