Optimal distributed control with stability guarantees by training a network of neural closed-loop maps

Danilo Saccani, Leonardo Massai, Luca Furieri and Giancarlo Ferrari-Trecate

Abstract— This paper proposes a novel approach to improve the performance of distributed nonlinear control systems while preserving stability by leveraging Deep Neural Networks (DNNs). We build upon the Neural System Level Synthesis (Neur-SLS) framework and introduce a method to parameterize stabilizing control policies that are distributed across a network topology. A distinctive feature is that we iteratively minimize an arbitrary control cost function through an unconstrained optimization algorithm, all while preserving the stability of the overall network architecture by design. This is achieved through two key steps. First, we establish a method to parameterize interconnected Recurrent Equilibrium Networks (RENs) that guarantees a bounded $\mathcal{L}_2$ gain at the network level. This ensures stability. Second, we demonstrate how the information flow within the network is preserved, enabling a fully distributed implementation where each subsystem only communicates with its neighbors. To showcase the effectiveness of our approach, we present a simulation of a distributed formation control problem for a fleet of vehicles. The simulation demonstrates how the proposed neural controller enables the vehicles to maintain a desired formation while navigating obstacles and avoiding collisions, all while guaranteeing network stability.

I. Introduction

Controlling interconnected systems with non-linear dynamics is a crucial task across various fields ranging from robotics, and power grids, to social sciences [1]–[4]. However, achieving networked control becomes significantly more challenging when the system dynamics are non-linear and a convex function does not accurately capture the desired performance metric.

Indeed, several Optimal Distributed Control (ODC) approaches exist for the case of linear systems with convex performance metrics [5]–[9]; however, for non-convex problems, solutions are scarce. Dissipativity-based approaches, like passivity control [10], [11] and control by interconnection [12], offer ways to design stabilizing distributed control policies without minimizing a specific cost. However, these methods become prohibitive when it comes to minimizing a user-defined cost function. Nonlinear Distributed Model Predictive Control (DMPC) [13], [14] gives an alternative, but computing crucial terminal ingredients in a distributed manner remains unsolved for non-linear systems. Additionally, suitable distributed optimization algorithms for online implementation are limited, often relying on the Alternating Direction Method of Multipliers (ADMM) [15], whose potential lack of convergence poses limitations in real-time applications.

Recently, there has been a surge of interest in using Deep Neural Networks (DNNs) to parameterize highly non-linear distributed controllers. Approaches like Graph Neural Networks (GNNs) have shown promise in designing such policies [16], [17] in a scalable way. Nevertheless, stability guarantees are often limited by restrictive assumptions. For instance, some methods require a system to be initially linear and open-loop stable, with the trained network maintaining small Lipschitz constants across layers [17]. These conditions might not hold during exploration, potentially leading to system failures before a safe policy is learned [18].

An alternative approach using deep Hamiltonian control policies offers built-in closed-loop stability guarantees for specific systems [19]. However, these policies only apply to a narrow class of systems with a particular skew-symmetric interconnection structure.

Contributions: With this work, we provide two main contributions. First, we propose a novel distributed state-feedback control scheme leveraging the Neural System Level Synthesis (Neur-SLS) framework and dissipativity theory. The scheme is composed of a parametrized family of nonlinear sub-operators, interconnected according to a prescribed network structure that mimics the sparsity pattern of the distributed system itself. This means that communication between sub-operators only occurs between interconnected agents in the physical system. We prove that closed-loop stability is guaranteed as long as the parameters satisfy a Linear Matrix Inequality (LMI). Secondly, we present a *free* parametrization of this networked ensemble of nonlinear sub-operators, with each sub-operator modeled as a Recurrent Equilibrium Network (REN). The parameterization is *free* in the sense that the resulting distributed controller guarantees the closed-loop stability for any value of the parameters. This eliminates the need to enforce LMIs, thereby enabling the training of a distributed controller using unconstrained learning algorithms akin to neural networks. This sparse neural controller structure facilitates distributed implementation across individual subsystems while preserving communication topology among the subsystems.

Notation

Throughout the paper, we denote with $\mathbb{N}$ the set of non-negative integers. I_a (0_a) is the identity (zero) square matrix with dimensions $a \times a$, $0_{a \times b}$ is the zero matrix with dimension $a \times b$ and $\mathbb{1}$ is the vector of all ones, where the dimension is clear from the context. We denote with $\text{diag}(v)$

a square diagonal matrix with the elements of vector v on the main diagonal, while blkdiag(A, B) denotes a block diagonal matrix created by aligning the matrices A and B along the diagonal. Positive semidefinite matrices A are denoted as $A \succeq 0$. The set of all sequences $\mathbf{v} = (v_0, v_1, v_2, \dots)$, where $v_t \in \mathbb{R}^n$ for all $t \in \mathbb{N}$, is denoted as l^n. Moreover, $\mathbf{v}$ belongs to $l_2^n \subset l^n$ if $\|\mathbf{v}\|_2 = \left(\sum_{t=0}^{\infty} |v_t|^2 \right)^{\frac{1}{2}} < \infty$. We refer to $\mathbf{v_T}$ to denote a truncation of $\mathbf{v}$ with t ranging from 0 to T. An operator $\mathbf{A} : l^n \to l^m$ is said to be *causal* if $\mathbf{A}(\mathbf{x}) = (A_0(\mathbf{x_0}), A_1(\mathbf{x_1}), \dots, A_t(\mathbf{x_t}), \dots)$. If $A_t(\mathbf{x_t}) = A_t(0, \mathbf{x_{t-1}})$, then $\mathbf{A}$ is said to be *strictly* causal. An operator $\mathbf{A} : l^n \to l^m$ is said to be $\mathcal{L}_2$ stable if it is causal and $\mathbf{A}(\mathbf{a}) \in l_2^m$ for all $\mathbf{a} \in l_2^n$. Equivalently, we write $\mathbf{A} \in \mathcal{L}_2$. Given an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ described by the set of nodes $\mathcal{V}$ and the set of edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$, we denote set of neighbors of node i, including i itself by $\mathcal{N}_i = \{i\} \cup \{j \mid \{i, j\} \in \mathcal{E}\} \subseteq \mathcal{V}$. We denote with $\text{col}_{j \in \mathcal{V}}(v^{[j]})$ a vector which consists of the stacked subvectors $v^{[j]}$ from $j = 1$ to $j = |\mathcal{V}|$ and with $v^{[\mathcal{N}_i]}$ a vector composed by the stacked subvectors $v^{[i]}$, i.e. $v^{[\mathcal{N}_i]} = \text{col}_{i \in \mathcal{N}_i}(v^{[i]})$.

II. PRELIMINARIES

A. Distributed nonlinear systems

We consider a network of N interconnected nonlinear subsystems. The coupling network among the subsystems is defined as an undirected communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V} = \{1, \dots, N\}$ representing the subsystems in the network, and the set of edges $\mathcal{E}$ containing the pairs of subsystems $\{i, j\}$, which can communicate with each other. Each subsystem is of the form

$$x_t^{[i]} = f^{[i]}(x_{t-1}^{[\mathcal{N}_i]}, u_{t-1}^{[i]}) + w_t^{[i]}, \quad \forall t \geq 1, \qquad (1)$$

where state and input of each subsystem $i \in \mathcal{V}$ are denoted by $x^{[i]} \in \mathbb{R}^{n_i}$ and $u^{[i]} \in \mathbb{R}^{m_i}$ respectively. Moreover, $w^{[i]} \in \mathbb{R}^{n_i}$ is an unknown process noise with $w_0^{[i]} = x_0^{[i]}$.
In operator form, we can express the subsystem (1) as:

$$\mathbf{x}^{[i]} = \mathbf{F}^{[i]}(\mathbf{x}^{[\mathcal{N}_i]}, \mathbf{u}^{[i]}) + \mathbf{w}^{[i]}, \qquad (2)$$

where $\mathbf{F}^{[i]} : l^{n_{\mathcal{N}_i}} \times l^{m_i} \to l^{n_i}$ is a strictly causal operator defining the subsystem dynamics such that $\mathbf{F}^{[i]}(\mathbf{x}^{[\mathcal{N}_i]}, \mathbf{u}^{[i]}) = \left(0, f^{[i]}\left(x_0^{[\mathcal{N}_i]}, u_0^{[i]}\right), \dots, f^{[i]}\left(x_t^{[\mathcal{N}_i]}, u_t^{[i]}\right), \dots \right)$.
By combining the local system dynamics in (1), the dynamics of the global system result in

$$x_t = f(x_{t-1}, u_{t-1}) + w_t, \quad \forall t \geq 0, \qquad (3)$$

where $x = \text{col}_{i \in \mathcal{V}}(x^{[i]}) \in \mathbb{R}^n$, $u = \text{col}_{i \in \mathcal{V}}(u^{[i]}) \in \mathbb{R}^m$, $w = \text{col}_{i \in \mathcal{V}}(w^{[i]}) \in \mathbb{R}^n$. Similarly to the subsystems, we can rewrite system (3) in operator form as:

$$\mathbf{x} = \mathbf{F}(\mathbf{x}, \mathbf{u}) + \mathbf{w}, \qquad (4)$$

We make the following Assumption on the system to be controlled:

Assumption 1: We assume that the causal operator $\mathbf{F}$ is such that the map $(\mathbf{w}, \mathbf{u}) \mapsto \mathbf{x}$ lies in $\mathcal{L}_2$, and that the process noise $w(t) \sim \mathcal{D}$ is distributed according to an unknown distribution $\mathcal{D}$, and that $\mathbf{w}$ belongs to l_2^n.

The Assumption above means that the interconnection is stable or locally controlled to achieve $\mathcal{L}_2$ stability. This is typically true in practical applications where systems are either pre-stabilized or can be stabilized using simple controllers. While these controllers ensure stability, they often result in suboptimal closed-loop performance. Our goal is to enhance performance while maintaining stability.

To control the system, we consider nonlinear state-feedback causal policies parametrized as follows

$$\mathbf{u} = \mathbf{K}(\mathbf{x}, \mathbf{u}) = (K_0(x_0), \dots, K_t(\mathbf{x_t}, \mathbf{u_{t-1}}), \dots). \qquad (5)$$

Thanks to causality, each sequence of disturbances, denoted as $\mathbf{w} \in l_2^n$, results in unique trajectories for the closed-loop system (4)-(5). This implies that the mappings from disturbances to state and control, denoted as $\mathbf{\Phi^x}[\mathbf{F}, \mathbf{K}]$ and $\mathbf{\Phi^u}[\mathbf{F}, \mathbf{K}]$ respectively, are well-defined for any given system $\mathbf{F}$ and controller $\mathbf{K}$. Thus, we express $\mathbf{x} = \mathbf{\Phi^x}[\mathbf{F}, \mathbf{K}](\mathbf{w})$ and $\mathbf{u} = \mathbf{\Phi^u}[\mathbf{F}, \mathbf{K}](\mathbf{w})$ for all $\mathbf{w} \in l_2^n$.

B. Problem formulation

Our goal is to address the following challenges in designing a control policy $\mathbf{K}(\mathbf{x}, \mathbf{u})$:

C-1 Ensuring the stability of the closed-loop mappings $\mathbf{\Phi^x}[\mathbf{F}, \mathbf{K}]$ and $\mathbf{\Phi^u}[\mathbf{F}, \mathbf{K}]$ in the $\mathcal{L}_2$ sense.

C-2 Enhancing the performance of the closed-loop system by minimizing a loss function defined as:

$$J = \mathbb{E}_{\mathbf{w} \sim \mathcal{D}} \left[\ell(\mathbf{x_T}, \mathbf{u_T}) \right], \qquad (6)$$

where ℓ is differentiable and $\ell(\mathbf{x_T}, \mathbf{u_T}) \geq 0$ for all $T \in \mathbb{N}$ and all $(x, u) \in \mathbb{R}^{n+m}$.

C-3 Ensuring that the resulting policy $\mathbf{K}(\mathbf{x}, \mathbf{u})$ can be implemented in a distributed manner exploiting a neighbor-to-neighbor communication. Specifically, we require that

$$\mathbf{u}^{[i]} = \mathbf{K}^{[i]}(\mathbf{x}^{[\mathcal{N}_i]}, \mathbf{u}^{[\mathcal{N}_i]}), \quad \forall i \in \mathcal{V}. \qquad (7)$$

Remark 1: Unlike most optimal control and dynamic programming problems that rely on a sum of stage costs over time, our approach is more flexible. We only require the loss function to be differentiable in its arguments. This allows us to incorporate complex objectives typical of many robotics and reinforcement learning tasks, such as logical specifications formulated using Signal Temporal Logic (STL) [20].

We can now define the finite horizon ODC problem we aim to solve:

$$\min_{\mathbf{K}(\cdot)} \qquad \mathbb{E}_{\mathbf{w} \sim \mathcal{D}} \left[\ell(\mathbf{x_T}, \mathbf{u_T}) \right] \qquad (8a)$$

$$\text{subject to} \quad w_0 = x_0, \quad (1), (5), (7), \ \forall i \in \mathcal{V}, \qquad (8b)$$

$$(\mathbf{\Phi^x}[\mathbf{F}, \mathbf{K}], \mathbf{\Phi^u}[\mathbf{F}, \mathbf{K}]) \in \mathcal{L}_2, \qquad (8c)$$

where (5), (7) impose the causality and sparsity structure of the obtained policy while ensuring the satisfaction of (8c) guarantees the closed-loop stability of the distributed system at a global network level.

Remark 2: Under Assumption 1, the problem (8) is always feasible (consider for example $\mathbf{K} = \mathbf{0}$).

Remark 3: Critically, (8c) enforces strict closed-loop stability at the network level for all optimized distributed
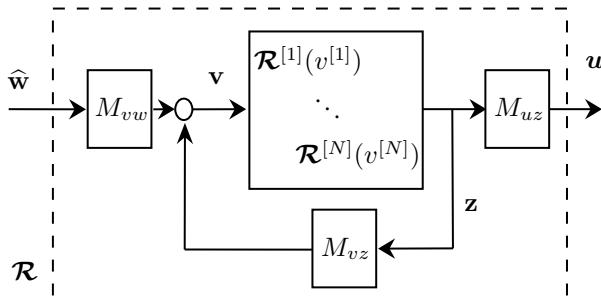
Fig. 1: Interconnection of N operators $\mathcal{R}^{[i]}$.

policies. However, in this paper we consider a stronger requirement since we are interested in iteratively solving the optimization problem (8) while ensuring *fail-safe learning*. This means remaining within the feasible region defined by (8b) and (8c) at every iteration, while improving the obtained policy by minimizing the loss function (8a).

III. MAIN RESULT: FREE PARAMETRIZATION OF DISTRIBUTED STABLE NEURAL CLOSED-LOOP MAPS

In this section, we will address the three main objectives we aim to achieve. Firstly, we will begin by recalling the process of parametrizing stabilizing control policies for the closed-loop system to establish a framework capable of satisfying **C**-1 with a distributed structure **C**-3. Then, we will demonstrate how a proper parametrization of the controller enables us to fulfill **C**-2 in an unconstrained fashion.

A. Networked controller

To achieve a distributed controller ensuring closed-loop stability, we first recall the findings introduced in [21] for centralized control. Using insights from internal model control [22], the main result of [21] is that under Assumption 1, *all and only* stabilizing control policies for the system $\mathbf{F}$ can be parameterized in terms of one stable operator $\mathcal{R} \in \mathcal{L}_2$ by implementing the control input $\mathbf{u} = \mathcal{R}(\widehat{\mathbf{w}})$ where $\widehat{\mathbf{w}} = \mathbf{x} - \mathbf{F}(\mathbf{x}, \mathbf{u})$ reconstructs the true disturbances $\mathbf{w}$. The resulting closed loop system is given by

$$\begin{aligned} \mathbf{x} &= \mathbf{F}(\mathbf{x}, \mathbf{u}) + \mathbf{w}, \\ \mathbf{u} &= \mathcal{R}(\mathbf{x} - \mathbf{F}(\mathbf{x}, \mathbf{u})), \quad \mathcal{R} \in \mathcal{L}_2, \end{aligned} \quad (9)$$

where $\mathcal{R}$ defines an overall control policy $\mathbf{K}$ such that $\mathbf{u} = \mathbf{K}(\mathbf{x}, \mathbf{u}) = \mathcal{R}(\widehat{\mathbf{w}})$.

Beyond achieving closed-loop stability, the information constraints (7) of the ODC problem (8) bring in additional challenges. Indeed, implementing a controller $\mathbf{u} = \mathcal{R}(\widehat{\mathbf{w}})$ requires, at each node $i \in \mathcal{V}$, information about other $\widehat{\mathbf{w}}^{[i]}$'s, which in turn are reconstructed based on the neighboring states $x^{[\mathcal{N}_i]}$'s and $u^{[i]}$'s, that is, $\mathbf{w}^{[i]} = \mathbf{x}^{[i]} - \mathbf{F}^{[i]}(\mathbf{x}^{[\mathcal{N}_i]}, \mathbf{u}^{[i]})$. This raises the question of how the constraints in (7) impact the design of $\mathcal{R}$.

Our solution is based on designing N networked $\mathcal{L}_2$ sub-operators $\mathcal{R}^{[i]} : l^{q_i} \to l^{r_i}$ whose interconnection structure is shown in Fig. 1 and resulting in the overall local controller shown in Fig. 2. Specifically, we aim to design sub-operators

with input $v^{[i]} \in \mathbb{R}^{q_i}$ and output $z^{[i]} \in \mathbb{R}^{r_i}$ with $q_i \geq n_i$ and $r_i \geq m_i$, $\forall i \in \mathcal{V}$ whose state-space definition is as follows:

$$\begin{aligned} \xi_{t+1}^{[i]} &= \rho^{[i]}\left(\xi_t^{[i]}, v_t^{[i]}\right) \\ z_t^{[i]} &= \chi^{[i]}\left(\xi_t^{[i]}, v_t^{[i]}\right), \quad \forall i \in \mathcal{V} \qquad \forall t \geq 0, \qquad (10) \end{aligned}$$

where $\xi_t^{[i]} \in \mathbb{R}^c$ is the internal state. We assume that each sub-operator has a finite $\mathcal{L}_2$-gain not grater than $\gamma^{[i]}$, meaning that there exists a non-negative local storage function $V^{[i]}$ such that for all $t \in \mathbb{N}$

$$V^{[i]}(\xi_{t+1}^{[i]}) - V^{[i]}(\xi_t^{[i]}) \leq s^{[i]}(v_t^{[i]}, z_t^{[i]}), \qquad (11)$$

where $s^{[i]}(v^{[i]}, z^{[i]})$ is the quadratic supply rate

$$s^{[i]}(v^{[i]}, z^{[i]}) = \begin{bmatrix} v^{[i]} \\ z^{[i]} \end{bmatrix}^\top \underbrace{\begin{bmatrix} \gamma^{[i]2} I_{q_i} & 0_{q_i \times r_i} \\ 0_{r_i \times q_i} & -I_{r_i} \end{bmatrix}}_{X^{[i]}} \begin{bmatrix} v^{[i]} \\ z^{[i]} \end{bmatrix}.$$
$$(12)$$

It is worth mentioning that (11) is the definition of dissipativity for the system (10) with respect to the generic supply rate $s^{[i]}(v^{[i]}, z^{[i]})$. For the specific supply given in (12), we retrieve the well-known concept of finite $\mathcal{L}_2$-gain property, which is indeed a particular case of dissipativity (see [23]).

The interconnection structure of sub-operators shown in Fig. 1 is formalized as follows:

$$\begin{bmatrix} v \\ u \end{bmatrix} = \bar{M} \begin{bmatrix} z \\ \widehat{w} \end{bmatrix} = \begin{bmatrix} M_{vz} & M_{vw} \\ M_{uz} & 0_{m \times n} \end{bmatrix} \begin{bmatrix} z \\ \widehat{w} \end{bmatrix}, \quad (13)$$

where $v \in \mathbb{R}^q$ are the inputs to local controllers and $z \in \mathbb{R}^r$ the outputs. The matrix $\bar{M}$ describes how each sub-operator input $v^{[i]}$ is coupled with each sub-operator output $z^{[i]}$ as well as with the exogenous input $w^{[i]}$ and output $u^{[i]}$. Specifically, the matrix M_{vz} describes how the input of each sub-operator is linked to the outputs of the others. In practice, M_{vz} is defined coherently with the network structure defined by the graph $\mathcal{G}$ so that v_i depends on $z_{\mathcal{N}_i}$.

To obtain an overall operator $\mathcal{R}$ with a finite $\mathcal{L}_2$ gain $\gamma_R > 0$ between $\widehat{w}$ and u, we need to guarantee the dissipativity of the interconnected operator $\mathcal{R}$, with respect to the quadratic supply rate:

$$\begin{bmatrix} \widehat{w} \\ u \end{bmatrix}^\top \underbrace{\begin{bmatrix} \gamma_R^2 I_n & 0_{n \times m} \\ 0_{m \times n} & -I_m \end{bmatrix}}_{S} \begin{bmatrix} \widehat{w} \\ u \end{bmatrix}. \quad (14)$$

We are now in the position to state the following proposition guaranteeing the $\mathcal{L}_2$ stability of the closed-loop (9).

Proposition 1: Consider the closed-loop system given by (9) where the controller operator $\mathcal{R}(\widehat{w})$ is composed of sub-operators interconnected according to (13). Let Assumption 1 hold, then the system (9) is $\mathcal{L}_2$ stable, i.e., $(\mathbf{\Phi}^{\mathbf{x}}[\mathbf{F}, \mathbf{K}], \mathbf{\Phi}^{\mathbf{u}}[\mathbf{F}, \mathbf{K}]) \in \mathcal{L}_2$ if there exist scalars $\alpha^{[i]} \geq 0, i \in \mathcal{V}$ such that the following inequality holds:

$$\begin{bmatrix} M_{vz} & M_{vw} \\ I & 0 \\ 0 & I \\ M_{uz} & 0 \end{bmatrix}^\top \begin{bmatrix} \mathbf{X}(\alpha^{[i]} X^{[i]}) & 0 \\ 0 & -S \end{bmatrix} \begin{bmatrix} M_{vz} & M_{vw} \\ I & 0 \\ 0 & I \\ M_{uz} & 0 \end{bmatrix} \preceq 0,$$
$$(15)$$
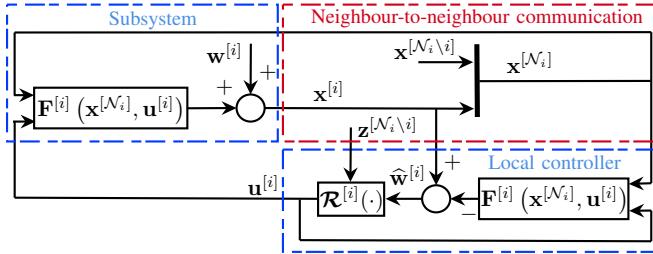
Fig. 2: Proposed local controller parametrizing stabilizing controllers in terms of freely chosen sub-operator $\mathcal{R}^{[i]} \in \mathcal{L}_2$.

where $\mathbf{X}(\alpha^{[i]} X^{[i]})$ is a matrix composed of the weighted matrices that define the local supply functions $X^{[i]}$ (see Appendix A), and the matrices M_{vz}, M_{vw}, M_{uz} and S are defined in (13) and (15).

Proof: Thanks to Assumption 1, from [21, Thm. 2] we have that $(\mathbf{\Phi^x}[\mathbf{F}, \mathbf{K}], \mathbf{\Phi^u}[\mathbf{F}, \mathbf{K}]) \in \mathcal{L}_2$ for any $\mathcal{R} \in \mathcal{L}_2$. Now we have to prove that the operator $\mathcal{R}$ composed by interconnecting $\mathcal{L}_2$ stable operators $\mathcal{R}^{[i]}$ as in (13) has a finite $\mathcal{L}_2$ gain if the inequality (15) is satisfied. Following [11], we consider as candidate storage function the weighted sum of the local storage functions for the whole interconnection $V(\xi) = \alpha^{[1]} V^{[1]}(\xi^{[1]}) + \cdots + \alpha^{[N]} V^{[N]}(\xi^{[N]})$, with $\alpha^{[i]} \geq 0$ and $\xi = col_{i \in \mathcal{V}}(\xi^{[i]})$. Thus, to obtain a finite $\mathcal{L}_2$ gain between $\widehat{w}$ and u we want that the interconnection is dissipative with respect to the weighted sum of the supply rates (12) and that this term is dominated by the performance supply rate (14), i.e.,

$$\sum_{i=1}^{N} \alpha^{[i]} \left[V^{[i]}(\xi_{t+1}^{[i]}) - V^{[i]}(\xi_t^{[i]}) \right] \leq$$

$$\sum_{i=1}^{N} \alpha^{[i]} \begin{bmatrix} v^{[i]} \\ z^{[i]} \end{bmatrix}^{\top} X^{[i]} \begin{bmatrix} v^{[i]} \\ z^{[i]} \end{bmatrix} \leq$$

$$\begin{bmatrix} \widehat{w} \\ u \end{bmatrix}^{\top} \underbrace{\begin{bmatrix} \gamma_R^2 I_n & 0_{n \times m} \\ 0_{m \times n} & -I_m \end{bmatrix}}_{S} \begin{bmatrix} \widehat{w} \\ u \end{bmatrix} \leq 0. \quad (16)$$

By rearranging the right-hand side of (16) we can rewrite it as:

$$\begin{bmatrix} v \\ z \\ \widehat{w} \\ u \end{bmatrix}^{\top} \begin{bmatrix} \mathbf{X}(\alpha^{[i]} X^{[i]}) & 0 \\ 0 & -S \end{bmatrix} \begin{bmatrix} v \\ z \\ \widehat{w} \\ u \end{bmatrix} \leq 0. \quad (17)$$

By substituting the interconnection (13), we obtain (15). ∎

Proposition 1 allows us to fulfill **C**-1 and **C**-3 by providing a condition that ensures the closed-loop stability at the network level (9) with a sparse and causal controller. We will address **C**-2 in the next section.

B. Free parametrization of networked RENs

Recently, finite-dimensional DNN approximations of non-linear $\mathcal{L}_p$ operators [21], [24] have been proposed. Specifically, as suggested in [21] an operator $\mathcal{R} : l^n \to l^m$ is a

REN if $\mathbf{u} = \mathcal{R}(\widehat{\mathbf{w}})$ is generated by the following dynamical system with input $\widehat{w} \in \mathbb{R}^n$ and output $u \in \mathbb{R}^m$:

$$\begin{bmatrix} \xi_t \\ \nu_t \\ u_t \end{bmatrix} = \underbrace{\begin{bmatrix} A_1 & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix}}_{Z(\theta, \gamma)} \begin{bmatrix} \xi_{t-1} \\ \sigma(\nu_t) \\ \widehat{w}_t \end{bmatrix}, \quad \xi_{-1} = 0,$$
(18)

where $\xi \in \mathbb{R}^c$ and $\nu \in \mathbb{R}^s$ are the hidden states and the input of neurons of the REN, while $\sigma : \mathbb{R} \to \mathbb{R}$ is the activation function applied element-wise. Further, $\sigma(\cdot)$ must be piecewise differentiable and with first derivatives restricted to the interval $[0, 1]$. As highlighted in [24], RENs are as expressive as various existing DNN architectures. However, when picking any random value for their parameters, the operator $\mathcal{R}$ defined by (18) might not lie in the space $\mathcal{L}_2$. In [24] the authors introduce a smooth function Z that takes unconstrained training parameters $\theta \in \mathbb{R}^{n_\theta}$, for a suitable n_θ, and a prescribed $\mathcal{L}_2$ gain $\gamma > 0$ map them to a matrix $Z(\theta, \gamma)$, which defines (18). Notably, they ensure that the resulting operator $\mathcal{R}[\theta, \gamma]$ is in $\mathcal{L}_2$ and has $\mathcal{L}_2$-gain at most γ for any value of the parameters θ. For this reason the parametrization induced by Z is termed *free*.

Since our focus is on using a networked operator as explained in Section III-A, we consider an interconnection of RENs that replicates the sparsity of the system. To meet requirement **C**-2 without dealing with a large-dimensional constrained optimization problem, we aim to develop a free parametrization of the networked operator. This means finding a parametrization that allows any parameter value while still meeting the constraint. To achieve this, we consider the following dynamics of the interconnected sub-operators $\mathcal{R}^{[i]}$:

$$\begin{bmatrix} \xi_t^{[i]} \\ \nu_t^{[i]} \\ z_t^{[i]} \end{bmatrix} = \underbrace{\begin{bmatrix} A_1^{[i]} & B_1^{[i]} & B_2^{[i]} \\ C_1^{[i]} & D_{11}^{[i]} & D_{12}^{[i]} \\ C_2^{[i]} & D_{21}^{[i]} & D_{22}^{[i]} \end{bmatrix}}_{Z^{[i]}(\theta^{[i]}, \gamma^{[i]})} \begin{bmatrix} \xi_{t-1}^{[i]} \\ \sigma(\nu_t^{[i]}) \\ v_t^{[i]} \end{bmatrix}, \quad (19)$$

where $\xi_{-1}^{[i]} = 0$, $\forall i \in \mathcal{V}$, and $Z^{[i]}(\theta^{[i]}, \gamma^{[i]})$ defining the free parametrization of the local REN model (19). Before entering the details of the proposed parametrization, we consider the following Assumption on the chosen interconnection for the network of RENs.

Assumption 2: We assume a particular structure for the matrices M_{vw} and M_{uz}, more in detail:

(a) we assume that there exists a permutation matrix $C \in [0, 1]^{q \times q}$ such that

$$C M_{vw} = \begin{bmatrix} I_n \\ 0_{(q-n) \times n} \end{bmatrix}. \quad (20)$$

Notice that this also implies that M_{vw} is a positive-valued semi-orthogonal matrix satisfying $M_{vw}^{\top} M_{vw} = I_n$.

(b) We assume that M_{uz} is semi-orthogonal, i.e., it satisfies

$$M_{uz}^{\top} M_{uz} = H, \quad H = \text{diag}(h), \ h \in \mathbb{R}^r. \quad (21)$$

Remark 4: A few comments on Assumption 2 are in order. With (a) we allow the input of one or more sub-operators to depend on the exogenous input. More in detail, the assumption covers all cases where each sub-system can have inputs depending on different exogenous inputs. What it does not cover, is the case in which different sub-systems have inputs depending on the combination of *common* exogenous inputs. We aim to relax this assumption in future work. Condition (b) states that the columns of M_{uz} are mutually orthogonal, which in turn implies that the output of the operator $\mathcal{R}$ is comprised of linearly independent combinations of the sub-operators output z. This is fairly general and gives one enough leeway on how the output u is chosen. As a particular case, we have $M_{uz} = I_r$ where the output of the operator coincides with the output of all sub-operators.

Before presenting our main result, it is useful to define the set of such sub-operators that takes at least one exogenous input, which will be called *out-connected*. First of all, we define

$$\mathcal{A}_{u_i} = \text{set of indices of } u \text{ associated with } i \in \mathcal{V}, \quad (22)$$
$$\mathcal{A}_{z_i} = \text{set of indices of } z \text{ associated with } i \in \mathcal{V}. \quad (23)$$

With this, we can define

$$\mathcal{A}^1_{u_i} = \{k \in \mathcal{A}_{u_i} \mid (M_{vw}\mathbb{1})_k = 1\}, \quad i \in \mathcal{V}, \quad (24)$$
$$\mathcal{A}^0_{u_i} = \{k \in \mathcal{A}_{u_i} \mid (M_{vw}\mathbb{1})_k = 0\}, \quad i \in \mathcal{V}. \quad (25)$$

These are those indices of u_i that are associated with the exogenous inputs and those that are not respectively. If $\mathcal{A}^1_{u_i}$ is not empty, we say that the sub-operator i is out-connected. The set of out-connected operators is

$$\mathcal{V}^1 = \left\{i \in \mathcal{V} \mid \mathcal{A}^1_{u_i} \neq \emptyset\right\}. \quad (26)$$

We are now able to present our main result concerning a free parametrization of distributed $\mathcal{L}_2$-stable operators $\mathcal{R}$ composed by a network of RENs.

Theorem 1: Consider a set $\mathcal{V} = \{1, \dots, N\}$ of sub-operators $\mathcal{R}^{[i]}[\theta^{[i]}, \gamma^{[i]}]$, $i \in \mathcal{V}$, each dissipative with respect to the supply (12), and interconnected according to (13). By choosing the $\mathcal{L}_2$ gain $\gamma^{[i]}$ of each sub-operator $i \in \mathcal{V}$ as follows:

$$\gamma^{[i]} = \eta(b^{[i]}) = \begin{cases} \left(\frac{1}{\alpha^{[i]}} \min\left\{\frac{\gamma_R^2}{\max_{j \in \mathcal{A}^1_{u_i}}\left(\sum_k |m_{jk}|\right)\gamma_R^2 + 1}, \frac{1}{\max_{j \in \mathcal{A}^0_{u_i}}\left(\sum_k |m_{jk}|\right)}\right\}\right)^{\frac{1}{2}}, & \text{if } i \in \mathcal{V}^1 \\ \left(\frac{1}{\alpha^{[i]} \max_{j \in \mathcal{A}^0_{u_i}}\left(\sum_k |m_{jk}|\right)}\right)^{\frac{1}{2}}, & \text{otherwise} \end{cases} \quad (27)$$

where

$$\alpha^{[i]} = h^{[i]} + \max_{j \in \mathcal{A}_{z_i}}\left(\sum_k |m_{kj}|\right) + {b^{[i]}}^2, \ \forall i \in \mathcal{V}, \quad (28)$$

the interconnected system composed by the sub-operators $\mathcal{R}^{[i]}[\theta^{[i]}, \eta(b^{[i]})]$ satisfy condition (15) for all $\theta^{[i]}, b^{[i]}$, and the resulting operator $\mathcal{R}[\theta, \gamma_R]$ lies in $\mathcal{L}_2$.

The proof of Theorem 1 is moved to the Appendix for the sake of readability. Theorem 1, together with Proposition 1, allows us to fulfill **C**-2 by enabling an unconstrained parametrization of $\mathcal{L}_2$-stable distributed operators, effectively removing the need to enforce the hard constraint 8c which, thanks to Proposition 1, led to the inequality (15). This is important as it allows the use of parametrized families of highly non-linear operators such as RENs, which can be optimized very efficiently by solving an unconstrained optimization problem over the free parameters $\theta^{[i]}, b^{[i]}$ with standard backpropagation and unconstrained gradient descent methods.

Remark 5: Notice that Theorem 1 generalizes [25, Thm. 1]. In that paper, we considered the special case where $M_{uz}^\top M_{uz} = I_r$ and $M_{vw} = I_q$. The latter condition implies that w can only be an additive term in each sub-operator's input v (note that this also implies that w is constrained to have the same dimension as the input v). In such a special case, we have $\mathcal{V}^1 = \mathcal{V}, \mathcal{A}^1_{u_i} = \mathcal{A}_{u_i}$ and $\mathcal{A}^0_{u_i} = \emptyset \ \forall i \in \mathcal{V}$, therefore (27) reduces to

$$\eta(b^{[i]}) = \sqrt{\frac{1}{\alpha^{[i]}}\frac{\gamma_R^2}{\max_{j \in \mathcal{A}_{u_i}}\left(\sum_k |m_{jk}|\right)\gamma_R^2 + 1}}, \ \forall i \in \mathcal{V}, \quad (29)$$

which is the parametrization originally proposed in [25].

A detailed description of how Theorem 1 can be leveraged for the training of a network of $\mathcal{L}_2$-bounded RENs, is provided by Algorithm 1.

Remark 6: There is usually a "price to pay" for transforming the problem (8) into an unconstrained one via a free parametrization like the one we derived with Theorem 1. Mainly, this stems from the fact that the map η will only output a subset of all possible $\mathcal{L}_2$-gains such that the overall operator $\mathcal{R}$ lies in $\mathcal{L}_2$, which in turn means that we are restricting ourselves to optimize over a subset of $\mathcal{L}_2$-bounded operators. In particular, this comes from the use of sufficient but not necessary conditions, like Gershgorin Theorem, to ensure that (15) is satisfied and the reader can take a look at the proof in the Appendix for the details. Finding less conservative methods for guaranteeing the satisfaction of (15) is still a matter of ongoing research. Despite this, as it is shown in [24], the advantages of having a free parametrization greatly outweigh such a drawback.

Remark 7: Notice that, due to the assumption made regarding the structure of the matrices M_{vz} and M_{vw}, the computation of each $\gamma^{[i]}$ according to (1) only necessitates knowledge of the coupling matrices M_{uz} and M_{vz}. Therefore, by assuming suitable separability properties of the costs among agents, the proposed algorithm could be adapted to a distributed implementation. The distributed design of local controllers is not within the scope of this paper and will be addressed in future research.

Algorithm 1 Training of networked $\mathcal{L}_2$ stable RENs

---

Input: Input data: initial conditions $\{w_0^j\}_{j=1}^{n_{exp}}$ sampled from $\mathcal{D}$, system model $f(x,u)$, learning rate η, number of epochs E.

Initialize randomly the parameters of each subsystem: $\theta^{[i]}$, $b^{[i]}$, $i \in \mathcal{V}$ and γ_R .

for $e = 1$ to E **do**
    for $j = 1$ to n_{exp} **do**
        Initialize hidden state $\xi^{[i]}$, $\forall i \in \mathcal{V}$.
        Initialize control action $u^{[i]}$, $\forall i \in \mathcal{V}$.
        Compute $\gamma^{[i]} = \eta(b^{[i]})$, $\forall i \in \mathcal{V}$ using Eq. (27).
        for $t = 0$ to T **do**
            Simulate system (3)
            Reconstruct the noise $\widehat{w}$
            Compute the inputs:
            $u_t^{[i]} = M_{uz}^{[i]} \left(C_2^{[i]} \xi^{[i]}_{t-1} + D_{21}^{[i]} \sigma(\nu_t^{[i]}) + D_{22}^{[i]} \widehat{w}_t^{[i]} \right)$
        end for
    end for
    Compute loss $\mathbb{E}_{\mathbf{w} \sim \mathcal{D}} [\ell(\mathbf{x_T}, \mathbf{u_T})]$.
    Backpropagate gradients through time (see [26]) to update parameters via gradient descent with learning rate η.
end for
Output: Trained model parameters $\theta_i, b^{[i]}$, $i \in \mathcal{V}$.

$m^{[i]}$	$1\ kg$	$c^{[i]}$	$1\ Ns/m$
$k^{[ij]}$	$1\ N/m$	$k^{[ir]}$	$1\ N/m$
$\delta^{[1]} = \delta^{[3]}$	$4\ m$	$\delta^{[2]} = \delta^{[4]}$	$1.5\ m$
T_s	$0.05\ s$	Q	blkdiag$(I_{16}, 0.01 \cdot I_8)$
η	0.001	E	1500
$\xi^{[i]}$	$25, \forall i \in \mathcal{V}$	$\nu^{[i]}$	25

TABLE I: List of parameters employed in the simulation $\forall i, j \in \mathcal{V}$.

IV. NUMERICAL EXAMPLE

In this section, we demonstrate the effectiveness of the proposed approach in addressing a complex control problem. We use PyTorch to build and train our distributed neural controller, employing stochastic gradient descent method. The implementation of our approach can be found at `https://github.com/DecodEPFL/Distributed_neurSLS`. The system we are considering consists of four point-mass vehicles (see Fig. 3), each with a position denoted as $p^{[i]} \in \mathbb{R}^2$ and velocity represented as $v^{[i]} \in \mathbb{R}^2$. Each vehicle experiences friction forces, which leads to the discrete-time model:

$$\begin{bmatrix} p_{t+1}^{[i]} \\ v_{t+1}^{[i]} \end{bmatrix} = \begin{bmatrix} p_t^{[i]} \\ v_t^{[i]} \end{bmatrix} + T_s \begin{bmatrix} v_t^{[i]} \\ m^{[i]-1} \left(-c^{[i]} |v_t^{[i]}|_2 + F_t^{[i]} + u_t^{[i]} \right) \end{bmatrix}, \tag{30}$$

where, $m^{[i]} > 0$ is the vehicle mass, $c^{[i]} > 0$ is the friction coefficient, $u^{[i]}$ is the local control input, $T_s > 0$ represents the sampling time and $F^{[i]} \in \mathbb{R}^2$ stands for the action of a local base controller whose goal is to maintain a specific formation between the agents, which is described by a fixed distance $\delta^{[i]} > 0$ between them. More specifically, the base controller stabilizes each agent around a final reference point $\bar{p}^{[i]}$, by applying the following force:

$$F^{[i]} = - \sum_{j=1}^{|\mathcal{N}_i \setminus i|} k^{[ij]} (|p^{[i]} - p^{[j]}|_2 - \delta^{[i]}) - k^{[ir]} |\bar{p}^{[i]} - p^{[i]}|_2, \tag{31}$$

where $, k^{[ij]}, k^{[ir]} > 0$ are the gains used in the local controllers. The overall system obtained thus consists of nonlinear coupled subsystems (30)- (31). These subsystems can be written as (1), for which the origin is a stable equilibrium. We chose the loss function in (6) as:

$$\ell(\mathbf{x_T}, \mathbf{u_T}) = \sum_{t=0}^{T} \ell(x_t, u_t), \tag{32}$$

$$\ell(x_t, u_t) = \ell_{traj}(x_t, u_t) + \ell_{ca}(x_t) + \ell_{obs}(x_t) + \ell_{form}(x_t),$$

where, $\ell_{traj}(x, u) = [x^\top\ u^\top] Q [x^\top\ u^\top]^\top$ with $Q \succeq 0$ penalizes the distance between agents and their reference, as well as the magnitude of the control action. The other terms, $\ell_{ca}(x)$, $\ell_{obs}(x)$ and $\ell_{form}(x_t)$ instead, are barrier functions that penalize collisions between agents and obstacles, and incentivize maintaining the desired formation, respectively (see Fig. 3). The expected value in the cost function (8a) is approximated by an empirical average (see [21]) over 10 randomly sampled initial conditions $\{w_0^s\}_{s=1}^{10}$. The parameters employed in the simulation are listed in Tab I. Fig. 3 shows the closed loop trajectories of the controlled system. The goal promoted by the cost $\ell(x, u)$ is to coordinate the formation in order to pass through a narrow valley avoiding collision with each others while keeping the connectivity required by the formation control. The agents start from a randomly sampled initial position marked with "∘" and have to reach the target position marked with "⋆".

V. CONCLUSIONS

This paper introduces a novel distributed control method for interconnected nonlinear systems. By harnessing the framework of neural system level synthesis and dissipativity theory, we achieve optimized performance for user-defined cost functions while guaranteeing stability throughout the training process at the global network level. A key feature is the ability to train local control policies using unconstrained optimization algorithms. We hope that the proposed framework will pave the way for applications in several domains where achieving both stability and optimal performance has traditionally been difficult, such as smart-grid control [27], flocking control for autonomous vehicles [28], and distributed optimization algorithm design [29]. Further important theoretical developments include the addition of noisy outputs, unknown system dynamics, and parallelizing the training across agents.
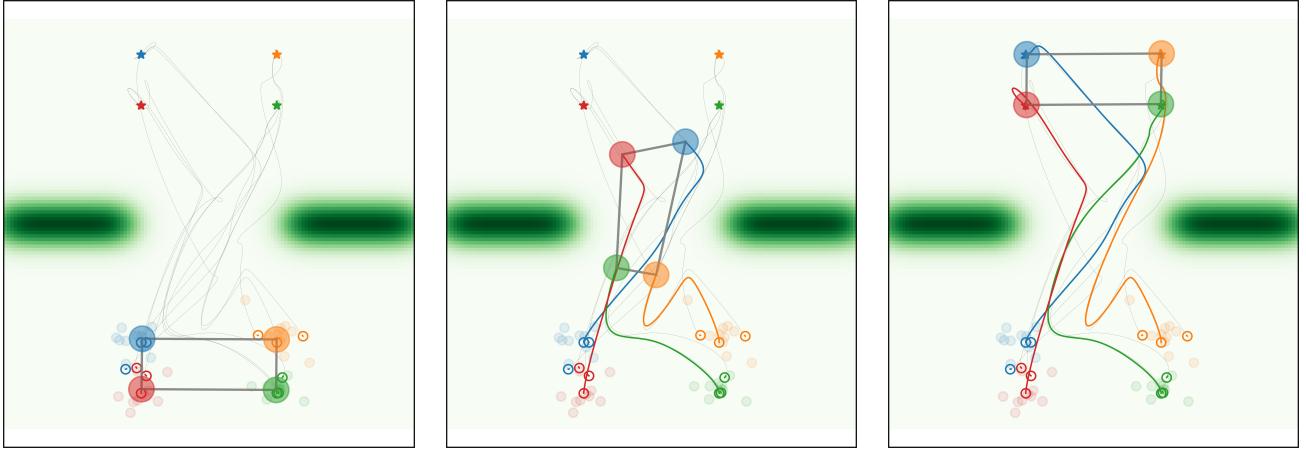
Fig. 3: Closed-loop trajectories after training over randomly sampled initial conditions marked with "○" (filled transparent "○" are the sampled w_0 used for the training). Images captured at time instants $\tau_1 = 0.3\ s$, $\tau_2 = 8.5\ s$ and $\tau_3 = 25\ s$. Colored (gray) lines depict the trajectories within $[0, \tau_i]$ ($[\tau_i, \infty)$). Colored circles (and their radius) represent the agents (and their size for collision avoidance), while colored "⋆" symbols represent the reference position and green ellipsoids are the obstacles.

APPENDIX

A. Proof of Theorem 1

Firstly we rewrite $\mathbf{X}(\alpha^{[i]} X^{[i]})$ in a more compact form as:

$$\mathbf{X}(\alpha^{[i]} X^{[i]}) = \begin{bmatrix} \Pi_{N,v} & 0 \\ 0 & -\Pi_{N,z} \end{bmatrix}, \qquad (33)$$

with $\Pi_{N,v} = \mathrm{blkdiag}\{\alpha^{[1]}\gamma^{[1]^2} I_{q_1}, \ldots, \alpha^{[N]}\gamma^{[N]^2} I_{q_N}\}$ and $\Pi_{N,z} = \mathrm{blkdiag}\{\alpha^{[1]} I_{r_1}, \ldots, \alpha^{[N]} I_{r_N}\}$. By Proposition 1, (15) needs to be satisfied, yielding:

$$\begin{bmatrix} -M_{vz}^\top \Pi_{N,v} M_{vz} + \Pi_{N,z} - M_{uz}^\top M_{uz} & -M_{vz}^\top \Pi_{N,v} M_{vw} \\ -M_{vw}^\top \Pi_{N,v} M_{vz} & -M_{vw}^\top \Pi_{N,v} M_{vw} + \gamma_R^2 I_n \end{bmatrix} \succ 0. \qquad (34)$$

By applying the Schur complement, the above condition is equivalent to

$$-M_{vz}^\top \Pi_{N,v} M_{vz} + \Pi_{N,z} - M_{uz}^\top M_{uz}$$
$$-M_{vz}^\top \Pi_{N,v} M_{vw}[-M_{vw}^\top \Pi_{N,v} M_{vw} \qquad (35)$$
$$+\gamma_R^2 I_n]^{-1} M_{vw}^\top \Pi_{N,v} M_{vz} \succ 0$$

$$-M_{vw}^\top \Pi_{N,v} M_{vw} + \gamma_R^2 I_n \succ 0. \qquad (36)$$

Using the structure of M_{vw} given by Assumption 2, easy computations show that $M_{vw}^\top \Pi_{N,v} M_{vw} = P_1 \in \mathbb{R}^{n \times n}$ is a diagonal matrix where the diagonal elements are given by $\alpha^{[i]}\gamma^{[i]^2}$, $\forall i \in \mathcal{V}^1$, each one repeated as many times as the cardinality of $\mathcal{A}_{u_i}^1$. Therefore, condition (36) reads as

$$-M_{vw}^\top \Pi_{N,v} M_{vw} + \gamma_R^2 I_n \succ 0 \qquad (37)$$
$$-P_1 + \gamma_R^2 I_n \succ 0$$
$$\iff -\alpha^{[i]}\gamma^{[i]^2} + \gamma_R^2 > 0$$
$$\alpha^{[i]}\gamma^{[i]^2} < \gamma_R^2, \forall i \in \mathcal{V}^1. \qquad (38)$$

As for condition (35) by collecting M_{vz} we obtain

$$M_{vz}^\top(-\Pi_{N,v} - \Pi_{N,v} M_{vw}[-M_{vw}^\top \Pi_{N,v} M_{vw}$$
$$+\gamma_R^2 I_n]^{-1} M_{vw}^\top \Pi_{N,v}) M_{vz} + \Pi_{N,z} - M_{uz}^\top M_{uz} \succ 0 \qquad (39)$$

By applying the Schur complement again this is equivalent to:

$$\begin{bmatrix} \Pi_{N,z} - H & M_{vz}^\top \\ M_{vz} & -D^{-1} \end{bmatrix} \succ 0 \qquad (40)$$

$$D \prec 0, \qquad (41)$$

where we recall that $M_{uz}^\top M_{uz} = H$ is a diagonal matrix by Assumption 2 and

$$D = (-\Pi_{N,v} - \Pi_{N,v} M_{vw}[-M_{vw}^\top \Pi_{N,v} M_{vw}$$
$$+\gamma_R^2 I_n]^{-1} M_{vw}^\top \Pi_{N,v}). \qquad (42)$$

Using again the structure of M_{vw} given in Assumption 2, it is easy to show that $M_{vw}[-M_{vw}^\top \Pi_{N,v} M_{vw} + \gamma_R^2 I_n]^{-1} M_{vw}^\top = P_2 \in \mathbb{R}^{q \times q}$ is a diagonal matrix such that $(P_2)_{jj} = (-\alpha^{[i]}\gamma^{[i]^2} + \gamma_R^2)^{-1}$ if $j \in \mathcal{A}_{u_i}^1$, $i \in \mathcal{V}^1$ and $(P_2)_{jj} = 0$ otherwise. From this, we can see that D is a diagonal matrix given by

$$(D)_{jj} = \begin{cases} \dfrac{\alpha^{[i]}\gamma^{[i]^2}\gamma_R^2}{\alpha^{[i]}\gamma^{[i]^2} - \gamma_R^2} & \text{for } j \in \mathcal{A}_{u_i}^1,\ i \in \mathcal{V}^1 \\ -\alpha^{[i]}\gamma^{[i]^2} & \text{for } j \in \mathcal{A}_{u_i}^0,\ i \in \mathcal{V}. \end{cases} \qquad (43)$$

Notice that condition (41) is always satisfied when (38) holds. Finally, we enforce (40) by applying the Gershgorin Theorem [30]. In particular, let us focus on the first p rows of the matrix (40). We want all the Gershgorin circles to lie

in the non-negative half-plane, i.e.

$$\underbrace{(\Pi_{N,z} - H)_{jj}}_{\text{Center}} - \underbrace{\sum_k |(M_{vz}^\top)_{jk}|}_{\text{Radius}} \geq 0, \quad \forall j \in \mathcal{A}_{z_i}, i \in \mathcal{V}. \tag{44}$$

Notice that the Gershgorin center is $\alpha^{[i]} - 1$ for every $j \in \mathcal{A}_{z_i}$, hence condition (44) can be enforced by imposing $\alpha^{[i]}$ such that:

$$\alpha^{[i]} = h^{[i]} + \max_{j \in \mathcal{A}_{z_i}} \left(\sum_k |m_{kj}| \right) + b^{[i]^2}, \forall i \in \mathcal{V}, \tag{45}$$

where $b^{[i]} \in \mathbb{R}$. Similarly, we can impose the other Gershgorin circles associated with the remaining rows of the matrix (40) to lie in the non-negative half of the plane. Using (43), this yields the conditions:

$$0 \leq \alpha^{[i]} \gamma^{[i]^2} \leq \frac{\gamma_R^2}{\max_{j \in \mathcal{A}_{u_i^w}} \left(\sum_k |m_{jk}| \right) \gamma_R^2 + 1}, \forall i \in \mathcal{V}^1 \tag{46}$$

$$0 \leq \alpha^{[i]} \gamma^{[i]^2} \leq \frac{1}{\max_{j \in \mathcal{A}_{u_i^0}} \left(\sum_k |m_{jk}| \right)}, \forall i \in \mathcal{V}, \tag{47}$$

notice that (46) ensures that (38) holds true. The conditions above can be rewritten as

$$0 \leq \alpha^{[i]} \gamma^{[i]^2} \leq \min \left\{ \frac{\gamma_R^2}{\max_{j \in \mathcal{A}_{u_i}^1} \left(\sum_k |m_{jk}| \right) \gamma_R^2 + 1}, \right.$$
$$\left. \frac{1}{\max_{j \in \mathcal{A}_{u_i}^0} \left(\sum_k |m_{jk}| \right)} \right\}, \forall i \in \mathcal{V}^1 \tag{48}$$

$$0 \leq \alpha^{[i]} \gamma^{[i]^2} \leq \frac{1}{\max_{j \in \mathcal{A}_{u_i}^0} \left(\sum_k |m_{jk}| \right)}, \forall i \in \mathcal{V} \setminus \mathcal{V}^1, \tag{49}$$

where $\alpha^{[i]}$ is given by (45). From this, we readily obtain the parametrization of the $\mathcal{L}_2$ gains given in (27). ∎

REFERENCES

[1] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *nature*, vol. 473, no. 7346, pp. 167–173, 2011.

[2] G. Antonelli, "Interconnected dynamic systems: An overview on distributed control," *IEEE Control Systems Magazine*, vol. 33, no. 1, pp. 76–88, 2013.

[3] A. Barrat, M. Barthelemy, and A. Vespignani, *Dynamical processes on complex networks*. Cambridge university press, 2008.

[4] B. Sinopoli, C. Sharp, L. Schenato, S. Schaffert, and S. S. Sastry, "Distributed control applications within sensor networks," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1235–1246, 2003.

[5] M. Rotkowitz and S. Lall, "A characterization of convex problems in decentralized control," *IEEE transactions on Automatic Control*, vol. 50, no. 12, pp. 1984–1996, 2005.

[6] L. Lessard and S. Lall, "Quadratic invariance is necessary and sufficient for convexity," in *Proceedings of the 2011 American Control Conference*. IEEE, 2011, pp. 5360–5362.

[7] Y.-S. Wang, N. Matni, and J. C. Doyle, "A system-level approach to controller synthesis," *IEEE Transactions on Automatic Control*, vol. 64, no. 10, pp. 4079–4093, 2019.

[8] L. Furieri, Y. Zheng, A. Papachristodoulou, and M. Kamgarpour, "On separable quadratic lyapunov functions for convex design of distributed controllers," in *2019 18th European control conference (ECC)*. IEEE, 2019, pp. 42–49.

[9] ——, "Sparsity invariance for convex design of distributed controllers," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 4, pp. 1836–1847, 2020.

[10] A. Van der Schaft, *L2-gain and passivity techniques in nonlinear control*. Springer, 2000.

[11] M. Arcak, C. Meissen, and A. Packard, *Networks of dissipative systems: compositional certification of stability, performance, and safety*. Springer, 2016.

[12] R. Ortega, A. Van Der Schaft, F. Castanos, and A. Astolfi, "Control by interconnection and standard passivity-based control of port-hamiltonian systems," *IEEE Transactions on Automatic control*, vol. 53, no. 11, pp. 2527–2542, 2008.

[13] R. Scattolini, "Architectures for distributed and hierarchical model predictive control–a review," *Journal of process control*, vol. 19, no. 5, pp. 723–731, 2009.

[14] J. Liu, D. Muñoz de la Peña, and P. D. Christofides, "Distributed model predictive control of nonlinear process systems," *AIChE journal*, vol. 55, no. 5, pp. 1171–1184.

[15] R. Nishihara, L. Lessard, B. Recht, A. Packard, and M. Jordan, "A general analysis of the convergence of admm," in *International conference on machine learning*. PMLR, 2015, pp. 343–352.

[16] E. Tolstaya, F. Gama, J. Paulos, G. Pappas, V. Kumar, and A. Ribeiro, "Learning decentralized controllers for robot swarms with graph neural networks," in *Conference on robot learning*. PMLR, 2020, pp. 671–682.

[17] F. Gama and S. Sojoudi, "Graph neural networks for distributed linear-quadratic control," in *Learning for Dynamics and Control*. PMLR, 2021, pp. 111–124.

[18] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick, "End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 3387–3395.

[19] L. Furieri, C. L. Galimberti, M. Zakwan, and G. Ferrari-Trecate, "Distributed neural network control with dependability guarantees: a compositional port-hamiltonian approach," in *Learning for Dynamics and Control Conference*. PMLR, 2022, pp. 571–583.

[20] K. Leung, N. Aréchiga, and M. Pavone, "Backpropagation through signal temporal logic specifications: Infusing logical structure into gradient-based methods," *The International Journal of Robotics Research*, vol. 42, no. 6, pp. 356–370, 2023.

[21] L. Furieri, C. L. Galimberti, and G. Ferrari-Trecate, "Neural system level synthesis: Learning over all stabilizing policies for nonlinear systems," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 2765–2770.

[22] C. G. Economou, M. Morari, and B. O. Palsson, "Internal model control: Extension to nonlinear system," *Industrial & Engineering Chemistry Process Design and Development*, vol. 25, no. 2, pp. 403–411, 1986.

[23] P. J. Koelewijn and R. Tóth, "Incremental stability and performance analysis of discrete-time nonlinear systems using the lpv framework," *IFAC-PapersOnLine*, vol. 54, no. 8, pp. 75–82, 2021.

[24] M. Revay, R. Wang, and I. R. Manchester, "Recurrent equilibrium networks: Flexible dynamic models with guaranteed stability and robustness," *IEEE Transactions on Automatic Control*, 2023.

[25] L. Massai, D. Saccani, L. Furieri, and G. Ferrari-Trecate, "Unconstrained learning of networked nonlinear systems via free parametrization of stable interconnected operators," *arXiv preprint arXiv:2311.13967*, 2023.

[26] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[27] P. Nahata, R. Soloperto, M. Tucci, A. Martinelli, and G. Ferrari-Trecate, "A passivity-based approach to voltage stabilization in dc microgrids with zip loads," *Automatica*, vol. 113, p. 108770, 2020.

[28] A. Carron, D. Saccani, L. Fagiano, and M. N. Zeilinger, "Multi-agent distributed model predictive control with connectivity constraint," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 3806–3811, 2023.

[29] A. Martin and L. Furieri, "Learning to optimize with convergence guarantees using nonlinear system theory," *arXiv preprint arXiv:2403.09389*, 2024.

[30] H. N. Salas, "Gershgorin's theorem for matrices of operators," *Linear algebra and its applications*, vol. 291, no. 1-3, pp. 15–36, 1999.