

A Non-Cooperative Distributed Model Predictive Control Using Laguerre Functions for Large-Scale Interconnected Systems



Kamel Menighed^{1*}, Joseph Julien Yamé², Issam Chekakta³

¹ Department of Petrochemicals, University of 20 août 1955, Skikda 21000, Algeria

² Université de Lorraine, CNRS, CRAN, F-54000, Nancy, France

³ DAC HR Laboratory, Ferhat ABBAS University, Sétif 1 19000, Algeria

Corresponding Author Email: k.menighed@univ-skikda.dz

<https://doi.org/10.18280/jesa.550501>

ABSTRACT

Received: 14 August 2022

Accepted: 10 October 2022

Keywords:

distributed model predictive control, discrete-time Laguerre functions, large-scale interconnected systems, non-cooperative strategy, optimization problem

This paper deals with a new non-cooperative distributed controller for linear large-scale systems based on designing multiple local Model Predictive Control (MPC) algorithms using Laguerre functions to enhance the global performance of the overall closed-loop system. In this distributed control scheme, that does not require a coordinator, local MPC algorithms might transmit and receive information from other sub-controllers by means of the communication network to perform their control decisions independently on each other. Thanks to the exchanged information, the sub-controllers have in this way the ability to work together in a collaborative manner towards achieving a good overall system performance. To decrease drastically the computational load in the small-size optimization problem with a short prediction horizon, discrete-time Laguerre functions are used to tightly approximate the optimal control sequence. For evaluating the proposed distribution control framework, a simulation example is proposed to show the effectiveness of the proposed scheme and its applicability for large-scale interconnected systems. The obtained simulation results are provided to demonstrate clearly that the proposed Non-Cooperative Distributed MPC (NC-DMPC) outperforms Decentralized MPC (De-MPC) and achieves performance comparable to centralized MPC with a reduced computing time. The system performance of the proposed distributed model predictive control is given.

1. INTRODUCTION

In the last few decades there has been an increasing importance in the formulation of control strategies for the class of large-scale and spatially distributed systems, which consist of finite number of (interconnected or independent) subsystems and possibly located at different sites. An archetype of large-scale systems are transportation systems such as power networks, water distribution networks or traffic [1, 2], chemical process networks [3-5] and power flow systems [6]. These systems, generally called large-scale interconnected systems, are composed of many subsystems, in such a way that a subsystem is significantly interacting with other subsystems through their control inputs and/or states. Indeed, due to the growing requirements in terms of global performance of the whole closed-loop system, the design of high performance controllers is often a nontrivial task. Therefore, several control strategies have been considered for these wide-plant systems to fulfil certain desired global performance.

Ideally, a centralized MPC control structure, as illustrated in Figure 1(a), can be able to provide a high global performance of the whole closed-loop system. Intuitively, a classical MPC strategy assumes that all available information regarding all the sub-systems are centralized. Indeed, MPC technique relies on a global dynamical model of the system to be controlled, whereby all subsystems' interactions are considered within the system, and it should be available for

control design. Moreover, all measurements of the whole system must be collected and gathered in a single controller (one location) to estimate all the states and compute all the sequences of future optimal control values to achieve a better performance. Unfortunately, when the number of input variables and states in the whole system becomes large, this centralized control scheme suffers from potential problems associated with heavy computational load of the centralized-single optimization problem. In addition, except the high risk of failure due to their centralized nature, using a single centralized agent can result in a high need of resources for computation and memory, which may further increase when the length of prediction horizon increases. Moreover, the amount of required resources also grows with the increasing of the system complexity. Accordingly, it demands exchange of vast amounts of information and use of high computing power. These drawbacks inherent to this control structure make the classical MPC approach often viewed by most engineers as inappropriate and impractical for control of large-scale interconnected systems. For this reason, it has progressively given way to non-centralized control strategies, including a decentralized (cf. Figure 1(b)) and distributed (cf. Figure 1(c)) control structures, for their notable decrease in system dimensionality using several local MPC controllers [7-11].

In the case of either decentralized or distributed MPC approach, the key idea is to divide the global optimization problem into many independent optimization sub-problems

also called local optimization problems. In other words, the global system model with large-size order is decomposed into several inter-acting subsystems with a small-size orders, where a local controller, also known as agent (cf. Figures 1(b) and 1(c)) is associated to each subsystem. Here, each independent controller is assigned to solve its own local optimization problem with local performance indices, relevant information and proper constraints. It should be stressed that there are some advantages of these control strategies, especially the benefit of being adaptable to the system structure, with less computational burden and no large amounts of information demands [12-16]. The important distinction between decentralized and distributed MPC architectures consists in the amount and type of information exchange by means of the communication network. In decentralized MPC strategy, the communication network is not used, then no shared information among the sub-controllers is possible, the sole type of information exchange is only between each subsystem and its sub-controller, i.e., local output measurements and control inputs. It is noteworthy that the design of decentralized control follows either an entirely decoupled system [17], or ignores the inter-subsystem interactions due to the weakly interacting dynamics [18] or models the interaction effects as unknown disturbances to be rejected, compensated using a robust MPC formulation [19]. Unfortunately, in the case where strong interactions would exist between the subsystems, the decentralization of the control leads generally to a poor performance of large-scale systems since no information about these interactions has been exchanged by means of the information network and taken into account by the agents in their control decisions. In contrast, the central idea of distributed control approach depends on the ability of exchanging some information between sub-controllers. For this architecture, each local controller obtains measurements from its subsystem and information from other sub-controllers. However, the type of the information exchange between agents, realized via inter-agents communication network, is the external control inputs and/or states from other subsystems. In addition, with shared information between the sub-controllers, the objective is to make them able to perform a certain degree of collaboration with each other with the aim of achieving the best global system performance.

Nowadays, the progress in communication network technologies and real-time distributed algorithms have allowed control methodologies to employ their potentials to handle more complex large-scale systems for dramatically enhancing the control performance. For this reason, the improvement of the global control performance of the closed-loop plant-wide systems using network information exchange has been recently a field of active research. In order to fulfil the overall objective for the whole system, communication between the agents over a communication network is needed. Thanks to the digital network, the required communication can be achieved via a shared information among the agents. From a control viewpoint, it is well known that MPC is able to handle hard constraints, multivariable-linear, nonlinear, uncertain or stochastic process [20, 21]. Moreover, this technique profits hugely from both advances in computational resources as well as advances in communication technology. Usually, in the context of distributed MPC design, different distributed MPC schemes have been proposed in the literature. Indeed, it allows for independent local controllers able to communicate and collaborate with other controllers to achieve

improved stability, robustness and global closed-loop performance in a distributed way.

From the literature review, various distributed MPC designs have been proposed in a number of works [9, 12, 22-24]. A study analysis of the control performance of distributed MPC has been discussed by Vaccarini et al. [23]. Adopting a distributed solution, the global computational burden can be reduced without significant deteriorations of the control performance and the fault tolerant issues can be addressed [22]. The authors propose a feasible-cooperation DMPC scheme, where the local MPC problem was resolved considering the effect of the local control actions on the performance of the remaining subsystems [25, 26]. The proposed distributed MPC can be classified into two approaches for collaboration between local controllers, namely a cooperation and non-cooperation MPC algorithms [24]. In cooperation-based DMPC algorithms, each local controller minimizes a global performance index [27]. However, when each local controller minimizes only a local performance index, it is called a non-cooperative-based DMPC algorithm [12, 28-30]. The control objective, in both approaches, is to enhance stability as well as optimality and makes the distributed control design very close to the centralized one in terms of control performance. The correlation between the complexity of non-centralized, i.e., decentralized and distributed, MPC schemes and their closed-loop performance have been analyzed [9]. The results conclude that, when the dynamical interaction effects between subsystems are weak, for the case of large-scale interconnected systems, a fully decentralized strategy can offer an acceptable control performance. In the other case, the distributed MPC scheme can improve the global performance of the whole closed-loop system, only if its communication implementation is feasible in practice. For the case of uncertain systems, a robust distributed MPC approach was proposed for a class of linear systems subject to structured time-varying uncertainties [31]. Moreover, for polytopic uncertain large-scale systems, a novel robust distributed model predictive control method has been presented by Shalmani et al. [32]. Viewing strong dynamic coupling effects as bounded disturbances between subsystems, the proposed algorithm can ensure some degree of stability robustness with respect to these disturbances. Since the works [7, 30], distributed MPC has become a field of active research, for detailed overviews on the current state of the art on this topic we refer to the papers [4, 33-36] and the references therein.

In this paper, we deal with unconstrained distributed MPC of large-scale interconnected systems to fulfill a global performance thanks to the use of a non-cooperative strategy between agents communicating through a digital network. The main contribution of the paper is an innovative solution for a distributed MPC framework relying on Laguerre functions with two key advantages consisting firstly of significantly reducing the computational load in the local receding optimization problems and secondly of allowing a small prediction horizon to closely approximate the predicted control trajectory. The rest of the paper is structured as follows. Mathematical models of large-scale systems are presented in section 2. In section 3, we state our problem to be resolved. Section 4 presents the proposed non-cooperation optimization-based distributed MPC (NC-DMPC). Section 5 summarizes some of the main properties of Laguerre functions to be used in section 6 to establish the algorithm used in the design of the proposed NC-DMPC scheme. An algorithm for non-cooperative distributed MPC is developed in section 7. A

numerical simulation example is presented in section 8 to illustrate the effectiveness of the proposed NC-DMPC algorithm with Laguerre functions incorporated. Finally, to sum up, some concluding remarks are drawn in section 9.

2. MATHEMATICAL MODELING OF LARGE-SCALE SYSTEMS

Let a discrete-time linear and time-invariant global system be given by

$$\mathcal{S} = \begin{cases} \mathbf{x}^{(k+1)} &= A\mathbf{x}^{(k)} + B\mathbf{u}^{(k)} \\ \mathbf{y}^{(k)} &= C\mathbf{x}^{(k)} \end{cases} \quad (1)$$

where, $\mathbf{x}^{(k)} \in \mathfrak{R}^{n_x}$ is the state of the system, $\mathbf{u}^{(k)} \in \mathfrak{R}^{n_u}$ is its input and $\mathbf{y}^{(k)} \in \mathfrak{R}^{n_y}$ is its output with $k > 0$ an integer index denoting discrete time. The whole system matrices are $A \in \mathfrak{R}^{n_x \times n_x}$, $B \in \mathfrak{R}^{n_x \times n_u}$ and $C \in \mathfrak{R}^{n_y \times n_x}$. Consider the decoupled subplants or decentralized models derived from (1),

$$\mathcal{S}_{ii} = \begin{cases} \mathbf{x}^{(k+1)} &= A_{ii}\mathbf{x}^{(k)} + B_{ii}\mathbf{u}^{(k)} \\ \mathbf{y}^{(k)} &= C_{ii}\mathbf{x}^{(k)} \quad i = \overline{1, \overline{N}} \end{cases} \quad (2)$$

where, vectors $\mathbf{x}^{(k)} \in \mathfrak{R}^{n_{x_i}}$, $\mathbf{u}^{(k)} \in \mathfrak{R}^{n_{u_i}}$ are the local states and inputs, respectively, of the i^{th} subsystem, while $\mathbf{y}^{(k)} \in \mathfrak{R}^{n_{y_i}}$ is its local output. The dimensions of these local vectors are such that

$$n_x = \sum_{i=1}^N n_{x_i}, n_u = \sum_{i=1}^N n_{u_i}, n_y = \sum_{i=1}^N n_{y_i}$$

and the global state vector can be written as $\mathbf{x} = \text{col}_{i \in \{1, \dots, N\}}(\mathbf{x}^{(i)})$, ($\xi = \text{col}_{i \in \{1, \dots, N\}}(\xi^{(i)})$): vector consisting of stacked subvectors $\xi^{(i)}$, the global input as $\mathbf{u} = \text{col}_{i \in \{1, \dots, N\}}(\mathbf{u}^{(i)})$, the global output as $\mathbf{y} = \text{col}_{i \in \{1, \dots, N\}}(\mathbf{y}^{(i)})$. Note that the interconnected system's elements are generally strongly coupled. Consequently, the decentralized control design leads to poor performance requirements or even instability because it has not the potential to take into account these interactions between the subsystems.

Throughout this manuscript, we suppose that the global system \mathcal{S} in (1) is composed of N subsystems \mathcal{S}_i . Therefore, any subsystem \mathcal{S}_i can be interacting with the other subsystems $\mathcal{S}_j, j \neq i$ through linear interconnections. Therefore, the local dynamics of any subsystem $\mathcal{S}_i, i = \overline{1, \overline{N}}$, is described as follows:

$$\mathcal{S}_i = \begin{cases} \mathbf{x}^{(k+1)} &= A_{ii}\mathbf{x}^{(k)} + B_{ii}\mathbf{u}^{(k)} + \sum_{j=1}^N [A_{ij}\mathbf{x}^{(j)} + B_{ij}\mathbf{u}^{(j)}] \\ \mathbf{y}^{(k)} &= C_{ii}\mathbf{x}^{(k)} + \sum_{j=1}^N C_{ij}\mathbf{x}^{(j)} \quad i = \overline{1, \overline{N}}, j \neq i \end{cases} \quad (3)$$

The interaction vectors are built upon state and input interactions produced on the subsystem \mathcal{S}_i by the other

subsystems $\mathcal{S}_j, j \neq i$ are described as follows:

$$\begin{aligned} \mathbf{v}^{(i)} &= \sum_{j=1}^N [A_{ij}\mathbf{x}^{(j)} + B_{ij}\mathbf{u}^{(j)}] \\ \mathbf{w}^{(i)} &= \sum_{j=1}^N C_{ij}\mathbf{x}^{(j)} \end{aligned} \quad (4)$$

According to this representation, the matrices A_{ij}, B_{ij} and C_{ij} , for $i \neq j$, define the dynamic coupling terms between the subsystems.

In general, not all the subsystems might have an influence on each other. To account for the influences between the subsystems, let us introduce the following definition

Definition 1 A subsystem \mathcal{S}_j interacts dynamically with the subsystem \mathcal{S}_i if and only if, at least, one of the matrices A_{ij}, B_{ij}, C_{ij} is non null, i.e., if and only if the states $\mathbf{x}^{(j)}$ and/or inputs $\mathbf{u}^{(j)}$ of \mathcal{S}_j affect the dynamics of \mathcal{S}_i .

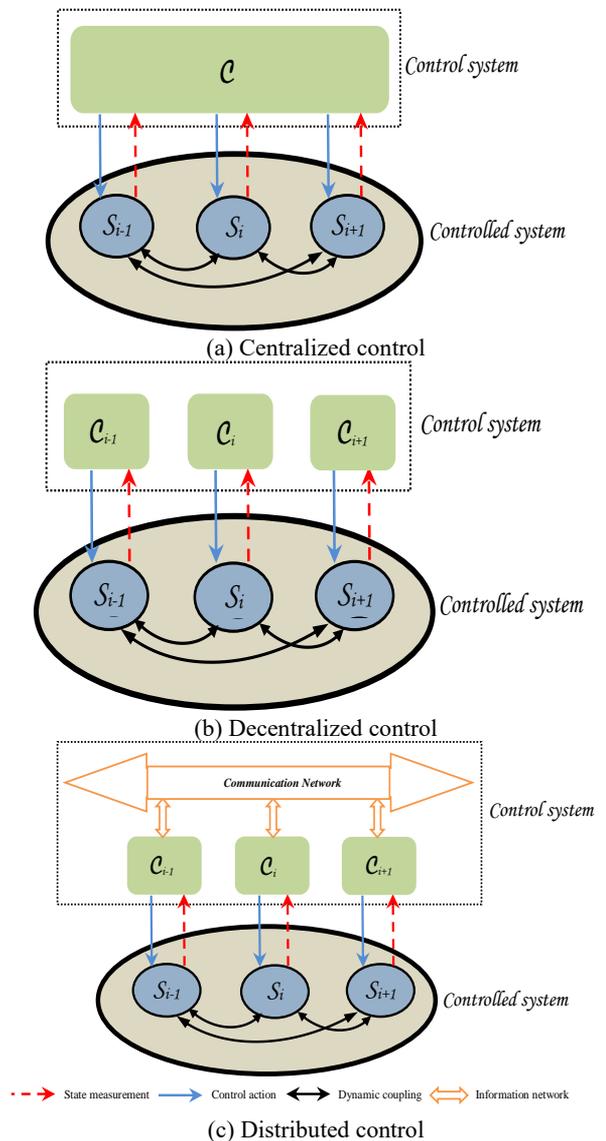


Figure 1. A schematic illustration of the principal control system architectures

The distributed control architecture dealt with throughout the paper has the generic structure given in Figure 1(c). This architecture consists of several local controllers, each dedicated to a subsystem of the overall plant, which can exchange information over a digital communication network. This architecture favors a non-cooperation based control strategy for large-scale interconnected systems (3)-(4) that aims to emulate the performance achievable with a centralized control scheme of Figure 1(a).

In a typical DMPC framework, \mathbb{N} subproblems are resolved, each one assigned to different sub-controllers \mathcal{C}_i , $i = \overline{1, \mathbb{N}}$, instead of a single centralized problem.

The following procedure achieved by the set of independent local controllers \mathcal{C}_i , at each time instant k , is given by: 1) acquire both local output measurements on \mathcal{S}_i and the received estimate of the interactions between \mathcal{S}_i and the other subsystems \mathcal{S}_j , $j = \overline{1, \mathbb{N}}$, $j \neq i$, transmitted through the communication network, to predicate the local state variable over the prediction horizon, 2) resolve the local optimization problem, 3) calculate the first control sample and apply it as a control input, 4) share both local optimal control sequence and future state prediction information with the other controllers through the communication network.

3. CONTROL PROBLEM FORMULATION

From (3) and (4), the l -step ahead state and output predictions at time instant k over a horizon of length N_p can be deduced easily, and are given as:

$$\hat{\mathbf{x}}_{(k+l|k)}^{[i]} = A_{ii}^l \hat{\mathbf{x}}_{(k|k)}^{[i]} + \sum_{s=1}^l [A_{ii}^{s-1} B_{ii} u_{(k+l-s|k)}^{[i]} + A_{ii}^{s-1} \hat{\mathbf{v}}_{(k+l-s|k-1)}^{[i]}] \quad (5a)$$

$$\hat{\mathbf{y}}_{(k+l|k)}^{[i]} = C_{ii} \hat{\mathbf{x}}_{(k+l|k)}^{[i]} + \hat{\mathbf{w}}_{(k+l-s|k-1)}^{[i]} \quad (5b)$$

for $l = \overline{1, N_p}$

where, $\hat{\mathbf{x}}_{(k+l|k)}^{[i]}$ ($\hat{\mathbf{y}}_{(k+l|k)}^{[i]}$) is the predicted state (output) variable at time instant $k+l$ having current plant information $\hat{\mathbf{x}}_{(k|k)}^{[i]}$ ($\hat{\mathbf{y}}_{(k|k)}^{[i]}$) at the current time instant k .

Suppose that the whole system \mathcal{S} , as illustrated in Figure 1(c), is composed of \mathbb{N} interacting subsystems \mathcal{S}_i , the unconstrained DMPC problem with N_p and N_u , respectively, the prediction and control horizons, $N_p \geq N_u$, lies in finding \mathbb{N} separate sub-controllers \mathcal{C}_i so that every \mathcal{C}_i solves, at each sampling instant k , an optimization problem with the following local quadratic performance index J_i , which penalizes output tracking errors and incremental controls $\Delta u^{[i]}$ [37].

$$J_i = \sum_{l=1}^{N_p} \|\hat{\mathbf{y}}_{(k+l|k)}^{[i]} - \mathbf{y}_{(k+l|k)}^{[i]sp}\|_{Q_i}^2 + \sum_{l=1}^{N_u} \|\Delta u_{(k+l-1|k)}^{[i]}\|_{R_i}^2 \quad (6)$$

s. t. the local dynamics model (5)

where, $\|\lambda\|_{\Psi}^2 = \lambda^T \Psi \lambda$, for a vector $\lambda \in \mathfrak{R}^n$ and a matrix $\Psi \in \mathfrak{R}^{n \times n}$.

$\mathbf{y}_{(k+l|k)}^{[i]sp}$ denotes the predicted values of the desired output at the future sampling instant $t+l$ known at time instant k .

$-\Delta u_{(k+l-1|k)}^{[i]}$ represents the future control increments (or control variations) signal at time instant k .

$-Q_i = Q_i^T \geq 0$ and $R_i = R_i^T > 0$ are real symmetric positive semi-definite and positive definite weight matrices, respectively.

$-N_p$ and $N_u \in \mathbf{N}^*$ are, respectively, the predictive and control horizons, used as tuning parameters.

Remark 1

i) The optimization problem is a quadratic programming problem and thus convex, thanks to the linearity of model (5), and the quadratic performance index J_i with R_i and Q_i respectively positive definite and semi-definite matrices.

ii) The design parameters of the optimization problem are the weighting matrices Q_i , R_i and the horizons N_p and N_u . These parameters might be tuned for achieving the desired closed-loop performance and the stability of the system under the unconstrained DMPC.

4. NON-COOPERATION BASED DISTRIBUTED MPC PROBLEM

In DMPC formulation, the whole-size optimization problem is replaced by \mathbb{N} small sub-problems working together towards achieving the best performance of the centralized control system. To obtain an explicit solution of the work on the nominal distributed optimization problem presented by Vaccarini et al. [12, 13], each sub-controller \mathcal{C}_i is composed of three parts: *an optimizer, a state predictor and an interaction predictor* (see Figure 3). The workflow of each sub-controller will be described later in algorithmic form in section 7.

For the sake of simplicity, the availability of the local states $\hat{\mathbf{x}}_{(k)}^{[i]}$ through measurements is assumed. Furthermore, denoting N_{pi} and N_{ui} the prediction and control horizons for the i^{th} sub-controller \mathcal{C}_i , the subsequent assumptions are adopted in the sequel:

Assumptions 1

a. the prediction and control horizons are equal, i.e., $N_p = N_{pi} = N_{pj}$ and $N_u = N_{ui} = N_{uj}$, $\forall j, i = \overline{1, \mathbb{N}}, i \neq j$;

b. the sub-controllers \mathcal{C}_i , for $i = \overline{1, \mathbb{N}}$, act in a synchronous way;

c. the information is transmitted (and received) by the local sub-controllers only once within each sampling time interval (non-iterative algorithm);

d. the digital network introduces one step communication delay;

e. the information exchange is not loosed during the transmission;

f. all the pairs (A_{ii}, B_{ii}) are stabilizable.

At each time instant k , neither $\mathbf{x}_{(k)}^{[j]}$ nor $\mathbf{u}_{(k)}^{[j]}$ for $j \neq i$ are available by the agent \mathcal{C}_i ; only their predictions, previously sent by the agents \mathcal{C}_j , are known. In fact, due to the unit delay introduced by the communication network (cf. Assumption 1-(d) above), the information of the other subsystems is available only after one sampling time interval. Therefore, the interaction predictions $\hat{\mathbf{v}}_{(k+l-s|k)}^{[i]}$ and $\hat{\mathbf{w}}_{(k+l-s|k)}^{[i]}$ are unavailable at current time k for the subsystem \mathcal{S}_i . Then, their values are available after one sampling time interval with respect to the instant they are broadcasted in. For this reason,

the predictions $\hat{v}_{(k+l-s|k-1)}^{\{i\}}$ and $\hat{w}_{(k+l-s|k-1)}^{\{i\}}$ will be used instead of $\hat{v}_{(k+l-s|k)}^{\{i\}}$ and $\hat{w}_{(k+l-s|k)}^{\{i\}}$ in the subsequent problem definition.

For mathematical notation simplicity, the following notations are used in this paper:

- $\text{diag}_\alpha[\Lambda] = \text{block-diag}[\underbrace{\Lambda \ \Lambda \ \dots \ \Lambda}_{\alpha \text{ times}}]$;
- $0_{\alpha \times \beta}$ ($I_{\alpha \times \beta}$) is the null (identity) matrix in $\mathfrak{R}^{\alpha \times \beta}$;
- 0_α (I_α) is the null (identity) matrix in $\mathfrak{R}^{\alpha \times \alpha}$;
- 0 is the null matrix or vector with appropriate dimensions.

4.1 State predictor

The l -step ahead predicted state variable of the i^{th} subsystem, at each time instant k , is given by:

$$\hat{x}_{(k+l|k)}^{\{i\}} = A_{ii}^l \hat{x}_{(k|k)}^{\{i\}} + \sum_{s=1}^l [A_{ii}^{s-1} B_{ii} u_{(k+l-s|k)}^{\{i\}} + A_{ii}^{s-1} \hat{v}_{(k+l-s|k-1)}^{\{i\}}] \quad (7)$$

where, $\hat{v}_{(k+l-s|k-1)}^{\{i\}}$ denotes the prediction of $v_{(k+l-s)}^{\{i\}}$ computed at the past time instant $k-1$.

4.2 Interaction prediction

The interaction predictions of the i^{th} subsystem, at each time instant k , are expressed as follows :

$$\begin{aligned} \hat{\mathcal{V}}_{(k, N_p|k-1)}^{\{i\}} &= \tilde{A}_i \tilde{X}_{(k, N_p|k-1)} + \tilde{B}_i \tilde{\Gamma}_i U_{(k-1, N_u|k-1)} \\ \hat{\mathcal{W}}_{(k, N_p|k-1)}^{\{i\}} &= \tilde{C}_i \tilde{X}_{(k, N_p|k-1)} \end{aligned} \quad (8)$$

Now, let us assume that:

$$\tilde{K}_i = [\text{diag}_p(K_{i1}) \dots \text{diag}_p(K_{i(i-1)}) \ 0 \ \text{diag}_p(K_{i(i+1)}) \dots \text{diag}_p(K_{iN})]$$

with $\tilde{K}_i \in \{\tilde{C}_i, \tilde{B}_i, \tilde{A}_i\}$

4.3 Optimal control sequence of each independent sub-controller \mathcal{C}_i

At each sampling time instant k , when a set of the estimation vectors $\hat{X}^{\{j\}}$ and $U^{\{j\}}$ are received from the other sub-controllers \mathcal{C}_j , $j = \overline{1, N}$, $j \neq i$ through the communication network (cf. Assumption 1-(d) above), the interaction predictor of each independent \mathcal{C}_i uses these information to estimate the interaction predictions. Then, these predictions are gathered with the local state value $x_{(k)}^{\{i\}}$ and all these information to be used by the optimizer in order to find a solution of the local optimization problem. Once the sequence of future control values $\Delta U^{*i} = \{\Delta u_{(k|k)}^{*i}, \dots, \Delta u_{(k+N_u-1|k)}^{*i}\}$ is computed by solving the finite-horizon optimal control problem (6), only the first sample of the computed optimal control sequence is retained and $u_{(k|k)}^{*i} = \Delta u_{(k|k)}^{*i} + u_{(k-1)}^{*i}$ is injected as the control action to the subsystem \mathcal{S}_i , while neglecting the rest of the elements constituting the computed optimal sequence. At the same time, by means (7), the state

predictor estimates the future state variable. Then, the optimal control sequence together with the state predictions over the prediction horizon N_p are broadcasted to the other sub-controllers \mathcal{C}_j , $j \neq i$, through the communication network. At the next time instant, $k+1$, the overall procedure is repeated, for each local controller, according to the so-called *receding horizon principle*.

Usually, the horizons N_p and N_u are two important tuning parameters of the MPC control design. However, the computational burden in MPC is directly depending upon them. Among the MPC formulations, the well known one is the classical scheme [38-40]. In this technique, for the possibility of high rapid sampling frequency, complex plant dynamics and/or high requirement on the global system performance, satisfactory approximation of the control increment $\Delta u_{(k)}^{\{i\}}$ could involve a vast number of parameters (especially N_u), resulting in poorly numerically conditioned solutions with a weighty computational burden [41]. Otherwise, a more suitable approach is to use Laguerre functions [42, 43] in the design of MPC.

5. PARAMETERIZATION OF THE CONTROL SIGNAL TRAJECTORY

A method of designing an MPC using orthonormal functions was proposed with the main advantage of reducing the number of tuning parameters used for the description of the control signal trajectory. This makes fewer computations compared to the traditional MPC approach [44-46]. The change in the control trajectory was achieved through the adjustment of the scaling factor incorporated in the orthonormal function.

In traditional MPC approach, at each time instant k , the state variable vector $x_{(k)}$ provides the current plant information obtained through measurements. In the case of a single input system, the optimal control sequence is then defined as follows

$$\Delta U_{(k, N_u|k)}^* = [\Delta u_{(k|k)}^*, \dots, \Delta u_{(k+N_u-2|k)}^*, \Delta u_{(k+N_u-1|k)}^*]^T$$

The control horizon N_u indicates the number of parameters used to capture the optimal control sequence. Having $x_{(k)}$, the future state variables are predicted for N_p future samples, N_p being the prediction horizon, it represents also the length of the optimization window. The future predicted state variables are then defined by :

$$\hat{X}_{(k+1, N_p|k)} = [\hat{x}_{(k+1|k)}^T, \dots, \hat{x}_{(k+N_u|k)}^T, \dots, \hat{x}_{(k+N_p|k)}^T]^T$$

According to the receding horizon strategy, the MPC technique takes only the first element of the computed optimal control sequence and apply it. Additionally, in the next sample time period, the new measurements are used to formulate the state vector for the computation of the new optimal control sequence.

Laguerre functions can be used to approximate the incremental terms contained in ΔU . The Z-transfer representation of the ℓ^{th} Laguerre function is given by:

$$\Gamma_{m(z)} = \sqrt{1-p^2} \frac{(z^{-1}-p)^{m-1}}{(1-pz^{-1})^m}, p \in [0 \ 1] \quad (9)$$

for $m = \overline{1, N}$

where, \mathcal{N} is the number of terms used in capturing the control signal and p is the pole location of the discrete-time Laguerre network, it is also called the scaling factor. The free parameter p is needed to be tuned by the designer to guarantee the stability of the network. It is worth to emphasize that the choice of the parameter p is very important to ensure the convergence rate of the Laguerre functions. Laguerre functions become a set of pulse operators when $p = 0$, which makes their using for MPC design equivalent to conventional MPC [43].

From (9), the structure of the discrete-time Laguerre network is illustrated in Figure 2.

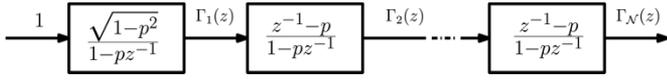


Figure 2. Illustration through a bloc-diagram representation of a discrete Laguerre network

The set of discrete Laguerre functions is defined for some $0 \leq p < 1$, by taking the inverse Z-transform of Eq. (9), that is:

$$\ell_{m(k)} = Z^{-1}\{\Gamma_{m(z)}\}$$

This set of Laguerre functions, at time instant k , can be expressed in a vector form as follows :

$$\mathcal{L}_{(k)} = [\ell_{1(k)} \ell_{2(k)} \dots \ell_{\mathcal{N}(k)}]^T \quad (10)$$

Finally, based on the relationship (9), the set of discrete Laguerre functions can be computed using the following state-space model:

$$\mathcal{L}_{(k+1)} = \mathcal{A}_l \mathcal{L}_{(k)} \quad (11)$$

where, the lower triangular matrix $\mathcal{A}_l(p, \mu)$ and the initial state $\mathcal{L}_{(0)}$ are given by:

$$\mathcal{A}_l(p, \mu) = \begin{bmatrix} p & 0 & 0 & 0 & \dots & 0 \\ \mu & p & 0 & 0 & \dots & 0 \\ -p\mu & \mu & p & 0 & \dots & 0 \\ p^2\mu & -p\mu & \mu & p & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ (-p)^{N-2}\mu & (-p)^{N-3}\mu & & \mu & p \end{bmatrix};$$

$$\mathcal{L}_{(0)} = \sqrt{\mu} \begin{bmatrix} 1 \\ -p \\ p^2 \\ -p^3 \\ \vdots \\ (-p)^{N-1} \end{bmatrix}$$

with $\mu = 1 - p^2$

For instance, in the case of $\mathcal{N} = 5$, the matrix \mathcal{A}_l and $\mathcal{L}_{(0)}$ are given by:

$$\mathcal{A}_l = \begin{bmatrix} p & 0 & 0 & 0 & 0 \\ \mu & p & 0 & 0 & 0 \\ -p\mu & \mu & p & 0 & 0 \\ p^2\mu & -p\mu & \mu & p & 0 \\ -p^3\mu & p^2\mu & -p\mu & \mu & p \end{bmatrix}; \quad \mathcal{L}_{(0)} = \sqrt{\mu} \begin{bmatrix} 1 \\ -p \\ p^2 \\ -p^3 \\ p^4 \end{bmatrix}$$

Moreover, the Laguerre functions are well known by their orthonormality, this property can be expressed by the following relationship:

$$\sum_{k=0}^{\infty} \ell_{f(k)} \ell_{h(k)} = \begin{cases} 0, & \text{for } f \neq h \\ 1, & \text{for } f = h \end{cases} \quad (12)$$

The orthonormality will be incorporated into the MPC scheme. The main idea in Laguerre functions-based MPC is to express each element of the future incremental control trajectory by a set of Laguerre functions.

At each time instant k , the future incremental control trajectory vector, i.e., $\Delta \mathbf{u}_{(k, N_u|k)}$ is captured by combining a set of Laguerre functions, defined in (10), with a set of the following Laguerre coefficients $\eta_{(k)}$ obtained from the online optimization:

$$\eta_{(k)} = [c_{1(k)} \ c_{2(k)} \ \dots \ c_{\mathcal{N}(k)}]^T \quad (13)$$

Then, all the components of the incremental control trajectory for a single input system, at the future time l , can be accurately approximated by a linear combination of Laguerre functions as follows

$$\Delta \mathbf{u}_{(k+l|k)} \cong \sum_{m=1}^{\mathcal{N}} \ell_{m(l)} c_{m(k)} \quad (14)$$

Hence, Eq. (14), for $l = \overline{0, N_u - 1}$, can be rewritten as follows

$$\Delta \mathbf{u}_{(k+l|k)} = \mathcal{L}_{(l)}^T \eta_{(k)} \quad (15)$$

According to (15), we can remark on the one hand, that the number of terms \mathcal{N} with the free parameter p are used implicitly to capture the optimal control sequence.

On the other hand, the coefficient vector $\eta_{(k)}$ depends only on the initial time k .

For the sake of simplicity, the expression of $\eta_{(k)}$ in (13) is abbreviated as follows

$$\eta = [c_1 \ c_2 \ \dots \ c_{\mathcal{N}}]^T$$

The control increment in (15) shows that the control horizon N_u , which is one of the tuning parameters of the classical MPC approach, is omitted from the control design. Consequently, with the discrete-time Laguerre functions, an alternative formulation of the performance index can be performed by using the coefficient vector η .

In a matter of fact, the second term of Eq. (6) can be rewritten as :

$$\begin{aligned} \|\Delta u_{(k+l-1|k)}\|_R^2 &= \sum_{l=1}^{N_u} \Delta u_{(k+l-1|k)}^T R \Delta u_{(k+l-1|k)} \\ &= \sum_{l=1}^{N_p} \Delta u_{(k+l-1|k)}^T R \Delta u_{(k+l-1|k)} \end{aligned} \quad (16)$$

In order to compute all the predictions over the prediction horizon N_p , all the incremental controls beyond the control horizon N_u are supposed to be zero, that is $\Delta u_{(k+l-1|k)} = 0$ for $N_u < l \leq N_p$.

Substituting (15) into (16) yields:

$$\begin{aligned} \|\Delta u_{(k+l-1|k)}\|_R^2 &= \sum_{l=1}^{N_u} \Delta u_{(k+l-1|k)}^T R \Delta u_{(k+l-1|k)} \\ &= \sum_{l=1}^{N_p} \Delta u_{(k+l-1|k)}^T R \Delta u_{(k+l-1|k)} \end{aligned} \quad (17)$$

where, $R_L = \text{block-diag}\{R \ R \ \dots \ R\} \in \mathfrak{R}^{N \times N}$.

This weighting matrix is used as a tuning parameter for the desired closed-loop response and N_p is chosen sufficiently large to satisfy the orthonormal property, that is:

$$\sum_{k=0}^{N_p} \ell_{f(k)} \ell_{h(k)} = \begin{cases} 0, & \text{for } f \neq h \\ 1, & \text{for } f = h \end{cases} \quad (18)$$

Then, the set of coefficients η in (13) is obtained from the online optimization. With this design framework, contrary to the classical MPC approach, the control horizon N_u is not needed. Note that the number of variables involved in the control vector according to (17) is just \mathcal{N} rather than N_u used in the original performance index. Typically, \mathcal{N} is smaller than N_u ($\mathcal{N} < N_u$) which may reduce the computational burden. Indeed, a larger control horizon N_u can lead to higher computational burden and memory storage. As it can be seen, the parameters in discrete MPC using Laguerre functions, namely p and \mathcal{N} , are used to capture the projected control signal and the prediction horizon N_p . The value of the Laguerre pole location p is included between 0 and 1 (i.e., $p \in [0,1]$) whereas the number of terms \mathcal{N} is selected to satisfy the impulse response. In this paper, a Laguerre functions-based MPC is used in the proposed distributed MPC scheme.

Remark 2: The stability and the desired closed-loop performance of the unconstrained DMPC based on Laguerre functions can be designed and tuned by adjusting the weighting matrices Q_i , R_i , the horizon N_p and the design parameters p_i , \mathcal{N}_i .

6. PROPOSED DISTRIBUTED MPC DESIGN USING LAGUERRE FUNCTIONS

6.1 The modified interaction prediction

Given the measurements until instant h , the future states, inputs and interactions vectors from instant k to instant $k + l - 1$, with $k \geq h$, are expressed by

$$\widehat{D}_{(k,l|h)}^{\{i\}} = \begin{bmatrix} \widehat{d}_{(k|h)}^{\{i\}} \\ \widehat{d}_{(k+1|h)}^{\{i\}} \\ \vdots \\ \widehat{d}_{(k+l-1|h)}^{\{i\}} \end{bmatrix}, \widehat{D}_{(k,l|h)} = \begin{bmatrix} \widehat{D}_{(k,l|h)}^{\{1\}} \\ \widehat{D}_{(k,l|h)}^{\{2\}} \\ \vdots \\ \widehat{D}_{(k,l|h)}^{\{N\}} \end{bmatrix}$$

where, $\widehat{d} \in \{\widehat{x}, \widehat{y}, \widehat{w}, \widehat{v}\}$ and $\widehat{D} \in \{\widehat{X}, \widehat{Y}, \widehat{W}, \widehat{V}\}$.

$$U_{(k,l|h)}^{\{i\}} = \begin{bmatrix} u_{(k|h)}^{\{i\}} \\ u_{(k+1|h)}^{\{i\}} \\ \vdots \\ u_{(k+l-1|h)}^{\{i\}} \end{bmatrix}, U_{(k,l|h)} = \begin{bmatrix} U_{(k,l|h)}^{\{1\}} \\ U_{(k,l|h)}^{\{2\}} \\ \vdots \\ U_{(k,l|h)}^{\{N\}} \end{bmatrix}$$

Furthermore, let us assume that:

$$\tilde{P}_i = [P_{i1} \ \dots \ P_{ii-1} \ 0 \ P_{ii+1} \ \dots \ P_{iN}]$$

where, $\tilde{P}_i \in \{\tilde{A}_i, \tilde{B}_i, \tilde{C}_i\}$.

For $i = \overline{1, N}$, the l -step ahead predictions of the local interaction vectors, at time instant k , can be described as follows :

$$\begin{aligned} \widehat{v}_{(k+l|k)}^{\{i\}} &= \sum_{j=1}^N [A_{ij} \widehat{x}_{(k+l|k)}^{\{j\}} + B_{ij} u_{(k+l|k)}^{\{j\}}] \\ \widehat{w}_{(k+l|k)}^{\{i\}} &= \sum_{j=1}^N C_{ij} \widehat{x}_{(k+l|k)}^{\{j\}} \end{aligned} \quad (19)$$

Using the previous definitions, the compact form of the interaction prediction vectors can be expressed as follows

$$\begin{aligned} \widehat{\mathcal{V}}_{(k,N_p|k-1)}^{\{i\}} &= \tilde{A}_i \widehat{X}_{(k,N_p|k-1)} + \tilde{B}_i U_{(k,N_p|k-1)} \\ \widehat{\mathcal{W}}_{(k,N_p|k-1)}^{\{i\}} &= \tilde{C}_i \widehat{X}_{(k,N_p|k-1)} \end{aligned} \quad (20)$$

Referring to Assumption 1-(d), the information of the other subsystems is available only after one sampling time interval. In this case, $U_{(k,N_p|k-1)}^{\{i\}}$ and $U_{(k,N_p|k-1)}$ can be expressed in terms of $U_{(k-1,N_p|k-1)}^{\{i\}}$ and $U_{(k-1,N_p|k-1)}$ respectively as follows:

$$U_{(k,N_p|k-1)}^{\{i\}} = \widehat{\Gamma}_i U_{(k-1,N_p|k-1)}^{\{i\}} \quad (21a)$$

$$U_{(k,N_p|k-1)} = \widehat{\Gamma} U_{(k-1,N_p|k-1)} \quad (21b)$$

where,

$$\widehat{\Gamma}_i = \begin{bmatrix} & I_{(N_p-1)n_{u_i}} \\ 0_{N_p n_{u_i} \times n_{u_i}} & I_{n_{u_i}} \end{bmatrix}$$

$$\widehat{\Gamma} = \text{diag}\{\widehat{\Gamma}_1, \widehat{\Gamma}_2, \dots, \widehat{\Gamma}_N\}$$

By substituting (21a) into (20), the interaction prediction vectors can be rewritten as follows:

$$\begin{aligned} \widehat{\mathcal{V}}_{(k,N_p|k-1)}^{\{i\}} &= \tilde{A}_i \widehat{X}_{(k,N_p|k-1)} + \tilde{B}_i \widehat{\Gamma}_i U_{(k-1,N_p|k-1)}^{\{i\}} \\ \widehat{\mathcal{W}}_{(k,N_p|k-1)}^{\{i\}} &= \tilde{C}_i \widehat{X}_{(k,N_p|k-1)} \end{aligned} \quad (22)$$

Note that the state predictions and the control actions contained in expression (22) are calculated at time instant $k - 1$ and transmitted through the digital network.

6.2 The modified state predictor

The state variable of the i^{th} subsystem, at sampling instant l , can be predicted as follows:

$$\hat{x}_{(k+l|k)}^{(i)} = A_{ii}^l \hat{x}_{(k|k)}^{(i)} + \sum_{s=1}^l A_{ii}^{s-1} [B_{ii} u_{(k+l-s|k)}^{(i)} + A_{ii}^{s-1} \hat{v}_{(k+l-s|k-1)}^{(i)}] \quad (23)$$

Let us now define, for $l = \overline{1, N_p}$

$$u_{(k+l-s|k)}^{(i)} = u_{(k-1)}^{(i)} + \sum_{n=s}^l \Delta u_{(k-s-1|k)}^{(i)} \text{ for } s = \overline{1, l} \quad (24)$$

By means (23) and (24), the states $\hat{x}_{(k+l|k)}^{(i)}$ become

$$\begin{aligned} \hat{x}_{(k+l|k)}^{(i)} &= A_{ii}^l \hat{x}_{(k|k)}^{(i)} + \sum_{s=1}^l A_{ii}^{s-1} B_{ii} u_{(k-1)}^{(i)} \\ &+ \sum_{s=1}^l \left(\sum_{n=s}^l A_{ii}^{l-n} B_{ii} \right) \Delta u_{(k+s-1|k)}^{(i)} \\ &+ \sum_{s=1}^l A_{ii}^{s-1} \hat{v}_{(k+l-s|k-1)}^{(i)} \end{aligned} \quad (25)$$

According to the expression (15), the incremental control trajectory along the prediction horizon at time instant k can be calculated as $\Delta u_{(k+s-1|k)}^{(i)} = \mathcal{L}_{i(s-1)}^T \eta_i$ in Laguerre formulation.

Then, the l -step ahead states can be predicted by

$$\begin{aligned} \hat{x}_{(k+l|k)}^{(i)} &= A_{ii}^l \hat{x}_{(k|k)}^{(i)} + \sum_{s=1}^l A_{ii}^{s-1} B_{ii} u_{(k-1)}^{(i)} \\ &+ \sum_{s=1}^l \left(\sum_{n=s}^l A_{ii}^{l-n} B_{ii} \right) \mathcal{L}_{i(s-1)}^T \eta_i \\ &+ \sum_{s=1}^l A_{ii}^{s-1} \hat{v}_{(k+l-s|k-1)}^{(i)} \end{aligned} \quad (26)$$

where, the predicted control is rewritten as

$$\begin{aligned} \sum_{s=1}^l A_{ii}^{s-1} B_{ii} u_{(k+l-s|k)}^{(i)} &= \sum_{s=1}^l A_{ii}^{s-1} B_{ii} u_{(k-1)}^{(i)} \\ &+ \sum_{s=1}^l \left(\sum_{n=s}^l A_{ii}^{l-n} B_{ii} \right) \mathcal{L}_{i(s-1)}^T \eta_i \end{aligned}$$

and the predicted output variable writes

$$\begin{aligned} \hat{y}_{(k+l|k)}^{(i)} &= C_{ii} A_{ii}^l \hat{x}_{(k|k)}^{(i)} + C_{ii} \sum_{s=1}^l A_{ii}^{s-1} B_{ii} u_{(k-1)}^{(i)} \\ &+ C_{ii} \sum_{s=1}^l \left(\sum_{n=s}^l A_{ii}^{l-n} B_{ii} \right) \mathcal{L}_{i(s-1)}^T \eta_i \end{aligned} \quad (27)$$

$$+ C_{ii} \sum_{s=1}^l A_{ii}^{s-1} \hat{v}_{(k+l-s|k-1)}^{(i)} + \hat{w}_{(k+l|k-1)}^{(i)}$$

The previous expressions of the predicted future state and output variables, show that the control signal $u_{(k)}^{(i)}$ is not needed and it can be replaced by the coefficient vector η_i . However, the main idea is finding the coefficient vector η_i that minimizes the performance index (31). In order to estimate the prediction, the following convolution sum

$$C_{si(l)} = \sum_{s=1}^l \left(\sum_{n=s}^l A_{ii}^{l-n} B_{ii} \right) \mathcal{L}_{i(s-1)}^T \quad (28)$$

requires to be evolved. For further details see [47].

Consequently, the expression (28) evaluated along the prediction horizon leads to

$$\begin{cases} C_{si(l)} = C_{si(l-1)} + \sum_{n=1}^l A_{ii}^{l-1} C_{si(1)} (\mathcal{A}_{li}^{l-n})^T \\ \text{with } C_{si(1)} = B_{ii} \mathcal{L}_{i(0)}^T \text{ for } l = \overline{1, N_p} \end{cases} \quad (29)$$

where, the matrix $\mathcal{A}_{li} \in \mathfrak{R}^{N_i \times N_i}$ is depending on the scaling factor p_i and μ_i , defined in (11).

Compact form representation:

The compact prediction of (26) has the following form

$$\begin{aligned} &\hat{x}_{(k+1, N_p|k)}^{(i)} \\ &= \bar{L}_i \hat{x}_{(k|k)}^{(i)} + \bar{M}_i u_{(k-1)}^{(i)} + \bar{N}_i \eta_i + \bar{S}_i \hat{v}_{(k, N_p|k-1)}^{(i)} \end{aligned} \quad (30)$$

where,

$$\bar{S}_i = \begin{bmatrix} A_{ii}^0 & \dots & 0_{n_{x_i}} \\ \vdots & \ddots & \vdots \\ A_{ii}^{N_p-1} & \dots & A_{ii}^0 \end{bmatrix}, \bar{L}_i = \bar{S}_i \begin{bmatrix} A_{ii} \\ 0_{N_p n_{x_i} \times n_{x_i}} \end{bmatrix}$$

$$\bar{M}_i = \bar{S}_i \begin{bmatrix} B_{ii} \\ B_{ii} \\ \vdots \\ B_{ii} \end{bmatrix}, \bar{N}_i = \begin{bmatrix} C_{si(1)} \\ A_{ii} C_{si(1)} + C_{si(1)} (\mathcal{A}_{li}^1)^T \\ A_{ii} C_{si(2)} + C_{si(1)} (\mathcal{A}_{li}^2)^T \\ \vdots \\ A_{ii} C_{si(N_p-1)} + C_{si(1)} (\mathcal{A}_{li}^{N_p-1})^T \end{bmatrix}$$

6.3 The modified optimizer

The local performance index in (6) can be rewritten using the coefficient vector η_i as follows:

$$\tilde{J}_i = \sum_{l=1}^{N_p} \|\hat{y}_{(k+l|k)}^{(i)} - y_{(k+l|k)}^{(i)sp}\|_{Q_i}^2 + \eta_i^T R_{L_i} \eta_i \quad (31)$$

The modified performance index shows that the control horizon N_u , one of the tuning parameters of MPC for the desired closed-loop response, is omitted.

Now, the objective is to find, at each time instant, the coefficient vector η_i that minimizes \tilde{J}_i . By substituting (27) into the performance index (31), we obtain

$$\begin{aligned} & \tilde{J}_i \\ = & \eta_i^T \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Omega}_{i(l)}^T + R_{L_i} \right) \eta_i \\ & + 2\eta_i^T \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Theta}_{i(l)} x_{(l)}^{(i)} \right) \\ & + 2\eta_i^T \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Psi}_{i(l)} u_{(k-1)}^{(i)} \right) + 2\eta_i^T \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Phi}_{i(l)} \right) \\ & + 2\eta_i^T \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \hat{w}_{(k+l-s|k-1)}^{(i)} \right) \\ & - 2\eta_i^T \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i y_{(k+l|k)}^{(i)sp} \right) \end{aligned} \quad (32)$$

To find the minimum of (32), without constraints, the first partial differentiation of the performance index can be used, leading to:

$$\begin{aligned} \frac{\partial \tilde{J}_i}{\partial \eta_i} = & 2 \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Omega}_{i(l)}^T + R_{L_i} \right) \eta_i \\ & + 2 \left(- \sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i y_{(k+l|k)}^{(i)sp} \right. \\ & + \sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Theta}_{i(l)} x_{(l)}^{(i)} \\ & + \sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Psi}_{i(l)} u_{(k-1)}^{(i)} \\ & + \sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Phi}_{i(l)} \\ & \left. + \sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \hat{w}_{(k+l-s|k-1)}^{(i)} \right) \end{aligned} \quad (33)$$

The necessary and sufficient optimality conditions of the minimum of \tilde{J}_i are obtained for:

$$\frac{\partial \tilde{J}_i}{\partial \eta_i} = 0$$

whereby the optimal parameter vector is computed as follows:

$$\begin{aligned} \hat{\eta}_i = \eta_i^{opt} = & \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Omega}_{i(l)}^T \right. \\ & + R_{L_i} \left. \right)^{-1} \times \left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i (y_{(k+l|k)}^{(i)sp} \right. \\ & - \bar{\Theta}_{i(l)} \hat{x}_{(k|k)}^{(i)} - \bar{\Psi}_{i(l)} u_{(k-1)}^{(i)} - \bar{\Phi}_{i(l)} \\ & \left. - \hat{w}_{(k+l-s|k-1)}^{(i)}) \right) \end{aligned} \quad (34)$$

with the assumption that the inversion of the *Hessian* matrix

$$\left(\sum_{l=1}^{N_p} \bar{\Omega}_{i(l)} Q_i \bar{\Omega}_{i(l)}^T + R_{L_i} \right)^{-1} \text{ exists.}$$

where,

$$\begin{aligned} \bar{\Theta}_{i(l)} &= C_{ii} A_{ii}^l, \quad \bar{\Psi}_{i(l)} = C_{ii} \sum_{s=1}^l A_{ii}^{s-1} B_{ii} \\ \bar{\Omega}_{i(l)}^T &= C_{ii} \Omega_{i(l)}^T, \quad \bar{\Phi}_{i(l)} = C_{ii} \sum_{s=1}^l A_{ii}^{s-1} \hat{v}_{(k+l-s|k-1)}^{(i)} \end{aligned}$$

Compact form representation:

In order to write the compact form of the solution of the optimization problem (31), the following notations are adopted:

$$\begin{aligned} Y_{(k,l|h)}^{(i)sp} &= \begin{bmatrix} y_{(k|h)}^{(i)sp} \\ y_{(k+1|h)}^{(i)sp} \\ \vdots \\ y_{(k+l-1|h)}^{(i)sp} \end{bmatrix}, \\ N_i &= \begin{bmatrix} C_{ii} C_{si(1)} \\ C_{ii} [A_{ii} C_{si(1)} + C_{si(1)} (\mathcal{A}_{ii}^1)^T] \\ C_{ii} [A_{ii} C_{si(2)} + C_{si(1)} (\mathcal{A}_{ii}^2)^T] \\ \vdots \\ C_{ii} [A_{ii} C_{si(N_p-1)} + C_{si(1)} (\mathcal{A}_{ii}^{N_p-1})^T] \end{bmatrix} \\ S_i &= \begin{bmatrix} C_{ii} A_{ii}^0 & \dots & 0_{n_{x_i}} \\ \vdots & \ddots & \vdots \\ C_{ii} A_{ii}^{N_p-1} & \dots & C_{ii} A_{ii}^0 \end{bmatrix}, \quad M_i = S_i \begin{bmatrix} B_{ii} \\ B_{ii} \\ \vdots \\ B_{ii} \end{bmatrix}, \\ L_i &= \begin{bmatrix} (C_{ii} A_{ii}^1)^T \\ \vdots \\ (C_{ii} A_{ii}^{N_p})^T \end{bmatrix}, \\ T_i &= \begin{bmatrix} 0_{n_{y_i}} & & & \\ \vdots & & & \\ 0_{n_{y_i}} & & I_{(N_p-1)n_{y_i}} & \\ \hline 0_{n_{y_i}} & 0_{n_{y_i}} & \dots & 0_{n_{y_i}} & I_{n_{y_i}} \end{bmatrix} \end{aligned}$$

Now, by means of these notations, it would be easy to write the compact form of the predicted output as follows:

$$\begin{aligned} \hat{Y}_{(k+1,N_p|k)}^{(i)} &= L_i \hat{x}_{(k|k)}^{(i)} + M_i u_{(k-1)}^{(i)} + N_i \eta_i \\ &+ S_i \hat{v}_{(k,N_p|k-1)}^{(i)} + T_i \hat{w}_{(k,N_p|k-1)}^{(i)} \end{aligned} \quad (35)$$

Consequently, the compact local performance index has the following form

$$\tilde{J}_i = \| \hat{Y}_{(k+1,N_p|k)}^{(i)} - Y_{(k+1,N_p|k)}^{(i)sp} \|_{Q_i}^2 + \eta_i^T R_{L_i} \eta_i \quad (36)$$

$$\begin{aligned} \tilde{J}_i &= [\hat{Y}_{(k+1,N_p|k)}^{(i)} - Y_{(k+1,N_p|k)}^{(i)sp}]^T \bar{Q}_i [\hat{Y}_{(k+1,N_p|k)}^{(i)} \\ &- Y_{(k+1,N_p|k)}^{(i)sp}] + \eta_i^T R_{L_i} \eta_i \end{aligned} \quad (37)$$

the process, whose dynamics is perfectly represented by the process model.

Let us now define the global matrices as follows:

$$\begin{aligned}\tilde{\mathcal{H}} &= [\tilde{\mathcal{H}}_1^T, \dots, \tilde{\mathcal{H}}_{\mathbb{N}}^T]^T, \quad \text{where } \tilde{\mathcal{H}} \in \{\tilde{A}, \tilde{B}, \tilde{C}\} \\ \Gamma_i' &= \begin{bmatrix} I_{n_{u_i}} \\ \vdots \\ I_{n_{u_i}} \end{bmatrix}, \Gamma_i = [I_{n_{u_i}} \ 0_{n_{u_i} \times (N_p - 1)n_{u_i}}] \\ \Lambda &= \text{diag}\{\Lambda_1, \dots, \Lambda_{\mathbb{N}}\}, \quad \text{where } \Lambda \in \{\Gamma', \Gamma, \hat{\Gamma}\} \\ \Delta U_{(k, N_p | k)}^{\{i\}} &= \begin{bmatrix} \Delta u_{(k|k)}^{\{i\}} \\ \Delta u_{(k+1|k)}^{\{i\}} \\ \Delta u_{(k+2|k)}^{\{i\}} \\ \vdots \\ \Delta u_{(k+N_p-1|k)}^{\{i\}} \end{bmatrix} = \begin{bmatrix} \mathcal{L}_{i(0)}^T \\ \mathcal{L}_{i(0)}^T (\mathcal{A}_{ii}^1)^T \\ \mathcal{L}_{i(0)}^T (\mathcal{A}_{ii}^2)^T \\ \vdots \\ \mathcal{L}_{i(0)}^T (\mathcal{A}_{ii}^{N_p-1})^T \end{bmatrix} \eta_{i(k)} = Z_i \eta_{i(k)} \\ Z_i &= \begin{bmatrix} \mathcal{L}_{i(0)}^T \\ \mathcal{L}_{i(0)}^T (\mathcal{A}_{ii}^1)^T \\ \mathcal{L}_{i(0)}^T (\mathcal{A}_{ii}^2)^T \\ \vdots \\ \mathcal{L}_{i(0)}^T (\mathcal{A}_{ii}^{N_p-1})^T \end{bmatrix}, \eta_{i(k)} = [\eta_{i1}^T(k), \dots, \eta_{iN}^T(k)]^T \\ \mathbb{Q} &= \text{diag}\{\mathbb{Q}_1, \dots, \mathbb{Q}_{\mathbb{N}}\}\end{aligned}$$

where, $\mathbb{Q} \in \{\bar{L}, L, \bar{M}, M, S, T, \bar{K}, \bar{N}, Z\}$.

Theorem 1 (Global stability condition for NC-DMPC scheme)

The closed-loop system including the open-loop system \mathcal{S} and the feedback distributed control solution of the NC-DMPC problem, composed by a set of independent sub-controllers $\mathcal{C}_i, i = 1, \dots, \mathbb{N}$, is asymptotically stable if and only if:

$$\begin{aligned}& \left| \lambda_j \left\{ \begin{bmatrix} A & 0 & |\lambda_j \{A_G\}| & 0 \\ \Sigma & \Pi & B\Gamma & F \\ (\bar{\Gamma}Z\theta A + \bar{\Gamma}Z\phi\Sigma) & \bar{\Gamma}Z\phi\Pi & (\bar{\Gamma}Z\theta B\Gamma + \Gamma'\Gamma + \bar{\Gamma}Z\rho) & \bar{\Gamma}Z\phi F \\ 0 & 0 & I_{N_p n_u} & 0 \end{bmatrix} \right\} \right| < 1 \\ & \forall j \in \{1, 2, \dots, N_G\}, \quad N_G = n_x + N_p n_x + 2N_p n_u \quad (42)\end{aligned}$$

where, A_G is the dynamic matrix of the global closed-loop system and N_G is its order.

The reader can refer to Appendix for details about the proof of this Theorem.

Corollary (Global stability condition for Decentralized MPC (De-MPC) scheme)

If the network communication is not used, then no shared information among the sub-controllers is possible, the global closed-loop system is asymptotically stable if and only if:

$$\begin{aligned}& \left| \lambda_j \left\{ \begin{bmatrix} A & B\Gamma \\ (\bar{\Gamma}Z\theta A) & (\bar{\Gamma}Z\theta B\Gamma + \Gamma'\Gamma + \bar{\Gamma}Z\rho) \end{bmatrix} \right\} \right| < 1 \\ & \forall j \in \{1, 2, \dots, \bar{N}_G\}, \quad \bar{N}_G = n_x + N_p n_u \quad (43)\end{aligned}$$

where, \bar{A}_G is the dynamic matrix of the global closed-loop system and \bar{N}_G is its order.

8. NUMERICAL SIMULATION RESULTS AND ANALYSIS

In this section, the proposed NC-DMPC algorithm, presented in the previous section, is tested and evaluated through a simulation example, related to popular case studies in the context of distributed control. This numerical simulation has been conducted to assess the efficiency of the proposed NC-DMPC algorithm. The proposed example for this evaluation is presented in the sequel.

8.1 System description and representation

A benchmark example, often used to assess the effectiveness of distributed control algorithms is the quadruple tank system, described in [48-50], and illustrated in Figure 4. The process consists of four interconnected water tanks and two pumps. It is a MIMO system with two manipulated variables and four state variables. The goal here is to control the water levels h_1 and h_3 in tanks 1 and 3 using the pumps command voltages v_1 and v_2 . Let us define $x = [x_1 \ x_2 \ x_3 \ x_4]^T = [h_1 \ h_2 \ h_3 \ h_4]^T$ and $u = [u_1 \ u_2]^T = [v_1 \ v_2]^T$ with h_i and v_j are the liquid levels and the voltages, respectively (cf. Figure 4).

Using mass balances and Bernoulli's law, we obtain

$$\begin{aligned}h_1(t) &= -\frac{a_1}{S_1} \sqrt{2gh_1(t)} + \frac{a_4}{S_4} \sqrt{2gh_4(t)} + \frac{\gamma_1 k_1}{S_1} v_1 \\ h_2(t) &= -\frac{a_2}{S_2} \sqrt{2gh_2(t)} + \frac{(1 - \gamma_1)k_1}{S_2} v_1 \\ h_3(t) &= -\frac{a_3}{S_3} \sqrt{2gh_3(t)} + \frac{a_2}{S_2} \sqrt{2gh_2(t)} + \frac{\gamma_2 k_2}{S_3} v_2 \\ h_4(t) &= -\frac{a_4}{S_4} \sqrt{2gh_4(t)} + \frac{(1 - \gamma_2)k_2}{S_4} v_2\end{aligned} \quad (44)$$

where, S_i : cross-section of Tank i ; a_i : cross-section of the outlet hole of Tank i ; $h_i(t)$: water level of Tank i ; g : acceleration of gravity.

The voltage applied to Pump i is v_i and the corresponding flow is $k_i v_i$. The parameters $\gamma_1, \gamma_2 \in [0 \ 1]$ are determined from how the valves are set prior to an experiment. The flow to tank 1 is $\gamma_1 k_1 v_1$ and the flow to tank 2 is $(1 - \gamma_1)k_1 v_1$ and similarly for tanks 3 and 4. The measured level signals are $k_c h_1$ and $k_c h_3$. According to [49], the values of the parameters considered in the following simulations are:

$$S_1 = S_4 = 28 [cm^2], \quad S_2 = S_3 = 32 [cm^2], \quad a_1 = a_4 = 0.071 [cm^2], \quad a_2 = a_3 = 0.057 [cm^2], \quad g = 981 [cm/s^2], \quad k_c = 1 [V/cm].$$

The chosen operating points correspond to the following parameter values: $(h_1^0, h_3^0) = (12.4, 12.7) [cm]$, $(h_2^0, h_4^0) = (1.4, 1.8) [cm]$, $(v_1, v_2) = (3, 3) [V]$, $(k_1, k_2) = (3.35, 3.33) [cm^3/Vs]$, $(\gamma_1, \gamma_2) = (0.7, 0.6)$.

The linearization of system (44) and its zero-order-hold discretization with sampling time $T_s = 0.5$ s, leads to the discrete-time state-space representation of the form (1), with $n_x = 4$ and $n_u = n_y = 2$. The obtained model matrices are given by the following:

$$A = \begin{bmatrix} 0.9921 & 0 & 0 & 0.0206 \\ 0 & 0.9835 & 0 & 0 \\ 0 & 0.0165 & 0.9945 & 0 \\ 0 & 0 & 0 & 0.9793 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.0417 & 2.47 \times 10^{-3} \\ 0.0156 & 0 \\ 1.30 \times 10^{-3} & 0.0311 \\ 0 & 0.0235 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

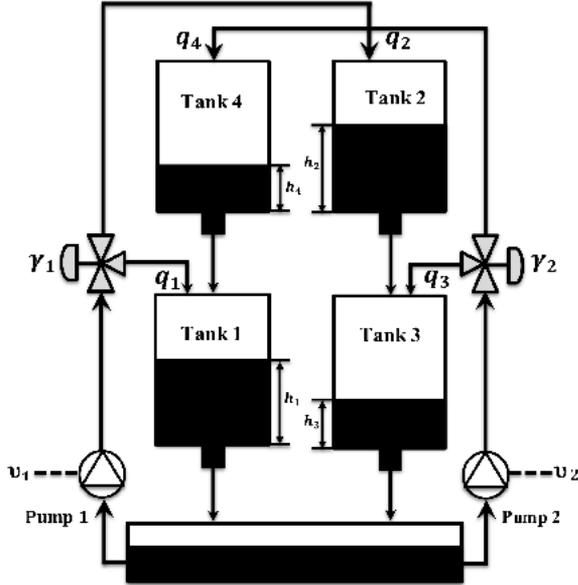


Figure 4. A schematic representation of the quadruple tank benchmark

In order to apply the proposed NC-DMPC control scheme, the whole system has been partitioned into two interconnected subsystems. The first one is composed of tank 1 and tank 2 and the second one is composed of tank 3 and tank 4. Therefore, the states and inputs are accordingly partitioned as follows:

$$x^{(1)} = [h_1 \quad h_2]^T \text{ and } u^{(1)} = [v_1];$$

$$x^{(2)} = [h_3 \quad h_4]^T \text{ and } u^{(2)} = [v_2]$$

Subsystem 1:

According to the form (3), we set the subsystem \mathcal{S}_1 with matrices $\{A_{11}, B_{11}, C_{11}, A_{12}, B_{12}\}$, where,

$$A_{11} = \begin{bmatrix} 0.9921 & 0 \\ 0 & 0.9835 \end{bmatrix}, B_{11} = \begin{bmatrix} 0.0417 \\ 0.0156 \end{bmatrix}, C_{11} = [1 \quad 0]$$

$$A_{12} = \begin{bmatrix} 0 & 0.0206 \\ 0 & 0 \end{bmatrix}, B_{12} = \begin{bmatrix} 2.47 \times 10^{-2} \\ 0 \end{bmatrix}$$

Subsystem 2:

Also, according to the form (3), we set the subsystem \mathcal{S}_2 with matrices $\{A_{22}, B_{22}, C_{22}, A_{21}, B_{21}\}$, where,

$$A_{22} = \begin{bmatrix} 0.9945 & 0 \\ 0 & 0.9793 \end{bmatrix}, B_{22} = \begin{bmatrix} 0.0311 \\ 0.0235 \end{bmatrix}, C_{22} = [1 \quad 0]$$

$$A_{21} = \begin{bmatrix} 0 & 0.0165 \\ 0 & 0 \end{bmatrix}, B_{21} = \begin{bmatrix} 1.30 \times 10^{-3} \\ 0 \end{bmatrix}$$

The interactions between the subsystems \mathcal{S}_1 and \mathcal{S}_2 are considered through the state and control input variables which will allow to test the effectiveness of the proposed DMPC structure in presence of both interactions. The proposed algorithm, described in section 7, was used to implement the two sub-controllers, where the control objective was to keep the levels of tank 1 and tank 3 at the reference values, expressed by their respective set-point signals, defined as follows

• For tank 1 (level 1):

- (a) from 0 s to 50 s, the set-point is 0.2 m;
- (b) from 51 s to 150 s, the set-point is 0.6 m;

• For tank 3 (level 3):

- (a) from 0 s to 20 s, the set-point is 0 m;
- (b) from 21 s to 80 s, the set-point is 0.4 m;
- (c) from 81 s to 150 s, the set-point is 0.1 m.

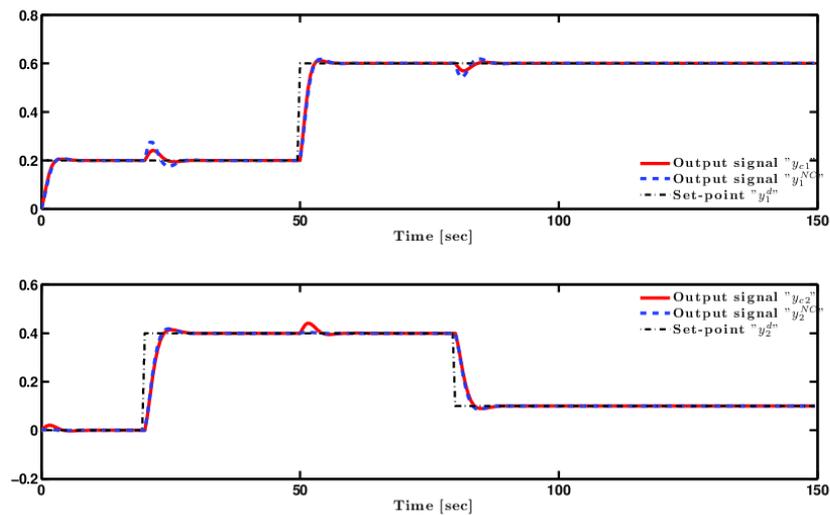


Figure 5. Evolution versus time of the output responses of Tank 1 and Tank 3, obtained with classical C-MPC and the proposed NC-DMPC for the quadruple tank system

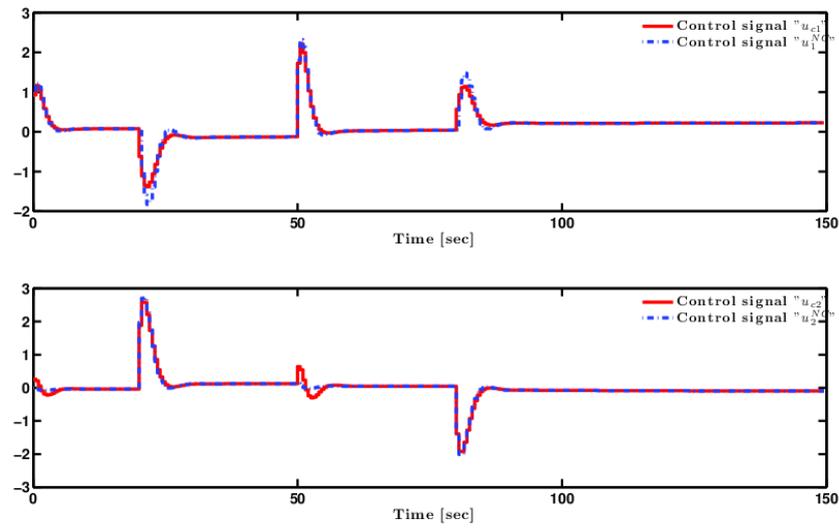


Figure 6. Evolution versus time of the manipulated variables (control inputs); v_1 and v_2 ; generated by classical C-MPC and the proposed NC-DMPC schemes for the control of the quadruple tank system

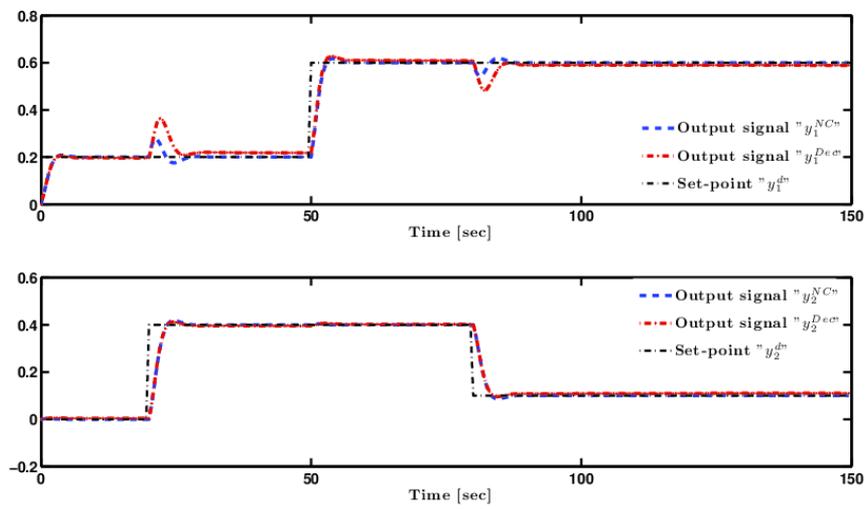


Figure 7. Evolution versus time of the output responses of Tank 1 and Tank 3, obtained with decentralized MPC and the proposed NC-DMPC for the quadruple tank system

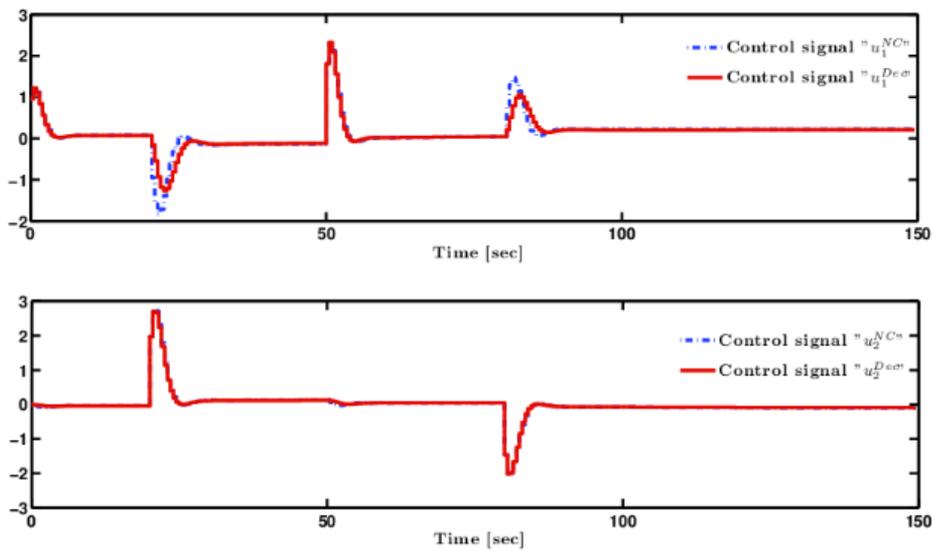


Figure 8. Evolution versus time of the manipulated variables (control inputs); v_1 and v_2 ; generated by decentralized MPC and the proposed NC-DMPC schemes for the control of the quadruple tank system

Table 1. Summary of the control design parameters of the different MPC controllers

Controllers	N_u	N_p	Q	R	\mathcal{N}	p	T_s	N^s
Classical-MPC (C-MPC)	5	10	$1 \times I_2$	$0.02 \times I_2$	-	-	0.5	300
NC-DMPC & De-MPC: \mathcal{C}_1	-	10	$1 \times I_2$	$0.02 \times I_2$	$\mathcal{N}_1 = 3$	$p_1=0.5$	0.5	300
NC-DMPC & De-MPC: \mathcal{C}_2	-	10	$1 \times I_2$	$0.02 \times I_2$	$\mathcal{N}_2 = 3$	$p_2=0.5$	0.5	300

Note that the design parameters N_p , R_i , Q_i , p_i and \mathcal{N}_i , for $i=1,2$ needed for the distributed optimization problem, have been manually (i.e., heuristically) tuned to obtain the best performance. To this end, several tests were performed to determine the most suitable values for the proposed case. Besides, the optimal values can be found using a multi-objective optimization procedure or one of the metaheuristic methods (such as swarm optimization technique), but this is out of the scope of this study. Usually, a trade-off between the two terms of the performance index should be ensured, depending on each particular circumstance (response rapidity or lower energy).

In order to make a fair comparison, the closed-loop performance of the proposed controller are compared with a Laguerre functions-based Decentralized MPC (DeMPC), where the network communication is omitted (Figure 1(b)), and a classical centralized MPC (C-MPC). The simulation parameter settings are summarized in Table 1.

where, N_u denotes the control horizon, it is selected by considering a good trade-off between tracking performance and the amount of the control effort; N_p is the prediction horizon, it is assumed sufficiently large, to ensure the closed-loop stability. Q and R are two constant matrices serving as weights of the tracking errors and control actions, respectively; p_i is the scaling factor, it is a tuning parameter for the closed-loop response speed; \mathcal{N}_i is the number of Laguerre functions representing the degrees of freedom in describing the control trajectory; T_s is the sampling time; N^s is the total number of samples of the whole simulation.

For this example, the obtained simulation results are depicted in Figures 5, 6, 7 and 8, including the system outputs and the generated control signals. The obtained simulation results, demonstrate that the proposed distributed control structure has a good control performance, since the quadruple tank system under the proposed DMPC strategy has the ability to track the set-point signals; with smooth control signals. Although, the control signals in this control strategy are obtained separately, the performance is still satisfactory due to the cooperative manner of the sub-controllers by using the local information and the information shared by the other agents via the communication network.

Furthermore, as depicted in Figures 5 and 6, the C-MPC has a similar performance as the proposed NC-DMPC, and both outperform the DeMPC strategy, as shown in Figures 7 and 8. This conclusion is not only valid for this four tank plant, but also for other plants, since the DeMPC strategies does not take the interactions into account, which leads to some degradation in performance. It is noteworthy to mention that the main drawback of decentralized MPC is that local MPC controllers would fail to fully compensate for the process interactions beyond the confines of the local subsystems. Such limitation can degrade the controller performance and possibly, leads to stability problems of the overall system.

The computational burden of the control schemes depends greatly on the number of inequality constraints and the number of control variables [51]. From this point of view, in the unconstrained centralized MPC, the optimization problem

increases significantly with the total number $u^{(i)}$ of system inputs as well as the length of the control horizon N_u , since a large value of the control horizon implies more control variables to be computed. In the proposed NC-DMPC strategy, the local control actions are parameterized by Laguerre functions. The total number of Laguerre functions, namely \mathcal{N} , used in capturing the future control trajectory vector is generally less than N_u . Thereby, each sub-controller needs smaller control variables to be computed, compared to the classical centralized control scheme.

Incidentally, by decomposing the problem into smaller subproblems and solving them in parallel, we can handle a much larger number of units. Furthermore, compared to the classical centralized algorithms, decomposition methods require less memory storage, and are mainly much faster. To sum up, the computational burden of the distributed implementation is much less than the centralized structure, especially for large-scale systems including a large number of subsystems.

8.2 Performance analysis discussion

The performance analysis of the control strategies can be described in terms of output tracking errors and control efforts. Therefore, a more accurate way to obtain a criterion reflecting the performance and differences between the different MPC controllers is to evaluate the following unified performance index,

$$\begin{aligned}
 J_{i(k)} &= \sum_{k=1}^{N^s} \|\hat{y}_{(k)}^{(i)} - y_{(k)}^{\{i\}sp}\|_{Q_i}^2 + \sum_{k=1}^{N^s} \|\Delta u_{(k)}^{(i)}\|_{R_i}^2 \\
 &= \sum_{k=1}^{N^s} J_{e_i(k)} + \sum_{k=1}^{N^s} J_{u_i(k)} \text{ for } i = 1,2.
 \end{aligned} \tag{45}$$

where, N^s represents the total number of samples of the whole simulation, $y^{\{i\}sp}$ denotes the set-point (reference) to be tracked and $y^{(i)}$ is the actual simulated output. The MPC performance index in (45) is formulated with two performance indices, namely the tracking error and the control effort. The performance index corresponding to the tracking error J_{e_i} is formulated using the squared difference between the set-point y^{sp} and the predicted plant behavior \hat{y} . Similar to the tracking error term, the performance index for the control effort J_{u_i} is also formulated as in (45). The control effort is determined by the squared difference between the present and past control actions.

It is worth to recall that one key concept in centralized control is that all the individual performance indices J_i are gathered in a single cost function, leading to:

$$J_{(k)}^{cost} = \sum_{i=1}^{N=2} J_{i(k)} = \sum_{i=1}^{N=2} \left(\sum_{k=1}^{N^s} J_{e_i(k)} + \sum_{k=1}^{N^s} J_{u_i(k)} \right) \tag{46a}$$

$$= \sum_{i=1}^{N=2} \sum_{k=1}^{N^s} J_{e_i(k)} + \sum_{i=1}^{N=2} \sum_{k=1}^{N^s} J_{u_i(k)} \quad (46b)$$

$$J_{(k)}^{cost} = J_{e(k)} + J_{u(k)} \quad (46c)$$

The evaluation results are summarized in Table 2.

Table 2. Comparison of the costs obtained with centralized MPC, distributed MPC and decentralized MPC

Controllers	J_e	J_u	J^{cost}
C-MPC	1.0307	0.3174	1.3481
The proposed distributed MPC	1.0452	0.3682	1.4133
De-MPC	1.2005	0.3234	1.5239

Table 2 shows the results obtained through the evaluation of the performance index (46) for the three MPC controllers. As it can be seen from this table, the same previous conclusion holds according to the obtained numerical values. It shows that decentralized MPC has the largest value of the global performance index J^{cost} , whereas the proposed distributed NC-MPC has a performance close to the centralized scheme. Taking into account each term of the performance index, the proposed DMPC has almost the same performance compared to the C-MPC controller; and both have a better performance related to the tracking error, but the De-MPC and C-MPC controllers have a slightly smaller control efforts compared to the proposed DMPC. Using the proposed NC-DMPC scheme, the tank levels track the set-point signals better than with the controller De-MPC. This is because the distributed MPC controller provides more control effort than decentralized MPC scheme. The obtained results of the above table, with Figures 7 and 8, corroborate this conclusion.

9. CONCLUSION AND FUTURE WORKS

In this paper, a novel distributed MPC approach for large-scale interconnected systems has been derived, leading to a non-cooperation based DMPC (NC-DMPC) algorithm. The main feature of the proposed algorithm lies in the use of orthonormal functions, viz. Laguerre functions, to parametrize the system trajectories, thereby reducing considerably the computational burden of the MPC optimization process under a one-step communication network delay and reducing significantly the set of exchanged information over the network. Moreover, the stability of the resulting plant-wide closed-loop system was analyzed and the performance of the proposed NC-DMPC algorithm was discussed. The approach methodology has been demonstrated on an example for a set-point tracking objective. Through the conducted numerical simulations, the obtained results show clearly that the proposed NC-DMPC scheme achieves similar performance as a centralized model predictive scheme while outperforming the decentralized MPC strategy. Future works may concentrate on extending the proposed methodology in the constrained case. Furthermore, the robustness issues with regards to model uncertainties and external disturbances may be investigated.

In this method the following important features can be emphasized:

- The global optimization problem was transformed into many independent local optimization problems, taking into

account the shared information about the interaction effects of the other subsystems in the local MPC decision.

- A number of distributed control algorithms based on the DMPC approach has been presented. They are designed for large and interconnected systems, described in state-space form and decomposable in several non-overlapping subsystems.

- The reduction of the communication among subsystems.

- A decrease in the computational burden associated with the solution of the local MPC problem.

REFERENCES

- [1] Camponogara, E. (2014). Distributed Model Predictive Control Made Easy. Chapter Distributed Optimization for MPC of Linear Dynamic Networks, Springer, Dordrecht, 193-208. <https://doi.org/10.1007/978-94-007-7006-5>
- [2] Negenborn, R.R., Hellendoorn, H., Lukszo, Z. (2010). Intelligent Infrastructures. Springer.
- [3] Cai, X., Suna, P., Chen, J., Xie, L. (2014). Rapid distributed model predictive control design using singular value decomposition for linear systems. Journal of Process Control, 24(7): 1135-1148. <http://dx.doi.org/10.1016/j.jprocont.2014.05.002>
- [4] Tippet, M.J., Bao, J. (2015). Distributed control of chemical process networks. International Journal of Automation and Computing, 12(4): 368-381. <https://doi.org/10.1007/s11633-015-0895-9>
- [5] Razavinasab, Z., Farsangi, M.M., Barkhordari, M. (2018). Robust output feedback distributed model predictive control of networked systems with communication delays in the presence of disturbance. ISA Transactions, 80: 12-21. <https://doi.org/10.1016/j.isatra.2018.07.003>
- [6] He, Y., Li, S. (2021). Distributed model predictive control with guaranteed performance for reconfigurable power flow systems based on passivity. Asian Journal of Control, 23(4): 1817-1830. <https://doi.org/10.1002/asjc.2333>
- [7] Du, X., Xi, Y., Li, S. (2001). Distributed model predictive control for large-scale systems. In Proceedings of the 2001 American Control Conference, 4: 3142-3143. <https://doi.org/10.1109/ACC.2001.946403>
- [8] Wei, Y., Li, S., Zheng, Y. (2020). Enhanced information reconfiguration for distributed model predictive control for cyber-physical networked systems. International Journal of Robust and Nonlinear Control, 30(1): 198-221. <https://doi.org/10.1002/rnc.4755>
- [9] Hermans, R.M., Jokić, A., Lazar, M., Alessio, A., van den Bosch, P.P., Hiskens, I.A., Bemporad, A. (2012). Assessment of non-centralised model predictive control techniques for electrical power networks. International Journal of Control, 85(8): 1162-1177. <https://doi.org/10.1080/00207179.2012.679972>
- [10] Skjong, E., Johansen, T.A., Molinas, M. (2019). Distributed control architecture for real-time model predictive control for system-level harmonic mitigation in power systems. ISA Transactions, 93: 231-243. <https://doi.org/10.1016/j.isatra.2019.01.043>
- [11] Alessio, A., Barcelli, D., Bemporad, A. (2011). Decentralized model predictive control of dynamically

- coupled linear systems. *Journal of Process Control*, 21(5): 705-714. <https://doi.org/10.1016/j.jprocont.2010.11.003>
- [12] Vaccarini, M., Longhi, S., Katebi, M.R. (2009). Unconstrained networked decentralized model predictive control. *Journal of Process Control*, 19(2): 328-339. <https://doi.org/10.1016/j.jprocont.2008.03.005>
- [13] Li, S., Zheng, Y. (2016). *Distributed model predictive control for plant-wide systems*. John Wiley & Sons.
- [14] Conte, C., Jones, C.N., Morari, M., Zeilinger, M.N. (2016). Distributed synthesis and stability of cooperative distributed model predictive control for linear systems. *Automatica*, 69: 117-125. <https://doi.org/10.1016/j.automatica.2016.02.009>
- [15] Rivero, S., Boem, F., Ferrari-Trecate, G., Parisini, T. (2016). Plug-and-play fault detection and control-reconfiguration for a class of nonlinear large-scale constrained systems. *IEEE Transactions on Automatic Control*, 61(12): 3963-3978. <https://doi.org/10.1109/TAC.2016.2535724>
- [16] Zhang, Y., Xu, C. (2014). Distributed model predictive control with one-step delay communication for large-scale systems and a case study. *Journal of Shanghai Jiaotong University (Science)*, 19(6): 747-754. <https://doi.org/10.1007/s12204-014-1575-6>
- [17] Zhang, R., Gao, F. (2015). An improved decoupling structure based state space mpc design with improved performance. *Systems & Control Letters*, 75: 77-83. <https://doi.org/10.1016/j.sysconle.2014.11.009>
- [18] Bakule, L. (2008). Decentralized control: An overview. *Annual Reviews in Control*, 32(1): 87-98. <https://doi.org/10.1016/j.arcontrol.2008.03.004>
- [19] Richards, A., How, J.P. (2007). Robust distributed model predictive control. *International Journal of Control*, 80(9): 1517-1531. <https://doi.org/10.1080/00207170701491070>
- [20] Grüne, L., Pannek, J. (2011). *Nonlinear Model Predictive Control: Theory and Algorithms*. Springer. <https://doi.org/10.1007/978-0-85729-501-9>
- [21] Kouvaritakis, B., Cannon, M. (2016). *Model predictive control: classical, robust and stochastic*. *Advanced Textbooks in Control and Signal Processing*- Springer. <https://doi.org/10.1007/978-3-319-24853-0>
- [22] Menighed, K., Yamé, J.J., Aubrun, C., Boussaid, B. (2011). Fault tolerant cooperative control: A distributed model predictive control approach. In *19th Mediterranean Conference on Control and Automation (MED)*, pp. 1094-1099. <https://doi.org/10.1109/MED.2011.5983199>
- [23] Vaccarini, M., Longhi, S., Katebi, M.R. (2006). Stability analysis tool for tuning unconstrained decentralized model predictive controllers. In *Proceedings of the International Conference Control, Glasgow, Scotland, UK*, pp. 159-164.
- [24] Rawlings, J.B., Mayne, D.Q., Diehl, M. (2017). *Model Predictive Control: Theory, Computation, and Design*. Nob Hill.
- [25] Venkat, A.N., Rawlings, J.B., Wright, S.J. (2006a). Stability and optimality of distributed, linear model predictive control. part i: State feedback. Technical report, 2006-03, TWMCC, Department of Chemical Engineering, University of Wisconsin-Madison.
- [26] Venkat, A.N., Rawlings, J.B., Wright, S.J. (2006b). Stability and optimality of distributed, linear model predictive control. part ii: Output feedback. Technical report, 2006-03, TWMCC, Department of Chemical Engineering, University of Wisconsin-Madison.
- [27] Morosan, P.D., Bourdais, R., Dumur, D., Buisson, J. (2010). Building temperature regulation using a distributed model predictive control. *Energy and Buildings*, 42(9): 1445-1452. <https://doi.org/10.1016/j.enbuild.2010.03.014>
- [28] Farina, M., Scattolini, R. (2012). Distributed predictive control: A non-cooperative algorithm with neighbor-to-neighbor communication for linear systems. *Automatica*, 48(6): 1088-1096. <https://doi.org/10.1016/j.automatica.2012.03.020>
- [29] Zhang, R., Liu, A., Yu, L., Zhang, W. (2015). Distributed model predictive control based on nash optimality for large scale irrigation systems. *International Federation of Automatic Control*, 48(8): 551-555. <https://doi.org/10.1016/j.ifacol.2015.09.025>
- [30] Camponogara, E., Jia, D., Krogh, B., Talukdar, S. (2002). Distributed model predictive control. *IEEE Control Systems Magazine*, 22(1): 44-52. <https://doi.org/10.1109/37.980246>
- [31] Zhang, L., Xie, W., Wang, J. (2017). Robust distributed model predictive control of linear systems with structured time-varying uncertainties. *International Journal of Control*, 90(11): 2449-2460. <https://doi.org/10.1080/00207179.2016.1250163>
- [32] Shalmani, A.R., Rahmani, M., Bigdeli, N. (2020). Nash-based robust distributed model predictive control for large-scale systems. *Journal of Process Control*, 88: 43-53. <https://doi.org/10.1016/j.jprocont.2020.02.005>
- [33] Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control-a review. *Journal of Process Control*, 19(5): 723-731. <https://doi.org/10.1016/j.jprocont.2009.02.003>
- [34] Müller, M.A., Allgöwer, F. (2017). Economic and distributed model predictive control: Recent developments in optimization-based control. *SICE Journal of Control, Measurement, and System Integration*, 10(2): 39-52. <https://doi.org/10.9746/jcmsi.10.39>
- [35] Yang, T., Yi, X., Wu, J., Yuan, Y., Wu, D., Meng, Z., Hong, Y., Wang, H., Lin, Z., Johansson, K.H. (2019). A survey of distributed optimization. *Annual Reviews in Control*, 47: 278-305. <https://doi.org/10.1016/j.arcontrol.2019.05.006>
- [36] Negenborn, R.R., De Schutter, B., Hellendoorn, J. (2009). Multi-agent model predictive control: A survey. Technical report, Technical report 04-010.eprintarXiv:0908.1076. <https://doi.org/10.48550/arXiv.0908.1076>
- [37] Maciejowski, J.M. (2000). Robust finite horizon mpc without terminal constraints. In *Proceedings of the 39th IEEE Conference on Decision and Control*, pp. 166-171. <https://doi.org/10.1109/CDC.2000.912751>
- [38] Camacho, E.F., Bordons, C. (2007). *Model predictive control*. Springer-Verlag, London. <http://dx.doi.org/10.1007/978-0-85729-398-5>
- [39] Rossiter, J.A. (2007). *Model-Based Predictive Control: A Practical Approach*. Springer Verlag. <https://doi.org/10.1201/9781315272610>
- [40] Morari, M., Lee, J.H. (1999). *Model predictive control: Past, present and future*. *Computers and Chemical Engineering*, 23(4-5): 667-682. [https://doi.org/10.1016/S0098-1354\(98\)00301-9](https://doi.org/10.1016/S0098-1354(98)00301-9)

- [41] Ben Abdennour, R. (2001). Identification et commande numérique des procédés industriels. Editions Technip.
- [42] Abdullah, M., Rossiter, J.A. (2021). Using laguerre functions to improve the tuning and performance of predictive functional control. *International Journal of Control*, 94(1): 202-214. <https://doi.org/10.1080/00207179.2019.1589650>
- [43] Wang, L. (2009). Model predictive control system design and implementation using MATLAB, *Advances in Industrial Control*. Springer. <https://doi.org/10.1007/978-1-84882-331-0>
- [44] Zheng, Y., Zhou, J., Xu, Y., Zhang, Y., Qian, Z. (2017). A distributed model predictive control based load frequency control scheme for multi-area interconnected power system using discrete-time laguerre functions. *ISA Transactions*, 68: 127-140. <https://doi.org/10.1016/j.isatra.2017.03.009>
- [45] Atam, E. (2017). Investigation of the computational speed of laguerre network-based mpc in the thermal control of energy-efficient buildings. *Turkish Journal of Electrical Engineering & Computer Sciences*, 25(5): 4369-4380. <https://doi.org/10.3906/elk-1702-244>
- [46] Dubravić, A., Šehić, Z., Burgić, M. (2014). Orthonormal functions based model predictive control of ph neutralization process. *Tehnički vjesnik*, 21(6): 1249-1253.
- [47] Menighed, K., Chemori, A., Boussaid, B. Yamé, J.J. (2021). Networked cooperation based distributed model predictive control using Laguerre functions for large-scale systems. *Proceedings of the 4th International Conference on Electrical Engineering and Control Applications*. ICEECA 2019. Lecture Notes in Electrical Engineering, Springer, Singapore, pp. 123-138. <https://doi.org/10.1007/978-981-15-6403-1>
- [48] Farina, M., Betti, G., Giulioni, L., Scattolini, R. (2014). An approach to distributed predictive control for tracking-theory and applications. *IEEE Transactions on Control Systems Technology*, 22(4): 1558-1566. <https://doi.org/10.1109/TCST.2013.2288409>
- [49] Johansson, K.H. (2000). The quadruple-tank process: A multivariable laboratory process with an adjustable zero. *IEEE Transactions On Control Systems Technology*, 8(3): 456-465. <https://doi.org/10.1109/87.845876>
- [50] Mercangoz, M., Doyle III, F.J. (2007). Distributed model predictive control of an experimental four-tank system. *Journal of Process Control*, 17(3): 297-308. <https://doi.org/10.1016/j.jprocont.2006.11.003>
- [51] Maxim, A., Copot, D., De Keyser, R., Ionescu, C.M. (2018). An industrially relevant formulation of a distributed model predictive control algorithm based on minimal process information. *Journal of Process Control*, 68: 240-253. <https://doi.org/10.1016/j.jprocont.2018.06.004>

APPENDIX

Proof of Theorem 1

Merging the process equation, the global prediction equation and the global optimal sequence, the closed-loop state-space model for the distributed control framework is obtained:

$$x_{(k+1)} = Ax_{(k)} + Bu_{(k)} \quad (A1a)$$

$$\begin{aligned} x_{(k)} &= Ax_{(k-1)} + Bu_{(k-1)} \\ &= Ax_{(k-1)} + B\Gamma U_{(k-1, N_p|k-1)} \eta_{(k)} \\ &= \bar{K}[Y_{(k+1, N_p|k)}^{SP} - L\hat{x}_{(k)}] \\ &\quad - Mu_{(k-1)} - S\hat{V}_{(k, N_p|k-1)} \\ &\quad - T\hat{W}_{(k, N_p|k-1)}] \end{aligned} \quad (A1b)$$

$$\begin{aligned} \eta_{(k)} &= \bar{K}[Y_{(k+1, N_p|k)}^{SP} - L\hat{x}_{(k)} \\ &\quad - M\Gamma U_{(k-1, N_p|k-1)} \\ &\quad - S\hat{V}_{(k, N_p|k-1)} - T\hat{W}_{(k, N_p|k-1)}] \end{aligned} \quad (A1c)$$

Therefore, the global interaction predictions can be expressed as follows:

$$\hat{V}_{(k, N_p|k-1)} = \tilde{A}\hat{X}_{(k, N_p|k-1)} + \tilde{B}\hat{\Gamma}U_{(k-1, N_p|k-1)} \quad (A1d)$$

$$\hat{W}_{(k, N_p|k-1)} = \tilde{C}\hat{X}_{(k, N_p|k-1)} \quad (A1e)$$

$$\begin{aligned} \eta_{(k)} &= \theta x_{(k)} + \phi \hat{X}_{(k, N_p|k-1)} + \rho U_{(k-1, N_p|k-1)} \\ &\quad + \bar{K}Y_{(k+1, N_p|k)}^{SP} \end{aligned} \quad (A1f)$$

$$\begin{aligned} \text{with } \theta &= -\bar{K}L, \quad \rho = -\bar{K}(M\Gamma + S\tilde{B}\hat{\Gamma}), \\ \phi &= -\bar{K}(S\tilde{A} + T\tilde{C}) \end{aligned}$$

$$\begin{aligned} \hat{X}_{(k+1, N_p|k)} &= \bar{L}\hat{x}_{(k)} + \bar{M}u_{(k-1)} + \bar{N}\eta_{(k)} \\ &\quad + \bar{S}\hat{V}_{(k, N_p|k-1)} \end{aligned} \quad (A1g)$$

$$\begin{aligned} \hat{X}_{(k+1, N_p|k)} &= \bar{L}\hat{x}_{(k)} + \bar{M}\Gamma U_{(k-1, N_p|k-1)} + \bar{N}\eta_{(k)} \\ &\quad + \bar{S}\hat{V}_{(k, N_p|k-1)} \end{aligned} \quad (A1h)$$

$$\begin{aligned} \hat{X}_{(k+1, N_p|k)} &= \bar{L}\hat{x}_{(k)} + \bar{S}\tilde{A}\hat{X}_{(k, N_p|k-1)} \\ &\quad + \Upsilon U_{(k-1, N_p|k-1)} + \bar{N}\eta_{(k)} \end{aligned} \quad (A1i)$$

$$\begin{aligned} \hat{X}_{(k, N_p|k-1)} &= \Sigma \hat{x}_{(k-1)} + \Pi \hat{X}_{(k-1, p|k-2)} \\ &\quad + F U_{(k-2, N_p|k-2)} \\ &\quad + \bar{N} \bar{K} Y_{(k, N_p|k-1)}^{SP} \end{aligned} \quad (A1j)$$

$$\text{with } \Sigma = \bar{L} + \bar{N}\theta, \quad \Pi = \bar{S}\tilde{A} + \bar{N}\phi, \quad F = \Upsilon + \bar{N}\rho,$$

$$\Upsilon = \bar{M}\Gamma + \bar{S}\tilde{B}\hat{\Gamma} \quad (A1k)$$

$$U_{(k, N_p|k)} = \Gamma' \Gamma U_{(k-1, N_p|k-1)} + \bar{\Gamma} \Delta U_{(k, N_p|k)} \quad (A1l)$$

$$U_{(k, N_p|k)} = \Gamma' \Gamma U_{(k-1, N_p|k-1)} + \bar{\Gamma} Z \eta_{(k)} \quad (A1m)$$

$$\begin{aligned} U_{(k, N_p|k)} &= \bar{\Gamma} Z \theta x_{(k)} + \bar{\Gamma} Z \phi \hat{X}_{(k, N_p|k-1)} + (\Gamma' \Gamma \\ &\quad + \bar{\Gamma} Z \rho) U_{(k-1, N_p|k-1)} \\ &\quad + \bar{\Gamma} Z \bar{K} Y_{(k+1, N_p|k)}^{SP} \end{aligned} \quad (A1n)$$

$$\begin{aligned} U_{(k, N_p|k)} &= [\bar{\Gamma} Z \theta A + \bar{\Gamma} Z \phi \Sigma] x_{(k-1)} \\ &\quad + \bar{\Gamma} Z \phi \Pi \hat{X}_{(k-1, p|k-2)} \end{aligned} \quad (A1o)$$

$$+(\bar{\Gamma}Z\theta B\Gamma + \Gamma'\Gamma + \bar{\Gamma}Z\rho)U_{(k-1, N_p|k-1)} + \bar{\Gamma}Z\phi FU_{(k-2, N_p|k-2)} + \beta Y_{(k, N_p|k-1)}^{sp}$$

$$\text{with } \beta = \bar{\Gamma}Z(\phi\bar{N}\bar{K} + \bar{K}\hat{\Gamma})$$

$$y_{(k)} = Cx_{(k)} \quad (\text{A1p})$$

Let us now define the augmented state vector as follows:

$$X_{G(k)} = [x_{(k)}^T, \hat{X}_{(k, N_p|k-1)}^T, U_{(k, N_p|k)}^T, U_{(k-1, N_p|k-1)}^T]^T$$

The state-space model of the whole closed-loop system is then given by:

$$S_G = \begin{cases} X_{G(k)} & = A_G X_{G(k-1)} + B_G Y_{(k, N_p|k-1)}^{sp} \\ y_{(k)} & = C_G X_{G(k)} \end{cases} \quad (\text{A1q})$$

where,

$$A_G = \begin{bmatrix} A & 0 & B\Gamma & 0 \\ \Sigma & \Pi & 0 & F \\ (\bar{\Gamma}Z\theta A + \bar{\Gamma}Z\phi\Sigma) & \bar{\Gamma}Z\phi\Pi & (\bar{\Gamma}Z\theta B\Gamma + \Gamma'\Gamma + \bar{\Gamma}Z\rho) & \bar{\Gamma}Z\phi F \\ 0 & 0 & I_{N_p, n_u} & 0 \end{bmatrix},$$

$$B_G = \begin{bmatrix} 0 \\ 0 \\ \beta \\ 0 \end{bmatrix}, C_G = [I \ 0 \ 0 \ 0]$$

Consequently, the stability of the overall system is determined by the eigenvalues of the dynamic matrix A_G .