POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Development of a unitary approach for physics-based learning and decentralized control with recurrent neural network models

Laurea Magistrale in Automation and Control Engineering

Author: Claudia Sbardi

Advisor: Prof. Marcello Farina

Co-advisor: Dott. Ing. Daniele Ravasio

Academic year: 2024-2025

---

1. Introduction

Large-scale systems are widely used in complex industrial contexts such as chemical processes, energy networks, and transportation systems. Their high dimensionality, coupling among variables, and nonlinearities make centralized control strategies hardly applicable.
To overcome these limitations, decentralized and distributed strategies provide scalable and computationally feasible solutions. However, their effectiveness relies on the ability of the models to capture complex system structure, which makes the derivation of physics-based models particularly challenging.
In this context, the need for a physics-inspired identification procedure arises and Recurrent Equilibrium Networks (RENs) define a suitable class of nonlinear models, to which structural constraints and physical properties can be enforced, so as to obtain suitable models for decentralized control. In its first part, this thesis addresses the identification of control-oriented models through different decomposition approaches and sampling time impact analysis. Three decomposition approaches [1] and the sampling time impact are defined and subsequently tested on a case study consisting of nonlinear chemical plant. Due to space limitations, only one decomposition procedure is presented in this executive summary; further analyses are detailed in the thesis.
In parallel, the decentralized control problem for the class of structured REN models is formulated and a novel solution, based on MPC, is proposed.
This executive summary outlines the two main aspects of the thesis. In particular, Section 2 presents Recurrent Equilibrium Network models, focusing on the structured identification approach, whereas Section 3 formulates the tube-based decentralized nonlinear MPC problem. Further details on both the identification and control contributions are reported in the the thesis.

2. RENs

2.1. The RENs model

The class of Recurrent Equilibrium Networks (RENs) [5] is a novel class of models designed to represent nonlinear dynamical systems. Their construction is based on the feedback interconnection between a linear dynamical system G and an implicit function, defined by a non-linear operator σ, as represented in Figure 1. Depending on σ, RENs may represent a deep neural network, while retaining a clear and compact definition.

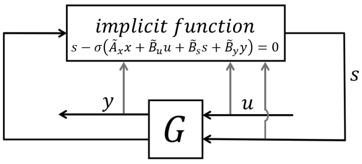

Figure 1: REN as a feedback interconnection of a linear system G and a nonlinear activation σ

The REN structure is a state-space representation expressed as:

$$x(t+1) = A_x x(t) + B_u u(t) + B_s s(t) + B_y y(t) \quad (1a)$$

$$s(t) = \sigma(\tilde{A}_x x(t) + \tilde{B}_u u(t) + \tilde{B}_s s(t) + \tilde{B}_y y(t)) \quad (1b)$$

$$y(t) = C x(t) + D_u u(t) + D_s s(t) \quad (1c)$$

where

- $x \in \mathbb{R}^{n_x}$ is the internal state vector of the network and it is characterized by matrices $A_x \in \mathbb{R}^{n_x \times n_x}$,$B_u \in \mathbb{R}^{n_x \times n_u}$, $B_s \in \mathbb{R}^{n_x \times n_s}$, and $B_y \in \mathbb{R}^{n_x \times n_y}$.
- $s \in \mathbb{R}^{n_s}$ is the solution to the equilibrium equation (1b), also known as "implicit network". The latter is characterized by matrices $\tilde{A}_x \in \mathbb{R}^{n_s \times n_x}$,$\tilde{B}_u \in \mathbb{R}^{n_s \times n_u}$, $\tilde{B}_s \in \mathbb{R}^{n_s \times n_s}$, $\tilde{B}_y \in \mathbb{R}^{n_s \times n_y}$, and by the activation function $\sigma = [\sigma_1(\cdot) \ \dots \ \sigma_{n_s}]^T$, that is a decentralized vector of scalar sigmoid functions.
- $u \in \mathbb{R}^{n_u}$ is the input vector.
- $y \in \mathbb{R}^{n_y}$ is the output vector.

The term "equilibrium" refers to the fact that any solution of the implicit equation (1b) defining $s(t)$ is an equilibrium point of the nonlinear internal dynamics, faster than the $x(t)$ dynamics, leading to a two-timescale interpretation. We define *Explicit RENs* [4] as the class of RENs having $\tilde{D}_s = 0$, that reduces the network (1b) to a single-layer neural network. These models are less flexible and representative of *Implicit RENs*, having $\tilde{D}_s \neq 0$, considered in this work. Depending on the value of $\tilde{D}_s$, the implicit Equation (1a) may or may not admit a unique solution $s(t)$ for given $x(t), u(t)$. A model (1) is said to be *well-posed* if it provides a unique solution of $s(t)$ for a given $x(t), u(t)$ (i.e., if it yields a unique response to any initial conditions and input). As stated in [5], the equilibrium network is well-posed if $\exists \Lambda > 0$ diagonal such that

$$2\Lambda - \Lambda\tilde{B}_s - \tilde{B}_s{}^T \Lambda > 0 \quad (2)$$

Note that, when $\widetilde{DB}_s$ and D_s have a triangular structure the model is defined as *acyclic RENs*, it is well-posed by definition and the elements of $s(t)$ can be explicitly computed row-by-row from (1b). The acyclic RENs are easier to implement and, in most cases, they provide models of excellent quality. The following assumption is also made on the function σ.

Assumption 2.1. Each component $\sigma_i(\cdot) : \mathbb{R} \to \mathbb{R}, i = 1, ..., n_s$ is a sigmoid function such that $\sigma_i(v_i(t)) \in [-1, 1], \forall v_i \in \mathbb{R}$. The activation function must be bounded, piecewise differentiable and slope-restricted in $[0, 1]$, i.e.,

$$0 \leq \frac{\sigma(b) - \sigma(a)}{b - a} \leq 1, \forall a, b \in \mathbb{R}, a \neq b \quad (3)$$

Also, $\sigma_i(\cdot)$ is Lipschitz continuous with unitary Lipschitz constant and one and only one inflection point in $v_i = 0$.

A key property used to derive useful Linear Matrix Inequalities (LMIs) is the incremental sector condition, provided by the following lemma [3].

Lemma 2.2. Under Assumption 2.1, $\forall h_i > 1, \exists \bar{v}_i(h_i) \geq 0$ such that $\forall (v_i, v_i + \Delta v_i) \in [-\bar{v}_i(h_i), \bar{v}_i(h_i)]^2$, it holds that

$$\Delta s_i(\Delta v_i - h_i \Delta s_i) \geq 0 \quad (4)$$

where $\Delta s_i = s_i(v_i + \Delta v_i) - s_i(v_i)$

2.2. Hyperparameter definition of REN models

The first objective of this work is to identify a suitable model from a set of candidates using data. In this thesis, to simplify the learning process of REN models, we adopt an identification approach, consisting of defining matrices A_x, $\tilde{A}_x$, B_u, $\tilde{B}_u$, B_y, $\tilde{B}_y$, B_s and $\tilde{B}_s$ as fixed random hyperparameters. Regarding A_x and $\tilde{A}_x$, we define them by, at the same time, imposing contractivity to the untrained model, which guarantees that trajectories of the system converge exponentially towards each other, independently of their initial conditions, thus ensuring the well-posedness (see Theorem 2.1 from [5]) of the identification process.

Under Assumption 2.1 described by (3), the following theorem gives the conditions for contracting RENs [5].

Theorem 2.1. *If there exists a matrix $P = P^T > 0$ and $\Lambda \in \mathbb{D}_+$, where $\mathbb{D}_+$ is the set of diagonal positive definite matrices, such that*

$$\begin{bmatrix} \bar{\alpha}^2 P & -\tilde{A}_x^T \Lambda \\ -\Lambda \tilde{A}_x & W \end{bmatrix} - \begin{bmatrix} A_x^T \\ B_s^T \end{bmatrix} P \begin{bmatrix} A_x^T \\ B_s^T \end{bmatrix}^T > 0 \quad (5)$$

where $W = 2\Lambda - \Lambda\tilde{B}_s - \tilde{B}_s^T \Lambda$, then the network is well-posed and contracting with some rate $\alpha < \bar{\alpha}$.

The maximum rate of convergence $\bar{\alpha}$, for instance, can be chosen as the spectral radius of the system (i.e., the maximum eigenvalue of the discrete-time linearized system), which can be inferred from the settling time of the system trajectories, e.g., in response to step-wise inputs.

Applying the Schur complement, the resulting LMI for the contractivity of the untrained model is the following:

$$\begin{bmatrix} \bar{\alpha}^2 P & -\tilde{A}_x^T \Lambda & A_x^T P \\ -\Lambda \tilde{A}_x & W & B_s^T P \\ P A_x & P B_s & P \end{bmatrix} > 0 \tag{6}$$

Depending on the problem, it is possible to influence the position of the eigenvalues of matrices A_x and $\tilde{A}_x$ by adding some constraints on the trace of the square matrix A_x and on the rectangular-trace[1] of the non-square matrix $\tilde{A}_x$.

2.3. Learning REN Models

After the definition of the model hyperparameters, the only free identifiable parameters are matrices C, D_u and D_s of (1c), defining $\theta = [C \quad D_u \quad D_s]$. The identification of θ is performed as follows:

1. Letting N_s be the number of the available data samples, the data $u(t)$ and $y(t)$ for $t = 1, ..., N_s$ are used to feed equations (1a, 1b), from a random initial condition $x(1)$. At this point all values of internal states $x(t)$ and nonlinearities $s(t)$ are available for $t = 2, ..., N_s$.
2. The goal is to select the value of θ that minimizes the quadratic error between the data and the predicted output:

$$J(\theta) = \sum_{t=1}^{N_s} \|y(t) - \hat{y}(t)\|^2 = \sum_{t=1}^{N_s} \left\| y(t) - \theta \begin{bmatrix} x(t) \\ u(t) \\ s(t) \end{bmatrix} \right\|^2 \tag{7}$$

Note that the optimization problem is quadratic and reduces to a Least-Squares formulation, avoiding the vanishing and exploding gradient issues.

Note that, the identification procedure proposed in this section adopts a hybrid approach that combines the Prediction Error Method (PEM), providing accurate one-step-ahead predictions but it may fail to reproduce the long-term dynamics of the system, and the Simulation Error Method (SEM), reproducing the overall system behavior through multi-step simulations, but may provide incorrect internal parametrization. Consequently, the proposed method exploits the short-term accuracy of PEM and the global consistency of SEM, capturing both the local and long-term dynamics of the system. After the learning procedure is completed, the learned REN model is:

$$x(t+1) = Ax(t) + Bu(t) + B_s^0 s(t) \tag{8a}$$

$$s(t) = \sigma\big(\tilde{A}x(t) + \tilde{B}u(t) + \tilde{B}_s^0 s(t)\big) \tag{8b}$$

$$y(t) = Cx(t) + D_u u(t) + D_s s(t) \tag{8c}$$

where $A = A_x + B_y C$, $B = B_u + B_y D$, $B_s^0 = B_s + B_y D_s$, $\tilde{A} = \tilde{A}_x + \tilde{B}_y C$, $\tilde{B} = \tilde{B}_u + \tilde{B}_y D$ and $\tilde{B}_s^0 = \tilde{B}_s + \tilde{B}_y D_s$.

Without structural constraints in the definition of hyperparameters and free parameters, the learned model may not retain the subsystem interconnections required for decentralized and distributed controller design. In particular, although the model may be accurate and dynamically consistent, modularity and scalability that characterize network-based structures may be absent. However, additional constraints incorporated into the optimization framework can enforce desired physical properties of the modeled system. Consequently, the overall identification procedure may adopt a physics-inspired perspective, enhancing the interpretability and physical consistency of the resulting model. In the following section we discuss how physical and structural information can be enforced on the learning process of RENs.

2.4. Structured Matrix Decomposition

The core idea of this approach is to perform the modelling approach described in Sections 2.2 and 2.3, while imposing a block-structured pattern on all matrices to reflect subsystem interconnections.

It is possible to define a physical graph $\mathscr{G}^0$, whose nodes are the subsystems, and whose edges represent the interconnections. More specifically, an edge (i, j) between $\mathscr{S}_j$ and $\mathscr{S}_i$ is present if some variables (e.g., the input u_j, the measured output y_j, or some elements of the internal non-measured state x_j) of $\mathscr{S}_j$ have a direct known physical influence on the dynamics of $\mathscr{S}_i$. As an alternative representation, we can define, for each $i = 1, ..., N$, the set $\mathscr{N}_i^0$ as the set of indices j such that (i, j) is an edge of the graph $\mathscr{G}^0$ and which does not include i.

In this way, the influence of each subsystem on its neighbors is naturally represented within the model, without the need to explicitly introduce coupling variables, making this approach applicable even in case the interconnection variables are not measurable. In particular, the untrained REN model of the i-th subsystem assumes the following form:

$$x_i(t+1) = A_{x_i} x_i(t) + B_{u_i} u_i(t) + B_{s_i} s_i(t) + B_{y_i} y_i(t) \tag{9a}$$

$$s_i(t) = \sigma\big(\tilde{A}_{x_i} x_i(t) + \tilde{B}_{u_i} u_i(t) + \tilde{B}_{s_i} s_i(t) + \tilde{B}_{y_i} y_i(t)\big) \tag{9b}$$

$$\begin{aligned} y_i(t) &= C_i x_i(t) + D_{u_i} u_i(t) + D_{s_i} s_i(t) + \\ &+ \sum_{j \in \mathscr{N}_i^0} \big(C_{ij} x_j(t) + D_{u_{ij}} u_j(t)\big) \end{aligned} \tag{9c}$$

The overall state dimension of the identified model is obtained by aggregating the states of individual subsystems. Overall, consider N subsystems, the hyperparameters are defined as follows:

1. Define n_x (dimension of internal states vector $x(t)$), n_s (dimension of non-linearities vector $s(t)$), n_{x_i} and n_{s_i}, for $i = 1, ..., N$, representing

---

[1] **Rectangular-trace**: For a rectangular matrix $M \in \mathbb{R}^{n \times m}$, we define the generalized trace as

$$\text{tr}_r(M) = \sum_{i=1}^{\min(m,n)} M_{ii},$$

i.e., the sum of the diagonal entries up to the smallest dimension.

the dimensions of the blocks of the structured matrices, introduced in the following points.

2. $B_u, \tilde{B}_u, B_y, \tilde{B}_y, B_s$ and $\tilde{B}_s$ are defined as block diagonal matrices, composed of blocks $B_{u_i} \in \mathbb{R}^{n_{x_i} \times n_{u_i}}$, $\tilde{B}_{u_i} \in \mathbb{R}^{n_{s_i} \times n_{u_i}}$, $B_{y_i} \in \mathbb{R}^{n_{x_i} \times n_{y_i}}$, $\tilde{B}_{u_i} \in \mathbb{R}^{n_{s_i} \times n_{y_i}}$. $B_{s_i} \in \mathbb{R}^{n_{x_i} \times n_{s_i}}$ has sparse random blocks and $\tilde{B}_{s_i} \in \mathbb{R}^{n_{s_i} \times n_{s_i}}$ has random lower triangular blocks.
3. A_x and $\tilde{A}_x$ as block diagonal matrices, with blocks $A_{x_i} \in \mathbb{R}^{n_{x_i} \times n_{x_i}}$ and $\tilde{A}_{x_i} \in \mathbb{R}^{n_{s_i} \times n_{x_i}}$, respectively, generated randomly, but subjected to the contractivity constraint (5).

A given structure is imposed also on the free parameters C, D and D_s, so as to reflect the interconnections represented by the graph $\mathscr{G}^0$. All matrices must have suitable dimensions and must be coherent with each other and with overall system's interconnections. The identification of C, D and D_s is carried out by minimizing the cost function given in equation (7).

Considering the trained model equations derived by using (9c) into (9a) and (9b), the structure of all trained matrices is dictated by that of C, D, and D_s. This is a crucial result, as the tunable matrices C, D, and D_s encode the interconnection relationships within the system. Importantly, the admissible interaction structure is informed by the underlying physics of the system, while the level of interconnections is identified directly from data. As a consequence, the trained model inherits a structured representation that is both physics-consistent and data-driven, with matrices that preserve the same interconnection pattern as $\mathscr{G}^0$, ensuring a faithful representation of the system's physical and structural properties.

In this procedure, the validation step is performed using the model obtained by plugging (9c) into (9a) and (9b). The generated data are then compared with the validation dataset, and the resulting model is the one, among the candidate ones, that minimizes the cost function.

After training, we define matrices $A_i = A_{x_i} + B_{y_i} C_i, B_i = B_{u_i} + B_{y_i} D_{u_i}, B^0_{s_i} = B_{s_i} + B_{y_i} D_{u_i}, A_{ij} = B_{y_i} C_i, B_{ij} = B_{y_i} D_{u_{ij}}, \tilde{A}_i = \tilde{A}_{x_i} + \tilde{B}_{y_i} C_i, \tilde{B}_i = \tilde{B}_{u_i} + \tilde{B}_{y_i} D_{u_i}, \tilde{B}^0_{s_i} = \tilde{B}_{s_i} + \tilde{B}_{y_i} D_{u_i}, \tilde{A}_{ij} = \tilde{B}_{y_i} C_i, \tilde{B}_{ij} = \tilde{B}_{y_i} D_{u_{ij}}$.

In this way, the learned i-th submodel takes the form

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t) + B^0_{s_i} s_i(t) + \sum_{j \in \mathcal{N}^0_i} \left(A_{ij} x_j(t) + B_{ij} u_j(t) \right) \quad (10a)$$

$$s_i(t) = \sigma\Big(\tilde{A}_i x_i(t) + \tilde{B}_i u_i(t) + \tilde{B}^0_{s_i} s_i(t) + \sum_{j \in \mathcal{N}^0_i} \left(\tilde{A}_{ij} x_j(t) + \tilde{B}_{ij} u_j(t) \right) \Big) \quad (10b)$$

$$y_i(t) = C_i x_i(t) + D_i u_i(t) + D_{s_i} s_i(t) + \sum_{j \in \mathcal{N}^0_i} \left(C_{ij} x_j(t) + D_{ij} u_j(t) \right) \quad (10c)$$

3. Decentralized NMPC of Structured REN Models

Decentralized and distributed architectures represent alternatives developed to overcome centralized control limitations: they allow local controllers to operate using limited information, while still guaranteeing the control objective satisfaction. The objective of this section is to formulate the decentralized constrained tracking problem. For this purpose, we consider a system modeled by a Recurrent Equilibrium Network, described in Section 2 and by equations (1). Robust NMPC will be leveraged to decentralized and distributed controller design [2].

3.1. Problem statement

To formulate the decentralized NMPC problem, the system model must be decomposed into N interconnected subsystems as done in Section 2.4.

First of all, for tracking objectives, we define setpoints $\bar{y}_i$, $i = 1, ..., N$ and the corresponding steady-state inputs $\bar{u}_i$, states $\bar{x}_i$, and nonlinearities $\bar{s}_i$, i.e. such that, for $i = 1, ..., N$:

$$\begin{cases} \bar{x}_i = A_{x_i} \bar{x}_i + B_{u_i} \bar{u}_i + B_{s_i} \bar{s}_i + B_{y_i} \bar{y}_i \\ \bar{s}_i = \sigma\big(\tilde{A}_{x_i} \bar{x}_i + \tilde{B}_{u_i} \bar{u}_i + \tilde{B}_{s_i} \bar{s}_i + \tilde{B}_{y_i} \bar{y}_i \big) \\ \bar{y}_i = C_i \bar{x}_i + D_{u_i} \bar{u}_i + D_{s_i} \bar{s}_i + \sum_{j \in \mathcal{N}^0_i} \big(C_{ij} \bar{x}_j + D_{u_{ij}} \bar{u}_j \big) \end{cases}$$

The setpoints are assumed to be available to all local controllers, as they represent the desired steady-state operating condition.

We can rewrite (10a) and (10b) as

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t) + B^0_{s_i} s_i(t) + w_i(t) + \sum_{j \in \mathcal{N}^0_i} \left(A_{ij} \bar{x}_j + B_{ij} \bar{u}_j \right) \quad (11a)$$

$$s_i(t) = \sigma\Big(\tilde{A}_i x_i(t) + \tilde{B}_i u_i(t) + \tilde{B}^0_{s_i} s_i(t) + \tilde{w}_i(t) + \sum_{j \in \mathcal{N}^0_i} \left(\tilde{A}_{ij} \bar{x}_j + \tilde{B}_{ij} \bar{u}_j \right) \Big) \quad (11b)$$

where

$$w_i(t) = \sum_{j \in \mathcal{N}^0_i} \left(A_{ij} (x_j(t) - \bar{x}_j) + B_{ij} (u_j(t) - \bar{u}_j) \right) \quad (12a)$$

$$\tilde{w}_i(t) = \sum_{j \in \mathcal{N}^0_i} \left(\tilde{A}_{ij} (x_j(t) - \bar{x}_j) + \tilde{B}_{ij} (u_j(t) - \bar{u}_j) \right) \quad (12b)$$

Each subsystem can be represented by a local model (11a), (11b), where the equivalent perturbation (12a), (12b) is additive. In this way, the contributions provided by the deviation of the interconnection variables with respect to their steady-state values on the model can be regarded as disturbances to be properly attenuated by the dedicated tube-based controller. Before introducing the nominal model definition, used for the controller design, it is necessary to specify some assumptions.

Assumption 3.1. The state and input vectors are constrained, i.e., $x(t) \in \mathbb{X}$ and $u(t) \in \mathbb{U}$. Sets $\mathbb{X}$ and $\mathbb{U}$ must be compact and convex.
In particular, for all $j = 1, ..., N$, we have $x_j(t) \in \mathbb{X}_j$ and $u_j(t) \in \mathbb{U}_j$, where

$$\mathbb{X}_j := \{x_j \in \mathbb{R}^{n_{x_j}} : (x_j(t) - \bar{x}_j)^T Q^0_{x_j}(x_j(t) - \bar{x}_j) \leq 1\} \tag{13}$$

$$\mathbb{U}_j := \{u_j \in \mathbb{R}^{n_{u_j}} : (u_j(t) - \bar{u}_j)^T Q^0_{u_j}(u_j(t) - \bar{u}_j) \leq 1\} \tag{14}$$

and where $Q^0_{x_j}$ and $Q^0_{u_j}$ are symmetric and positive definite matrices of suitable dimensions.

3.2. Nominal model and tube-based definition

The first step for the application of tube-based NMPC to (11a), (11b) is to define the nominal model, corresponding to the original one, but discarding the perturbation terms w_i and $\tilde{w}_i$ defined in (12a) and (12b), respectively. Note, in passing, that, in view of (13), (14), and (12a), (12b), for all $i = 1, ..., N$, the disturbances w_i and $\tilde{w}_i$ are bounded to lie in bounded sets $\mathbb{W}_i$ and $\tilde{\mathbb{W}}_i$. The latter sets need to be defined in such a way that, for all $i = 1, ..., N$, for all $u_j \in \mathbb{U}_j$ and $x_j \in \mathbb{X}_j$, $j \in \mathcal{N}^0_i$:

$$\sum_{j \in \mathcal{N}^0_i} \big(A_{ij}(x_j(t) - \bar{x}_j) + B_{ij}(u_j(t) - \bar{u}_j)\big) \in \mathbb{W}_i \tag{15a}$$

$$\sum_{j \in \mathcal{N}^0_i} \big(\tilde{A}_{ij}(x_j(t) - \bar{x}_j) + \tilde{B}_{ij}(u_j(t) - \bar{u}_j)\big) \in \tilde{\mathbb{W}}_i \tag{15b}$$

The nominal model is the following.

$$\begin{aligned} \hat{x}_i(t+1) = A_i\hat{x}_i(t) + B_i\hat{u}_i(t) + B^0_{s_i}\hat{s}_i(t) + \\ + \sum_{j \in \mathcal{N}^0_i} \big(A_{ij}\bar{x}_j + B_{ij}\bar{u}_j\big) \end{aligned} \tag{16a}$$

$$\begin{aligned} \hat{s}_i(t) = \sigma\Big(\tilde{A}_i\hat{x}_i(t) + \tilde{B}_i\hat{u}_i(t) + \tilde{B}^0_{s_i}\hat{s}_i(t) + \\ + \sum_{j \in \mathcal{N}^0_i} \big(\tilde{A}_{ij}\bar{x}_j + \tilde{B}_{ij}\bar{u}_j\big)\Big) \end{aligned} \tag{16b}$$

The input u_i is defined as follows:

$$u_i(t) = \hat{u}_i(t) + K^i_x\big(x_i(t) - \hat{x}_i(t)\big) + K^i_s\big(s_i(t) - \hat{s}_i(t)\big) \tag{17}$$

Note that the adoption of two control gains K^i_x and K^i_s provides flexibility and improves corrective action in the controller design.
From (11a, 11b), (16a, 16b), and (17),

$$\begin{aligned} x_i(t+1) - \hat{x}_i(t+1) = (A_i + B_iK^i_x)(x_i(t) - \hat{x}_i(t)) + \\ + (B^0_{s_i} + B_iK^i_s)(s_i(t) - \hat{s}_i(t)) + w_i(t) \end{aligned} \tag{18}$$

Through proper choices of K^i_x and K^i_s, and under Assumption 3.1, we can define the RPI set $\delta\mathbb{X}_i$:

Definition 3.2. For the system (11a)–(11b) subject to a bounded disturbance $w_i(t) \in \mathbb{W}_i$, a set $\delta\mathbb{X}_i \subseteq \mathbb{R}^{n_i}$ is said to be *Robust Positively Invariant (RPI)* if, for all $t \geq 0$,

$$\delta x_i(t) \in \delta\mathbb{X}_i \Rightarrow \delta x_i(t+1) \in \delta\mathbb{X}_i, \quad \forall w_i(t) \in \mathbb{W}_i.$$

Similarly, we will define, for all $i = 1, ..., N$ sets $\delta\mathbb{S}_i$ and $\delta\mathbb{U}_i$, such that

$$u_i(t) - \hat{u}_i(t) \in \delta\mathbb{U}_i, \quad s_i(t) - \hat{s}_i(t) \in \delta\mathbb{S}_i \tag{19}$$

At time t, the MPC optimal control problem addresses the nominal system variables. It is possible to define tightened constraints, for all subsystems, to ensure the satisfaction of original constraints, even in the worst case effect of bounded disturbances $w(t)$. The tightened constraints are defined as

$$\hat{x}_i(t) \in \hat{\mathbb{X}}_i, \quad \hat{u}_i(t) \in \hat{\mathbb{U}}_i, \quad \forall i = 1, ..., N, \forall t \geq 0$$

The sets $\hat{\mathbb{X}}_i$ and $\hat{\mathbb{U}}_i$ will be obtained by properly tightening the original sets $\mathbb{X}_i$ and $\mathbb{U}_i$, using Pontryagin difference, as follows:

$$\hat{\mathbb{X}}_i \subseteq \mathbb{X}_i \ominus \delta\mathbb{X}_i, \quad \hat{\mathbb{U}}_i \subseteq \mathbb{U}_i \ominus (K^i_x\delta\mathbb{X}_i \oplus K^i_s\delta\mathbb{S}_i)$$

3.3. Online optimal control problem

As discussed, the decentralized MPC algorithm reduces to solving N separated robust MPC problems. For each subsystem, a FHOCP to be solved at each time step t, consists in finding the optimization variables coinciding with the input sequence $\hat{u}_i(t), ..., \hat{u}_i(t + N_h - 1)$ over a prediction horizon of length N_h and nominal state $\hat{x}_i(t)$.
Under the auxiliary control law, the FHOCP for the i-th subsystem is formulated as follows

$$\min_{\hat{x}_i(t), \hat{u}_i(t),\ \dots\ , \hat{u}_i(t+N_h-1)} J_i \tag{20}$$

subject to:

$$x_i(t) - \hat{x}_i(t) \in \delta\mathbb{X}_i, \tag{21}$$

$$\forall k = 0, \dots, N_h - 1 :$$

$$\begin{aligned} \hat{x}_i(t+k+1) = (A_i + B_iK^i_x)\hat{x}_i(t+k) + \\ + (B_{s_i}0 + B_iK^i_s)\hat{s}_i(t+k) + \hat{d}_i(t+k), \end{aligned} \tag{22}$$

$$\hat{u}_i(t+k) \in \hat{\mathbb{U}}_i, \tag{23}$$

$$\hat{x}_i(t+k) \in \hat{\mathbb{X}}_i, \tag{24}$$

$$\hat{x}_i(t+N_h) \in \mathbb{X}_{f_i}. \tag{25}$$

where $\hat{\mathbb{X}}_i$ and $\hat{\mathbb{U}}_i$ are obtained by properly tightening the original sets, as previously specified.
The i-th controller minimizes the following cost function:

$$\begin{aligned} J_i = \sum_{k=0}^{N_h-1} \Big(||\hat{x}_i(t+k) - \bar{x}_i||^2_{T_i} + ||\hat{u}_i(t+k) - \bar{u}_i||^2_{R_i}\Big) + \\ + V_{f_i}(\hat{x}_i(t+N_h) - \bar{x}_i) \end{aligned} \tag{26}$$

where the terms $T_i \in \mathbb{R}^{n_{x_i} \times n_{x_i}}$ and $R_i \in \mathbb{R}^{n_{u_i} \times n_{u_i}}$ must be properly chosen to ensure an appropriate balance between tracking performance and control effort, and the terminal cost V_{f_i} is commonly defined based on the system's Lyapunov function, under an appropriate stabilizing auxiliary control law.
The solution to the FHOCP is iteratively evaluated, according to receding horizon procedure, and at time t it is represented by $\hat{x}_i(t|t), \hat{u}_i([t : N_h - 1]|t)$.

3.4. Terminal ingredients

The two fundamental properties of convergence and recursive feasibility are ensured by the design of the terminal ingredients for each subsystem i: the terminal set $\mathbb{X}_{f,i}$ and the terminal cost $V_{f,i}(\hat{x}_i(t+N)-\bar{x}_i)$.
For tracking purposes, the terminal ingredients are defined based on a local auxiliary control law of the type

$$\hat{u}_i(t) = K_x^i\big(\hat{x}_i(t) - \bar{x}_i\big) + K_s^i\big(\hat{s}_i(t) - \bar{s}_i\big) + \bar{u}_i \quad (27)$$

where K_x^i and K_s^i guarantee asymptotic stability of the equilibrium $\bar{x}_i$.
The terminal cost and set are chosen such that, under this control law, the decrease condition

$$\begin{aligned} V_{f_i}\big(\hat{x}_i(t+1) - \bar{x}_i\big) \le V_{f_i}\big(\hat{x}_i(t) - \bar{x}_i\big) - \\ - \|\hat{x}_i(t) - \bar{x}_i\|_{T_i}^2 - \|\hat{u}_i(t) - \bar{u}_i\|_{R_i}^2 \end{aligned} \quad (28)$$

holds, when the nominal system (16a), (16b) is controlled using (27), and $\mathbb{X}_{f_i}$ is positively invariant.
In the proposed robust decentralized MPC scheme, the terminal ingredients are constructed consistently with the tube-based formulation.
In particular, the terminal cost is defined as

$$V_{f_i} = \|\hat{x}_i(t) - \bar{x}_i(t)\|_{T_{f_i}}^2,$$

where T_{f_i} is obtained as a suitably scaled version of the matrix P_i, where P_i is the solution of the nominal Lyapunov equation associated with the local stabilizing feedback law (27).
Similarly, the terminal set $\mathbb{X}_{f_i}$ is chosen as a properly selected subset of the set $\delta\mathbb{X}_i$. More precisely, $\mathbb{X}_{f_i}$ is designed so as to be a robust positively invariant set for the closed-loop error dynamics under the local auxiliary controller and is chosen so as to be compatible with the tube-based constraint tightening.
In particular, it satisfies $\mathbb{X}_{f_i} \oplus \delta\mathbb{X}_i \subseteq \mathbb{X}_i$ so that, once the nominal state enters $\mathbb{X}_{f_i}$, the real state and input trajectories satisfy the original constraints for all admissible disturbances.
With this construction, recursive feasibility and convergence of each local closed-loop subsystem are guaranteed once the subsystem state enters $\mathbb{X}_{f_i}$, the auxiliary controller steers it to the reference while satisfying all constraints.

4. Conclusions

This work has addressed the problem of physics-inspired scalable modeling and control design for large-scale nonlinear systems: these two components are presented in an unified framework as two complementary parts of the same problem. The main outcome is the demonstration that physics-inspired structured neural models can be effectively used as control-oriented representations for large-scale systems, enabling the design of decentralized robust MPC schemes with guaranteed stability and constraint satisfaction. This points out the importance of integrating modeling and control objectives within a single framework, rather than treating them as separate and sequential steps. Some aspects of the proposed approach could be further investigated in future research. In this work, the LMI-based design procedure has been used to derive the local stabilizing feedback gains and to guarantee the required incremental stability and consistency properties. The next step is the integration of these results into a full receding-horizon optimization framework, in which the decentralized optimal control problem is solved in real time. This next implementation will assess the closed-loop performance of the proposed architecture and evaluate the computational advantages provided by the structured REN models. Another relevant direction concerns the validation of the proposed methodology on experimental plants and the integration of online learning strategies for the REN models, so as to handle slowly varying dynamics and modeling uncertainties.

References

[1] Marcello Farina and Riccardo Scattolini. Distributed mpc for large-scale systems. In *Handbook of Model Predictive Control*, pages 239–258. Springer, 2018.

[2] Daniele Ravasio, Marcello Farina, and Andrea Ballarino. Lmi-based design of a robust model predictive controller for a class of recurrent neural networks with guaranteed properties. *IEEE Control Systems Letters*, 8:1126–1131, 2024.

[3] Daniele Ravasio, Marcello Farina, Alessio La Bella, and Andrea Ballarino. Recurrent neural network-based robust control systems with closed-loop regional incremental iss and application to mpc design, 2025.

[4] Max Revay, Ruigang Wang, and Ian R Manchester. Recurrent equilibrium networks: Unconstrained learning of stable and robust dynamical models. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 2282–2287. IEEE, 2021.

[5] Max Revay, Ruigang Wang, and Ian R Manchester. Recurrent equilibrium networks: Flexible dynamic models with guaranteed stability and robustness. *IEEE Transactions on Automatic Control*, 69(5):2855–2870, 2023.