High-Performance Cooperative Distributed Model Predictive Control for Linear Systems

Georgios Darivianakis*, Salar Fattahi†, John Lygeros* and Javad Lavaei†

Abstract— **Distributed model predictive control (DMPC) has been proven a successful method in regulating the operation of large-scale networks of constrained dynamical systems. This paper is concerned with cooperative DMPC in which the control actions of the systems are derived by the solution of a system-wide optimization problem. To exploit the merits of distributed computation algorithms, we investigate how to approximate this system-wide optimization problem by a number of loosely coupled subproblems. In this context, the main challenge is to design appropriate terminal cost-to-go functions and invariant sets that comply with the coupling pattern of the network. The goal of this paper is to present a unified framework for the synthesis of a terminal distributed controller, cost-to-go function and invariant set based on an existing optimal centralized terminal controller. We construct an objective function for the synthesis problem, which mathematically quantifies the closeness of the given centralized and distributed control systems. This objective function is formulated using the optimizer of a robust optimization problem. Conditions for global Lyapunov stability are imposed in the synthesis problem in a way that allows the terminal cost-to-go function and invariant set to admit the desired distributed structure. We illustrate the effectiveness of the proposed method on a benchmark spring-mass-damper problem.**

I. Introduction

Control of large-scale networks of interacting dynamical systems is an active field of research due to its high impact on real-world applications, e.g., power networks [1] and building districts [2], [3]. Even though the design of a centralized controller to regulate the operation of these networks of systems is often feasible, it is sometimes difficult to practically implement the controller due to computation and communication limitations in the network. In such cases, it is desirable to design interacting local controllers with a prescribed structure which rely only on local information and computational resources. Even though the problem of synthesizing optimal distributed controllers is known NP-hard [4] in its general form, for certain network structures it has been shown to admit either a closed-form solution [5] or an exact convex reformulation [6]. For general network structures, the usual practice is to resort to linear matrix inequality (LMI) relaxations [7], [8] or semidefinite programming (SDP) relaxations [9], [10] to obtain suboptimal distributed controllers with performance guarantees.

A downside of these static distributed controllers is their inability to cope with state and input constraints. Model predictive control (MPC) is an optimization based methodology that is well-suited for constrained systems [11]. Distributed MPC (DMPC) approaches are typically distinguished into non-cooperative [12] and cooperative [13], [14]. In the former, each system considers the effect of neighboring systems as a disturbance in its own dynamics. Though computationally simple and effective in practice, non-cooperative approaches can be conservative in presence of strong coupling. On the other hand, cooperative distributed MPC approaches require substantial communication infrastructure and computation resources since a system-wide MPC problem is formulated and solved using distributed computation algorithms, e.g., the alternating direction method of multipliers (ADMM) [15].

To guarantee stability of the closed-loop system, the MPC approach relies on the existence of a stabilizing static terminal controller that respects the system constraints when operated in a terminal invariant set. The infinite-horizon cost of this terminal controller is captured by an appropriate cost-to-go function [11]. Even if this terminal controller has been designed as to admit a distributed structure, the respective invariant set and cost-to-go function do not necessarily exhibit the same distributed structure since these are constructed based on global Lyapunov stability and invariance concepts. This arbitrary structure for the terminal invariant set and cost-to-go function results in a cooperative DMPC formulation which is not amendable to distributed computation algorithms. Current approaches in the literature deal with this issue by simultaneously designing the terminal controller, cost-to-go function and invariant set. This, however, is only achieved by imposing restrictions during the synthesis phase, e.g., the dynamical coupling of the systems is considered as disturbance [16], [17], or the cost-to-go functions and invariant sets admit a completely decoupled structure [18]. These restrictions go beyond the necessity of simply enforcing a distributed structure on the terminal cost-to-go function and invariant set and may result in suboptimal cooperative DMPC formulations with invariant terminal sets that are small or even empty. In addition, none of these approaches provides performance guarantees and optimality bounds with respect to the optimal centralized design.

Here we present a unified framework for the synthesis of a distributed terminal controller, cost-to-go function and invariant set starting with an existing optimal centralized design. We rely on recent findings of [19] to construct

This research was partially supported by CTI under SCCER FEEB&D, SNSF under IMES, DARPA YFA, ONR YIP and AFOSR YIP Awards. The authors would like to thank Professor R.S. Smith for helpful discussions.

*The authors are with the Automatic Control Laboratory, Department of Electrical Engineering and Information Technology, ETH, Zurich, Switzerland. {gdarivia, jlygeros}@control.ee.ethz.ch

†The authors are with the Department of Industrial Engineering and Operations Research, University of California, Berkeley, California, USA. {fattahi, lavaei}@berkeley.edu

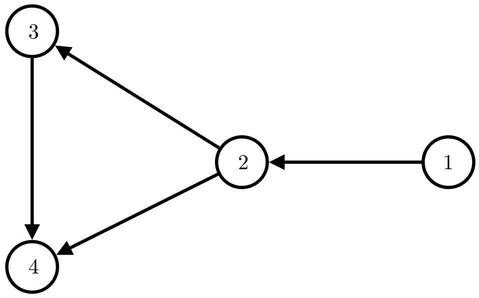

Fig. 1. Dynamical coupling graph of four systems.

an objective function that mathematically quantifies the closeness of the given centralized and distributed control systems. This objective function is formulated using the optimizer of a robust optimization problem, which is solved by reformulating it into a convex program. Global Lyapunov stability conditions are explicitly imposed on the synthesis problem. We use similar theoretical tools as those developed in [20] to express these stability conditions as LMIs. The optimization variables associated with the controller parameters are independent of the symmetric matrix that defines a quadratic Lyapunov function used to guarantee stability and approximate the infinite-horizon cost. This allows the terminal cost-to-go function and invariant set to admit the desired distributed structure that is amendable to distributed computation algorithms.

The cooperative DMPC problem is formulated in Section II. The main contributions are presented in Section III, where the techniques associated with the derivation of a distributed terminal controller, cost-to-go function and invariant set based on an existing optimal centralized controller are presented. Section IV provides a numerical study to assess the efficacy of the proposed method.

Notation: The set of real numbers is denoted by $\mathbb{R}$. The symbol $\text{trace}\{W\}$ denotes the trace of a matrix W. The notation I refers to the identity matrix of appropriate dimension. The symbol $(.)^T$ denotes transpose. The symbol $\|W\|_2$ is used to denote the 2-norm of W. The notation $W \succeq 0$ is used to show that a symmetric matrix W is positive semidefinite. For a set of vectors or matrices X_i where $i \in \mathcal{R} = \{1, 2, ..., r\}$, the operator $[X_i]_{i \in \mathcal{R}}$ takes their column-wise concatenation.

II. Problem formulation

We consider a network of M interacting dynamical systems that may be coupled through their dynamics. For each system $i \in \mathcal{M} = \{1, \ldots, M\}$, we denote by $\mathcal{N}_i \subseteq \mathcal{M}$ the set that includes itself and all its neighboring systems, i.e., the set of systems that affect its dynamics. In the illustrative example shown in Fig. 1, a network of $M = 4$ interacting dynamical systems is shown in which an arc connecting system j to system i indicates that the states of the j-th system affect the dynamics of the i-th one. Hence, $\mathcal{N}_1 = \{1\}$, $\mathcal{N}_2 = \{1, 2\}$, $\mathcal{N}_3 = \{2, 3\}$, and $\mathcal{N}_4 = \{2, 3, 4\}$. Moreover, we assume that the constraints and objective functions of the systems are appropriately formulated as to respect the coupling structure of the network.

A. Dynamics and constraints

The state-space dynamical evolution of the i-th system is given as follows:

$$x_{i,t+1} = \sum_{j \in \mathcal{N}_i} A_{ij} x_{j,t} + B_i u_{i,t}, \tag{1}$$

where $x_{i,t} \in \mathbb{R}^{n_i}$ and $u_{i,t} \in \mathbb{R}^{m_i}$ denote the state and input of system i at time t, respectively. The matrices $A_{ij} \in \mathbb{R}^{n_i \times n_j}$ and $B_i \in \mathbb{R}^{n_i \times m_i}$ are assumed to be known. To simplify notation, we compactly rewrite (1) as

$$x_{i,t+1} = A_{\mathcal{N}_i} x_{\mathcal{N}_i,t} + B_i u_{i,t}, \tag{2}$$

where $x_{\mathcal{N}_i,t} = [x_{j,t}]_{j \in \mathcal{N}_i} \in \mathbb{R}^{n_{\mathcal{N}_i}}$, $A_{\mathcal{N}_i} = [A_{ij}^\top]_{j \in \mathcal{N}_i}^\top \in \mathbb{R}^{n_i \times n_{\mathcal{N}_i}}$, and $n_{\mathcal{N}_i} = \sum_{j \in \mathcal{N}_i} n_j$.

In addition, we consider linear state and input constraints of the form

$$x_{\mathcal{N}_i,t} \in \mathcal{X}_{\mathcal{N}_i} = \{x \in \mathbb{R}^{n_{\mathcal{N}_i}} \ : \ G_{\mathcal{N}_i} x \leq g_{\mathcal{N}_i}\}, \tag{3a}$$
$$u_{i,t} \in \mathcal{U}_i = \{u \in \mathbb{R}^{m_i} \ : \ H_i u \leq h_i\}, \tag{3b}$$

where the matrices $G_{\mathcal{N}_i} \in \mathbb{R}^{k_i \times n_{\mathcal{N}_i}}$, $g_{\mathcal{N}_i} \in \mathbb{R}^{k_i}$, $H_i \in \mathbb{R}^{p_i \times m_i}$ and $h_i \in \mathbb{R}^{p_i}$ are known. For simplicity, we assume that the sets $\mathcal{X}_{\mathcal{N}_i}$ and $\mathcal{U}_i$ contain the origin in their interior.

By combining the dynamics of all the systems in the network, we derive the dynamical equations of the global system as

$$x_{t+1} = Ax_t + Bu_t, \tag{4}$$

where $x_t = [x_{i,t}]_{i \in \mathcal{M}} \in \mathbb{R}^n$ and $u_i = [u_{i,t}]_{i \in \mathcal{M}} \in \mathbb{R}^m$ are the state and input, respectively. The matrices $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times m}$ are readily derived from the state-space equations (2) of the individual systems. Similarly, we define the constraint sets $\mathcal{X}$ and $\mathcal{U}$ of the global system as

$$\mathcal{X} = \{x \in \mathbb{R}^n \ : \ Gx \leq g\}, \tag{5a}$$
$$\mathcal{U} = \{u \in \mathbb{R}^m \ : \ Hu \leq h\}, \tag{5b}$$

where the matrices $G \in \mathbb{R}^{k \times n}$, $g \in \mathbb{R}^k$, $H \in \mathbb{R}^{p \times m}$ and $h \in \mathbb{R}^p$ are derived by the concatenation of the corresponding coefficients in (3).

B. Objective function

Given the initial state x_0, the system-wide goal is to minimize the objective function

$$J_\infty = \sum_{i=1}^{M} \sum_{t=0}^{\infty} \ell_i(x_{\mathcal{N}_i,t}, u_{i,t}), \tag{6}$$

while satisfying the state and input constraints given in (5). Here, the stage cost $\ell_i(\cdot)$ in (6) is given as

$$\ell_i(x_{\mathcal{N}_i,t}, u_{i,t}) = x_{\mathcal{N}_i,t}^\top Q_{\mathcal{N}_i} x_{\mathcal{N}_i,t} + u_{i,t}^\top R_i u_{i,t}, \tag{7}$$

where $Q_{\mathcal{N}_i} \in \mathbb{R}^{n_{\mathcal{N}_i} \times n_{\mathcal{N}_i}}$ and $R_i \in \mathbb{R}^{m_i \times m_i}$ are known positive semi-definite and positive definite matrices, respectively.

C. Centralized MPC

In the spirit of MPC, we rely on a cost-to-go function $V_f(\cdot)$ and a positively invariant set $\mathcal{X}_f$ to formulate a finite-horizon optimization problem. The original infinite horizon objective function J_∞ is now approximated by

$$\widetilde{J}_\infty = V_f(x_T) + \sum_{i\in\mathcal{M}}\sum_{t\in\mathcal{T}} \ell_i(x_{\mathcal{N}_i,t}, u_{i,t}).$$

where $\mathcal{T} = \{0,\dots,T-1\}$, with T denoting the prediction horizon.

For the time being, we ignore the input and state constraints and assume that a stabilizing controller $u_t = K_c x_t$, with $K_c \in \mathbb{R}^{m\times n}$, exists for the unconstrained system. If we assume that this controller is applied to the (unconstrained) system for all $t \geq T$, a cost-to-go function of the form

$$V_f(x_T) = x_T^\top P_c x_T,$$

with $P_c \in \mathbb{R}^{n\times n}$, can be used to approximate the infinite horizon cost of operating the system with initial condition x_T. To ensure the stability of the closed-loop system, K_c and P_c need to satisfy the Lyapunov inequality:

$$V_f(x_{t+1}) - V_f(x_t) + \sum_{i=1}^{M} \ell_i(x_{\mathcal{N}_i,t}, u_{i,t}) \leq 0, \;\; \forall t \geq T. \quad (8)$$

This condition guarantees that the terminal controller stabilizes the system and $V_f(\cdot)$ is an upper approximation of the true cost-to-go function. Computationally efficient algorithms, such at the discrete-time algebraic Riccati equation (DARE), exist to compute K_c and P_c that satisfy (8) with equality. Typically, the resulting matrices do not admit any distributed structure, even if the problem dynamics and constraints do.

To re-introduce the state and input constraints we resort to positively invariant sets.

Definition 1. The set $\mathcal{X}_f$ is *positively invariant* under the controller $u_t = K_c x_t$ if for all $x_t \in \mathcal{X}_f$, we have that $(A + BK_c)x_t \in \mathcal{X}_f$ and $K_c x_t \in \mathcal{U}$.

To ensure state and input constraints satisfaction under the terminal controller $u_t = K_c x_t$ for $t \geq T$, we enforce the terminal state x_T to lie in a set $\mathcal{X}_f \subseteq \mathcal{X}$ that is positively invariant. A typical choice for $\mathcal{X}_f$ is

$$\mathcal{X}_f = \{x \in \mathbb{R}^n : V_f(x) \leq \gamma\}, \quad (9)$$

where $\gamma \in \mathbb{R}$ is a positive constant. To increase system flexibility, one typically seeks the maximum volume ellipsoidal set $\mathcal{X}_f$. As shown in [21], this can be obtained by solving the following convex optimization problem:

$$\begin{array}{lll} \max & \zeta & \\ \text{s.t.} & \|P_c^{-1/2} G_s^\top\|_2^2\,\zeta \leq g_s^2, & \forall s = 1,\dots,k, \\ & \|P_c^{-1/2} K_c^\top H_l^\top\|_2^2\,\zeta \leq h_l^2, & \forall l = 1,\dots,p, \end{array} \quad (10)$$

and setting $\gamma = \zeta^2$.

Using the terminal set and cost-to-go function, we now formulate the centralized MPC problem as follows:

$$\begin{aligned} \min\;\; & V_f(x_T) + \sum_{i\in\mathcal{M}}\sum_{t\in\mathcal{T}} \ell_i(x_{\mathcal{N}_i,t}, u_{i,t}) \\ \text{s.t.}\;\; & x_{t+1} = Ax_t + Bu_t, \\ & (x_t, u_t) \in \mathcal{X}\times\mathcal{U}, \\ & x_T \in \mathcal{X}_f. \end{aligned} \quad (11)$$

with optimization variables (x_t, u_t) for all $t \in \mathcal{T}$.

Problem (11) is a quadratic optimization program with linear and conic constraints. Common algorithms for solving this type of problem are mostly based on centralized computation, which makes them less practical if the number of systems or the horizon length is large. To deal with this issue, distributed computation algorithms, such as ADMM, may be used to efficiently solve Problem (11). It is well-known that these methods are computationally far more efficient when the respective optimization problem admits a partially-distributed structure. However, this is not the case for Problem (11) due to the possibly dense structure of the terminal set and cost-to-go function.

D. Distributed MPC

In the distributed MPC framework, we enforce the terminal controller of the i-th system to admit a partial state feedback structure such that

$$u_{i,t} = K_{\mathcal{N}_i} x_{\mathcal{N}_i}, \;\; \forall t \geq T,$$

where $K_{\mathcal{N}_i} \in \mathbb{R}^{m_i\times n_{\mathcal{N}_i}}$. In this context, the cost-to-go function, denoted by $\widehat{V}_f(\cdot)$, admits a distributed structure

$$\widehat{V}_f(x_T) = \sum_{i=1}^{M} \widehat{V}_f^i(x_{\mathcal{N}_i,T}),$$

with each $\widehat{V}_f^i(\cdot)$ being formulated as

$$\widehat{V}_f^i(x_{\mathcal{N}_i,T}) = x_{\mathcal{N}_i,T}^\top P_{\mathcal{N}_i} x_{\mathcal{N}_i,T},$$

where $P_{\mathcal{N}_i} \in \mathbb{R}^{n_{\mathcal{N}_i}\times n_{\mathcal{N}_i}}$. Moreover, the terminal set $\widehat{\mathcal{X}}_f$ also admits a distributed structure

$$\widehat{\mathcal{X}}_f = \widehat{\mathcal{X}}_f^1 \cap \dots \cap \widehat{\mathcal{X}}_f^M$$

with

$$\widehat{\mathcal{X}}_f^i = \{x_{\mathcal{N}_i} \in \mathbb{R}^{n_{\mathcal{N}_i}} : \widehat{V}_f^i(x_{\mathcal{N}_i}) \leq \widehat{\gamma}_i\},$$

and $\widehat{\gamma}_i$ a positive constant computed by an optimization problem similar to (10).

We now formulate the distributed MPC problem as follows:

$$\begin{aligned} \min\;\; & \sum_{i\in\mathcal{M}} \left(\widehat{V}_f^i(x_{\mathcal{N}_i,T}) + \sum_{t\in\mathcal{T}} \ell_i(x_{\mathcal{N}_i,t}, u_{i,t}) \right) \\ \text{s.t.}\;\; & \left.\begin{aligned} & x_{i,t+1} = A_{\mathcal{N}_i} x_{\mathcal{N}_i,t} + B_i u_{i,t}, \\ & (x_{\mathcal{N}_i,t}, u_{i,t}) \in \mathcal{X}_{\mathcal{N}_i} \times \mathcal{U}_i, \\ & x_{\mathcal{N}_i,T} \in \widehat{\mathcal{X}}_f^i. \end{aligned}\right\} \forall i \in \mathcal{M} \end{aligned} \quad (12)$$

with optimization variables $(x_{\mathcal{N}_i,t}, u_{i,t})$ for all $i \in \mathcal{M}$, $t \in \mathcal{T}$.

Problem (12) exhibits a distributed structure that can be exploited by distributed computation algorithms to efficiently solve it. In what follows, we will develop a new machinery to synthesize $K_{\mathcal{N}_i}$ and $P_{\mathcal{N}_i}$ based on the optimal centralized control system, while ensuring that the distributed nature of the problem is maintained. To ease the notation, we define $K_d \in \mathbb{R}^{m\times n}$ as

$$u_t = [K_{\mathcal{N}_i} x_{\mathcal{N}_i,t}]_{i\in\mathcal{M}} = K_d x_t, \ \ \forall t \geq T.$$

We denote by $\mathcal{K}_d$ the linear subspace induced by the distributed structure of K_d, i.e., $K_d \in \mathcal{K}_d \subseteq \mathbb{R}^{m\times n}$. Moreover, we define the positive definite matrix $P_d \in \mathbb{R}^{n\times n}$ to be

$$\widehat{V}_f(x_T) = \sum_{i=1}^{M} x_{\mathcal{N}_i,T}^\top P_{\mathcal{N}_i} x_{\mathcal{N}_i,T} = x_T^\top P_d x_T.$$

and we denote by $\mathcal{P}_d$ the linear subspace of $\mathbb{R}^{n\times n}$ generated by the distributed structure of P_d, i.e., $P_d \in \mathcal{P}_d$.

III. Terminal controller and set synthesis

In this section, we discuss the proposed methodology to synthesize the distributed terminal controller $K_{\mathcal{N}_i}$ and cost-to-go function $P_{\mathcal{N}_i}$ for all $i \in \mathcal{M}$. We split the synthesis phase into two parts: (i) performance and (ii) stability. In the former, we discuss the conditions that allow the distributed terminal controller to approximate the performance of the centralized one. In the latter, we provide conditions to guarantee that the closed-loop system is asymptotically stable.

A. Performance criteria

Our goal is to design a distributed terminal controller K_d that approximates the performance of the centralized terminal controller K_c.

Definition 2. (Terminal Controller Equivalence) Given $x_T \in \mathcal{X}_f$, the gain K_c is equivalent to K_d if and only if $u_t = K_c x_t = K_d x_t$ for all $t \geq T$.

Given $x_T \in \mathcal{X}_f$, let $P_x \in \mathbb{R}^{n\times n}$ be the unique positive semidefinite solution of the Lyapunov equation

$$\mathcal{L}(P_x, x_T) = 0$$

where

$$\mathcal{L}(P, x) = (A + BK_c)P(A + BK_c)^\top - P + xx^\top.$$

Note that such a unique solution $P_x \succeq 0$ exists because the terminal controller K_c is stabilizing.

Theorem 1. ([19, Thm. 2]) For a given $x_T \in \mathcal{X}_f$, there exists $K_d \in \mathcal{K}_d$ that is equivalent to K_c if and only if the optimal objective of the optimization problem

$$\begin{aligned} \min_{K_d} \quad & \operatorname{trace}\left\{(K_c - K_d)P_x(K_c - K_d)^\top\right\} \\ \text{s.t.} \quad & K_d \in \mathcal{K}_d \end{aligned} \tag{13}$$

is zero.

As shown in [19], a small objective value for Problem (13) implies that K_d closely approximates the performance of K_c for a given initial condition x_T. Ideally, it is desirable that the equivalence between K_c and K_d is met for every x_T in the centralized terminal set $\mathcal{X}_f$ defined in (9). Since a matrix $P_x \succeq 0$ that satisfies the Lyapunov equation

$$\mathcal{L}(P_x, x_T) = 0, \quad \forall x_T \in \mathcal{X}_f, \tag{14}$$

may not exist, we accomplish this through the following robust optimization problem:

$$\begin{aligned} (P_x, \alpha_x) = \arg\min_{P,\alpha} \quad & \alpha \\ \text{s.t.} \quad & -\alpha I \preceq \mathcal{L}(P, x) \preceq \alpha I, \quad \forall x \in \mathcal{X}_f, \\ & P \succeq 0, \end{aligned} \tag{15}$$

It can easily be verified that if the optimal objective value α_x is zero, then P_x solves (14); otherwise, $\mathcal{L}(P_x, x_T)$ is maintained as close as possible to zero for every $x_T \in \mathcal{X}_f$.

Problem (15) is computationally intractable in this form due to having an infinite number of constraints. A tractable finite-dimensional convex reformulation is given below:

$$\begin{aligned} (P_x, \alpha_x) = \arg\min_{P,\alpha} \quad & \alpha \\ \text{s.t.} \quad & \mathcal{L}(P, 0) + \gamma P_c^{-1} \preceq \alpha I, \\ & -\alpha I \preceq \mathcal{L}(P, 0) \\ & P \succeq 0, \end{aligned} \tag{16}$$

Theorem 2. Problems (16) and (15) are equivalent.

Proof. The proof is provided in the Appendix. ■

B. Stability conditions

Similar to the centralized case, we impose the following Lyapunov inequality on the design of the distributed terminal controller:

$$V_f(x_{t+1}) - V_f(x_t) \leq -x_t^\top \left(Q + K_d^\top R K_d\right) x_t, \ \forall t > T. \tag{17}$$

This condition is enforced mainly for two reasons: (i) it guarantees the asymptotically stability of the closed-loop system, and (ii) it ensures that $V_f(\cdot)$ is an upper approximation of the true cost-to-go function.

Lemma 1. Inequality (17) holds if and only if there exist $K_d \in \mathcal{K}_d$ and $P_d \in \mathcal{P}_d$ such that the following matrix inequality holds:

$$\left[\begin{array}{c:c:c:c} T_d & (A_d + B_d K_d)T_d & 0 & 0 \\ T_d(A_d + B_d K_d)^\top & T_d & T_d Q^{1/2} & T_d K_d^\top R^{1/2} \\ 0 & Q^{1/2} T_d & I & 0 \\ 0 & R^{1/2} K_d T_d & 0 & I \end{array}\right] \succeq 0, \tag{18}$$

where the substitution $T_d = P_d^{-1}$ is used.

Proof. The proof is provided in the Appendix. ■

Inequality (18) is non-convex due to the bilinear term $K_d T_d$. Upon defining the new variable $L_d = K_d T_d$, inequality (18) is translated to a linear matrix inequality in terms of (T_d, L_d), and K_d can be retrieved as $K_d = L_d T_d^{-1}$. Note that in general, it is not an easy task to exactly translate the sparsity structure on K_d into a set of convex constraints on L_d and T_d. However, a set of sufficient convex constraints that ensures K_d lies in $\mathcal{K}_d$ is $L_d \in \mathcal{K}_d$ and $T_d \in \mathcal{I}$, where $\mathcal{I}$

denotes the linear subspace of diagonal matrices. One drawback of enforcing such convex constraints is that $P_d \in \mathcal{I}$, instead of $P_d \in \mathcal{P}_d$, which introduces conservativeness both on the design of the distributed terminal controller and on the approximation of the true cost-to-go function. Using techniques similar to [20], the following theorem establishes an equivalent formulation that alleviates this issue.

Theorem 3. Inequality (17) holds if and only if there exist $K_d \in \mathcal{K}_d$, $P_d \in \mathcal{P}_d$ and $S_d \in \mathbb{R}^{n\times n}$ such that the following matrix inequality holds:

$$\left[\begin{array}{c:c:c:c} T_d & (A_d+B_dK_d)S_d & 0 & 0 \\ S_d^\top(A_d+B_dK_d)^\top & S_d+S_d^\top-T_d & S_d^\top Q^{1/2} & S_d^\top K_d^\top R^{1/2} \\ 0 & Q^{1/2}S_d & I & 0 \\ 0 & R^{1/2}K_dS_d & 0 & I \end{array}\right] \succeq 0, \tag{19}$$

where $T_d = P_d^{-1}$ and $T_d^{-1} \in \mathcal{P}_d$.

Proof. The proof is provided in the Appendix. ∎

We now define the non-convex transformation, $L_d = K_dS_d$, to re-write (19) as an LMI constraint. Note that (19) implies $S_d + S_d^\top \succeq T_d \succeq 0$, which leads to the non-singularity of S_d. Therefore, the distributed controller K_d can be recovered as $K_d = L_dS_d^{-1}$. Similar to the previous case, it can be verified that one sufficient condition for $K_d \in \mathcal{K}_d$ is to impose $S_d \in \mathcal{I}$ and $L_d \in \mathcal{K}_d$. However, note that, due to Theorem 3, T_d^{-1} should belong to the linear subspace $\mathcal{P}_d$, which is non-convex in T_d. This non-convexity will be remedied in the next section.

C. Distributed terminal controller synthesis

As already mentioned, it is desirable for the terminal set $\widehat{\mathcal{X}}_f$ to be the maximum volume ellipsoid confined in the feasible region $\mathcal{X}$. To achieve this, we combine (19) with the minimization of the objective function

$$J_{\text{ast}}(T_d) = -\log\det(T_d),$$

which maximizes the volume of the 1-level set ellipsoid $\mathcal{E}^{(1)} = \{x \in \mathbb{R}^n : x^\top P_d x \le 1\}$ [22].

We now define the feasible set C_{ast} to guarantee the asymptotic stability of the terminal closed-loop system:

$$C_{\text{ast}} = \{(T_d, S_d, L_d) : \text{LMI (19) holds}\}.$$

Next, we combine the performance criteria with the condition for asymptotic stability in order to formulate the synthesis problem of the distributed terminal controller and cost-to-go function. To do so, the objective function in (13) should be written in terms of S_d and L_d, i.e.,

$$J_{\text{pf}}(S_d, L_d) = \text{trace}\{(K_cS_d - L_d)S_d^{-1}P_xS_d^{-1}(K_cS_d - L_d)^\top\}.$$

It can be observed that the above objective function is non-convex in (S_d, L_d). Therefore, we resort to a convex relaxation of the objective function. Specifically, we employ the epigraph representation to relax this nonlinear objective function into a set of LMI constraints

$$C_{\text{pf}} = \left\{ (S_d, L_d, D_d) : \exists W_d \in \mathbb{R}^{n\times n} \text{ s.t. } \begin{bmatrix} W_d & S_d \\ S_d & P_x \end{bmatrix} \succeq 0, \begin{bmatrix} D_d & (K_cS_d - L_d) \\ (K_cS_d - L_d)^\top & W_d \end{bmatrix} \succeq 0 \right\},$$

and an objective function

$$\widetilde{J}_{\text{pf}}(D_d) = \text{trace}\{D_d\}.$$

To obtain the above formulation, we use the Schur complement and relax the equality constraint $W_d = S_dP_x^{-1}S_d$ into $W_d \ge S_dP_x^{-1}S_d$. This relaxation is necessary for the set C_{pf} to admit a convex structure.

To obtain a trade-off between performance and the size of the terminal invariant set, a linear combination of $J_{\text{ast}}(T_d)$ and $\widetilde{J}_{\text{pf}}(D_d)$ is minimized based on the following optimization problem:

$$\begin{array}{rl} \min & \beta J_{\text{ast}}(T_d) + (1-\beta)\widetilde{J}_{\text{pf}}(D_d) \\ \text{s.t.} & T_d^{-1} \in \mathcal{P}_d,\ S_d \in \mathcal{I},\ L_d \in \mathcal{K}_d, D_d \in \mathbb{R}^{m\times m}, \\ & (T_d, S_d, L_d) \in C_{\text{ast}}, \\ & (S_d, L_d, D_d) \in C_{\text{pf}}. \end{array} \tag{20}$$

where $0 \le \beta \le 1$. Problem (20) is still non-convex due to the constraint $T_d^{-1} \in \mathcal{P}_d$. To deal with this issue, first recall that T_d is introduced as an intermediate variable and it plays the role of P_d^{-1} throughout the developed method. We relax the constraint $T_d^{-1} \in \mathcal{P}_d$ by allowing $T_d \in \mathbb{R}^{n\times n}$. After solving the relaxed problem, we solve the following optimization problem in order to find a matrix P_d that is the *approximate* inverse of T_d and belongs to the subspace $\mathcal{P}_d$:

$$\begin{array}{rl} \min & -\text{logdet}(P_d) + \text{trace}(T_d^* P_d) \\ \text{s.t.} & P_d \in \mathcal{P}_d,\ P_d \succeq T_d^{*-1}. \end{array} \tag{21}$$

where T_d^* is the optimal solution of (20) corresponding to the variable T_d. Note that if $\mathcal{P}_d = \mathbb{R}^{n\times n}$, the optimal solution of (21) satisfies $P_d = T_d^{*-1}$ and if $\mathcal{P}_d$ is a structured subspace of $\mathbb{R}^{n\times n}$, then P_d is an approximation of T_d^{*-1} which results in a conservative estimation of the true cost-to-go function. The optimization problem (21) is widely used for estimating the structured precision matrix (i.e., the inverse of the covariance matrix) of a random vector with an underlying Gaussian distribution (see [23], [24] for more details).

IV. Numerical example

In this numerical study, we consider systems composed of masses that are connected by springs and dampers and arranged in a chain formation, exemplified in Fig. 2. The values of the masses, spring constants and damping coefficients are chosen uniformly at random from the intervals [5, 10]kg, [0.8, 1.2]N/m and [0.8, 1.2]Ns/m, respectively. We assume that each i-th mass is an individual system with its state vector $x_{i,t} \in \mathbb{R}^2$ representing the position and velocity deviation from the system's equilibrium state, and its input $u_{i,t} \in \mathbb{R}$ denoting the force applied to the i-th mass. We assume that the states and inputs are constrained such that $\|x_{i,t}\|_\infty \le 2$ and $\|u_{i,t}\|_\infty \le 5$ for all times t. The masses are

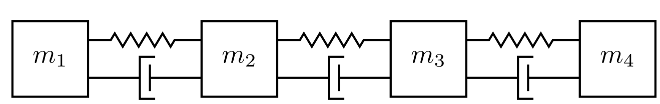

Fig. 2. A chain of four masses connected by springs and dampers.

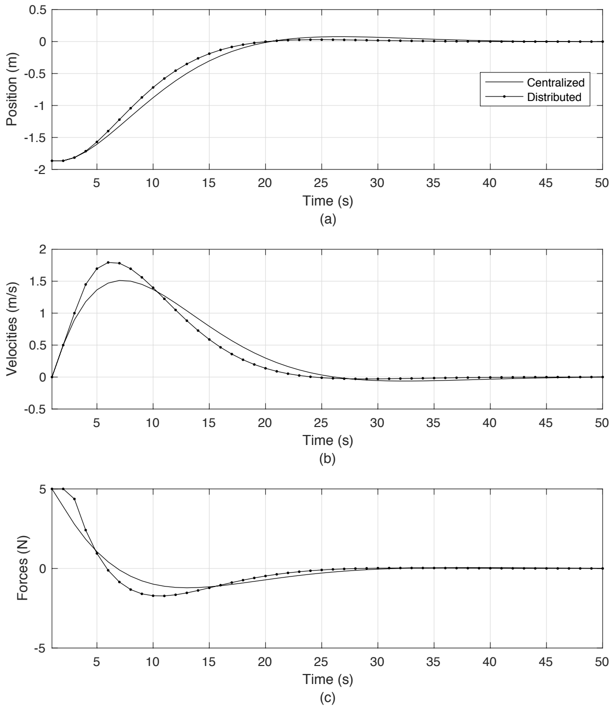

Fig. 3. Closed-loop trajectories of position (a), velocity (b) and force (c) for mass 1 during receding horizon simulations.

initially at rest and positioned uniformly at random within the intervals $[-2, -1.8]$m and $[1.8, 2]$m.

The continuous-time dynamics of this interconnected dynamical system naturally admits a distributed structure. The prediction control model is obtained by the discretization of the system's continuous dynamics using forward Euler with the sampling time 0.1s. Although inexact, Euler discretization is chosen as to preserve the distributed structure of the system. On the contrary, the discrete-time simulation model of the system is obtained using the exact zero-order hold discretization method with the sampling time 0.1s. The objective function of each system is of the form (7) with $Q_i = \text{diag}(1, 1)$ and $R_i = 0.1$. The centralized design to obtain K_c and P_c is performed by solving the discrete algebraic Riccati equation. The distributed design to obtain K_d and P_d is performed using the methods introduced in Section III of this paper.

The performance of the system is evaluated on a receding horizon implementation, i.e., the first input resulting from the respective centralized and distributed optimization problem is applied to the exact system dynamics, and the next state is evaluated. We use as a metric the cost of operating the system until convergence to the system's equilibrium state. Initially, we conduct a closed-loop simulation experiment for a system comprising five masses and the prediction horizon of $T = 12$. In Fig. 3, the trajectories generated for the centralized and distributed designs are shown. We observe that these trajectories are very similar, which illustrates the proximity in performance between the centralized and distributed designs.

To better quantify the performance comparison between the centralized and distributed designs, we conducted several

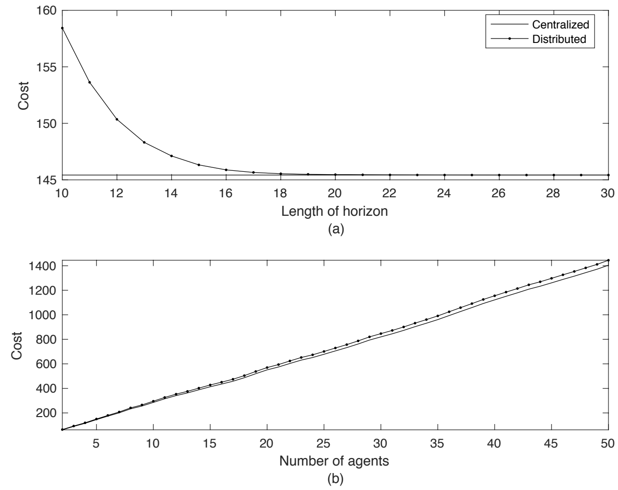

Fig. 4. Performance comparison between centralized and distributed designs for (a) a system of five masses and varying T, and (b) a system of varying number of masses and $T = 12$.

simulation experiments for systems with different horizons and number of masses. Fig. 4(a) shows the cost associated with the length of prediction horizon for a system comprising five masses. We observe that the increase of the horizon length results in cost convergence for the two methods. This can be explained by considering that large horizons make the use of terminal sets and cost-to-go functions obsolete since the system is capable of steering its states to origin within the prediction horizon time. In Fig. 4(b), the effect of the number of masses on the cost is depicted. As expected, the suboptimality gap increases with respect to the number of masses in the system. We note, however, that in all instances this suboptimality gap is fairly small, which indicates the efficiency of the proposed distributed synthesis method.

The main reason of this growth in the suboptimality gap with respect to the number of masses in the system is due

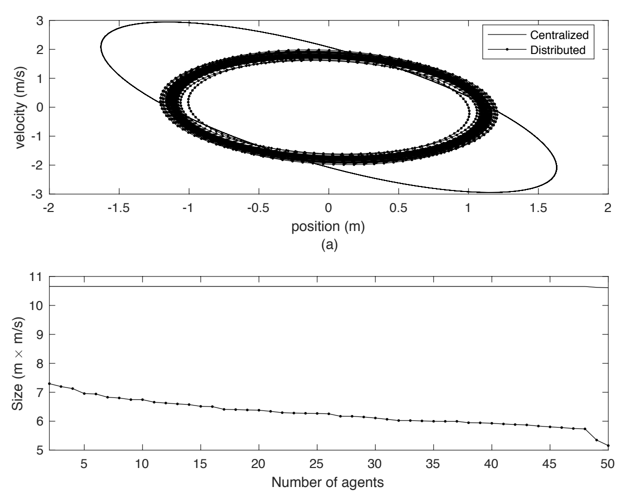

Fig. 5. Evolution in region (a) and size (b) of the distributed and centralized terminal invariant set of mass 1 with respect to the number of masses in the system.

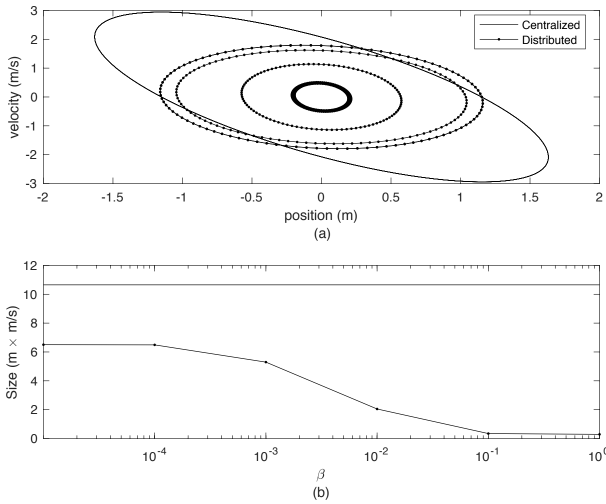

Fig. 6. Effect of the tuning parameter $\beta \in (0, 1)$ on the evolution in region (a) and size (b) of the distributed terminal invariant set of mass 1.

to the corresponding decrease in the size of the distributed terminal invariant sets, which is graphically depicted in Fig. 5. We observe that the number of masses in the system has little effect on the size of the terminal centralized invariant set, which is attributed to the flexibility associated with the full structure of the centralized terminal controller. On the other hand, imposing a distributed structure on the terminal controller does not provide enough flexibility to each individual system to completely compensate the effect of the increase in the number of masses in the system.

Finally, we investigate the effect of the tuning parameter β in the objective function (20), which weighs the closeness of distributed and centralized controllers versus the volume maximization of the terminal invariant set. We observe in Fig. 6 that if $\beta \simeq 1$, then the resulting distributed terminal invariant sets are considerably smaller in size. This seemingly counterintuitive result can be understood by recalling that maximizing the volume of the 1-level set ellipsoid is equivalent to demanding a superior performance from the distributed terminal controller, i.e., a faster convergence to the origin. This implies that the distributed terminal controller would take steeper actions for the same terminal state, which has as a repercussion a higher difficulty in satisfying the input constraint of the system. This in turn makes the distributed terminal invariant set to be of a smaller size than in the case for which $\beta \simeq 0$. This is an important observation that highlights the role of equipping the design of the distributed terminal controller with an objective function that measures its closeness to the optimal centralized controller.

V. Conclusion

In this paper, we presented a synthesis approach for distributed cooperative MPC that correlates the design of the distributed terminal controller, cost-to-go function and invariant set with an existing optimal centralized controller. An objective function that measures the closeness of the designed distributed controller to the optimal centralized one is derived by the solution of a robust optimization problem. The latter problem is then solved by reformulating it into a convex optimization problem using standard robust optimization techniques. Conditions for global Lyapunov stability are imposed in such a way that the terminal cost-to-go function and invariant set admit an appropriate distributed structure. This allows the resulting distributed cooperative MPC problem to be solvable by distributed computation. Conservativeness in the design is reduced by exploiting methods that allow the optimization variables associated with the controller parameters to be independent of the symmetric matrix that defines the quadratic Lyapunov function used to guarantee stability and approximate the infinite-horizon cost. We illustrated the merits of the proposed approach for a large-scale system that is composed of masses connected by springs and dampers, where the closed-loop performance of the distributed cooperative MPC closely approximates the one of the centralized MPC.

Future work involves the appropriate extension of the developed method so as the synthesis to be executed in a completely distributed way. This is a particularly important feature in case of varying network topologies for which the centralized design can trivially be derived while only the new and a few of the existing distributed controllers need to be redesigned.

Appendix

Proof of Theorem 2:

We begin this proof by noting that the terminal set $\mathcal{X}_f$, given as

$$\mathcal{X}_f = \{x \in \mathbb{R}^n \,:\, x^\top P_c x \leq \gamma\},$$

can be equivalently written as follows:

$$\begin{aligned}\mathcal{X}_f = \{x \in \mathbb{R}^n \,: \exists w \in \mathbb{R}^{n\times n} \text{ such that} \\ x = \gamma^{1/2} P_c^{-1/2} w,\, \|w\|_2 \leq 1\}.\end{aligned}$$

We prove the equivalence, in terms of feasible set and optimal value, between Problem (15) and Problem (16) in a constructive manner. Using standard duality arguments, the robust constraint

$$\mathcal{L}(P, x) \preceq \alpha I, \;\; \forall x \in \mathcal{X}_f,$$

is equivalent to

$$y^\top \mathcal{L}(P, x) y \leq \alpha,\; \forall x \in \mathcal{X}_f,\; \forall y \in \mathcal{Y} = \{y : \|y\|_2 = 1\},$$

which can be simplified as

$$y^\top \mathcal{L}(P, 0) y + \max_{x \in \mathcal{X}_f} \|x^\top y\|_2^2 \leq \alpha,\; \forall y \in \mathcal{Y}. \tag{E.1}$$

Now consider the following optimization problem:

$$\max_{x \in \mathcal{X}_f} \|x^\top y\|_2^2 \;\;\Leftrightarrow\;\; \begin{array}{cl} \max & \|x^\top y\|_2^2 \\ \text{s.t.} & w \in \mathbb{R}^{n\times n}, \\ & x = \gamma^{1/2} P_c^{-1/2} w, \\ & \|w\|_2 \leq 1. \end{array} \tag{E.2}$$

As shown in [21], the optimal solution of (E.2) is equal to $\|\gamma^{1/2}P_c^{-1/2}y\|_2^2$. Therefore, (E.1) yields that

$$\begin{aligned}&y^\top \mathcal{L}(P,0)y + \|\gamma^{1/2}P_c^{-1/2}y\|_2^2 \le \alpha,\ \forall y\in\mathcal{Y}, \Leftrightarrow\\ &y^\top\left(\mathcal{L}(P,0)+\gamma P_c^{-1}\right)y \le \alpha,\ \forall y\in\mathcal{Y}, \Leftrightarrow\\ &\mathcal{L}(P,0)+\gamma P_c^{-1} \preceq \alpha I.\end{aligned}$$

In a similar fashion, the robust constraint

$$\mathcal{L}(P_x,x) \succeq -\alpha I, \quad \forall x\in\mathcal{X}_f,$$

can be equivalently written as

$$\begin{aligned}&y^\top\mathcal{L}(P,x)y \ge -\alpha,\ \forall x\in\mathcal{X}_f,\ \forall y\in\mathcal{Y} \Leftrightarrow\\ &y^\top\mathcal{L}(P,0)y + \min_{x\in\mathcal{X}_f}\|x^\top y\|_2^2 \ge -\alpha,\ \forall y\in\mathcal{Y}.\end{aligned} \tag{E.3}$$

It can easily be verified that $\min_{x\in\mathcal{X}_f}\|x^\top y\|_2^2$ is equal to 0. Therefore, (E.3) yields that,

$$\begin{aligned}&y^\top\mathcal{L}(P,0)y \ge -\alpha,\ \forall y\in\mathcal{Y} \Leftrightarrow\\ &\mathcal{L}(P,0)\succeq -\alpha I.\end{aligned}$$

Using these results, the robust optimization problem (15) can be written as follows:

$$\begin{aligned}\min_{\alpha,P}\quad & \alpha\\ \text{s.t.}\quad & \mathcal{L}(P,0)+\gamma P_c^{-1}\preceq \alpha I,\\ & -\alpha I \preceq \mathcal{L}(P,0),\\ & P\succeq 0,\end{aligned}$$

This equivalence argument concludes the proof. ∎

Proof of Lemma 1:

Notice that

$$V_f(x_{t+1}) - V_f(x_t) \le -x_t^\top\left(Q+K_d^\top R K_d\right)x_t \Leftrightarrow$$

$$P_d - (A_d+B_dK_d)^\top P_d(A_d+B_dK_d) - (Q+K_d^\top RK_d) \ge 0 \Leftrightarrow$$

$$\begin{bmatrix} P_d^{-1} & (A_d+B_dK_d)\\ (A_d+B_dK_d)^\top & P_d\end{bmatrix} - \begin{bmatrix}0 & 0\\ Q^{1/2} & K_d^\top R^{1/2}\end{bmatrix}\begin{bmatrix}0 & Q^{1/2}\\ 0 & R^{1/2}K_d\end{bmatrix} \succeq 0.$$

Post-multiplying the inequality by $T=\operatorname{diag}(I,P_d^{-1})$ and pre-multiplying it by $T^\top$, and then applying once again the Schur complement, leads to

$$\begin{bmatrix} T_d & (A_d+B_dK_d)T_d & 0 & 0\\ T_d(A_d+B_dK_d)^\top & T_d & T_dQ^{1/2} & T_dK_d^\top R^{1/2}\\ 0 & Q^{1/2}T_d & I & 0\\ 0 & R^{1/2}K_dT_d & 0 & I\end{bmatrix}\succeq 0,$$

where the substitution $T_d = P_d^{-1}$ is used. ∎

Proof of Theorem 3:

This proof follows similar arguments to [20, Thm. 1]. In particular, the *necessity* can be shown by choosing $S_d = S_d^\top = T_d$. To prove *sufficiency*, we assume that (19) holds. Then, $S_d + S_d^\top \succeq T_d$. Note that this implies that S_d is non-singular. Since T_d is positive definite, the inequality $(T_d - S_d)^\top T_d^{-1}(T_d - S_d)\succeq 0$ holds. This implies that $S_d^\top T_d^{-1}S_d \succeq S_d + S_d^\top - T_d$ which leads to,

$$\begin{bmatrix} T_d & (A_d+B_dK_d)S_d & 0 & 0\\ S_d^\top(A_d+B_dK_d)^\top & S_d^\top T_d^{-1}S_d & S_d^\top Q^{1/2} & S_d^\top K_d^\top R^{1/2}\\ 0 & Q^{1/2}S_d & I & 0\\ 0 & R^{1/2}K_dS_d & 0 & I\end{bmatrix}\succeq 0.$$

Since S_d is no-singular and by post-multiplying the inequality by $T=\operatorname{diag}(I, S_d^{-1}T_d, I, I)$ and pre-multiplying it by $T^\top$, we can recover the LMI inequality (18). ∎

References

[1] A. Venkat, I. Hiskens, J. Rawlings, and S. Wright, "Distributed MPC strategies with application to power system automatic generation control," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 6, pp. 1192–1206, 2008.

[2] G. Darivianakis, A. Georghiou, R. Smith, and J. Lygeros, "The power of diversity: Data-driven robust predictive control for energy efficient buildings and districts," *arXiv:1607.05441*, 2016.

[3] G. Darivianakis, A. Georghiou, A. Eichler, R. Smith, and J. Lygeros, "Scalability through decentralization: A robust control approach for the energy management of a building community," in *IFAC World Congress*, Toulouse, France, 2017.

[4] J. Tsitsiklis and M. Athans, "On the complexity of decentralized decision making and detection problems," *IEEE Trans. Autom. Control*, vol. 30, no. 5, pp. 440–446, 1985.

[5] M. Fardad, F. Lin, and M. Jovanovic, "On the optimal design of structured feedback gains for interconnected systems," in *Proc. IEEE Conf. Decision Control*, Shanghai, China, 2009, pp. 978–983.

[6] M. Rotkowitz and S. Lall, "A characterization of convex problems in decentralized control ^{a}st," *IEEE Trans. Autom. Control*, vol. 51, no. 2, pp. 274–286, 2006.

[7] C. Langbort, R. Chandra, and R. D'Andrea, "Distributed control design for systems interconnected over an arbitrary graph," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1502–1519, 2004.

[8] A. Zecevic and D. Siljak, *Control of complex systems: Structural constraints and uncertainty*. Springer, 2010.

[9] J. Lavaei, "Decentralized implementation of centralized controllers for interconnected systems," *IEEE Trans. Autom. Control*, vol. 57, no. 7, pp. 1860–1865, 2012.

[10] G. Fazelnia, R. Madani, A. Kalbat, and J. Lavaei, "Convex relaxation for optimal distributed control problems," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 206–221, 2017.

[11] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[12] M. Farina and R. Scattolini, "Distributed predictive control: A non-cooperative algorithm with neighbor-to-neighbor communication for linear systems," *Automatica*, vol. 48, no. 6, pp. 1088–1096, 2012.

[13] A. Venkat, J. Rawlings, and S. Wright, "Stability and optimality of distributed model predictive control," in *Proc. IEEE Conf. Decision Control*, Seville, Spain, 2005, pp. 6680–6685.

[14] P. Giselsson and A. Rantzer, "On feasibility, stability and performance in distributed model predictive control," *IEEE Trans. Autom. Control*, vol. 59, no. 4, pp. 1031–1036, 2014.

[15] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends. Network.*, vol. 3, no. 1, pp. 1–122, 2011.

[16] S. Riverso, M. Farina, and G. Ferrari-Trecate, "Plug-and-play model predictive control based on robust control invariant sets," *Automatica*, vol. 50, no. 8, pp. 2179–2186, 2014.

[17] J. Maestre, D. De La Pena, E. Camacho, and T. Alamo, "Distributed model predictive control based on agent negotiation," *J. Process Control*, vol. 21, no. 5, pp. 685–697, 2011.

[18] C. Conte, N. Voellmy, M. Zeilinger, M. Morari, and C. Jones, "Distributed synthesis and control of constrained linear systems," in *Proc. Amer. Control Conf.*, Montreal, Canada, 2012, pp. 6017–6022.

[19] S. Fattahi, G. Fazelnia, and J. Lavaei, "Transformation of optimal centralized controllers into near-global static distributed controllers," in *Proc. IEEE Conf. Decision Control*, Osaka, Japan, 2015, pp. 4915–4922.

[20] M. De Oliveira, J. Geromel, and J. Bernussou, "Extended H_2 and H_∞ characterizations and controller parametrizations for discrete-time systems," *Int. J. Control*, vol. 75, no. 9, pp. 666–679, 2002.

[21] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[22] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[23] O. Banerjee, L. El Ghaoui, and A. d' Aspremont, "Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data," *J. Mach. Learn. Res*, vol. 9, pp. 485–516, 2008.

[24] J. Friedman, H. Trevor, and T. Robert, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.