AUT Journal of Modeling and Simulation

AUT J. Model. Simul., 55(1) (2023) 127-138
DOI: 10.22060/miscj.2023.22087.5312

Robust Distributed $\mathcal{L}$asso-Model Predictive Control Design: A Case Study on Large-Scale Multi-Robot Systems

H. Ahmadian, I. Sharifi*, H. A. Talebi

Department of Electrical Engineering, Amirkabir University of Technology, Tehran, Iran

ABSTRACT: The complexity and dynamic order of large-scale systems is continuously increasing. Considering the many challenges that exist for these systems, it is very important to provide a robust distributed controller that performs well against uncertainties, computation volume, and interaction between subsystems. A robust-distributed $\mathcal{L}$asso-MPC (RD-LMPC) approach is suggested in this study for multi-robot systems in the presence of polytopic uncertainty. In addition, a distributed Kalman filter is used to capture interactions between subsystems. To evaluate and perform the effectiveness of the suggested approach, the results obtained on the multi-robot system are compared with the results of the predictive control methods of the centralized, distributed model, and $\mathcal{L}$1 adaptive control.

Review History:

Received: Jan. 11, 2023
Revised: Jul. 11, 2023
Accepted: Sep. 26, 2023
Available Online: Oct. 10, 2023

Keywords:

Distributed MPC
Large Scale Multi-Robot Systems
$\mathcal{L}$asso Regression
$\mathcal{L}$asso- MPC
Model Predictive Control (MPC)
Robust MPC

1- Preliminaries

1- 1- Key Concepts

Concept I Centralized Control System (CCS): The system is considered as a single system and it is assumed that all the global information of the system is available and there is only one central controller.

Concept II Decentralized Control System (DeCS): The system is divided into several subsystems and a separate control is considered for each subsystem.

Concept III Hierarchical Control System (HCS): In this structure, the system is divided into several subsystems and the control is done in a multi-level manner.

Concept IV Distributed Control System (DCS): This structure is similar to decentralized control with the difference that subsystems interact and exchange information.

Concept V Lasso Regression (LR): LR is a method of model adjustment to prevent preprocessing in regression.

1- 2- Introduction

The MPC has absorbed a lot of attention in both applied and theoretical areas during the recent decades. It has been utilized successfully in different industrial operations [1]. For further information, see [2] and the references therein.

In recent years, system size and complexity have increased, making centralized control approaches increasingly challenging, time-consuming, and impracticable. During centralized control, all system parameters are gathered in one control unit, in which calculations are made to ensure the best operation all around. The method may frequently fail when a large-scale system is actually scattered in this way [3]. Furthermore, if a certain process step had a challenge or failed, the entire system would be shut down.

To prevent issues related to centralized control and decrease computational complexity, different algorithms such as distributed, hierarchical, and decentralized ones have been presented throughout the years. These approaches show that a large-scale system is composed of many smaller systems, each with a local controller. Several strategies have been suggested to divide the initial framework into connected components with low coupling; for example, see [4, 5]. In a decentralized approach, each component is managed independently, whereas other components' impacts are disregarded or seen as model defects or disruptions. There are several decentralized MPC methods for coupled systems published in the literature. Alessio et al. looked at the level of sub-model decoupling in [6] and how it affected the computational load and overall system performance while taking input and output constraints into account. Vaccarini et al. published a decentralized strategy for systems of fast networks in [7], which, despite the subsystems' tight dependence, ensures system stability. This approach's states, output, and input are unrestricted. [8, 9] also discusses stabilizing decentralized MPC for nonlinear

*Corresponding author's email: imansharifi@aut.ac.ir

systems.

A control subsystem is taken into consideration for each subsystem in distributed techniques. Each agent has the ability to interact with others. If the information exchange happened more than once throughout the sample period, the process would be iterative. Cooperative and Nash-based model predictive control, for example [10-13], has been suggested via the use of iterative information sharing and a range of cost functions. Using both local and transmitted information, a cooperative approach solves the overall optimization problem by assigning a common objective function to all agents that account for all system variables. In this method, agents must be completely connected to one another and share all information [10]. In a Nash-based algorithm, the Nash optimum solution is found by constantly sharing information between each agent, which has an objective function that depends on local variables [10, 11]. The distributed MPC suggested in [13] achieves the centralized MPC optimum point. It does this by working together and resolving issues in stages. Maestre et al. provided an MPC approach with two stages of solution based on game theory [14]. Each controller in the first phase resolves its local optimization issue and communicates the results to the other controllers. To ensure optimum system performance, subsystems select the most favorable suboptimal alternative in the second step. Following the introduction of a decomposition approach for large-scale systems in [15], Camponogara et al. presented an MPC approach with distributed communication. Liu and Christofides [16] also suggested a distributed MPC based on Lyapunov that keeps the closed-loop stability of the original system even when shocks happen.

Numerous robust distributed MPC (RD-MPC) strategies have also been developed for uncertain large-scale systems [17, 18, 19]. The problem's stability and viability of all agents are provided while taking the necessity for robustness with external disturbance into consideration, as described in [20], which describes a RD-MPC strategy for systems of nonlinear. In [21], Shi and Li concentrated on creating RD-MPC for constrained nonlinear systems with communication delays. Based on the method described in [17], Al-Gherwi et al. [22] suggested a RD-MPC where agents' total objective function minimizes and gives feedback control to the related subsystem. In this case, all system data should be available to every agent. It is usually not possible to construct a structure that fully describes the system, and even if it were, the computations of the method would be very expensive and time-consuming. These shortcomings prompt us to develop a unique distributed MPC methodology that is equivalent to the centralized method in terms of optimality, performance, and cost-effectiveness.

In this work, a novel robust-distributed $\mathcal{L}$asso-MPC algorithm (called RD-MPC) is suggested for LTV systems in the presence of polytopic uncertainty. There are M state-coupled subsystems in the system. A RD-LMPC has been made to decrease the over limit on the worst-case value of the cost function inside the polytope uncertainty. This is done by taking into account the polytope uncertainty. This method views the control rule of each agent as a distinct state feedback form and feedforward interaction. This control input both assures the closed-loop system's quadratic bound stability and reduces the undesirable impacts of surrounding subsystems. For the suggested LMPC to work, a *distributed Kalman filter* is also made to show predictors and interactions for each agent. This filter takes advantage of local data as well as network-provided measures from surrounding subsystems. In order to assess the efficacy of the proposed approach, it was compared to centralized MPC, distributed MPC, and $\mathcal{L}_1$ adaptive control techniques on a multi-robot system.

In light of the above debate, the contribution of the paper is as follows:

1. To provide a distributed system global structure with connections and polytopic uncertainty.
2. To propose a robust distributed LMPC based on LMI for large-scale systems under uncertainty.
3. The distributed LMPC's control law to directly consist of the interaction signals.
4. To solve a multi-robot system using the offered method in order to assess the utility and viability of the solution.

2- RD $\mathcal{L}$asso Model Predictive Control

2- 1- Problem Statement

Consider the following linear time-variant system [23]:

$$\begin{aligned} &x(k+1)=A(k)x(k)+B(k)u(k)+E(k)w(k),\\ &y(k)=C(k)x(k)+v(k)\\ &[A(k),B(k)]\in\Omega \end{aligned} \tag{1}$$

where, $x\in\mathbb{R}^{n_x}$, $y\in\mathbb{R}^{n_y}$, $u\in\mathbb{R}^{n_u}$ represent the state, output, and control input vectors, respectively. Measurement and process noises are $w\in\mathbb{R}^{n_w}, v\in\mathbb{R}^{n_v}$, respectively. The polytope Ù has the discrete-time index k defined as follows:

$$\Omega=\mathrm{Co}\{[A^1,B^1],[A^2,B^2],\dots,[A^L,B^L]\} \tag{2}$$

here $\mathrm{Co}\{\}$ denotes the convex hull. A polytope called Any $[A(k)B(k)]$ may have its vertices merged in the following ways [23]:

$$[A(k)B(k)]=\sum_{l=1}^{L}\lambda^l[A^lB^l],\ \sum_{l=1}^{L}\lambda^l=1. \tag{3}$$

In this work, the multi-robot system is divided into M agents using the same method as in [24]. In this case, the created subsystems are connected and the agent model i -th may be represented as follows, where $i,j=1,\dots,M; i\neq j$.

$$x_i(k+1) = A_{ii}(k)x_i(k) + B_i(k)u_i(k) + E_i(k)w_i(k) + \sum_{j=1,j\neq i}^{M} A_{ij}(k)x_j(k), = A_{ii}(k)x_i(k) + B_i(k)u_i(k) + E_i(k)w_i(k) + D_i(k)z_i(k) \quad (4)$$

The neighboring subsystem states that affect on the i -th agent are blended linearly to provide the interaction signals ($z_i(k)$, $w_i \in \mathbb{R}^{n_{w_i}}$, $u_i \in \mathbb{R}^{n_{u_i}}$, and $x_i \in \mathbb{R}^{n_{x_i}}$):

$$z_i(k) = \left[x_1^T(k) \ldots x_{i-1}^T(k) x_{i+1}^T(k) \ldots x_M^T(k)\right]^T, \quad D_i = \left[A_{i1} \ldots A_{i,i-1} A_{i,i+1} \ldots A_{iM}\right] \quad (5)$$

Be aware that the j -th agent is a neighbor of the i -th agent when $A_{ij} \neq 0$. The i -th agent is thought to include the following polytopic uncertainties:

$$\left[A_{ii}(k), A_{ij}(k), B_i(k)\right] \in \Omega_i = \sum_{l=1}^{L} \lambda_i^l \left[A_{ii}^l, A_{ij}^l, B_i^l\right], \quad i = 1,\ldots,M; j = 1,\ldots,M; j \neq i, \quad (6)$$

When a *distributed Kalman filter* (DKF) is used to anticipate the states and interactions that result from distinct subsystems, it is expected that a local control subsystem would give the proper control input for the concerned agent. To estimate the missing variables, the DKF employs both the local data and the data supplied by other subsystems across the network.

Additionally, the following constraint likely applies to the control signal of the i -th subsystem:

$$\left|u_i^h(k+n \mid k)\right| \leq u_{i,\max}^h, h = 1,2,\ldots,n_{u_i} \quad (7)$$

In the predictive control of the robust distributed model, with an infinite horizon, each control agent seeks to reduce the overbound on the worst-case value of the cost function. The minimization problem is fixed at each iteration, and the demands of the problem are satisfied. The following is the prediction model for the i -th agent that is utilized to solve the LMPC issue since an estimator makes the state values accessible [20]:

$$\hat{x}_i(k+n+1 \mid k) = A_{ii}(k+n)\hat{x}_i(k+n \mid k) + B_i(k+n)u_i(k+n \mid k) + D_i(k+n)\hat{z}_i(k+n \mid k) \quad (8)$$

in which $\hat{z}_i(k+n|k)$ and $\hat{x}_i(k+n|k)$ represent, respectively, the vector of surrounding states and the estimate of $x_i(k+n)$ in time step k, both of which are given to agent i by the associated estimator. Agent i 's control signal is consider as follows:

$$u_i(k+l \mid k) = F_i\hat{x}_i(k+n \mid k) + K_i\hat{z}_i(k+n \mid k) \quad (9)$$

where K_i is the online-calculated feed-forward gain of the interactive effect and F_i is the feedback output gain. Each factor additionally specifies the input condition as follows:

$$|u_i^h(k+n \mid k)| \leq u_{(i,max)}^h, h = 1,2,\ldots,n_{u_i}. \quad (10)$$

where $u_{(i,max)}^h$ is the upper limit on the h -th of the input of $u_i^h(k)$.

2- 2- Objective Function

Definition 1.: $\mathcal{L}asso - \mathcal{MPC}$

The model predictive control problem based on the $\mathcal{L}asso$ regression theory is expressed as follows:

$$V_N^o(\tilde{q}) = \min_{\underline{\tau_a}} \left\{ V_N(\tilde{q}, \underline{\tau_a}) \triangleq F(\tilde{q}_N) + \sum_{j=0}^{N-1} \ell(\tilde{q}_j, \tau_{aj}) \right\} \quad (11)$$
$$s.t.: \quad \tau_{aj} \in \mathbb{U},\ \tilde{q}_j \in \mathbb{X},\ j = 0,\ldots,N-1$$

with

$$\ell(\tilde{q}_j, \tau_{aj}) = \tilde{q}_j^T Q \tilde{q}_j + \tau_{aj}^T R \tau_{aj} + \|S\tau_{aj}\|_1, \quad (12)$$

with $\underline{\tau_a}^T = \left[\tau_{a_0}^T \ \cdots \ \tau_{a_{N-1}}^T\right]$, and S is a constant matrix. The $\mathcal{L}asso - \mathcal{MPC}$ executes the first action of the optimum policy, $\tau_a(k) = \tau_{a_0}^{\star}$, at the current state, $q = q(k)$, at each iteration k. This action is determined by the online solution of (10), (12). The obtained implicit control law is also considered as $K_N(\tilde{q}) \equiv \tau_{a_0}^{\star}$ [22].

In general, LMPC is a technique for *allocating control* while minimizing the goal function in accordance with limitations on the system states, output, and input. Since the system being studied is uncertain, the robust LMPC technique changes the challenge of reducing the cost function into a problem of minimizing or maximizing. It establishes the upper limit for the worst possible value of the minimal objective function. In this case, the max-min problem for an agent is phrased as follows [23]:

$$\min_{u_i(k+l|k)} \max_{[A_{ii}(k),B_i(k),A_{ij}(k)]\in\Omega_i} J_i(k)$$
$$\text{subject to}$$
$$u_i(k+l\mid k) = F_i\hat{x}_i(k+n\mid k) + K_i\hat{z}_i(k+n\mid k) \tag{13}$$
$$\left|u_i^h(k+n|k)\right| \le u_{(i,max)}^h, h = 1,2,\ldots,n_{u_i}.$$

The cost function for the i -th agent's infinite horizon is written as follows:

$$J(k) = \sum_{i=1}^{M} J_i(k),$$
$$J_i(k) = \sum_{n=0}^{\infty} \left\{ \begin{array}{c} \hat{x}(k+n|k)^T M_i \hat{x}(k+n|k) \\ +u_i(k+n|k)^T R_i u_i(k+n|k) \end{array} \right\} \tag{14}$$
$$+\|Su_i(k+n|k)\|_1.$$

The worst-case scenario indicates that for each of the multiple vertices, the outcomes of each vertex's prediction are different due to the proper weight functions M_i and R_i. We will therefore have various cost-function values. The cost function may thus determine the highest value by setting the values of one of the polyhedron's vertices. As a result, the minimization issue, which will be described in further detail later, has been resolved for all of the polyhedron's vertices. Reduce the largest's upper limit as well.

2- 3- Proposed Control Algorithm

Consider $V\left(\hat{x}_i\right) = \hat{x}_i^T P_i \hat{x}_i$ as the Lyapunov function. Based on the theorem of quadratic constraint in [25, 26], P_i is a positive definite matrix that fulfills the following conditions of quadratic constraint for $\left[A_i i\left(k\right), B_i\left(k\right), A_i j\left(k\right)\right] \in \Omega_i$:

$$V_i\big(\hat{x}_i(k+n|k)\big) = \hat{x}_i(k+n|k)^T P_i \hat{x}_i(k+n|k) \le \gamma_i \tag{15}$$

which, under the requirements of the quadratic limitation, can be written:

$$V_i(\hat{x}_i(k+n+1|k)) - V_i(\hat{x}_i(k+n|k)) \le -\hat{x}_i(k+n|k)^T M_i \hat{x}_i(k+n|k) +u_i(k+n|k)^T R_i u_i(k+n|k) \tag{16}$$

In actuality, the quadratic restriction for each subsystem is weaker than the aforementioned formula. To prove the robust condition's stability, $x(\infty\,|\,k) = 0$, which is identical to $V_i(x_i(\infty\,|\,k))$, must hold. The two previously stated sides of the inequality will now add up to:

$$\sum_{n=0}^{\infty} V_i(\hat{x}_i(k+n+1|k)) - V_i(\hat{x}_i(k+n|k)) \le -\sum_{n=0}^{\infty} \{\hat{x}_i(k+n|k)^T M_i \hat{x}_i(k+n|k) + u_i(k+n|k)^T R_i u_i(k+n|k)\} \tag{17}$$
$$V_i(\hat{x}_i(k|k)) = \hat{x}_i^T(k|k) P_i \hat{x}_i(k|k) \le \gamma_i$$

to which it is equal:

$$\max_{[A_i i(k),B_i(k),A_i j(k)]\in\Omega_i} J_i(k) \le V_i(\hat{x}_i(k\mid k)) \le \gamma_i \tag{18}$$

γ_i can be expressed as follows and is a non-negative suitable variable that needs to be minimized:

$$\min_{\gamma_i,K_i,Q_i} \gamma_i \tag{19}$$

Calculating the control input (9), which is the outcome of reducing the upper band on $V_i(x_i(k\,|\,k))$, is the robust LMPC algorithm's goal. Therefore, the first computed input, $u_i(k\,|\,k)$, is applied to the system to calculate the input with an infinite horizon. The estimate of $x_i\left(k+1\right)$ is acquired for the subsequent time sample, and the optimization is carried out once again to determine K_i and F_i. The following theorem lays forth the prerequisites for P_i being appropriate in the context of research, together with interaction effect feedback K_i and output feedback F_i for agent i.

Theorem 1. The minimization problem for the LTI subsystem (1) must be solved in accordance with the input constraint (10), optimization problem (19), feedforward K_i, and feedback gain F_i for $n \ge 0$ in each time sample k. Then The following LMI are obtained:

$$\min_{Y_i,K_i,Q_i} \gamma_i$$
$$Subject\ to$$
$$\begin{bmatrix} 1 & \hat{x}_i^T(k \mid k) \\ \hat{x}_i(k \mid k) & Q_i \end{bmatrix} \geq 0$$
$$\begin{bmatrix} (1-\alpha_i\beta_{i,k})Q_i & 0 & (V_i^l)^T & (R_i^{\frac{1}{2}}Y_i)^T & (M_i^{\frac{1}{2}}Q_i)^T \\ 0 & \alpha_i I & (N_i^l)^T & (R_i^{\frac{1}{2}}K_i)^T & 0 \\ V_i^l & N_i^l & Q_i & 0 & 0 \\ R_i^{\frac{1}{2}}Y_i & R_i^{\frac{1}{2}}K_i & 0 & \gamma_i I & 0 \\ M_i^{\frac{1}{2}}Q_i & 0 & 0 & 0 & \gamma_i I \end{bmatrix} \tag{20}$$
$$\geq 0, l \in \{1,\ldots,L\}$$
$$\begin{bmatrix} \rho_i & \sqrt{2}Y_i & \sqrt{2}K_i \\ \sqrt{2}Y_i^T & Q_i & 0 \\ \sqrt{2}K_i^T & 0 & \beta_{i,k}^{-\frac{1}{2}}I \end{bmatrix} \geq 0,$$
$$\rho_i = \mathrm{diag}\left[(u_{i,\max}^1)^2,(u_{i,\max}^2)^2,\ldots,(u_{i,\max}^n)^2\right]$$

where $V_i^l = A_{ii}^l Q_i + B_i^l Y_i$ and $N_i^l = D_i^l + B_i^l K_i$.

2- 4- Feasibility Conditions

Now, to claim robust stability of the problem, we need to prove fixed elliptic and feasibility theorems for the proposed algorithm. First, to prove the constant elliptic theorem of the proposed algorithm, since the quadratic constraint condition is used, the result of establishing the quadratic constraint is the following theorem, which is proved in [27] and finally it is used in the constant elliptic theorem for the proposed control algorithm.

Theorem 2. [28] If (15) is quadratic bounded for the Lyapunov matrix P, then it can be said: The ellipse $\varepsilon_p \equiv \{z \in R^n : z^T Pz \leq 1\}$ is a robust positive fixed set for the (15).

Now for system (4) using the above theorem, the following theorem can be expressed.

Theorem 3. It considers the time-varying linear system (4) and Ω_i is an indefinite set for it. Suppose that the optimization problem (20) has a solution from which Q_i, Y_i and K_i are determined. Then, for all future inputs obtained from equation (9) and $P_i = \gamma_i Q_i^{-1}$, it can be concluded that:

$$\max_{\left[A_{il}(k+n),B_i(k+n),A_{ij}(k+n)\right]\in\Omega_i} \hat{x}_i^T \times (k+n|k)P_i\hat{x}_i(k+n|k) \leq \gamma_i \tag{21}$$

Therefore, $\varepsilon_i \equiv \{z_i \in R^{n_i} : z_i^T Pz_i \leq \gamma_i\}$ is a fixed ellipse for all future states of the uncertain system (4) obtained from equation (9) and if the optimization problem (20) has a feasible solution at sampling time k, then this solution is feasible for all times t > k.

2- 5- Robust Stability Check

Theorem 4. *(Robust Stability):* The feasible answer obtained from the optimization problem (20) guarantees asymptotic stability: if for $k \to \infty$, $x_i(k) \to 0$ is established.

Proof. According to the feasibility of the problem, at sampling time $k+1$, one of the possible answers for γ_{i+1} is γ_i, but by solving the optimization problem at time $k+1$, the statement $\gamma_{i+1} \geq \gamma_i$ is fulfilled and γ_i is the upper limit of $x_i(k|k)^T P_{i,k} x_i|(k|k)$, which is uniformly does not decrease but for large k, γ_i becomes less than a certain value, which means that $x_i(k)$ and $u_i(k)$ converge to the neighborhood of zero and remain in that region. It should be noted that [27]. is fulfilled for very small Q_i and γ_i, and as a result, $x_i(k+1)$ is placed in a small ellipse, and since $x_i(k) \to x_i(k)$ is for $k \to \infty$, the result is $x_i(k) \to 0$.

In the $\mathcal{L}$*asso-model predictive control of robust distribution*, each control agent satisfies the following condition by solving the optimization problem and due to robust stability.

$$\hat{x}_i(k+1 \mid k+1)^T P_i \hat{x}_i(k+1 \mid k+1) \leq \hat{x}_i(k \mid k)^T P_i \hat{x}_i(k \mid k) \tag{22}$$

which implies

$$\|\hat{x}_i(k+1 \mid k+1)\| \leq \|\hat{x}_i(k \mid k)\| \tag{23}$$

By establishing the above relationship in each agent, it can be claimed that the interactive effect vector in the i-th subsystem will be as follows:

$$\|\hat{z}_i(k+1 \mid k+1)\| \leq \|\hat{z}_i(k \mid k)\| \tag{24}$$

3- Distributed Kalman Filter

The dynamics of distributed subsystems are different from a simple system, so for estimation, it is necessary to design a filter that can estimate the states of the subsystem according to the presence of the effect of neighboring subsystems. In this way, the Kalman filter needs to use information from other subsystems in addition to local measurements. In the design of the Kalman filter, the nominal model of each subsystem is used, which means that it is the polyhedral center resulting from uncertainty. The i-th model of the subsystem is considered as follows:

$$x_i(k+1) = A_{ii}x_i(k) + B_iu_i(k)$$
$$+E_iw_i(k) + \sum_{j=1, j\neq i}^{M} A_{ij}x_j(k) \quad (25)$$
$$y_i(k) = C_i(k)y_i(k) + v_i(k)$$

so that the process noise $w_i(k)$ is a discrete-time white noise signal whose covariance matrix is defined below:

$$E\{w_i(k)w_i(k)^T\} = Q_{wi} \quad (26)$$

The measurement noise $v_i(k)$ is also a discrete-time white noise signal. Its covariance matrix is expressed as follows:

$$E\{v_i(k)v_i(k)^T\} = R_{vi} \quad (27)$$

The prediction equation is modified using the Kalman gain L_{ii} and the measurement it has at time k (which is available) as follows:

$$\hat{x}_i(k+1 \mid k) = A_{ii}\hat{x}_i(k \mid k-1) + B_iu_i(k)$$
$$+\sum_{\substack{j=1\\ j\neq i}}^{M} A_{ij}\hat{x}_j(k \mid k-1) +$$
$$L_{ii}(k)[y_i(k) - C_i(k)\hat{x}_i(k \mid k-1)] \quad (28)$$
$$+\sum_{\substack{j=1\\ j\neq i}}^{M} L_{ij}(k)\left[y_j(k) - C_j(k)\hat{x}_j(k \mid k-1)\right]$$

In the above expression, $L_{ij}(k)$ is the Kalman gain of the j -th agent, which is considered as the correction gain of the interaction effect in the j -th agent. Also, $x_i(k|k-1)$ is the estimate of the state at the previous time in the corresponding agent, which is available to the i -th Kalman filter. The Kalman gain $L_{ii}(k)$ is chosen to minimize the estimation covariance error matrix. The estimation error $k+1$ at time k is considered as follows:

$$e_i(k+1 \mid k) = x_i(k+1 \mid k) - \hat{x}_i(k+1 \mid k) \quad (29)$$

By inserting (27) and (24) in the above equation, the error covariance matrix will be obtained as follows:

$$\Sigma_i(k+|1|k) =$$
$$\text{Cov}\{e_i(k+1 \mid k)\} =$$
$$A_{ii}\Sigma_i(k \mid k-1)A_{ii}^T + E_iQ_{wi}E_i^T$$
$$+\sum_{\substack{j=1\\ j\neq i}}^{M} A_{ij}\Sigma_j(k-1 \mid k-1)A_{ij}^T \quad (30)$$
$$+\{L_{ii}(k)C_i\}\Sigma_i(k \mid k-1)$$
$$\{L_{ii}(k)C_i\}^T + L_{ii}(k)R_{vi}L_{ii}(k)^T -$$
$$2\{L_{ii}(k)C_i\}\Sigma_i(k \mid k-1)A_{ii}^T$$

Now the derivative of the estimation error covariance matrix is calculated relative to $L_{ii}(k)$:

$$\frac{\partial}{\partial L_{ii}(k)}\Sigma_i(k+1 \mid k) =$$
$$\frac{\partial}{\partial L_{ii}(k)}\{A_{ii}\Sigma_i(k \mid k-1)A_{ii}^T + E_iQ_{wi}E_i^T$$
$$+\underset{\substack{j=1\\ j\neq i}}{M} A_{ij}\Sigma_j(k-1 \mid k-1)A_{ij}^T \quad (31)$$
$$+\{L_{ii}(k)C_i\}\Sigma_i(k \mid k-1)$$
$$\{L_{ii}(k)C_i\}^T + L_{ii}(k)R_{vi}L_{ii}(k)^T$$
$$-2\{L_{ii}(k)C_i\}\Sigma_i(k \mid k-1)A_{ii}^T\}$$

In this way, the $L_{ii}(k)$ that minimizes the covariance matrix of the estimation error satisfies the following equation:

$$-2A_{ii}\Sigma_i(k \mid k-1)C_i^T$$
$$+22\{L_{ii}(k)C_i)\}\Sigma_i(k \mid k-1)C_i^T \quad (32)$$
$$+2L_{ii}(k)R_{vi} = 0$$

Finally, the Kalman gain for the $i-th$ subsystem will be as follows:

$$L_{ii}(k) = A_{ii}\Sigma_i(k \mid k-1)C_i^T \times \left[R_{vi} + C_i\Sigma_i(k \mid k-1)C_i^T\right]^{-1} \quad (33)$$

To calculate $L_{ij}(k)$, we can also write:

$$\sum_{\substack{j=1\\ j\neq i}}^{M} \Sigma_j(k \mid k) =$$

$$\sum_{\substack{j=1\\ j\neq i}}^{M} E\{e_j(k \mid k)e_j(k \mid k)^T\} =$$

$$\sum_{\substack{j=1\\ j\neq i}}^{M} A_{ij}\Sigma_j(k \mid k-1)A_{ij}^T \quad (33)$$

$$+\sum_{\substack{j=1\\ j\neq i}}^{M} \{L_{ii}(k)C_i\}\Sigma_i(k \mid k-1)\{L_{ii}(k)C_i\}^T$$

$$-2\sum_{\substack{j=1\\ j\neq i}}^{M} A_{ij}\Sigma_j(k \mid k-1)\{L_{ii}(k)C_i\}^T$$

The minimum of the above expression is as follows:

$$\frac{\partial}{\partial L_{ij}(k)}\sum_{\substack{j=1\\ j\neq i}}^{M} \Sigma_j(k \mid k) =$$

$$\frac{\partial}{\partial L_{ij}(k)}\sum_{\substack{j=1\\ j\neq i}}^{M} A_{ij}\Sigma_j(k \mid k-1)A_{ij}^T$$

$$+\sum_{\substack{j=1\\ j\neq i}}^{M} \{L_{ij}(k)C_j\} \quad (34)$$

$$\Sigma_j(k \mid k-1)\{L_{ij}(k)C_j\}^T$$

$$-2\sum_{\substack{j=1\\ j\neq i}}^{M} A_{ij}\Sigma_j(k \mid k-1)\{L_{ij}(k)C_j\}^T,$$

$$L_{ij}(k) = A_{ij}C_j^T\left[C_jC_j^T\right]^{-1}$$

In the distributed Kalman filter design, each of the local Kalman filters by predicting one time unit ahead of the local states, provide timely information to the controllers for decision making.

4- Simulation-Based Validation of the Proposed Method

4- 1- Case Study

In order to evaluate the proposed method, a three-link flexible manipulator is considered, where each link acts like a Quanser robot and the links interfere with each other. According to [29], each link can be considered as a mass-spring-damper system as shown in Figure 1. In this case, the model of each link is obtained as follows:

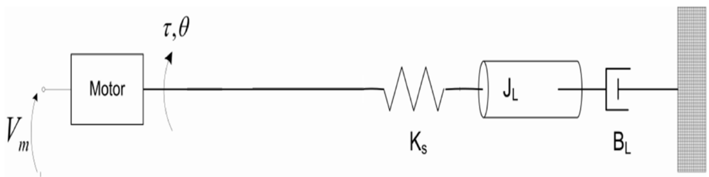

Fig. 1. Schematic model of one-link flexible manipulator.

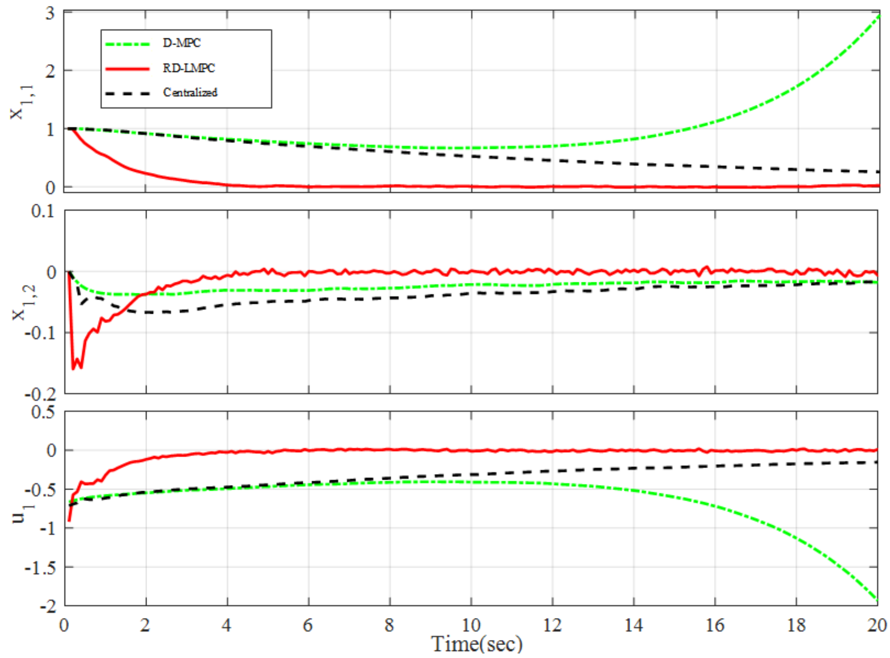

Fig. 2. Simulation results with RD-LMPC, D-MPC, and centralized MPC: The control signal and state path for the first subsystem.

$$\tau = J_L\ddot{\theta} + B_L\dot{\theta} + K_s\theta \tag{35}$$

Now, assuming $B_L = 0, \theta = x_1, \dot{\theta} = x_2$, the above differential equation can be rewritten as follows:

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\frac{K_s}{J_L} & 0\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{J_L}\end{bmatrix}\tau(t) \tag{36}$$

Therefore, by assuming $J_L = 1$ and $K_s = 0.4$ and adding uncertainty and the effect of interaction of agents (links) on each other, the following general model is obtained for each agent:

$$\begin{aligned}
&\begin{bmatrix}\dot{x}_{1,1}(t)\\ \dot{x}_{1,2}(t)\end{bmatrix} = \begin{bmatrix}0 & 1+0.2\alpha_1(t)\\ -0.4 & 0\end{bmatrix}\begin{bmatrix}x_{1,1}(t)\\ x_{1,2}(t)\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}u_1(t) + \begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix}\begin{bmatrix}x_{2,1}(t)\\ x_{2,2}(t)\end{bmatrix}\\
&y_1(t) = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_{1,1}(t)\\ x_{1,2}(t)\end{bmatrix}\\
&\begin{bmatrix}\dot{x}_{2,1}(t)\\ \dot{x}_{2,2}(t)\end{bmatrix} = \begin{bmatrix}0 & 1+0.2\alpha_2(t)\\ -0.4 & 0\end{bmatrix}\begin{bmatrix}x_{2,1}(t)\\ x_{2,2}(t)\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}u_2(t) + \begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix}\begin{bmatrix}x_{1,1}(t)\\ x_{1,2}(t)\end{bmatrix} + \begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix}\begin{bmatrix}x_{3,1}(t)\\ x_{3,2}(t)\end{bmatrix}\\
&y_2(t) = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_{2,1}(t)\\ x_{2,2}(t)\end{bmatrix}\\
&\begin{bmatrix}\dot{x}_{3,1}(t)\\ \dot{x}_{3,2}(t)\end{bmatrix} = \begin{bmatrix}0 & 1+0.2\alpha_3(t)\\ -0.4 & 0\end{bmatrix}\begin{bmatrix}x_{1,1}(t)\\ x_{1,2}(t)\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}u_3(t) + \begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix}\begin{bmatrix}x_{2,1}(t)\\ x_{2,2}(t)\end{bmatrix}\\
&y_3(t) = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_{3,1}(t)\\ x_{3,2}(t)\end{bmatrix}
\end{aligned} \tag{37}$$

Since the suggested control approach is for systems with multifaceted uncertainty, the following 20% of variations around the present uncertain nominal values are taken into consideration:

$$-1 \leq \alpha_i(t) \leq 1 \tag{38}$$

The weight matrices for all regions are taken into consideration as $M_i = I_{n_i}$ and $R_i = 1$, with initial state $x_i(0) = [0.1 \quad 0]^T$ and $epsilon_i = 0.01$, while designing the controller. The input condition is also for all areas $|u_i(k+n|k)| \leq 4$.

4- 2- Simulation and Validation

The simulation results are shown in Figures 2-4. According to the simulation findings, the distributed technique performs the worst. The DMPC controller is unable to solve it because of the strong interaction impact, and the system converges. Additionally, the distributed technique has a relatively high average control cost, but the problem-solving time is much reduced since each region is handled separately. Considering

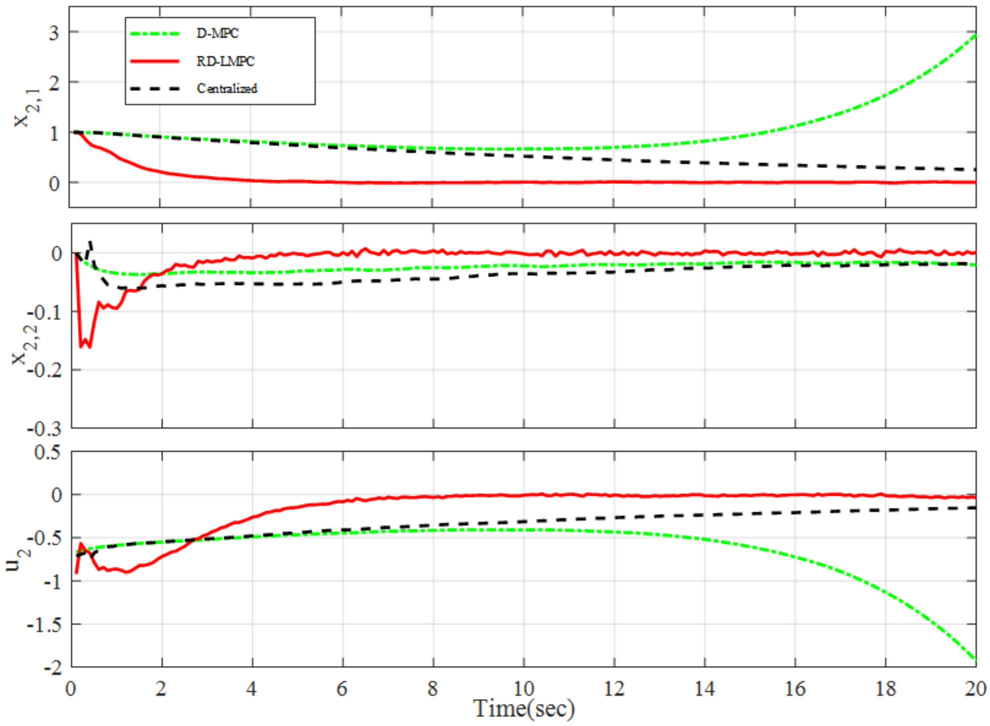

Fig. 3. Simulation results with RD-LMPC, D-MPC, and centralized MPC: The control signal and state path for the second subsystem.

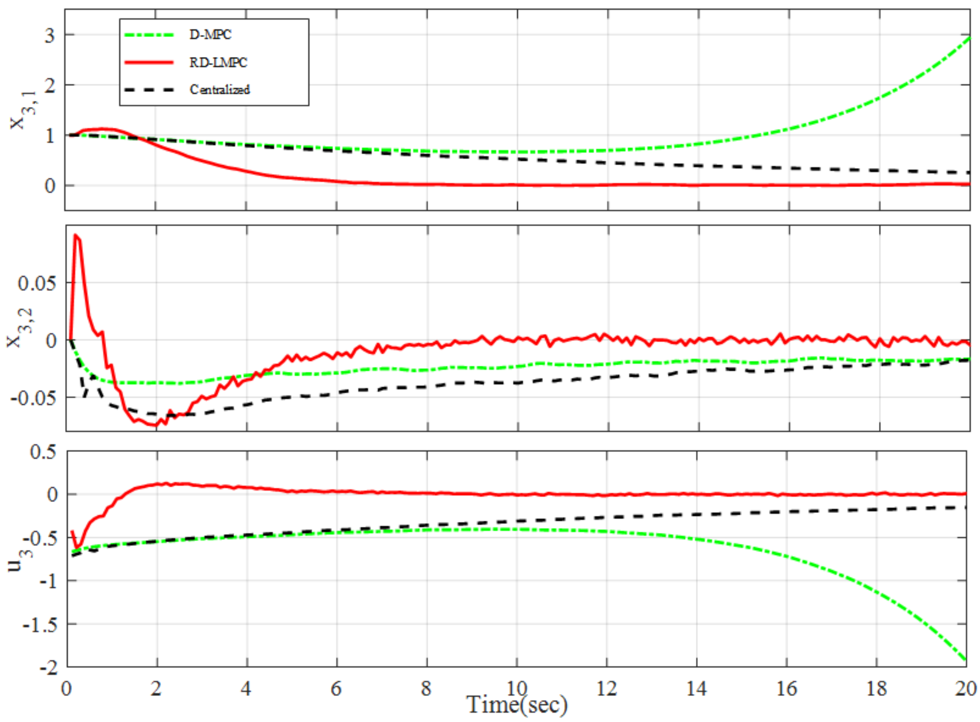

Fig. 4. Simulation results with RD-LMPC, D-MPC, and centralized MPC: The control signal and state path for the third subsystem.

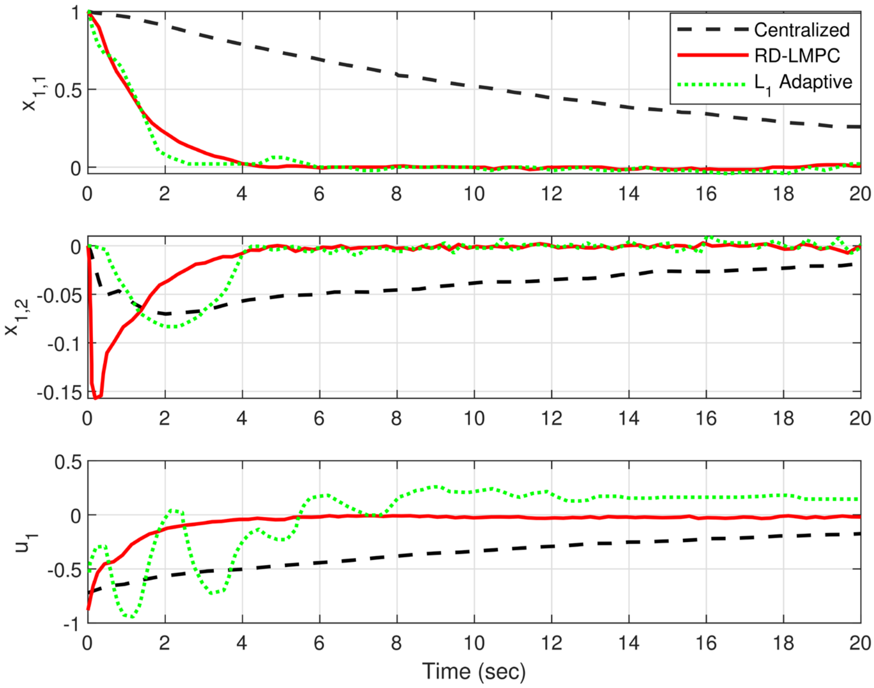

Fig. 5. Simulation results to compare the proposed method with the L1 adaptive method: The control signal and state path for the first subsystem.

that the approach used in this research requires more computation time than the distributed mode, it performs almost as well as the centralized mode, allowing the cost of solving the sub-problems repeatedly and transferring information to be reduced. Control has been diminished to the same degree as the centralized approach. The RD-LMPC method is the best option because its computational load is insignificant despite the fact that the performance benefits and the control cost is low. Based on the instructions for this solution method, solving the problem centrally is almost impossible in practice for such a system and has high computational complexity.

4- 3- Comparison

As seen in the simulation results, the proposed method has better performance and stability compared to the distributed predictive control method (presented in references [12, 15]) and is very close to the centralized control performance. For further comparison, the $\mathcal{L}_1$ adaptive control method (see [30]) was tested on the studied system and its results are shown in Figure 5 along with the proposed method (for states $x_{1,1}, x_{1,2}$). As seen in Figure 5, the performance of the two methods is close to each other, but the proposed method has a smoother and better control signal. For a better evaluation, the cost value of the methods is also given in Table 1 according to the cost functions of the methods. As can be seen from this table, the cost value of the proposed method is much closer to the centralized predictive control method, and according to the simulation results, it can be said that the proposed method has similar behavior to the centralized control method.

5- Conclusion

In this study, a novel robust distributed $\mathcal{L}$asso-MPC algorithm (RD-LMPC) is proposed for LTV systems with polytopic uncertainty. There are M state-coupled subsystems in the system. A robust LMPC has been made to reduce the over limit on the worst-case value of the objective function inside the polytope uncertainty. This is done by taking into account the polytope uncertainty. This approach views the control rule of each agent as a distinct feedforward interaction and state feedback form. This control input both assures the closed-loop system's quadratic bound stability and reduces the undesirable impacts of surrounding subsystems. For the suggested LMPC to work, a *distributed Kalman filter* is also made to show predictors and interactions for each agent. This filter uses local data as well as measurements from nearby subsystems given by the network. To assess the efficacy of the proposed approach, it was compared to centralized MPC, distributed MPC, and $\mathcal{L}_1$ adaptive control techniques on a multi-robot system.

Table 1. Comparison between cost values (mean).

Methods	Cost value
$\mathcal{C}-\mathcal{MPC}$	6.17
$\mathcal{D}-\mathcal{MPC}$	6.67
Proposed method ($\mathcal{RD}-\mathcal{LMPC}$)	6.21

References

[1] T. Gholaminejad, A. Khaki-Sedigh, and P. Bagheri, ”Adaptive Tuning of Model Predictive Control Parameters Based on Analytical Results,” AUT Journal of Modeling and Simulation, vol. 50, no. 2, pp. 109-116, 2018.

[2] Q. Sun, K. Zhang, and Y. Shi, ”Resilient model predictive control of cyber–physical systems under DoS attacks,” IEEE Transactions on Industrial Informatics, vol. 16, no. 7, pp. 4920-4927, 2019.

[3] S. Hashemipour, N. Vasegh, and A. Khaki Sedigh, ”Decentralized Model Reference Adaptive Control of Large Scale Interconnected Systems with Both State and Input Delays,” AUT Journal of Modeling and Simulation, vol. 50, no. 1, pp. 3-12, 2018.

[4] Y. Zhu and E. Fridman, ”Predictor methods for decentralized control of large-scale systems with input delays,” Automatica, vol. 116, p. 108903, 2020.

[5] X.-B. Chen and S. S. Stankovi´c, ”Decomposition and decentralized control of systems with multi-overlapping structure,” Automatica, vol. 41, no. 10, pp. 1765-1772, 2005.

[6] Z. Karami, Q. Shafee, Y. Khayat, M. Yaribeygi, T. Dragiˇcevi´c, and H. Bevrani, ”Decentralized model predictive control of DC microgrids with constant power load,” IEEE Journal of Emerging and Selected Topics in Power Electronics, vol. 9, no. 1, pp. 451-460, 2019.

[7] P. Ojaghi, N. Bigdeli, and M. Rahmani, ”An LMI approach to robust model predictive control of nonlinear systems with state-dependent uncertainties,” Journal of Process Control, vol. 47, pp. 1-10, 2016.

[8] C. Liu, H. Li, J. Gao, and D. Xu, ”Robust self-triggered min–max model predictive control for discrete-time nonlinear systems,” Automatica, vol. 89, pp. 333-339, 2018.

[9] P. Ojaghi and M. Rahmani, ”LMI-based robust predictive load frequency control for power systems with communication delays,” IEEE Transactions on Power Systems, vol. 32, no. 5, pp. 4091-4100, 2017.

[10] S. Yu, M. Hirche, Y. Huang, H. Chen, and F. Allg¨ower, ”Model predictive control for autonomous ground vehicles: A review,” Autonomous Intelligent Systems, vol. 1, no. 1, pp. 1-17, 2021.

[11] D. R. Robinson, R. T. Mar, K. Estabridis, and G. Hewer, ”An efcient algorithm for optimal trajectory generation for heterogeneous multi-agent systems in non-convex environments,” IEEE Robotics and Automation Let ters, vol. 3, no. 2, pp. 1215-1222, 2018.

[12] J. A. Preiss, W. H¨onig, N. Ayanian, and G. S. Sukhatme, ”Downwash- aware trajectory planning for large quadrotor teams,” in 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017:IEEE, pp. 250-257.

[13] C. E. Luis, M. Vukosavljev, and A. P. Schoellig, ”Online trajectory generation with distributed model predictive control for multi-robot motion planning,” IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 604-611, 2020.

[14] R. A. Shalmani, M. Rahmani, and N. Bigdeli, ”Nash-based robust distributed model predictive control for large-scale systems,” Journal of Process Control, vol. 88, pp. 43-53, 2020.

[15] S. Tang and V. Kumar, ”A complete algorithm for generating safe trajectories for multi-robot teams,” in Robotics Research: Springer, 2018, pp.599-616.

[16] R. Van Parys and G. Pipeleers, ”Distributed model predictive formation control with inter-vehicle collision avoidance,” in 2017 11th Asian Control Conference (ASCC), 2017: IEEE, pp. 2399-2404.

[17] L. Dai, Q. Cao, Y. Xia, and Y. Gao, ”Distributed MPC for formation of multi-agent systems with collision avoidance and obstacle avoidance,” Journal of the Franklin Institute, vol. 354, no. 4, pp. 2068-2085, 2017.

[18] P. Wang and B. Ding, ”A synthesis approach of distributed model predictive control for homogeneous multi-agent system with collision avoidance,” International Journal of Control, vol. 87, no. 1, pp. 52-63, 2014.

[19] G. Xu, T. Long, Z. Wang, and J. Sun, ”Trust-region fltered sequential convex programming for multi-UAV trajectory planning and collision avoid ance,” ISA transactions, vol. 128, pp. 664-676, 2022.

[20] G. K. Larsen, N. D. Van Foreest, and J. M. Scherpen, ”Distributed MPC applied to a network of households with micro-CHP and heat storage,” IEEE Transactions on Smart Grid, vol. 5, no. 4, pp. 2106-2114, 2014.

[21] Z. Feng, G. Wen, and G. Hu, ”Distributed secure coordinated control for multiagent systems under strategic attacks,” IEEE transactions on cyber- netics, vol. 47, no. 5, pp. 1273-1284, 2016.

[22] M. Gallieri, ”Principles of LASSO MPC,” in Lasso-MPC–Predictive Control with L1-Regularised Least Squares: Springer, 2016, pp. 47-63.

[23] J. Zhan, Y. Chen, A. Aleksandrov, and X. Li, ”Robust distributed model predictive control based consensus of general linear multi-agent systems,” in 2019 IEEE International Symposium on Circuits and Systems (ISCAS), 2019: IEEE, pp. 1-5.

[24] H. Li and Y. Shi, ”Distributed model predictive control of constrained non- linear systems with communication delays,” Systems and Control Letters, vol. 62, no. 10, pp. 819-826, 2013.

[25] A. Katriniok, B. Rosarius, and P. M¨ah¨onen, ”Fully distributed model pre- dictive control of connected automated vehicles in intersections: Theory and vehicle experiments,” IEEE Transactions on Intelligent Transportation Systems, 2022.

[26] C. D. Lellis and P. M. Topping, ”Almost-Schur lemma,” Calculus of Variations and Partial Di-erential Equations, vol. 43, pp. 347-354, 2012.

[27] R. Vadigepalli and F. J. Doyle, ”A distributed state estimation and control algorithm for plantwide processes,” IEEETransactions on Control Systems Technology, vol. 11, pp. 119-127, 2003.

[28] W. Hackbusch, Elliptic di-erential equations: theory and numerical treat- ment. Springer, 2017.

[29] W. Alam, N. Ali, H. M. W. Aziz, and J. Iqbal, ”Control of flexible joint robotic manipulator: Design and prototyping,” in 2018 International Con- ference on Electrical Engineering (ICEE), 2018: IEEE, pp. 1-6.

[30] H. Ahmadian, M. Lotf, M. B. Menhaj, H. A. Talebi, and I. Sharif, ”A novel L1 adaptive-hybrid control with guaranteed stability for a class of uncertain nonlinear systems: A case study on SA330 Puma,” Journal of the Franklin Institute, vol. 359, no. 17, pp. 9860-9885, 2022.

HOW TO CITE THIS ARTICLE

H. Ahmadian, I. Sharifi, H. A. Talebi, Robust Distributed $\mathcal{L}$asso-Model Predictive Control Design: A Case Study on Large-Scale Multi-Robot Systems, AUT J. Model. Simul., 55(1) (2023) 127-138.

DOI: 10.22060/miscj.2023.22087.5312