Distributed Model Predictive Control for Linear Systems With Adaptive Terminal Sets

Journal Article

Author(s):
Darivianakis, Georgios; Eichler, Annika; Lygeros, John

Publication date:
2020-03

Permanent link:
https://doi.org/10.3929/ethz-b-000406434

Originally published in:
IEEE Transactions on Automatic Control 65(3), https://doi.org/10.1109/TAC.2019.2916774

Distributed Model Predictive Control for Linear Systems with Adaptive Terminal Sets

Georgios Darivianakis*, *Member, IEEE,* Annika Eichler†, *Member, IEEE* and John Lygeros‡, *Fellow, IEEE*

***Abstract*—We propose a distributed model predictive control scheme for linear time-invariant constrained systems which admit a separable structure. To exploit the merits of distributed computation algorithms, the terminal cost and invariant terminal set of the optimal control problem need to respect the coupling structure of the system. Existing methods to address this issue typically separate the synthesis of terminal controllers and costs from the one of terminal sets, and do not explicitly consider the effect of the current and predicted system states on this synthesis process. These limitations can adversely affect performance due to small or even empty terminal sets. Here, we present a unified framework to encapsulate the synthesis of both the stabilizing terminal controller and invariant terminal set into the same optimization problem. Conditions for Lyapunov stability and invariance are imposed in the synthesis problem in a way that allows the terminal cost and invariant terminal set to admit the desired distributed structure. We illustrate the effectiveness of the proposed method on several numerical examples.**

***Index Terms*—Predictive control, Large-scale systems, Cooperative control, Robust adaptive control, Optimal control.**

I. INTRODUCTION

OPERATION of large-scale networks of interacting dynamical systems remains an active field of research due to its high impact on real-world applications, e.g., regulation of power networks [1] and energy management of building districts [2]. For systems of this scale, the design and deployment of centralized controllers is often a difficult task due to computation and communication limitations in the network. Hence, 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 [3] in its general form, for certain network structures it has been shown to admit either a closed-form solution [4] or an exact convex reformulation [5]. For unstructured network topologies, the usual practice is to resort to semidefinite programming relaxations [6]–[10] to obtain suboptimal distributed controllers with performance guarantees.

A downside of these static distributed controllers is their inability to efficiently cope with state and input constraints of the systems. Model predictive control (MPC) is an optimization based methodology that is well-suited for constrained linear systems [11]. Despite recent advances on computation and communication technologies, formulating and solving a large optimization problem in real time remains a challenging task. To circumvent this, several methods have been proposed in the literature to leverage the distributed structure of the network to approximate the original optimization problem through a set of loosely coupled subproblems. Distributed model predictive control (DMPC) approaches are typically categorized into non-cooperative [12]–[18] and cooperative ones [19]–[26]. In the former, each system considers the effect of neighboring systems as a disturbance in its own dynamics and constraints. Typical cases of non-cooperative DMPC approaches are tube-based methods, where the states and inputs of neighboring subsystems are confined in a precomputed [12]–[14] or adaptive [15]–[18] bounded set and each subsystem needs to account for all possible impacts of its neighbors occurring within these bounded sets. Though computationally simple and effective in practice, non-cooperative approaches can be conservative in the 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. Approaches discussed in the literature [19]–[22] typically involve the communication of planned control sequences or state trajectories between neighboring systems. Unlike non-cooperative methods, cooperative approaches can guarantee convergence to the optimal solution of the centralized optimization problem.

In all these approaches the existence of a stabilizing static terminal controller is assumed to guarantee recursive feasibility and stability of the closed-loop system. This terminal controller respects the state and input constraints of the system when operated in an invariant terminal set. The infinite-horizon cost associated with this terminal controller is upper bounded by a terminal cost [11]. In the DMPC framework adopted here, the terminal controller, cost and invariant terminal set are designed to respect the distributed structure of the system [12]–[26], making the resulting DMPC optimization problem amendable to distributed computation algorithms such as the alternating direction method of multipliers (ADDM) [27]. To achieve this, current approaches in the literature typically split the design phase into two sequential parts: (i) the terminal controller and cost are synthesized based on Lyapunov stability concepts, then (ii) the invariant terminal set is constructed to satisfy the state and input constraints for the closed-loop system under the given terminal controller. However, the resulting invariant terminal set can be a small (or even empty) inner approximation of the maximum invariant terminal set due to

Research supported by the Swiss Innovation Agency Innosuisse under the Swiss Competence Center for Energy Research SCCER FEEB&D.

*G. Darivianakis is with the ABB Corporate Research, 5405 Baden-Dättwil, Switzerland (e-mail: georgios.darivianakis@ch.abb.com),

†A. Eichler is with the Deutsches Elektronen-Synchroton DESY, 22607 Hamburg, Germany (e-mail: annika.eichler@desy.de),

‡J. Lygeros is with the Automatic Control Laboratory, Department of Electrical Engineering and Information Technology, ETH Zurich, 8092 Zurich, Switzerland. (e-mail: jlygeros@control.ee.ethz.ch).

the imposed restrictions on its structure and the decoupled design phases. This can lead the resulting DMPC scheme to severe performance degradation.

In this paper, we propose a novel approach to design a distributed stabilizing terminal controller and invariant terminal set that can be adapted in every iteration based on the current and predicted state of the system. The necessity of online adaptation of terminal sets based on the predicted system evolution has previously been identified on several works (e.g., [17], [18], [23], [25], [28]). The key difference of our approach is that the design of the stabilizing terminal controller, the construction of the invariant terminal set, and the derivation of the optimal input trajectory for the finite-time horizon DMPC problem are the result of one single optimization problem. In other words, unlike current approaches, the construction of the terminal controller and invariant set is shifted from the design phase to the online phase. This is achieved by each system treating the influence of its neighboring systems as a disturbance; thus, we resort to robust optimization techniques to formulate the resulting synthesis problem. The proposed method falls in the category of cooperative schemes since the optimization variables of the proposed DMPC scheme need to be agreed among all the involved systems in the network.

The rest of the paper is organized as follows. In Section II, the dynamical system is analyzed and the standard DMPC problem formulation is reviewed. The main contributions are presented in Section III, where the methods to encapsulate the design of the distributed stabilizing terminal controller, cost and invariant terminal set, based on Lyapunov stability and invariance conditions, in the DMPC problem formulation are discussed. Section IV provides numerical studies to assess the efficacy and scalability of the proposed method. Concluding remarks are provided in Section V.

Notation: Let $\mathbb{R}$, $\mathbb{R}_+$ and $\mathbb{N}_+$ denote the set of real numbers, non-negative real numbers and non-negative integers, respectively. For a vector $v \in \mathbb{R}^n$, we denote by $v^\top$ its transpose and $\|v\|$ its Euclidean norm. For given vectors $v_i \in \mathbb{R}^{k_i}$ with $k_i \in \mathbb{N}$, $i \in \mathcal{M} = \{1, \dots, m\}$, we define $[v_i]_{i\in\mathcal{M}} = [v_1^\top \dots v_m^\top]^\top \in \mathbb{R}^k$ with $k = \sum_{i=1}^m k_i$ as their vector concatenation, and $\operatorname{diag}(v_1, \dots, v_M)$ as the block diagonal matrix with $v_1, \dots, v_M$ on the diagonal and zeros elsewhere. The notation $W \succeq 0$ is used to show that a symmetric matrix W is positive semidefinite. A function $f : \mathbb{R}_+ \to \mathbb{R}_+$ belongs to class $\mathcal{K}$ if it is continuous, strictly increasing and $f(0) = 0$. A function $f : \mathbb{R}_+ \to \mathbb{R}_+$ belongs to class $\mathcal{K}_\infty$ if $f \in \mathcal{K}$ and $\lim_{x\to\infty} f(x) = \infty$. For given matrices $A \in \mathbb{R}^{n\times m}$, $B \in \mathbb{R}^{n\times n}$ and $C \in \mathbb{R}^{m\times m}$, we define the conventions $\star^\top BA = A^\top BA$ and $\begin{bmatrix} B & A \\ \star^\top & C \end{bmatrix} = \begin{bmatrix} B & A \\ A^\top & C \end{bmatrix}$.

II. Problem formulation

A. Dynamically coupled constrained linear systems

Consider a discrete-time linear time-invariant system with state dynamics at time $t \in \mathbb{N}_+$ given by

$$x_{t+1} = Ax_t + Bu_t. \tag{1a}$$

Here, $x_t \in \mathbb{R}^n$ denotes the states with x_0 known and $u_t \in \mathbb{R}^m$ the control inputs. The system matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ are assumed known. The states and inputs of the system are subject to linear constraints

$$x_t \in \mathcal{X} = \{x \in \mathbb{R}^n \ : \ Gx \leq g\}, \tag{1b}$$
$$u_t \in \mathcal{U} = \{u \in \mathbb{R}^m \ : \ Hu \leq h\}, \tag{1c}$$

where $G \in \mathbb{R}^{p\times n}$, $g \in \mathbb{R}^p$, $H \in \mathbb{R}^{k\times m}$ and $h \in \mathbb{R}^k$ are known matrices. The system objective is to minimize the infinite-horizon cost function

$$J_\infty = \sum_{t=0}^{\infty} \ell(x_t, u_t), \tag{1d}$$

while satisfying its dynamics and constraints. The stage cost $\ell(\cdot)$ is given by

$$\ell(x_t, u_t) = x_t^\top Q x_t + u_t^\top R u_t, \tag{1e}$$

with $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ known positive semi-definite and positive definite matrices, respectively.

Assumption 1. *(A, B) is a stabilizable pair, $(Q^{1/2}, A)$ is an observable pair, and the sets $\mathcal{X}$ and $\mathcal{U}$ contain the origin in their interior.*

Assumption 1 is standard in the MPC literature [11] and is made to guarantee that convergence to the origin, which is the system equilibrium state, can be achieved. Henceforth, we refer to the set of initial conditions, x_0, for which the optimizer of this infinite-horizon optimization problem exists as its feasibility set.

We consider linear systems of the form (1) whose matrices A, B, G, H, Q and R admit a structure that allows us to decompose the original system into an ordered set $\mathcal{M} = \{1, \dots, M\}$ of M subsystems, which are coupled by their states, i.e., B, H and R are block diagonal matrices.

Assumption 2. *The subsystem dynamics, constraints and objective functions are coupled only through the subsystem states and not through the subsystem inputs.*

Assumption 2 is not restrictive since any linear system with coupled state and input constraints can be brought to the desired form by considering its coupled inputs as additional states, and their deviation in time as the corresponding new inputs [25]. This increases the dimension of the system, since integrator states need to be added. The system states x_t and inputs u_t are partitioned as $x_t = [x_{1,t}^\top, \dots, x_{M,t}^\top]^\top$ and $u_t = [u_{1,t}^\top, \dots, u_{M,t}^\top]^\top$ where $x_{i,t} \in \mathbb{R}^{n_i}$ and $u_{i,t} \in \mathbb{R}^{m_i}$ denote the local states and inputs of i-th subsystem, respectively. For each i-th subsystem, we define $\mathcal{N}_i \subseteq \mathcal{M}$ as the set of subsystems whose states, $x_{\mathcal{N}_i,t} = [x_j]_{j\in\mathcal{N}_i} \in \mathbb{R}^{n_{\mathcal{N}_i}}$, affect either the dynamics, constraints, or objective function of the i-th subsystem. Notice that $\mathcal{N}_i$ also contains the i-th subsystem. To simplify notation, we also define the projection matrices $X_i \in \{0,1\}^{n_i\times n}$, $X_{\mathcal{N}_i} \in \{0,1\}^{n_{\mathcal{N}_i}\times n}$ and $U_i \in \{0,1\}^{m_i\times m}$ such that

$$x_{i,t} = X_i x_t, \ x_{\mathcal{N}_i,t} = X_{\mathcal{N}_i} x_t \text{ and } u_{i,t} = U_i u_t. \tag{2a}$$

The i-th subsystem is now defined by state dynamics

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

constraints

$$x_{\mathcal{N}_i,t} \in \mathcal{X}_{\mathcal{N}_i} = \{x_{\mathcal{N}_i} \in \mathbb{R}^{n_{\mathcal{N}_i}} \ : \ G_{\mathcal{N}_i} x_{\mathcal{N}_i} \leq g_{\mathcal{N}_i}\}, \tag{2c}$$

$$u_{i,t} \in \mathcal{U}_i = \{u_i \in \mathbb{R}^{m_i} \ : \ H_i u_i \leq h_i\}, \tag{2d}$$

and objective function

$$J_\infty = \sum_{t=0}^{\infty} \sum_{i=1}^{M} \ell_i(x_{\mathcal{N}_i,t}, u_{i,t}), \tag{2e}$$

where $\ell_i(x_{\mathcal{N}_i,t}, u_{i,t}) = x_{\mathcal{N}_i,t}^\top Q_{\mathcal{N}_i} x_{\mathcal{N}_i,t} + u_{i,t}^\top R_i u_{i,t}$. Here, the matrices $A_{\mathcal{N}_i}$, B_i, $G_{\mathcal{N}_i}$, $g_{\mathcal{N}_i}$, H_i, h_i, $Q_{\mathcal{N}_i}$ and R_i are constructed as $A_{\mathcal{N}_i} = X_i A X_{\mathcal{N}_i}^\top$, $B_i = X_i B {U_i}^\top$, $G_{\mathcal{N}_i}$ are the non-zero rows of $G X_{\mathcal{N}_i}^\top$ with $g_{\mathcal{N}_i}$ the respective rows of g, H_i are the non-zero rows of $H X_i^\top$ with h_i the respective rows of h, $Q_{\mathcal{N}_i} = X_{\mathcal{N}_i} Q X_{\mathcal{N}_i}^\top$ and $R_i = U_i R {U_i}^\top$.

B. Centralized MPC formulation

In the spirit of MPC, we fix a prediction horizon T and a time window $\mathcal{T} = \{0, \dots, T-1\}$, and introduce a terminal cost $V : \mathbb{R}^n \to \mathbb{R}_+$ to approximate the original infinite-horizon objective function by

$$\widetilde{J}_\infty = V(x_T) + \sum_{t \in \mathcal{T}} \ell(x_t, u_t).$$

The terminal cost is chosen to upper bound the infinite-horizon cost associated with the terminal closed-loop system

$$x_{t+1} = (A + BK_c)x_t, \tag{3}$$

for all $t \geq T$. To enforce state and input constraints, we require x_T to lie in the set $\mathcal{X}_f \subseteq \mathcal{X}$ that is positive invariant under the terminal closed-loop dynamics in (3). In other words, for all $t \geq T$ and $x_t \in \mathcal{X}_f \subseteq \mathcal{X}$ then $(A + BK_c)x_t \in \mathcal{X}_f$ and $K_c x_t \in \mathcal{U}$. The following theorem establishes the necessary conditions for $V(\cdot)$ to be a Lyapunov function of this terminal closed-loop system.

Theorem 1 (Lyapunov stability [11, §3])**.** *If there exist functions $\sigma^1(\cdot)$, $\sigma^2(\cdot)$ and $\sigma^3(\cdot) \in \mathcal{K}_\infty$ such that for all $x \in \mathcal{X}_f$ it holds*

$$\sigma^1(\|x\|) \leq V(x) \leq \sigma^2(\|x\|), \tag{4a}$$

$$\sigma^3(\|x\|) \leq \ell(x, K_c x), \tag{4b}$$

$$V\big((A + BK_c)x\big) - V(x) \leq -\ell(x, K_c x), \tag{4c}$$

then the function $V(\cdot)$ is a Lyapunov function for the terminal closed-loop system in (3)*.*

Note that due to condition (4c), the control law $u_t = K_c x_t$ stabilizes the system and $V(\cdot)$ is an upper bound to the infinite-horizon cost. Here, we consider a terminal cost of the form

$$V(x) = x^\top P_c x, \tag{5}$$

where $P_c \in \mathbb{R}^{n \times n}$ is a positive definite matrix. Then, conditions (4a) and (4b) are satisfied by construction, while condition (4c) can be reformulated as a linear matrix inequality (LMI) of the form

$$\begin{bmatrix} I_n & 0 & 0 & Q^{1/2}E \\ 0 & I_m & 0 & R^{1/2}Y \\ 0 & 0 & E & AE + BY \\ (\star)^\top & (\star)^\top & (\star)^\top & E \end{bmatrix} \succeq 0, \tag{6}$$

where $E = P_c^{-1}$ and $Y = K_c P_c^{-1}$. To find K_c and P_c that satisfy the conditions of Theorem 1 and minimize the infinite-horizon cost of the unconstrained closed-loop system in (3), the problem of maximizing trace(E) under (6) is solved [29]. The solution to this problem, which is unique and known to match the one obtained by the discrete algebraic Riccati equation (DARE), implies that (4c) holds with equality; hence, the derived $V(\cdot)$ captures the actual infinite-horizon cost.

Given these K_c and P_c, a positive invariant set $\mathcal{X}_f$ needs to be computed to further guarantee state and input constraint satisfaction for the terminal closed-loop system in (3). For $\mathcal{X}_f$, two standard forms have been proposed in the literature, ellipsoidal and polytopic. Ellipsoidal sets of the form

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

where α is a positive scalar, are the most commonly used, since the problem of finding the maximum α such that $\mathcal{X}_f$ remains a positive invariant set can be cast as a linear optimization problem [29, §5.2]. However, ellipsoidal sets typically provide an inner approximation of the maximum positive invariant set, denoted by $\mathcal{X}_\infty$. Alternatively, one can consider polytopic invariant sets of the form

$$\mathcal{X}_f = \{x \in \mathbb{R}^n : A_f x \leq b_f\}.$$

Indeed, it is shown in [30] that for linear and stable systems confined by compact constraint sets containing the origin, as the terminal closed-loop system in (3), the maximum positive invariant set, $\mathcal{X}_\infty$, admits a polytopic form and can be computed through a finite iterative procedure, though the computation is often prohibitive for larger systems.

In summary, in its centralized form, the MPC optimization problem is given by

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

with optimization variables (x_t, u_t) for all $t \in \mathcal{T}$. Problem $(\mathcal{C})$ is solved at every time step and is designed to make the infinite-horizon problem in (1) amenable to finite-dimensional optimization. However, its optimal cost is greater and its feasibility set smaller compared to (1), making it a conservative approximation. Problem $(\mathcal{C})$ is not amendable to distributed computation algorithms since its terminal cost $V(\cdot)$ and set $\mathcal{X}_f$ will not in general respect the distributed structure afforded by the problem dynamics and constraints.

C. Distributed MPC formulation

To enforce decoupling along the system structure, the terminal controller of each i-th subsystem can be designed as

$$u_{i,t} = K_{\mathcal{N}_i} x_{\mathcal{N}_i,t},$$

where $K_{\mathcal{N}_i} \in \mathbb{R}^{m_i \times n_{\mathcal{N}_i}}$, such that its closed-loop dynamics are given by

$$x_{i,t+1} = (A_{\mathcal{N}_i} + B_i K_{\mathcal{N}_i}) x_{\mathcal{N}_i,t}. \tag{7}$$

In this setting, the system input, u_t, for all $t \geq T$ is formed as

$$u_t = \left(\sum_{i=1}^{M} U_i{}^\top K_{\mathcal{N}_i} X_{\mathcal{N}_i} \right) x_t = K_d x_t.$$

and the terminal closed-loop dynamics as

$$x_{t+1} = (A + BK_d) x_t. \tag{8}$$

For each i-th subsystem, we define

$$\widehat{V}_i(x_i) = x_i^\top P_i x_i,$$

where $P_i \in \mathbb{R}^{n_i \times n_i}$ is a positive definite matrix, such that the terminal cost associated with the terminal closed-loop system in (8), denoted by $\widehat{V}(\cdot)$ to distinguish it from the non-separable one in (5), is now given by

$$\begin{aligned} \widehat{V}(x_T) &= \sum_{i=1}^{M} \widehat{V}_i(x_{i,T}) \\ &= x_T^\top \left(\sum_{i=1}^{M} X_i^\top P_i X_i \right) x_T = x_T^\top P_d x_T. \end{aligned}$$

Similarly to the centralized case, to find $K_{\mathcal{N}_i}$ and P_i for all $i \in \mathcal{M}$, or equivalently K_d and P_d, that satisfy the stability conditions of Theorem 1 and minimize the infinite-horizon cost of the unconstrained closed-loop system in (8), the problem of maximizing trace(E) under (6), where now $E = P_d^{-1}$ and $Y = K_d P_d^{-1}$, is solved [29]. Unlike the centralized case, condition (4c) does not in general hold with equality because of the structure enforced on K_d and P_d. In other words, the resulting terminal cost, $\widehat{V}(\cdot)$, is an upper bound to the cost of operating the system under the static terminal controller $u_t = K_d x_t$ for all $t \geq T$.

Given K_d and P_d, a positive invariant set $\widehat{\mathcal{X}}_f$ needs to be computed such that the state and input constraints of the system are satisfied under the terminal closed-loop dynamics in (8). To retain the system's decoupled structure, $\widehat{\mathcal{X}}_f$ is designed as

$$\widehat{\mathcal{X}}_f = \widehat{\mathcal{X}}_{f,1} \times \cdots \times \widehat{\mathcal{X}}_{f,M},$$

with $\widehat{\mathcal{X}}_{f,i}$ ellipsoidal

$$\widehat{\mathcal{X}}_{f,i} = \{x_i \in \mathbb{R}^{n_i} : x_i^\top P_i x_i \leq \alpha_i\} \text{ (e.g., [23], [25], [26]),}$$

or polytopic

$$\widehat{\mathcal{X}}_{f,i} = \{x_i \in \mathbb{R}^{n_i} : A_{f,i} x_i \leq b_{f,i}\} \text{ (e.g., [16], [19]).}$$

In this distributed framework, the MPC optimization problem is given by

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

with optimization variables $(x_{\mathcal{N}_i,t}, u_{i,t})$ for all $i \in \mathcal{M}$, $t \in \mathcal{T}$. Problem ($\mathcal{D}$) exhibits the desired distributed structure which is amendable to distributed computation algorithms (e.g., the alternating method of multipliers [27]). However, Problem ($\mathcal{D}$) will in general be a conservative approximation, in terms of optimal cost and feasibility set, of Problem ($\mathcal{C}$) due to the structure imposed on its terminal cost and invariant set. Intuitively, if the terminal region is small then the effort needed to push x_T into it can be large or in some instances even not feasible. Besides structural restrictions, further conservativeness is introduced in the computation of the terminal set $\widehat{\mathcal{X}}_f$ by not considering the system constraints during the design of the distributed terminal controller K_d, although K_d directly affects the shape of $\widehat{\mathcal{X}}_f$. In current state-of-the art approaches [12]–[26], the design of K_d typically relies on satisfying the stability conditions stated in Theorem 1 while the computation of $\widehat{\mathcal{X}}_f$ is performed afterwards. In what follows, we propose a new method that allows us to couple the design of the stabilizing terminal controller K_d and positive invariant set $\widehat{\mathcal{X}}_f$ under the same DMPC optimization problem.

III. Adaptive distributed MPC

In this section, we consider the construction of an adaptive terminal set and show how its computation can become tractable by exploiting robust optimization techniques. Similar to the DMPC formulation described above, we calculate P_d by only considering Lyapunov stability, but allow K_d and $\widehat{\mathcal{X}}_f$ to adapt in every iteration based on the current and predicted state of the system. We introduce additional constraints to ensure that the underlying adaptive K_d stabilizes the terminal closed-loop system. We conclude our analysis by bringing parts together to formulate the proposed adaptive DMPC scheme for which we establish recursive feasibility and stability under receding horizon implementation.

A. Positive invariant terminal sets

For each $i \in \mathcal{M}$, we consider ellipsoidal terminal sets of the form

$$\widehat{\mathcal{X}}_{f,i}(\alpha_i) = \{x_i \in \mathbb{R}^{n_i} : x_i^\top Z_i x_i \leq \alpha_i\},$$

where Z_i is a predefined, fixed, positive definite matrix and α_i is a positive scalar decision variable. To ease exposition, we define the decision-dependent matrices $\alpha = \text{diag}(\alpha_1, \ldots, \alpha_M)$ and $\alpha_{\mathcal{N}_i} = X_{\mathcal{N}_i} \alpha X_{\mathcal{N}_i}^\top$, and the terminal set $\widehat{\mathcal{X}}_{f,\mathcal{N}_i}(\alpha_{\mathcal{N}_i}) = \bigtimes_{j \in \mathcal{N}_i} \widehat{\mathcal{X}}_{f,j}(\alpha_j)$. The following proposition provides the necessary conditions for $\widehat{\mathcal{X}}_{f,i}(\alpha_i)$ to be positive invariant.

Proposition 1. *If for each subsystem $i \in \mathcal{M}$ and for all $x_{\mathcal{N}_i} \in \widehat{\mathcal{X}}_{f,\mathcal{N}_i}(\alpha_{\mathcal{N}_i})$ the following conditions hold*

$$(A_{\mathcal{N}_i} + B_i K_{\mathcal{N}_i}) x_{\mathcal{N}_i} \in \widehat{\mathcal{X}}_{f,i}(\alpha_i), \tag{9a}$$

$$x_{\mathcal{N}_i} \in \mathcal{X}_{\mathcal{N}_i}, \tag{9b}$$

$$K_{\mathcal{N}_i} x_{\mathcal{N}_i} \in \mathcal{U}_i, \tag{9c}$$

then each set $\widehat{\mathcal{X}}_{f,i}(\alpha_i)$ is positive invariant under the i-th subsystem closed-loop dynamics in (7); hence, $\widehat{\mathcal{X}}_f(\alpha) = \widehat{\mathcal{X}}_{f,1}(\alpha_1) \times \cdots \times \widehat{\mathcal{X}}_{f,M}(\alpha_M)$ is also positive invariant under the system closed-loop dynamics in (8).

Proof. Conditions (9b) and (9c) guarantee that the state and input constraints of the i-th subsystem are satisfied. To prove that $\widehat{\mathcal{X}}_{f,i}(\alpha_i)$ is a positive invariant set, it remains to show that for all $x_{i,t} \in \widehat{\mathcal{X}}_{f,i}(\alpha_i)$ then $x_{i,t+1} \in \widehat{\mathcal{X}}_{f,i}(\alpha_i)$. This is indeed guaranteed by condition (9a) as long as the set $\bigtimes_{j\in\mathcal{N}_i\setminus\{i\}} \widehat{\mathcal{X}}_{f,j}(\alpha_j)$ is also positive invariant. The last argument directly follows since the Cartesian product of positive invariant sets is itself a positive invariant set, and condition (9a) is imposed simultaneously for all $i \in \mathcal{M}$. ■

Roughly speaking, the conditions of Proposition 1 are equivalent to assuming that each subsystem treats the mutual dependencies of its neighboring subsystems as bounded disturbances to its own dynamics. Under this assumption the terminal set $\widehat{\mathcal{X}}_{f,i}(\alpha_i)$ can be considered as a robust positive invariant set and the terminal controller $K_{\mathcal{N}_i}$ as a disturbance feedback. Thus, we employ robust optimization tools to express the Lyapunov stability and invariance conditions explicitly on the DMPC optimization problem in the form of LMIs that respect the existing coupling structure of the system.

In the sequel, we provide the reformulations of the individual robust constraints in Proposition 1 into LMIs. To do so, we make the substitution $Y_{\mathcal{N}_i} = K_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}$ where $Y_{\mathcal{N}_i}$ is a matrix of decision variables, and introduce the notational conventions $Z_{ij} = X_{\mathcal{N}_i}X_j^\top Z_j X_j X_{\mathcal{N}_i}^\top$, $P_{ij} = X_{\mathcal{N}_i}X_j^\top P_j X_j X_{\mathcal{N}_i}^\top$. Recall that Z_j and P_j are predefined fixed matrices. We now start by reformulating the invariance constraint in (9a).

Proposition 2. *For each $i \in \mathcal{M}$, condition*

$$(A_{\mathcal{N}_i} + B_i K_{\mathcal{N}_i})x_{\mathcal{N}_i} \in \widehat{\mathcal{X}}_{f,i}(\alpha_i) \text{ for all } x_{\mathcal{N}_i} \in \widehat{\mathcal{X}}_{f,\mathcal{N}_i}(\alpha_{\mathcal{N}_i}),$$

holds if there exists $\Lambda_i = [\lambda_{ij}]_{j\in\mathcal{N}_i} \in \mathbb{R}_+^{|\mathcal{N}_i|}$ such that

$$\begin{bmatrix} Z_i^{-1}\alpha_i^{1/2} & A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2} + B_i Y_{\mathcal{N}_i} \\ \star^\top & \sum_{j\in\mathcal{N}_i}\lambda_{ij}Z_{ij} \end{bmatrix} \succeq 0, \tag{10a}$$

and

$$\sum_{j\in\mathcal{N}_i}\lambda_{ij} \le \alpha_i^{1/2}. \tag{10b}$$

Proof. Parsing the expression of invariance for the i-th subsystem, we get

$$x_{\mathcal{N}_i}^\top(\star)^\top Z_i(A_{\mathcal{N}_i} + B_i K_{\mathcal{N}_i})x_{\mathcal{N}_i} \le \alpha_i \quad \text{for all } x_j^\top Z_j x_j \le \alpha_j \text{ with } j \in \mathcal{N}_i.$$

We use the auxiliary variables $s_i \in \mathbb{R}^{n_i}$ to make the substitutions $x_i = \alpha_i^{1/2}s_i$ for all $i \in \mathcal{M}$. Using this, the robust constraint above is equivalently written as

$$\begin{aligned} & s_{\mathcal{N}_i}^\top(\star)^\top Z_i(A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2} + B_i Y_{\mathcal{N}_i})s_{\mathcal{N}_i} \le \alpha_i \\ & \qquad \text{for all } s_j^\top Z_j s_j \le 1 \text{ with } j \in \mathcal{N}_i \\ \Leftrightarrow\ & s_{\mathcal{N}_i}^\top(\star)^\top Z_i\alpha_i^{-1/2}(A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2} + B_i Y_{\mathcal{N}_i})s_{\mathcal{N}_i} \le \alpha_i^{1/2} \\ & \qquad \text{for all } s_{\mathcal{N}_i}^\top Z_{ij} s_{\mathcal{N}_i} \le 1 \text{ with } j \in \mathcal{N}_i. \end{aligned}$$

Now using the S-lemma [29, §2.6.3], the robust constraint above holds if there exists $\lambda_{ij} \ge 0$ with $j \in \mathcal{N}_i$ such that

$$(\star)^\top Z_i\alpha_i^{-1/2}(A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2} + B_i Y_{\mathcal{N}_i}) \preceq \sum_{j\in\mathcal{N}_i}\lambda_{ij}Z_{ij},$$

and

$$\sum_{j\in\mathcal{N}_i}\lambda_{ij} \le \alpha_i^{1/2}.$$

By applying the Schur-complement, we obtain (10). ■

We continue by providing tractable approximations to conditions (9b) and (9c) of Proposition 1 that guarantee that the state and input constraints of the system are satisfied by the terminal controller. To do so, we denote the ℓ-th row (out of p_i rows) of the $G_{\mathcal{N}_i}$ and $g_{\mathcal{N}_i}$ state constraint matrices by $G_{\mathcal{N}_i}^\ell$ and $g_{\mathcal{N}_i}^\ell$, respectively. Similarly, we denote the ℓ-th row (out of k_i rows) of the H_i and h_i input constraint matrices by $H_{\mathcal{N}_i}^\ell$ and h_i^ℓ, respectively.

Proposition 3. *For each $i \in \mathcal{M}$, the state constraints*

$$G_{\mathcal{N}_i}x_{\mathcal{N}_i} \le g_{\mathcal{N}_i} \text{ for all } x_{\mathcal{N}_i} \in \widehat{\mathcal{X}}_{f,\mathcal{N}_i}(\alpha_{\mathcal{N}_i}),$$

hold, if there exists matrix $\Phi_i \in \mathbb{R}_+^{|\mathcal{N}_i|\times p_i}$ with ℓ-th column $\Phi_i^\ell = [\phi_{ij}^\ell]_{j\in\mathcal{N}_i}$ such that

$$\begin{bmatrix} g_{\mathcal{N}_i}^\ell & G_{\mathcal{N}_i}^\ell\alpha_{\mathcal{N}_i}^{1/2} \\ \star^\top & \sum_{j\in\mathcal{N}_i}\phi_{ij}^\ell Z_{ij} \end{bmatrix} \succeq 0, \tag{11a}$$

and

$$\sum_{j\in\mathcal{N}_i}\phi_{ij}^\ell \le g_{\mathcal{N}_i}^\ell, \tag{11b}$$

for all $\ell = 1,\dots,p_i$.

Similarly, the input constraints

$$H_i K_{\mathcal{N}_i}x_{\mathcal{N}_i} \le h_i \text{ for all } x_{\mathcal{N}_i} \in \widehat{\mathcal{X}}_{f,\mathcal{N}_i}(\alpha_{\mathcal{N}_i}),$$

hold, if there exists matrix $\Psi_i \in \mathbb{R}_+^{|\mathcal{N}_i|\times k_i}$ with ℓ-th column $\Psi_i^\ell = [\psi_{ij}^\ell]_{j\in\mathcal{N}_i}$ such that

$$\begin{bmatrix} h_i^\ell & H_i^\ell Y_{\mathcal{N}_i} \\ \star^\top & \sum_{j\in\mathcal{N}_i}\psi_{ij}^\ell Z_{ij} \end{bmatrix} \succeq 0, \tag{12a}$$

and

$$\sum_{j\in\mathcal{N}_i}\psi_{ij}^\ell \le h_i^\ell, \tag{12b}$$

for all $\ell = 1,\dots,k_i$.

Proof. For each $i \in \mathcal{M}$, the ℓ-th state constraint in (9b) is given by

$$G_{\mathcal{N}_i}^\ell x_{\mathcal{N}_i} \le g_{\mathcal{N}_i}^\ell \text{ for all } x_j^\top Z_j x_j \le \alpha_j \text{ with } j \in \mathcal{N}_i.$$

We consider the auxiliary variables $s_i \in \mathbb{R}^{n_i}$ and make the substitutions $x_i = \alpha_i^{1/2}s_i$ for all $i \in \mathcal{M}$. The robust constraint above is now equivalently written as

$$G_{\mathcal{N}_i}^\ell\alpha_{\mathcal{N}_i}^{1/2}s_{\mathcal{N}_i,t} \le g_{\mathcal{N}_i}^\ell \text{ for all } s_j^\top Z_j s_j \le 1 \text{ with } j \in \mathcal{N}_i.$$

It is easy to verify that in case of ellipsoidal sets the robust constraint above is equivalent to

$$\begin{aligned} & \|G_{\mathcal{N}_i}^\ell\alpha_{\mathcal{N}_i}^{1/2}s_{\mathcal{N}_i,t}\|_2 \le g_{\mathcal{N}_i}^\ell \\ & \qquad \text{for all } s_j^\top Z_j s_j \le 1 \text{ with } j \in \mathcal{N}_i \\ \Leftrightarrow\ & s_{\mathcal{N}_i,t}^\top(\star)^\top g_{\mathcal{N}_i}^{\ell\,-1}(G_{\mathcal{N}_i}^j\alpha_{\mathcal{N}_i}^{1/2})s_{\mathcal{N}_i,t} \le g_{\mathcal{N}_i}^\ell \\ & \qquad \text{for all } s_{\mathcal{N}_i}^\top Z_{ij}s_{\mathcal{N}_i} \le 1 \text{ with } j \in \mathcal{N}_i. \end{aligned}$$

Applying the S-lemma, this robust constraint holds if there exists $\phi_{ij}^\ell \ge 0$ with $j \in \mathcal{N}_i$ such that

$$(\star)^\top g_i^{\ell\,-1}G_{\mathcal{N}_i}^\ell(\alpha_{\mathcal{N}_i}^{1/2}) \preceq \sum_{j\in\mathcal{N}_i}\phi_{ij}^\ell Z_{ij},$$

and

$$\sum_{j\in\mathcal{N}_i}\phi_{ij}^{\ell}\le\alpha_i^{1/2}.$$

Then, we apply the Schur complement to obtain (11). Following similar derivation arguments, one can prove that the ℓ-th input constraint in (9c), that is equivalent to

$$H_i^{\ell}K_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}s_{\mathcal{N}_i,t}\le h_i^{\ell}\text{ for all }s_j^\top Z_js_j\le 1\text{ with }j\in\mathcal{N}_i,$$

holds, if (12) is satisfied. ∎

B. Stability of the terminal closed-loop system

To ensure stability of the terminal closed-loop system in (8), the adaptive K_d needs to be designed to also satisfy the stability conditions in Theorem 1, for a given Lyapunov matrix P_d. The non-decoupled structure of these conditions makes them unsuitable for explicit consideration in the formulation of Problem ($\mathcal{D}$). Instead, we adopt the notion of structured control Lyapunov functions, introduced in [31], which allow us to consider the conditions for stability in a way that respects the distributed structure of our system.

Theorem 2. *([31, §3.2]) If for each $i\in\mathcal{M}$ there exist functions $\sigma_i^1(\cdot)$, $\sigma_i^2(\cdot)$, $\sigma_i^3(\cdot)\in\mathcal{K}_\infty$, and $\gamma_i(x_{\mathcal{N}_i}):\mathbb{R}^{n_{\mathcal{N}_i}}\to\mathbb{R}$ such that for all $x_i\in\widehat{\mathcal{X}}_{f,i}(\alpha_i)$:*

$$\sigma_i^1(\|x_i\|)\le\widehat{V}_i(x_i)\le\sigma_i^2(\|x_i\|)\quad\text{(13a)}$$

$$\sigma_i^3(\|x_i\|)\le\ell_i(x_{\mathcal{N}_i},K_{\mathcal{N}_i}x_{\mathcal{N}_i})\quad\text{(13b)}$$

$$\begin{aligned}\widehat{V}_i((A_{\mathcal{N}_i}+B_iK_{\mathcal{N}_i})x_{\mathcal{N}_i})-\widehat{V}_i(x_i)&\\ \le-\ell_i(x_{\mathcal{N}_i},K_{\mathcal{N}_i}x_{\mathcal{N}_i})+\gamma_i(x_{\mathcal{N}_i})&\end{aligned}\quad\text{(13c)}$$

$$\sum_{i=1}^{M}\gamma_i(x_{\mathcal{N}_i})\le 0\quad\text{(13d)}$$

then $\widehat{V}(x_i)=\sum_{i=1}^{M}\widehat{V}_i(x_i)$ is a Lyapunov function for the terminal closed-loop system in (8).

Notice that Theorem 2 implies the more general Lyapunov stability Theorem 1, when that is formulated using K_d and P_d. Conditions (13a) and (13b) in Theorem 2 are satisfied by the terminal and stage costs constructions. Conditions (13c) and (13d) guarantee that the distributed terminal controllers stabilize the system and $\widehat{V}(\cdot)$ is an upper bound on the actual value function. Note that the definition above does not impose that each function $\widehat{V}_i(\cdot)$ is a control Lyapunov function for the corresponding subsystem in $\widehat{\mathcal{X}}_{f,i}(\alpha_i)$. Roughly speaking, this condition allows a local terminal cost to increase as long as at the same time the sum of all terminal costs in (13d) decreases.

Proposition 4. *Conditions* (13c) *and* (13d) *hold if there exists $\Pi_{\mathcal{N}_i}\in\mathbb{R}^{n_{\mathcal{N}_i}\times n_{\mathcal{N}_i}}$ such that*

$$\begin{bmatrix}\alpha_i^{1/2}I_{\mathcal{N}_i} & 0 & 0 & Q_{\mathcal{N}_i}^{1/2}\alpha_{\mathcal{N}_i}^{1/2}\\ 0 & \alpha_i^{1/2} & 0 & R_i^{1/2}Y_{\mathcal{N}_i}\\ 0 & 0 & P_i^{-1}\alpha_i^{1/2} & A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}+B_iY_{\mathcal{N}_i}\\ (\star)^\top & (\star)^\top & (\star)^\top & P_{ii}\alpha_i^{1/2}+\Pi_{\mathcal{N}_i}\end{bmatrix}\succeq 0.\quad\text{(14)}$$

and

$$\sum_{i=1}^{M}X_{\mathcal{N}_i}^\top\Pi_{\mathcal{N}_i}X_{\mathcal{N}_i}\preceq 0.\quad\text{(15)}$$

Proof. Condition (13c) formulated with $\gamma_i(x_{\mathcal{N}_i})=x_{\mathcal{N}_i}^\top\Gamma_{\mathcal{N}_i}x_{\mathcal{N}_i}$, where $\Gamma_{\mathcal{N}_i}\in\mathbb{R}^{n_{\mathcal{N}_i}\times n_{\mathcal{N}_i}}$, is written as

$$\begin{aligned}&x_{\mathcal{N}_i}^\top\big(P_{ii}-(\star)^\top P_i(A_{\mathcal{N}_i}+B_iK_{\mathcal{N}_i})-Q_{\mathcal{N}_i}-K_{\mathcal{N}_i}^\top R_iK_{\mathcal{N}_i}\\ &\quad+\Gamma_{\mathcal{N}_i}\big)x_{\mathcal{N}_i}\ge 0\text{ for all }x_j^\top Z_jx_j\le\alpha_j\text{ with }j\in\mathcal{N}_i.\end{aligned}$$

Once again, we use the auxiliary variable $s_i\in\mathbb{R}^{n_i}$ to make the substitution $x_i=\alpha_i^{1/2}s_i$. Then, the robust constraint above is equivalently written as

$$\begin{aligned}&s_{\mathcal{N}_i}^\top\Big(\alpha_{\mathcal{N}_i}^{1/2}P_{ii}\alpha_{\mathcal{N}_i}^{1/2}-(\star)^\top P_i(A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}+B_iY_{\mathcal{N}_i})\\ &\quad-\alpha_{\mathcal{N}_i}^{1/2}Q_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}-Y_{\mathcal{N}_i}^\top R_iY_{\mathcal{N}_i}+\alpha_{\mathcal{N}_i}^{1/2}\Gamma_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}\Big)s_{\mathcal{N}_i}\ge 0\\ &\qquad\text{for all }s_{\mathcal{N}_i}^\top Z_{ij}s_{\mathcal{N}_i}\le 1\text{ with }j\in\mathcal{N}_i.\end{aligned}$$

Making use of the substitutions $\alpha_{\mathcal{N}_i}^{1/2}P_{ii}\alpha_{\mathcal{N}_i}^{1/2}=\alpha_i^{1/2}P_{ii}\alpha_i^{1/2}$ and $\Pi_{\mathcal{N}_i}=\alpha_{\mathcal{N}_i}^{1/2}\Gamma_{\mathcal{N}_i}\alpha_i^{-1/2}\alpha_{\mathcal{N}_i}^{1/2}$, and then applying the S-lemma, we have that the robust constraint above holds, if there exist $\tau_{ij}\ge 0$ with $j\in\mathcal{N}_i$ such that

$$\left[\begin{array}{c|c}0 & 0\\ \hline 0 & \begin{array}{c}P_{ii}\alpha_i^{1/2}-(\star)^\top P_i\alpha_i^{-1/2}(A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}+B_iY_{\mathcal{N}_i})\\ -\alpha_{\mathcal{N}_i}^{1/2}Q_{\mathcal{N}_i}\alpha_i^{-1/2}\alpha_{\mathcal{N}_i}^{1/2}-Y_{\mathcal{N}_i}^\top R_i\alpha_i^{-1/2}Y_{\mathcal{N}_i}+\Pi_{\mathcal{N}_i}\end{array}\end{array}\right]\succeq\sum_{j\in\mathcal{N}_i}\tau_{ij}\begin{bmatrix}1 & 0\\ 0 & -Z_{ij}\end{bmatrix}.$$

This implies $\tau_{ij}=0$ for all $j\in\mathcal{N}_i$; hence, the matrix inequality constraint above is equivalently written as

$$\begin{aligned}&P_{ii}\alpha_i^{1/2}-(\star)^\top P_i\alpha_i^{-1/2}(A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}+B_iY_{\mathcal{N}_i})\\ &-\alpha_{\mathcal{N}_i}^{1/2}Q_{\mathcal{N}_i}\alpha_i^{-1/2}\alpha_{\mathcal{N}_i}^{1/2}-Y_{\mathcal{N}_i}^\top R_i\alpha_i^{-1/2}Y_{\mathcal{N}_i}+\Pi_{\mathcal{N}_i}\succeq 0.\end{aligned}$$

We use the Schur complement to write this expression as,

$$\begin{bmatrix}P_i^{-1}\alpha_i^{1/2} & (A_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}^{1/2}+B_iY_{\mathcal{N}_i})\\ (\star)^\top & P_{ii}\alpha_i^{1/2}+\Pi_{\mathcal{N}_i}\end{bmatrix}$$

$$-\begin{bmatrix}\star & \star\\ \star & \star\end{bmatrix}^\top\begin{bmatrix}\alpha_i^{-1/2}I_{\mathcal{N}_i} & 0\\ 0 & \alpha_i^{-1/2}\end{bmatrix}\begin{bmatrix}0 & Q_{\mathcal{N}_i}^{1/2}\alpha_{\mathcal{N}_i}^{1/2}\\ 0 & R^{1/2}Y_{\mathcal{N}_i}\end{bmatrix}\succeq 0.$$

Applying, once again, the Schur complement, leads to (14).

Finally, conditions (13d) is written as:

$$\begin{aligned}&x^\top\Big(\sum_{i=1}^{M}X_{\mathcal{N}_i}^\top\Gamma_{\mathcal{N}_i}X_{\mathcal{N}_i}\Big)x\le 0\\ &\qquad\text{for all }x_i^\top Z_ix_i\le\alpha_i\text{ with }i=1,\dots,M\\ \Leftrightarrow\;&s^\top\Big(\sum_{i=1}^{M}\alpha X_{\mathcal{N}_i}^\top\Gamma_{\mathcal{N}_i}X_{\mathcal{N}_i}\alpha\Big)s\le 0\\ &\qquad\text{for all }s_i^\top Z_is_i\le 1\text{ with }i=1,\dots,M\\ \Leftrightarrow\;&s^\top\Big(\sum_{i=1}^{M}X_{\mathcal{N}_i}^\top\alpha_{\mathcal{N}_i}\Gamma_{\mathcal{N}_i}\alpha_{\mathcal{N}_i}X_{\mathcal{N}_i}\Big)s\le 0\\ &\qquad\text{for all }s_i^\top Z_is_i\le 1\text{ with }i=1,\dots,M\\ \Leftrightarrow\;&s^\top\Big(\sum_{i=1}^{M}X_{\mathcal{N}_i}^\top\alpha_{\mathcal{N}_i}\Gamma_{\mathcal{N}_i}\alpha_i^{-1/2}\alpha_{\mathcal{N}_i}X_{\mathcal{N}_i}\Big)s\le 0\\ &\qquad\text{for all }s_i^\top Z_is_i\le 1\text{ with }i=1,\dots,M\\ \Leftrightarrow\;&s^\top\Big(\sum_{i=1}^{M}X_{\mathcal{N}_i}^\top\Pi_{\mathcal{N}_i}X_{\mathcal{N}_i}\Big)s\le 0\\ &\qquad\text{for all }s_i^\top Z_is_i\le 1\text{ with }i=1,\dots,M\end{aligned}$$

Using the S-lemma, we have that the robust constraints above holds if there exists $\rho_{ij} \geq 0$ with $j \in \mathcal{M}$ such that

$$\begin{bmatrix} 0 & 0 \\ 0 & \sum_{i=1}^{M} X_{\mathcal{N}_i}^\top \Pi_{\mathcal{N}_i} X_{\mathcal{N}_i} \end{bmatrix} \succeq \sum_{j\in\mathcal{M}} \rho_{ij} \begin{bmatrix} 1 & 0 \\ 0 & -X_j^\top Z_j X_j \end{bmatrix}$$

This implies $\rho_{ij} = 0$ for all $j \in \mathcal{N}_i$; hence, the matrix inequality constraint above can be written as an LMI of the form (15) which concludes the proof. ■

To be compatible with the LMI invariance and stability conditions presented previously, we rewrite condition $x_{i,T} \in \widehat{\mathcal{X}}_{f,i}(\alpha_i)$ in terms of the square root of the decision variable $\alpha_i^{1/2}$, given by

$$x_{i,T} \in \widehat{\mathcal{X}}_{f,i}(\alpha_i) \Leftrightarrow \begin{bmatrix} Z_i^{-1}\alpha_i^{1/2} & x_{i,T} \\ x_{i,T}^\top & \alpha_i^{1/2} \end{bmatrix} \succeq 0, \tag{16}$$

where the Schur complement is applied. To this end, we define for each $i \in \mathcal{M}$ the set

$$\widetilde{\mathcal{X}}_{f,i}(\alpha_{\mathcal{N}_i}, Y_{\mathcal{N}_i}, \Lambda_i, \Phi_i, \Psi_i, \Pi_{\mathcal{N}_i}) \\ = \{x_i \in \mathbb{R}^{n_i} : \text{Conditions (10) to (12) and (14) to (16) hold}\}.$$

C. Stability and recursive feasibility

The decentralized MPC problem with adaptive terminal sets is given by

$$\begin{aligned} \min \; & \sum_{i\in\mathcal{M}} \left(\widehat{V}_i(x_{i,T}) + \sum_{t\in\mathcal{T}} \ell_i(x_{\mathcal{N}_i,t}, u_{i,t}) \right) \\ \text{s.t. } & \left. \begin{aligned} & x_{i,t+1} = A_{\mathcal{N}_i} x_{\mathcal{N}_i,t} + B_i u_{i,t} \\ & (x_{\mathcal{N}_i,t}, u_{i,t}) \in \mathcal{X}_{\mathcal{N}_i} \times \mathcal{U}_i \\ & x_{i,T} \in \widetilde{\mathcal{X}}_{f,i}(\alpha_{\mathcal{N}_i}, Y_{\mathcal{N}_i}, \Lambda_i, \Phi_i, \Psi_i, \Pi_{\mathcal{N}_i}) \end{aligned} \right\} \forall i \in \mathcal{M}, \end{aligned} \tag{$\mathcal{AD}$}$$

with optimization variables ($x_{\mathcal{N}_i,t}$, $u_{i,t}$, $\alpha_{\mathcal{N}_i}$, $Y_{\mathcal{N}_i}$, Λ_i, Φ_i, Ψ_i, $\Pi_{\mathcal{N}_i}$) for all $i \in \mathcal{M}$, $t \in \mathcal{T}$.

Following similar arguments to [11], we now show that establishing stability and recursive feasibility for the closed-loop system (1) under the receding horizon MPC controller defined in Problem ($\mathcal{AD}$) is equivalent to requiring that Problem ($\mathcal{AD}$) is feasible for the initial system state.

Theorem 3. *The closed loop system formed by solving Problem* ($\mathcal{AD}$) *in receding horizon enjoys the following properties:*

1) *It is recursively feasible, in the sense that if Problem* ($\mathcal{AD}$) *is feasible for the initial condition* x_0 *it remains feasible throughout the closed loop system evolution;*
2) *It is asymptotically stable.*

Proof. Assume that the optimization Problem ($\mathcal{AD}$) is feasible at time $t = t_0$. Then, we obtain a sequence of optimal inputs $[u_{i,t_0}, \dots, u_{i,t_0+T-1}]$ for all $i \in \mathcal{M}$, referred as "optimal" sequence, which satisfy the state, input and terminal constraints of the problem. Since $\widetilde{\mathcal{X}}_{f,i}(\cdot)$ is a positive invariant set thanks to Proposition 1, the sequence of inputs $[u_{i,t_0+1}, \dots, u_{i,t_0+T-1}, K_{\mathcal{N}_i} x_{\mathcal{N}_i,t_0+T}]$ for all $i \in \mathcal{M}$, referred as "tail" sequence, is a feasible solution for Problem ($\mathcal{AD}$) at time $t = t_0 + 1$. Hence, if the optimization Problem ($\mathcal{AD}$) has a solution at time t_0 then it is guaranteed to have a solution at time $t_0 + 1$. Since any solution to Problem ($\mathcal{AD}$) enforces the terminal set to be positive invariant, recursive feasibility is preserved even though the terminal sets are adapting on the current state of the system.

To prove stability of Problem ($\mathcal{AD}$), define the objective function cost J_{t_0} at time t_0 as

$$J_{t_0} = \sum_{i\in\mathcal{M}} \left(\widehat{V}_i(x_{i,T}) + \sum_{t=t_0}^{t_0+T-1} \ell_i(x_{\mathcal{N}_i,t}, u_{i,t}) \right).$$

Let now $J_{t_0}^*$ be the cost at time t_0 when applying the "optimal" sequence and $\widehat{J}_{t_0+1}$ be the cost when applying the "tail" sequences from time $t_0 + 1$. Then, we have that

$$\begin{aligned} \widehat{J}_{t_0+1} - J_{t_0}^* &\leq \widehat{V}(x_{\mathcal{N}_i,t_0+T+1}) + \ell(x_{\mathcal{N}_i,t_0+T}, K_{\mathcal{N}_i} x_{\mathcal{N}_i,t_0+T}) \\ &\qquad - \widehat{V}(x_{\mathcal{N}_i,t_0+T}) - \ell(x_{t_0}, u_{t_0}) \\ &\leq -\ell(x_{t_0}, u_{t_0}) \end{aligned}$$

since $\widehat{V}_i(\cdot)$ is a Lyapunov function thanks to Proposition 4. Moreover, noting that $\widehat{J}_{t_0+1} \geq J_{t_0+1}^*$ due to the suboptimality of the tail sequences gives

$$J_{t_0+1}^* - J_{t_0}^* \leq -\ell(x_{t_0}, u_{t_0})$$

implying that J^* is a Lyapunov function for the closed-loop system (1) under the receding horizon DMPC controller defined in Problem ($\mathcal{AD}$). ■

We now summarize in Algorithm 1 the main steps involved towards implementing the proposed DMPC scheme.

Algorithm 1 Receding horizon implementation of DMPC scheme with adaptive terminal sets

Offline phase:
1: Calculate $P_d \succeq 0$ by solving the LMI in (6)
2: Choose and fix $Z_i \succeq 0$ for all $i \in \mathcal{M}$
Online phase:
3: Measure current state x_0
4: Solve Problem ($\mathcal{AD}$)
5: Apply u_{i,t_0} for all $i \in \mathcal{M}$

In an offline phase, we calculate the distributed Lyapunov function $\widehat{V}(\cdot)$ by solving the LMI in (6) using the structured P_d and K_d, and we also fix, by predefining Z_i, the shapes of the decentralized terminal sets, which introduces conservatism. It is common practice in literature to select $Z_i = P_i^{-1}$ [22]–[25], although, this might not be the least conservative choice, as discussed in [26]. Note that similar offline calculation steps are involved in most DMPC schemes presented in the literature [12]–[26]. Our method, however, does not consider the K_d corresponding to the calculated Lypaunov matrix P_d as the terminal controller used to compute the positive invariant set $\widehat{\mathcal{X}}_f$. Instead, at every iteration of the online phase, K_d and $\widetilde{\mathcal{X}}_f(\cdot)$ are adapted because (10) to (12) and (14) to (16) take the current and predicted system states into account. This unified framework allows for additional flexibility during the online phase by considering the synthesis of the stabilizing terminal controllers and the scaling of positive invariant terminal sets under the same DMPC optimization Problem ($\mathcal{AD}$). Although global information is needed in steps 1 and 4 of Algorithm 1, we remark that both steps are amendable to distributed computation, without the need for additional communication links among the agents; for instance, [25] provides a distributed algorithm to calculate P_d, and [27] uses consensus ADMM as a suitable distributed optimization algorithm to solve Problem ($\mathcal{AD}$). Comparing to Problem ($\mathcal{D}$), one would expect increased communication overhead with adaptive terminal sets due to the requirement that subsystems additionally agree on the scaling, α_i, of the terminal set. Thus, while in Problem ($\mathcal{D}$), each subsystem needs to reach consensus on n_iT variables with each neighbor, in Problem ($\mathcal{AD}$), consensus needs to be achieve for n_iT+1 variables. In this context, the number of communication rounds needed to reach consensus in a distributed implementation is likely to increase. Formally quantifying the increase is hard, however, this will depend on the network structure, the subsystems characteristics and on the problem initialization.

IV. Numerical examples

In this section, we conduct a number of simulation-based studies to assess the efficacy of the proposed DMPC formulation with adaptive terminal sets. We focus our attention on two examples: (i) an illustrative two-dimensional system that allow us to assess, numerically and graphically, the benefits of invariant terminal sets that can adapt to the current and predicted states of the system, and (ii) a spring-mass-damper system which is a common benchmark (e.g., see [16], [23], [25], [26]) for studying the scalability and the closed-loop behavior in distributed control.

A. Illustrative example

We consider a linear time-invariant system with dynamics

$$x_{t+1} = \begin{bmatrix} 5 & 0.1 \\ 0.3 & 0.9 \end{bmatrix} x_t + \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix} u_t, \qquad (17a)$$

where $x_t \in \mathbb{R}^2$ denotes the states and $u_t \in \mathbb{R}^2$ the inputs. The system is subject to linear state and input constraints

$$\begin{bmatrix} -5 \\ -5 \end{bmatrix} \leq x_t \leq \begin{bmatrix} 5 \\ 5 \end{bmatrix} \text{ and } \begin{bmatrix} -1 \\ -1 \end{bmatrix} \leq u_t \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad (17b)$$

and its goal is to minimize the infinite-horizon objective function

$$J_\infty = \sum_{t=0}^{\infty} \left(x_t^\top Q x_t + u_t^\top R u_t \right), \qquad (17c)$$

where $Q = \operatorname{diag}(1,1)$ and $R = \operatorname{diag}(0.1, 0.1)$. We split the system into two dynamically coupled subsystems with states $x_{1,t}, x_{2,t} \in \mathbb{R}$ and inputs $u_{1,t}, u_{2,t} \in \mathbb{R}$ such that $x_t = [x_{1,t} \;\; x_{2,t}]^\top$ and $u_t = [u_{1,t} \;\; u_{2,t}]^\top$. The dynamics, constraints and objective functions of these subsystems can straightforwardly be constructed through (17).

We approximate the infinite-horizon objective function by

$$\widetilde{J}_\infty = V(x_T) + \sum_{t=0}^{T-1} \left(x_t^\top Q x_t + u_t^\top R u_t \right),$$

where $V(\cdot)$ denotes the terminal cost. In the centralized MPC formulation, $V(x_T) = x_T^\top P_c x_T$, where $P_c \in \mathbb{R}^{2\times 2}$, together with the underlying K_c, is computed by solving the LMI in (6) derived by the stability condition in Theorem 1, leading to

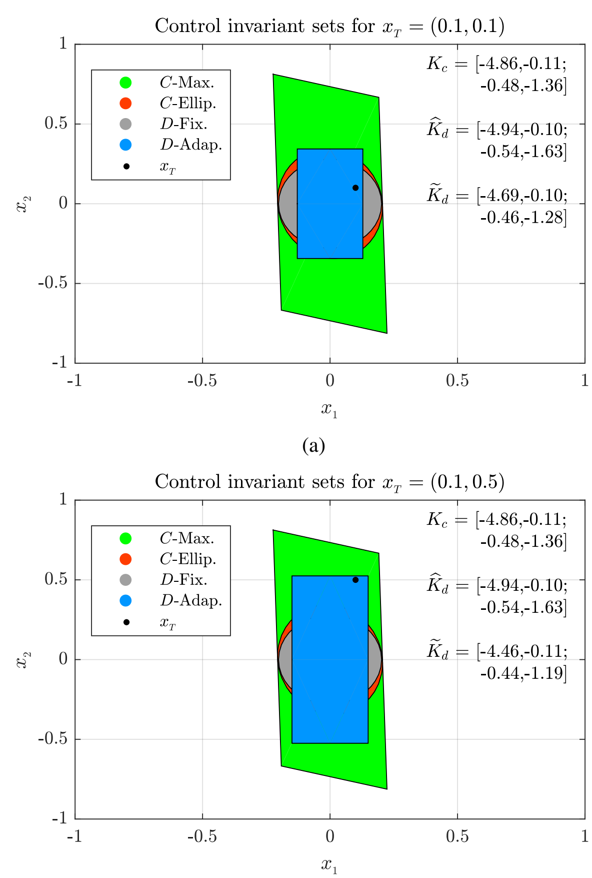
$$P_c = \begin{bmatrix} 3.46 & 0.13 \\ 0.13 & 1.25 \end{bmatrix} \text{ and } K_c = \begin{bmatrix} -4.86 & -0.11 \\ -0.48 & -1.36 \end{bmatrix}.$$

Given K_c, the maximum invariant terminal set, $\mathcal{X}_\infty$, is computed using routines developed in MPT 3.0 toolbox [32], while the ellipsoidal invariant terminal set, $\mathcal{X}_f$, is computed by solving a linear optimization problem, described in [29, §5.2]. We refer to Problem ($\mathcal{C}$) formulated with $\mathcal{X}_\infty$ as (C-Max.) and $\mathcal{X}_f$ as (C-Ellip.).

We compare these centralized formulations with the proposed distributed one where the terminal cost is given by $V(x_T) = x_{1,T}^\top P_{1,d} x_{1,T} + x_{2,T}^\top P_{2,d} x_{2,T} = x_T^\top P_d x_T$, where $P_{1,d}, P_{2,d} \in \mathbb{R}$. P_d, together with the underlying fixed distributed controller $\widehat{K}_d$, is computed by solving the LMI in (6) with $E = P_d^{-1}$ and $Y = \widehat{K}_d P^{-1}$ (c.f., step 1 of Algorithm 1), leading to

$$P_d = \begin{bmatrix} 8.07 & 0 \\ 0 & 4.25 \end{bmatrix} \text{ and } \widehat{K}_d = \begin{bmatrix} -4.94 & -0.10 \\ -0.54 & -1.63 \end{bmatrix}$$

As decentralized terminal sets, we consider two alternatives: (i) the fixed $\widehat{\mathcal{X}}_f = \{x \in \mathbb{R}^2 : x^\top P_d x \leq \alpha\}$, where α is a positive scalar computed by solving a linear optimization problem [29, §5.2]. We refer to Problem ($\mathcal{D}$) formulated with



(b)

Fig. 1. Shapes of $\mathcal{X}_\infty$ (green), $\mathcal{X}_f$ (red) $\widehat{\mathcal{X}}_f$ (gray) and $\widetilde{\mathcal{X}}_f(\alpha_1, \alpha_2)$ (blue) for different terminal states x_T.

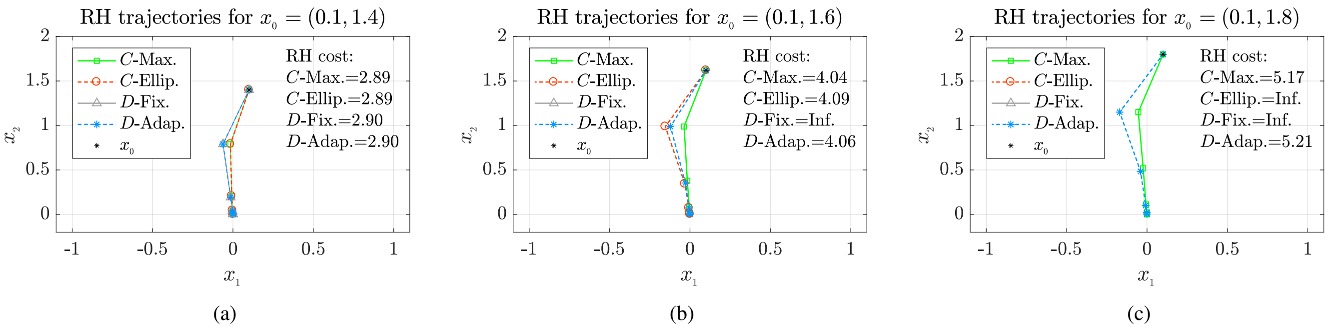


Fig. 2. Closed-loop receding horizon (RH) performance of the compared MPC schemes for different initial states x_0 and horizon $T = 2$.

$\widehat{\mathcal{X}}_f$ as (D-Fix.), and we remark that the cost of (D-Fix.) can be seen as a lower bound to the cost of the DMPC schemes in [23], [25] since $\widehat{\mathcal{X}}_f$ is an outer approximation to their optimally chosen decentralized sets; (ii) the adaptive terminal set $\widetilde{\mathcal{X}}_f(\alpha_1, \alpha_2) = \{(x_1, x_2) \in \mathbb{R}^2 : x_1^\top Z_1 x_1 \leq \alpha_1,\ x_2^\top Z_2 x_2 \leq \alpha_2\}$, where α_1, α_2 are positive scalars computed with the methods presented in Section III and $Z_1 = P_{1,d}$, $Z_2 = P_{2,d}$ (c.f., step 2 of Algorithm 1). In this adaptive process, a different feedback gain $\widetilde{K}_d$ is computed at each iteration that makes the corresponding $\widetilde{\mathcal{X}}_f(\alpha_1, \alpha_2)$ invariant (c.f., step 4 of Algorithm 1). We report the values of the different $\widetilde{K}_d$ in the numerical results below. Problem ($\mathcal{AD}$) formulated with $\widetilde{\mathcal{X}}_f(\alpha_1, \alpha_2)$ is referred as (D-Adap.). We formulate and solve the respective optimization problems using MATLAB with YALMIP and MOSEK on a computer equipped with 8 GB RAM and a single core 2.9 GHz processor.

In Figure 1, the shapes of $\mathcal{X}_\infty$ (green), $\mathcal{X}_f$ (red), $\widehat{\mathcal{X}}_f$ (gray) and $\widetilde{\mathcal{X}}_f(\alpha_1, \alpha_2)$ (blue) are depicted for different terminal states x_T. Note that $\widetilde{\mathcal{X}}_f(\alpha_1, \alpha_2)$ appears to be a box, because for this two dimensional example the combined terminal set is the product of two one dimensional, convex ellipsoidal sets. As convex ellipsoids are equivalent to intervals in one dimension, their product is equivalent to a box in two dimensions. If x_T lies outside the invariant terminal set, then the respective MPC problem is infeasible. That being said, this example highlights the ability of $\widetilde{\mathcal{X}}_f(\alpha_1, \alpha_2)$ to adapt and include terminal state x_T in its interior; hence, appropriately adapting the feasibility domain of Problem (D-Adap.). This adaptation also involves adjusting the values of the terminal controller $\widetilde{K}_d$, as reported in Figure 1.

To investigate the closed-loop behavior of the system for different initial conditions we choose a time horizon $T = 2$ and evaluate the performance of the system on a receding horizon implementation, i.e., repeating steps 3 to 5 of Algorithm 1. As a metric, the cost of operating the system until convergence within a ball of radius $\epsilon = 10^{-3}$ close to the origin is used. We report these comparison results in Figure 2. It can be observed that if the initial state x_0 is close to the origin, as in Figure 2(a), then MPC schemes (C-Max.) and (C-Ellip.) achieve the same cost, outperforming the decentralized approaches. This is not surprising since K_c is the optimal controller for the infinite horizon problem, thus, if the terminal state of (C-Max.) is inside the terminal set of (C-Ellip.), then the two approaches lead to the same solution. However, as the initial state is chosen further away from the origin, then higher control inputs are needed in the first two steps to ensure that the resulting x_T lies in the respective terminal set. For instance, in Figure 2(b), (D-Fix.) is infeasible while (D-Adap.) outperforms in terms of cost even the centralized MPC scheme (C-Ellip.). Note that (C-Max.) remains a strict lower bound on the achievable cost under comparable conditions, however, computing $\mathcal{X}_\infty$ offline is challenging and maybe prohibitive for more complex systems. The importance of considering adaptive terminal sets is further highlighted in Figure 2(c) where the DMPC scheme (D-Adap.) is feasible for initial states for which a solution for the centralized scheme (C-Ellip.) does not exist. This is attributed to the methods ability to modify the size of its terminal region by appropriately adapting its terminal controller $\widetilde{K}_d$ while satisfying the stability and invariance conditions.

B. Spring-mass-damper

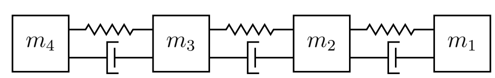


Fig. 3. A chain of four masses connected by springs and dampers.

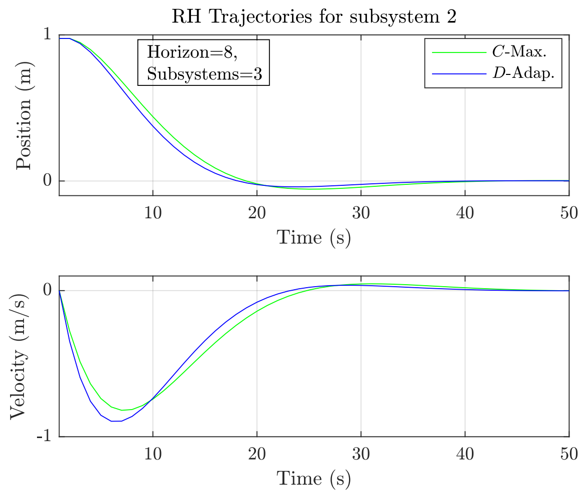
We now consider a series of masses that are connected by springs and dampers and arranged in a chain formation, exemplified in Figure 3. The values of the masses, spring constants and damping coefficients are chosen uniformly at random from the intervals $[5,\ 10]$kg, $[0.8,\ 1.2]$N/m and $[0.8,\ 1.2]$Kg/s, respectively. We assume that the i-th mass is a subsystem 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 $u_{i,t} \in \mathbb{R}$ an external force applied to it. We assume that the states and inputs are constrained by

$$\begin{bmatrix} -2 \\ -5 \end{bmatrix} \leq x_t \leq \begin{bmatrix} 2 \\ 5 \end{bmatrix} \text{ and } -u_c \leq u_t \leq u_c,$$

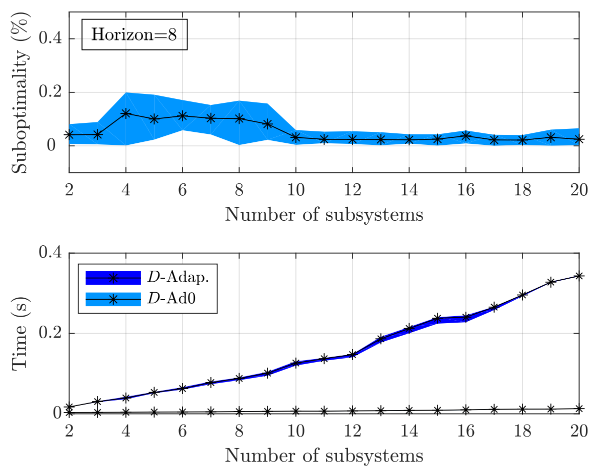
where u_c is chosen uniformly at random from the interval $[2,\ 4]$N. The subsystems are initially at rest and positioned uniformly at random within the intervals $[-2,\ -1.8]$m and $[1.8,\ 2]$m from their respective equilibrium positions.

The continuous-time dynamics of this interconnected dynamical system naturally admits a distributed structure. The prediction control model is obtained by the discretization of

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TAC.2019.2916774



(a)



(b)

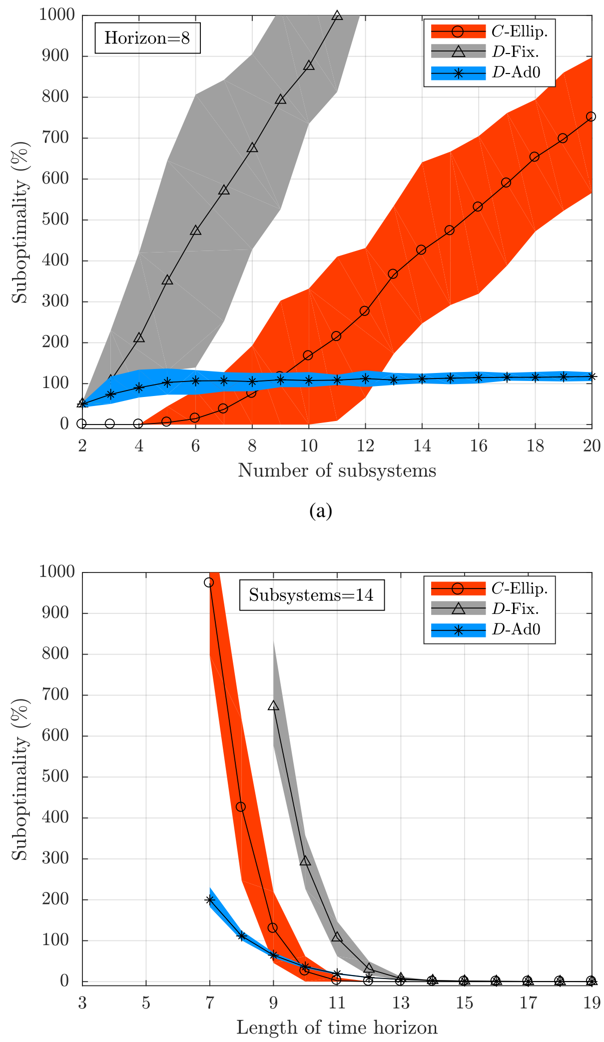
Fig. 4. (a) Receding horizon (RH) position and velocity trajectories for (C-Max.) and (D-Adap.) schemes; (b) Cost and solution time comparison for DMPC schemes with (D-Adap.) as basis for 100 randomly generated system instances. Spread of cost and solution time for the compared DMPC schemes are reported in their respective colors, while mean values in black.

the system's continuous dynamics using forward Euler with the sampling time 0.1s. Although inexact, Euler discretization is chosen 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 quadratic form (2e) with $Q_{\mathcal{N}_i} = \text{diag}(1, 1)$ and $R_i = 0.1$. Similar to Section IV-A, we formulate the respective LMIs from the Lyapunov conditions of Theorem 1 to compute P_c, K_c, $P_d = \text{diag}([P_{i,d}]_{i\in\mathcal{M}})$ with $P_{i,d} \in \mathbb{R}^2$, and $\widehat{K}_d$; then, we compute the sets $\mathcal{X}_\infty$, $\mathcal{X}_f$ and $\widehat{\mathcal{X}}_f$. We refer to Problem ($\mathcal{C}$) formulated with $\mathcal{X}_\infty$ as (C-Max.) and $\mathcal{X}_f$ as (C-Ellip.), Problem ($\mathcal{D}$) formulated with $\widehat{\mathcal{X}}_f$ as (D-Fix.), and Problem ($\mathcal{AD}$) with adaptive invariant terminal sets $\widetilde{\mathcal{X}}_f(\alpha)$, where $Z_i = P_{i,d}$ for all $i \in \mathcal{M}$, as (D-Adap.). We formulate and solve the respective optimization problems using MATLAB with YALMIP and MOSEK on a computer equipped with 8 GB RAM and a single core 2.9 GHz processor.

The performance of the system is evaluated on a receding horizon implementation. We use as a metric the cost of operating the system until convergence within a ball of radius $\epsilon = 10^{-3}$ close to the system's equilibrium state. Such a closed-loop simulation experiment for a system comprising three masses and a prediction horizon of $T = 8$ is shown in Figure 4(a) for the trajectories generated by the MPC (C-Max.) and DMPC (D-Adap.) schemes. We observe that these trajectories are very similar, indicating the proximity in performance between the centralized and distributed designs.

The proposed approach relies on the adaptation of the invariant terminal sets in each receding horizon simulation which involves the formulation and solution of a semi-definite program. To avoid the computational burden of this online phase (steps 3 to 5 in Algorithm 1), we compare the proposed fully adaptive method (D-Adap.) with its simplification (D-Ad0) in which the adaptation of the invariant terminal sets is only performed once at time $t = 0$ to account for the effect of the initial state of the system. Then, we enforce these computed terminal sets for the rest of the receding horizon simulations. That requires in step 4 of Algorithm 1 to only solve a quadratically constraints quadratic program instead of a semidefinite one. These two approaches are compared as the number of subsystems increases where for each topology we generate 100 random system instances. We use as metrics for this comparison the mean solution time and the cost of the receding horizon simulations until convergence to the origin is achieved. The results are reported in Figure 4(b). It is observed that adapting the invariant terminal sets in every iteration provides a slightly better solution quality with respect the case where the adaptation is only performed once at time $t = 0$. This is partially attributed to the dissipative nature of the spring-mass-damper system, the relatively small model mismatch and the absence of noise which make the initial displacement to be the determining factor for the shape of the terminal sets. Roughly speaking, once a feasible solution is found at $t = 0$, the receding horizon solution more or less follows the prediction you made at $t = 0$, hence, there is little need to adapt the sets to get feasibility. If there was noise or larger model mismatch, there would be occasional "outliers" with the system state to find itself far from the trajectory predicted at $t = 0$ during receding horizon implementation. At such times, it may be beneficial to adapt the sets again. If solve time permits, one could also do this in an event driven manner; if (D-Ad0) with the current terminal sets becomes infeasible, solve the full (D-Adap.) to get new sets. On the contrary, the computational benefit occurring when using the (D-Ad0) method is considerable since this simplified approach only requires a fraction of the time used by (D-Adap.) to generate the solution of the DMPC optimization problem.

To better quantify the performance comparison between the proposed adaptive DMPC approach and centralized MPC, we conducted several simulation experiments for systems with different horizons and number of subsystems. The comparison is performed on the suboptimality of the respective methods



(a)

(b)

Fig. 5. Cost comparison of different MPC schemes with (C-Max.) as basis for 100 randomly generated system instances as (a) the number of subsystems increases and (b) the time horizon of the MPC formulation increases. Spread of cost for each MPC scheme is reported in its respective color and mean value of cost in black.

during receding horizon simulations using as basis the cost associated with the centralized MPC Problem (C-Max.). Instead of (D-Adap.), we use the (D-Ad0); the online solution times are comparable for all four methods and not reported. Figure 5(a) shows the cost as the number of subsystems increases, with a simulation horizon of $T = 8$. We observe that the proposed method (D-Ad0) considerably outperforms even the centralized approach (C-Ellip.) as the number of subsystems increases. This is attributed to the proposed method's ability to adapt to the initial condition. As expected, the suboptimality gap increases with the number of subsystems. We note, however, that with (D-Ad0) the suboptimality gap remains bounded which indicates the efficiency of the proposed distributed design method. Finally, Figure 5(b) shows the cost associated with the length of prediction horizon for a system comprising fourteen masses. We observe that the increase of the horizon length results in cost convergence for the compared methods. Notably as the horizon increases the centralized methods outperform the decentralized ones. This is expected since large horizons make the use of terminal sets and costs obsolete as the system is capable of steering its states close enough to the equilibrium state within the prediction horizon.

V. Conclusion

In this paper, we presented a design approach for distributed cooperative MPC that encapsulates the design of the distributed terminal controller, cost and invariant set in the MPC formulation. Conditions for Lyapunov stability and invariance are imposed in the design problem in a way that allows the terminal cost and invariant set to admit the desired distributed structure. This allows the resulting distributed MPC problem to be amendable to distributed computation algorithms. The proposed distributed MPC method couples the design of the terminal stabilizing controllers and invariant terminal sets with the current and predicted states of the system. The closed-loop performance of the proposed distributed MPC approach is shown to outperform even the centralized MPC problem formulated with the ellipsoidal invariant terminal set for short prediction horizons.

Future work involves the theoretical investigation of the conjecture, verified in simulation, that the proposed method outperforms established centralized and distributed MPC approaches. In addition, we plan the extension of the proposed methodology to plug-and-play applications where only the new and a few of the existing distributed controllers need to be redesigned, and the conditions for Lyapunov stability and terminal set invariance need to be evaluated in a completely distributed way. In addition, the proposed method is not limited to distributed systems, where the decomposition is dictated by local information, neighbor communication, etc. It can also be seen as a way of exploiting structure in a centralized MPC to speed up or parallelize computation or capitalize on the potential advantage of using product of ellipsoids as terminal sets. In this sense it would be interesting to see whether one can develop methods to identify such beneficial decompositions when given a system.

Acknowledgment

The authors would like to thank Prof. Dr. Roy S. Smith for fruitful discussions on the topic.

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. S. Smith, and J. Lygeros, "The power of diversity: Data-driven robust predictive control for energy-efficient buildings and districts," *IEEE Trans. Control Syst. Technol.*, vol. PP, no. 99, pp. 1–14, 2017.
[3] 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.
[4] 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.

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

[6] 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.

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

[8] A. Eichler, C. Hoffmann, and H. Werner, "Robust control of decomposable lpv systems," *Automatica*, vol. 50, no. 12, pp. 3239–3245, 2014.

[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] D. Mayne, M. Seron, and S. Rakovic, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[13] A. Richards and J. P. How, "Robust distributed model predictive control," *Int. J. Control*, vol. 80, no. 9, pp. 1517–1531, 2007.

[14] P. Trodden and A. Richards, "Distributed model predictive control of linear systems with persistent disturbances," *Int. J. Control*, vol. 83, no. 8, pp. 1653–1663, 2010.

[15] 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.

[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] S. Lucia, M. Kögel, and R. Findeisen, "Contract-based predictive control of distributed systems with plug and play capabilities," *IFAC World Congress*, vol. 48, no. 23, pp. 205–211, 2015.

[18] P. Trodden and J. M. Maestre, "Distributed predictive control with minimization of mutual disturbances," *Automatica*, vol. 77, pp. 31–43, 2017.

[19] T. Keviczky, F. Borrelli, and G. Balas, "A study on decentralized receding horizon control for decoupled systems," in *Proc. Amer. Control Conf.*, vol. 6. IEEE, 2004, pp. 4921–4926.

[20] 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.

[21] B. Stewart, A. Venkat, J. Rawlings, S. Wright, and G. Pannocchia, "Cooperative distributed model predictive control," *Systems Control Letters*, vol. 59, no. 8, pp. 460–469, 2010.

[22] 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.

[23] 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.

[24] 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.

[25] C. Conte, C. N. Jones, M. Morari, and M. N. Zeilinger, "Distributed synthesis and stability of cooperative distributed model predictive control for linear systems," *Automatica*, vol. 69, pp. 117–125, 2016.

[26] G. Darivianakis, S. Fattahi, J. Lygeros, and J. Lavaei, "High-performance cooperative distributed model predictive control for linear systems," in *Proc. Amer. Control Conf.*, Milwaukee, USA, 2018, pp. –.

[27] 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.

[28] D. Simon, J. Löfberg, and T. Glad, "Reference tracking mpc using dynamic terminal set transformation," *IEEE Trans. Autom. Control*, vol. 59, no. 10, pp. 2790–2795, 2014.

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

[30] G. Gilbert and T. Tan, "Linear systems with state and control constraints: The theory and application of maximal output admissible sets," *IEEE Trans. Autom. Control*, vol. 36, no. 9, pp. 1008–1020, 1991.

[31] A. Jokic and M. Lazar, "On decentralized stabilization of discrete-time nonlinear systems," in *Proc. Amer. Control Conf.* IEEE, 2009, pp. 5777–5782.

[32] M. Herceg, M. Kvasnica, C. Jones, and M. Morari, "Multi-Parametric Toolbox 3.0," in *Proc. Eur. Control Conf.*, Zürich, Switzerland, July 17–19 2013, pp. 502–510.

Georgios Darivianakis received his Dipl.-Ing. degree in electrical and computer engineering from the National Technical University of Athens (NTUA), Greece in 2011 with the highest honour. He completed a M.Sc. degree in Robotics, Systems and Control in 2014 and a Ph.D. degree in Information Technology and Electrical Engineering in 2018, both at Swiss Federal Institute of Technology (ETH) Zurich, Switzerland. He is currently with the Power Conversion Systems Group at ABB's Corporate Research Center, Dättwil, Switzerland. His research interests include model predictive control, robust optimization, and control of large-scale distributed systems, with applications to aerial robots, energy-efficient buildings and districts, and power electronics.

Annika Eichler received the B.Sc. degree in General Engineering Science, in 2008, and the Dipl.-Ing. degree in Mechatronics, in 2011, from the Hamburg University of Technology, Germany, from where she received her Ph.D. degree, in 2015, at the Institute of Control Systems. From 2015 to 2017, she was a Post-Doctoral Researcher at the Automatic Control Laboratory, ETH Zurich, Switzerland, where she was working as Senior Researcher until 2019 focusing on the application of energy systems. Currently she is Senior Researcher at DESY, Hamburg, Germany. Her research interests include distributed, data-driven and robust control applying tools from model predictive control, dynamic programming and robust optimization.

John Lygeros completed a B.Eng. degree in electrical engineering in 1990 and an M.Sc. degree in Systems Control in 1991, both at Imperial College of Science Technology and Medicine, London, U.K.. In 1996 he obtained a Ph.D. degree from the Electrical Engineering and Computer Sciences Department, University of California, Berkeley. After postdoctoral researcher appointments at M.I.T. and U.C. Berkeley in 2000 he joined the Department of Engineering, University of Cambridge, U.K., as a Lecturer and Churchill College as a Fellow. Between 2003 and 2006 he was an Assistant Professor at the Department of Electrical and Computer Engineering, University of Patras, Greece. In July 2006 he joined the Automatic Control Laboratory at ETH Zurich, first as an Associate Professor, and since January 2010 as a Full Professor. Since 2009 he is serving as the Head of the Automatic Control Laboratory; between 2015 and 2018 he served as the Head of the Department of Information Technology and Electrical Engineering. His research interests include modelling, analysis, and control of hierarchical, hybrid, and stochastic systems, with applications to biochemical networks, automated highway systems, air traffic management, power grids and camera networks. John Lygeros is a Fellow of the IEEE, and a member of the IET and the Technical Chamber of Greece; since 2013 he is serving as the Treasurer of the International Federation of Automatic Control and a member of the IFAC Council.