Automatica 95 (2018) 481–487

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

An LMI-based approach to distributed model predictive control design for spatially-interconnected systems[☆]

Qin Liu[a], Hossam Seddik Abbas[b], Javad Mohammadpour Velni[a,*]

[a] School of Electrical and Computer Engineering, College of Engineering, University of Georgia, Athens, GA 30602, USA
[b] Electrical Engineering Department, Faculty of Engineering, Assiut University, Assiut 71515, Egypt

ARTICLE INFO

Article history:
Received 30 June 2016
Received in revised form 24 December 2017
Accepted 25 May 2018

Keywords:
Distributed model predictive control
Spatially-interconnected systems
Linear matrix inequalities
Consistency constraints

ABSTRACT

This paper proposes a new framework to distributed model predictive control (MPC) design for linear time- and space-invariant (LTSI) distributed systems subject to constraints. Given a two-dimensional, input–output model that describes the distributed dynamics among the subsystems, it is shown that a non-minimal state space realization leads to numerically tractable linear matrix inequality (LMI) based terminal state feedback controller design. The local online optimization problem is defined at the subsystem level with subsystems exchanging predictions through coupled states and can be solved in parallel at all subsystems non-iteratively. Stability and recursive feasibility are guaranteed in the presence of one-step delayed exchanging information among subsystems by imposing consistency constraints and terminal constraints. Attributed to the non-minimal state space realization, input–output properties are preserved in the MPC formulation, and hence no state estimator is needed for the online implementation. Simulation results using a heat equation demonstrate a satisfactory performance of the proposed distributed MPC design compared to centralized MPC schemes.

1. Introduction

Distributed model predictive control (MPC) design has attracted increasing attention for control of large-scale systems subject to constraints and time delays. This class of systems often consists of a large number of interconnected subsystems. Distributed MPC design decomposes a centralized optimization problem into a number of subsystem-based problems of smaller size and thus results in a lower computational complexity.

Considered here is the class of distributed parameter systems, whose underlying temporal and spatial dynamics are governed by partial differential equations (PDEs). Instead of preserving the continuous nature in space and the resulting infinite dimensions, the attachment of actuators and sensors induces a spatial discretization, so that the overall system can be treated as a physical interaction of subsystems on lattices, in which each subsystem is capable of actuating and sensing. If the spatially-discretized subsystems are homogeneous, and the interconnection of subsystems is periodic or infinite (see D'Andrea & Dullerud, 2003), we call these systems linear time- and space-invariant (LTSI). Obviously, infinite interconnected systems do not exist in practice. The invariant property can be affected by the boundary conditions due to the finite extent (Langbort & D'Andrea, 2005). Nevertheless, it has been experimentally validated in Liu and Werner (2013) that the dynamics of an interconnected system of a sufficiently large scale can be approximated by an LTSI model due to that the energy dissipated along space allows the space truncation. This supports to consider here rather general LTSI systems of periodic or finite interconnection, whose applications include PDE-governed road traffic systems (De Schutter, Hellendoorn, Hegyi, van den Berg, & Zegeye, 2010), flexible structures (Liu & Werner, 2016), distribution of heat or fluid in a given region, and irrigation canals (Negenborn & De Schutter, 2008) among many others.

In this work, we address the distributed MPC design problem for LTSI systems by extending the stability framework in Mayne, Rawlings, Rao, and Scokaert (2000). This paper is a follow-up to our previous work (Liu, Abbas, Mohammadpour, Wollnack, & Werner, 2016), where a non-iterative distributed MPC design approach for LTSI systems in input–output (I–O) form has been reported. In Liu et al. (2016), the terminal controller design in I–O form results in synthesis conditions in the form of bilinear matrix inequalities (BMIs). Although the *D-K* iteration approach (Ghaoui & Niculescu, 2000) has demonstrated its efficiency in solving BMI constraints,

[☆] The material in this paper was partially presented at the 2016 American Control Conference, July 6–8, 2016, Boston, MA, USA. This paper was recommended for publication in revised form by Associate Editor Tamas Keviczky under the direction of Editor Christos G. Cassandras.

* Corresponding author.
E-mail addresses: qin.liu@tuhh.de (Q. Liu), hossam.abbas@aun.edu.eg (H.S. Abbas), javadm@uga.edu (J. Mohammadpour Velni).

https://doi.org/10.1016/j.automatica.2018.06.024
0005-1098/

finding a stabilizing initial controller could be troublesome especially when dealing with complex systems. As the *first main contribution* of this paper, given the identified two-dimensional (in terms of time and space) I–O model of a spatially-distributed system, a non-minimal state space model can be constructed whose states are comprised of time- and space-shifted inputs and outputs. The advantages of this novel framework are threefold. Firstly, instead of solving BMI synthesis conditions as in Liu et al. (2016), the offline design of the terminal control law – a distributed state feedback controller – can be realized by solving computationally tractable linear matrix inequalities (LMIs) defined at the subsystem level. Secondly, given that the state vector consists of measurable time- and space-shifted inputs and outputs, no state estimator is required for the online implementation of the proposed MPC approach. Thirdly, due to its non-minimality, the state space model allows a fixed-structure terminal controller design; it gives the user the degrees of freedom to pre-specify the localized structure of the terminal control law.

It is well known that the parallel implementation of the local optimization problem of a distributed MPC results in one-step delayed exchanging information among the subsystems, due to that each subsystem exchanges the predicted state trajectory with its neighboring subsystems only once after the optimization problem is solved. As the *second contribution* of this paper, sufficient conditions for asymptotic stability and recursive feasibility of the closed-loop system are derived. Modified consistency constraints inspired by Dunbar (2007) and Keviczky, Borrelli, and Balas (2006) are developed to ensure that the actual state trajectory of any subsystem deviates not far from the trajectory assumed by its neighbors.

This paper is structured as follows: Section 2 presents the derivation of a non-minimal state space realization and the statement of a distributed MPC optimization problem. The offline computation of the state feedback controller and the terminal sets, as well as the formulation of the online optimization problem are presented in Section 3. Simulation results obtained from a heat equation are demonstrated in Section 4. Finally, conclusions are drawn in Section 5.

Notations: An identity matrix of dimension n is denoted by $I_n \in \mathbb{R}^{n\times n}$. We use $\mathbb{S}^n$ to define a symmetric matrix of dimension $n \times n$. The transpose of a block matrix or a vector is denoted by $*$. The Kronecker product is denoted by $\otimes$. Finally, $\|x\|_2$ denotes the vector 2-norm.

2. Preliminaries

2.1. Representation of distributed systems in I–O Form

Unlike temporal systems, whose signals are only functions of time, spatially-distributed systems are multidimensional. Involved signals in one spatial dimension are indexed by time k and the spatial variable s, e.g., $u(k,s)$. The localized dynamics of the subsystem s can be described as

$$\mathcal{G}(q_t,q_s): \quad \mathcal{A}(q_t,q_s)y(k,s) = \mathcal{B}(q_t,q_s)u(k,s), \tag{1}$$

where q_t and q_s are forward temporal and spatial shift operators, respectively, i.e., $q_t^{-1}q_s y(k,s) = y(k-1,s+1)$, and the polynomials $\mathcal{A}(q_t,q_s)$ and $\mathcal{B}(q_t,q_s)$ are defined as

$$\mathcal{A}(q_t,q_s) = 1 + \sum_{i_k=1}^{n_a}\sum_{i_s=-m_a}^{m_a} a_{(i_k,i_s)} q_t^{-i_k} q_s^{-i_s}, \tag{2a}$$

$$\mathcal{B}(q_t,q_s) = \sum_{j_k=1}^{n_b}\sum_{j_s=-m_b}^{m_b} b_{(j_k,j_s)} q_t^{-j_k} q_s^{-j_s}, \tag{2b}$$

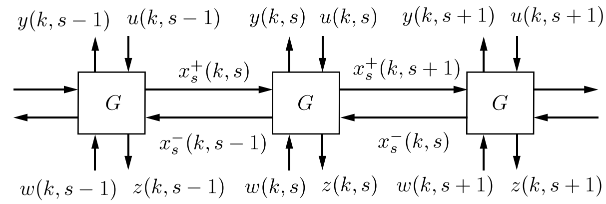


Fig. 1. Distributed systems in one-dimensional space.

where n_a, m_a, n_b, and m_b are the indices of time- and space-shifted outputs and inputs, respectively, $a_{(i_k,i_s)}$ and $b_{(j_k,j_s)}$ are constant coefficients. In this work, we focus on subsystems with single input and single output.

Let $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{B}}$ denote vectors of polynomial coefficients of $\mathcal{A}(q_t,q_s)$ and $\mathcal{B}(q_t,q_s)$, respectively. The I–O model (1) can then be rewritten as

$$\mathcal{G}(q_t,q_s): \quad \tilde{\mathcal{A}}\xi^y(k,s) = \tilde{\mathcal{B}}\xi^u(k,s), \tag{3}$$

where $\xi^y(k,s) \in \mathbb{R}^{n_{\xi^y}}$ and $\xi^u(k,s) \in \mathbb{R}^{n_{\xi^u}}$ are the vectors of time- and space-shifted outputs and inputs, respectively.

2.2. Non-minimal state space realization

Due to the localized dynamics of subsystems, each signal vector $\xi^i(k,s)$ $(i = y,u)$ in (3) can be partitioned into three parts as

$$\xi^i(k,s) = \begin{bmatrix} \xi_t^i(k,s) \\ \xi_s^{+,i}(k,s) \\ \xi_s^{-,i}(k,s) \end{bmatrix}, \tag{4}$$

where $\xi_t^i(k,s)$ (with $i_s = 0, j_s = 0$ in (2)) are the temporal variables, and $\xi_s^{+,i}(k,s)$ (with $i_s > 0, j_s > 0$) and $\xi_s^{-,i}(k,s)$ (with $i_s < 0, j_s < 0$) denote the positive and negative spatial variables, respectively.

Temporal variables consist of past inputs and outputs of subsystem s, while spatial variables are the received exchanging information from its nearest neighbors. Furthermore, by introducing an augmented shifted operator $\boldsymbol{\Delta} = \mathrm{diag}\{q_t I_{n_t}, q_s I_{n_{s+}}, q_s^{-1} I_{n_{s-}}\}$, a state-like vector $x(k,s) \in \mathbb{R}^{n_x}$ and its shifted version $\boldsymbol{\Delta}x(k,s) \in \mathbb{R}^{n_x}$ can be extracted from $\xi(k,s) = \begin{bmatrix}\xi^{y,\mathrm{T}}(k,s) & \xi^{u,\mathrm{T}}(k,s)\end{bmatrix}^{\mathrm{T}}$ by grouping the temporal, positive and negative spatial variables together, i.e.,

$$x(k,s) = \begin{bmatrix} x_t(k,s) \\ x_s^+(k,s) \\ x_s^-(k,s) \end{bmatrix}, \ \boldsymbol{\Delta}x(k,s) = \begin{bmatrix} x_t(k+1,s) \\ x_s^+(k,s+1) \\ x_s^-(k,s-1) \end{bmatrix}, \tag{5}$$

with $x_t(k,s) \in \mathbb{R}^{n_t}$, $x_s^+(k,s) \in \mathbb{R}^{n_{s+}}$ and $x_s^-(k,s) \in \mathbb{R}^{n_{s-}}$. To ensure size compatibility, let $n_{\xi^u} = n_{\xi^y}$ by filling coefficients of the lower order polynomials with zeros.

A generalized non-minimal state space model of the subsystem s can then be constructed as

$$\begin{bmatrix} x_t(k+1,s) \\ x_s^+(k,s+1) \\ x_s^-(k,s-1) \\ \hline z(k,s) \\ y(k,s) \end{bmatrix} = \left[\begin{array}{ccc|cc} A_{tt} & A_{ts}^+ & A_{ts}^- & B_{w,t} & B_t \\ A_{st}^+ & A_{ss}^{++} & A_{ss}^{+-} & B_{w,s}^+ & B_s^+ \\ A_{st}^- & A_{ss}^{-+} & A_{ss}^{--} & B_{w,s}^- & B_s^- \\ \hline C_{t,z} & C_{s,z}^+ & C_{s,z}^- & D_{zw} & D_z \\ C_{t,y} & C_{s,y}^+ & C_{s,y}^- & D_{yw} & D_y \end{array}\right] \begin{bmatrix} x_t(k,s) \\ x_s^+(k,s) \\ x_s^-(k,s) \\ \hline w(k,s) \\ u(k,s) \end{bmatrix}$$
$$:= \left[\begin{array}{c|cc} A & B_w & B \\ \hline C_z & D_{zw} & D_z \\ C_y & D_{yw} & D_y \end{array}\right] \begin{bmatrix} x(k,s) \\ \hline w(k,s) \\ u(k,s) \end{bmatrix}, \tag{6}$$

where $w(k,s)$ denotes generalized exogenous inputs, and $z(k,s)$ denotes the performance output. Subsystems exchange information through the spatial states $x_s(k,s)$ as shown in Fig. 1.

2.3. Distributed MPC formulation

Let the overall system be spatially-discretized into N_s subsystems. The following assumptions are made for the distributed MPC design:

(1) There are no disturbance and noise present in the closed-loop system, as well as no modeling error and uncertainty in the plant model.
(2) The subsystems are only subject to control constraints and the imposed constraints are identical, i.e., $\mathbb{U}^1 = \cdots = \mathbb{U}^{N_s}$, with $\mathbb{U}^s = \{u(k,s) \in \mathbb{R} | -\bar{u} \le u(k,s) \le \bar{u}\}$, where $\bar{u}$ ($\bar{u} > 0$) denotes the upper control limit.

The global control constraint set and the terminal set are then the Cartesian products of the subsystem ones, i.e.,

$$\mathbb{U} = \prod_{s=1}^{N_s} \mathbb{U}^s, \quad \mathbb{X}_f = \prod_{s=1}^{N_s} \mathbb{X}_f^s. \tag{7}$$

Let the control and the prediction horizons be the same and denoted as N. The distributed MPC optimization problem subject to control constraints is formulated as

$$\min_{\mathbf{u}(k,s)} J(k,s) = \sum_{i=0}^{N-1} \big([*]Qx_t(k+i|k,s) + [*]Ru(k+i|k,s)\big) + V_f(x_t(k+N|k,s)) \tag{8a}$$

$$\text{s.t.} \quad u(k+i|k,s) \in \mathbb{U}^s, \quad i = 0, 1, \ldots, N-1 \tag{8b}$$

$$x_t(k+N|k,s) \in \mathbb{X}_f^s, \tag{8c}$$

where $\mathbf{u}(k,s) := [u(k|k,s) \ldots u(k+N-1|k,s)]^{\mathrm{T}}$ is the optimization variable of the subsystem s at time k, $Q \in \mathbb{R}^{n_x \times n_x}$ and $R \in \mathbb{R}$ are positive-definite weights, and $V_f(\cdot)$ is the terminal cost defined as a positive-definite function of the terminal temporal state $x_t(k+N|k,s)$.

3. Distributed MPC design

In this section, we present the offline computation of the terminal controller and the terminal sets, as well as the online optimization problem for solving the proposed distributed MPC.

3.1. Offline terminal controller design

Consider now a Lyapunov function candidate as a function of the temporal state as

$$V(x(k,s)) = x_t^{\mathrm{T}}(k,s) P_t x_t(k,s), \tag{9}$$

where $P_t \in \mathbb{S}^{n_t}$ is a positive-definite Lyapunov matrix.

The objective of the terminal controller design is to find a distributed state feedback controller K over the terminal set $\mathbb{X}_f^s$ which takes the form

$$u(k,s) = Kx(k,s) := \begin{bmatrix} K_t & K_s^+ & K_s^- \end{bmatrix} x(k,s). \tag{10}$$

Due to that the state $x(k,s)$ is a vector of measurable inputs and outputs, the control law can be computed in a distributed fashion without the need of designing an observer.

Based on the results of Liu and Werner (2013), the stability condition can be derived independently of the number of subsystems as follows.

Theorem 1. *A distributed system in the form of (6) (with $w = 0$ for regulation problem) controlled by a distributed state feedback control law in the form of (10) is asymptotically stable and the energy of the subsystems decreases monotonically, if there exists a symmetric matrix P such that the following conditions are satisfied:*

$$P = \begin{bmatrix} P_t & 0 \\ 0 & P_s \end{bmatrix}, \quad P_t > 0, \quad \det(P_s) \ne 0, \tag{11a}$$

$$[*]P \begin{bmatrix} x_t(k+1,s) \\ x_s^+(k,s+1) \\ x_s^-(k,s) \end{bmatrix} - [*]P \begin{bmatrix} x_t(k,s) \\ x_s^+(k,s) \\ x_s^-(k,s-1) \end{bmatrix} \le -[*](Q + K^T RK)x(k,s), \tag{11b}$$

$$[*] \begin{bmatrix} P_t & 0 \\ 0 & -P_t \end{bmatrix} \begin{bmatrix} x_t(k+1,s) \\ x_t(k,s) \end{bmatrix} < 0. \tag{11c}$$

Proof. The proof of the stability condition under the terminal control law (10) follows immediately from Wu (2003). The monotonic decrease of the subsystem energy requires $\Delta V(x(k,s)) < 0$, which leads to (11c).

Due to the indefinite spatial Lyapunov matrix P_s, the controller synthesis problem in discrete time and space associated with Theorem 1 can only be solved in terms of BMIs (see Liu et al., 2016). However, this issue can be avoided by converting the discrete problem to its continuous equivalent. In the following, we show how this conversion leads to LMI-based synthesis conditions.

Analogous to the standard linear quadratic regulator (LQR) problem, we define

$$C_y = I, D_y = D_{yw} = 0, C_z = \begin{bmatrix} C_{z1} \\ 0 \end{bmatrix}, D_z = \begin{bmatrix} 0 \\ I \end{bmatrix}, D_{zw} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

with $C_{z1} = Q^{\frac{1}{2}}$.

A bilinear transformation (two-dimensional Tustin approximation, see D'Andrea and Dullerud 2003; Wu 2003) can be applied to convert the discrete system (6) into its continuous counterpart. The closed-loop system controlled by a continuous state feedback gain $\bar{K}$ is given by

$$\begin{bmatrix} \bar{\boldsymbol{\Delta}}\bar{x}(t,s) \\ \bar{z}_1(t,s) \\ \bar{z}_2(t,s) \end{bmatrix} = \left[\begin{array}{c|c} \bar{A} + \bar{B}\bar{K} & \bar{B}_w \\ \hline \bar{C}_{z1} + \bar{D}_{z1}\bar{K} & 0 \\ \bar{K} & 0 \end{array}\right] \begin{bmatrix} \bar{x}(t,s) \\ \hline \bar{w}(k,s) \end{bmatrix}, \tag{12}$$

in which t indicates continuous time, and $\bar{\boldsymbol{\Delta}}$ denotes the augmented shift operator for continuous systems. By a slight abuse of notation, we use s to denote the space index for both continuous and discrete systems. From here on, the overhead bar indicates variables or matrices of a (time and space) continuous system, and otherwise, a (time and space) discrete system.

Synthesis conditions of a distributed state feedback controller $\bar{K}$ in continuous time and space are stated as follows.

Theorem 2. *There exists a distributed state feedback controller in the form of (10) that asymptotically stabilizes the closed-loop system in the form of (12) (with $\bar{w}(k,s) = 0$ for regulation problem), and the energy of the subsystems decreases monotonically, if there exists a symmetric matrix Z such that*

$$Z = \begin{bmatrix} Z_t & 0 \\ 0 & Z_s \end{bmatrix}, \quad Z_t > 0, \quad \det(Z_s) \ne 0, \tag{13a}$$

$$\begin{bmatrix} Z\bar{A}^{\mathrm{T}} + \bar{A}P - \bar{B}R^{-1}\bar{B}^{\mathrm{T}} & Z\bar{C}_{z1}^{\mathrm{T}} - \bar{B}R^{-1}\bar{D}_{z1}^{\mathrm{T}} \\ * & -I - \bar{D}_{z1}R^{-1}\bar{D}_{z1}^{\mathrm{T}} \end{bmatrix} \le 0, \tag{13b}$$

$$\mathcal{N}^{\mathrm{T}}(\bar{B}_t^{\mathrm{T}}) \begin{bmatrix} I & \bar{A}_{tt} \end{bmatrix} \begin{bmatrix} 0 & Z_t \\ Z_t & 0 \end{bmatrix} \begin{bmatrix} I \\ \bar{A}_{tt}^{\mathrm{T}} \end{bmatrix} \mathcal{N}(\bar{B}_t^{\mathrm{T}}) < 0, \tag{13c}$$

where $\mathcal{N}(\bar{B}_t^{\mathrm{T}})$ is a null space basis matrix of $\bar{B}_t^{\mathrm{T}}$.

Proof. Given $\bar{P} = P$ as established in Wu (2003), the continuous counterpart of (11b) can be derived as

$$(\bar{A} + \bar{B}\bar{K})^{\mathrm{T}}P + P(\bar{A} + \bar{B}\bar{K}) + \bar{K}^{\mathrm{T}}R\bar{K} + (\bar{C}_{z1} + \bar{D}_{z1}\bar{K})^{\mathrm{T}}(\bar{C}_{z1} + \bar{D}_{z1}\bar{K}) \leq 0. \tag{14}$$

The application of Schur complement to (14) leads to (13b) by defining $Z := P^{-1}$. The constraint (13c) is the continuous equivalence of (11c).

After the Lyapunov matrix Z (or P) is computed, the continuous state feedback control law $\bar{K}$ can be easily obtained by solving an LMI problem. Under the reverse bilinear transformation from continuous to discrete system (Wu, 2003), the corresponding discrete control law can be computed by $K = \sqrt{2}\bar{K}(H - \bar{A} - \bar{B}\bar{K})$, with $H = \mathrm{diag}(I_{n_t}, I_{n_s^+}, -I_{n_s^-})$ as defined in D'Andrea and Dullerud (2003) and Wu (2003).

Moreover, the terminal cost $V_f(\cdot)$ in (8a) is chosen to take the form of the Lyapunov function as

$$V_f(x(k+N|k,s)) = x_t^{\mathrm{T}}(k+N|k,s)P_t x_t(k+N|k,s). \tag{15}$$

3.2. Offline computation of the terminal sets

Recursive feasibility requires that the terminal state enters an invariant terminal set. Due to the LTSI property of the plant dynamics and the static local control law, the terminal set can be computed offline. Enforcing the invariance of the global terminal set $\mathbb{X}_f$ leads to a centralized terminal state constraint $x(k+N|k) \in \mathbb{X}_f$, where the predicted terminal state of the global system is defined as $x(k+N|k) = \begin{bmatrix} x_t^{\mathrm{T}}(k+N|k,1) & \dots & x_t^{\mathrm{T}}(k+N|k,N_s) \end{bmatrix}^{\mathrm{T}}$. This formulation requires that each subsystem has access to the current states of other subsystems, whereas the subsystems in a distributed setting receive only localized information from their nearest neighbors. At the price of increased conservatism, the invariance of the subsystem terminal sets $\mathbb{X}_f^s$ is considered here to enable the distributed implementation of the online optimization problem.

The terminal set is chosen as a sublevel set of the Lyapunov function as

$$\mathbb{X}_f^s = \{x : x_t^{\mathrm{T}}(k,s)P_t x_t(k,s) \leq \alpha^2\}, \tag{16}$$

where $\alpha > 0$, and P_t is computed in Section 3.1.

The objective is to find the largest ellipsoid centered at the origin, where the control constraints are satisfied, i.e., $|Kx(k,s)| < \bar{u}$. After eliminating $x_t(k,s)$ and $x(k,s)$, the optimization problem is derived as

$$\max_{\alpha} \quad \alpha, \quad \text{s.t.} \quad \alpha \|P_t^{-\frac{1}{2}}K_t^{\mathrm{T}}\|_2 \leq \bar{u}. \tag{17}$$

We denote the solution to (17) by α_m. The resulting terminal set is an invariant set. This is due to the existence of a Lyapunov matrix that satisfies (13c), which indicates that any initial state trajectory starting in $\mathbb{X}_f^s$ remains inside and converges to the origin.

3.3. Recursive feasibility and asymptotic stability

We consider the parallel implementation of the local optimization problems at all subsystems and assume that the transmission of the predicted state trajectories among localized subsystems occurs only once at each sampling time after each subsystem has solved its own optimal control variables $\mathbf{u}(k,s)$. Let the predicted temporal state trajectory of the subsystem s at time k over the prediction horizon N be denoted by $x_t^p(k,s) = \begin{bmatrix} x_t^{\mathrm{T}}(k+1|k,s) & \dots & x_t^{\mathrm{T}}(k+N|k,s) \end{bmatrix}^{\mathrm{T}}$. According to (6), $x_t^p(k,s)$ can be computed as a function of the control variable $\mathbf{u}(k,s)$ and the predicted spatial state trajectories denoted by $\hat{x}_s(k,s)$ as

$$x_t^p(k,s) = \begin{bmatrix} A_{tt} & A_{ts} & 0 & \dots & 0 \\ A_{tt}^2 & A_{tt}A_{ts} & A_{ts} & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ A_{tt}^N & A_{tt}^{N-1}A_{ts} & A_{tt}^{N-2}A_{ts} & \dots & A_{ts} \end{bmatrix} \begin{bmatrix} x_t(k,s) \\ x_s(k,s) \\ \hat{x}_s(k,s) \end{bmatrix} + \begin{bmatrix} B_t & 0 & \dots & 0 \\ A_{tt}B_t & B_t & \dots & 0 \\ \vdots & \vdots & & \vdots \\ A_{tt}^{N-1}B_t & A_{tt}^{N-2}B_t & \dots & B_t \end{bmatrix} \mathbf{u}(k,s), \tag{18a}$$

$$:= f(x(k,s), \hat{x}_s(k,s), \mathbf{u}(k,s)). \tag{18b}$$

Due to the fact that all subsystems solve the local optimization problems (8) simultaneously and then transmit/receive the predicted state trajectories to/from their neighbors, the spatial information exhibits one-step delay, i.e.,

$$\hat{x}_s(k,s) = \begin{bmatrix} x_s^{\mathrm{T}}(k+1|k-1,s) & \dots & x_s^{\mathrm{T}}(k+N-1|k-1,s) \end{bmatrix}^{\mathrm{T}}.$$

Although the subsystem s assumes that at time k, its neighbors follow their trajectories predicted at $k-1$, the actual ones deviate in general since the neighbors solve for the optimal solution at k based on the information received at $k-1$. Inspired by Dunbar (2007) and Keviczky et al. (2006), modified consistency constraints are developed in the following to ensure stability and recursive feasibility of the closed-loop system in the presence of the discrepancy between the actual state trajectory and the one assumed by the neighboring subsystems.

Given the optimal control variables at k, i.e., $\mathbf{u}(k,s)$, a feasible solution at time $k+1$ denoted by $\tilde{\mathbf{u}}(k+1,s)$ can be constructed from the remainder of $\mathbf{u}(k,s)$ concatenated with the terminal control law as

$$\tilde{\mathbf{u}}(k+1,s) = \begin{bmatrix} u(k+1|k,s) \\ \vdots \\ u(k+N-1|k,s) \\ Kx(k+N|k,s) \end{bmatrix}, \tag{19}$$

with $x(k+N|k,s) = \begin{bmatrix} x_t^{\mathrm{T}}(k+N|k,s) & x_s^{\mathrm{T}}(k+N|k,s) \end{bmatrix}^{\mathrm{T}}$.

Due to the one-step delayed spatial information and the resulting deviation between the assumed and the actual predicted spatial state trajectories, the actual temporal state $x_t(k+1,s)$ is in general not identical to the one predicted at time k, i.e., $x_t(k+1|k,s)$. Given the feasible solution $\tilde{\mathbf{u}}(k+1,s)$, a feasible state trajectory can be computed by employing the prediction equation (18b), i.e.,

$$\tilde{x}_t(k+1,s) := \begin{bmatrix} x_t^{\mathrm{T}}(k+2|k+1,s) \\ \vdots \\ x_t^{\mathrm{T}}(k+N+1|k+1,s) \end{bmatrix}^{\mathrm{T}} = f(x(k+1,s), \hat{x}_s(k+1,s), \tilde{\mathbf{u}}(k+1,s)). \tag{20}$$

If it can be shown that the cost function $J(k,s)$ in (8a) is a Lyapunov function, which decreases along the closed-loop trajectories over time, asymptotic stability is implied. Let $\mathcal{J}(k,s)$ denote the minimal cost under the optimal control law $\mathbf{u}(k,s)$, and $\tilde{\mathcal{J}}(k+1,s)$ denote the cost generated by applying the constructed feasible control trajectory (19) and the feasible state (20). Following a similar line as in Keviczky et al. (2006), the difference between the feasible cost at $k+1$ and the minimal cost at k is denoted by ϵ, i.e.,

$$\tilde{\mathcal{J}}(k+1,s) - \mathcal{J}(k,s) = \epsilon, \tag{21}$$

with

$$\begin{aligned}
\epsilon &= \epsilon_1 + \epsilon_2 + \epsilon_3,\\
\epsilon_1 &= -[*]Q_t x_t(k,s) - [*]Q_s x_s(k,s) - [*]Ru(k|k,s),\\
\epsilon_2 &= -\sum_{i=1}^{N-1}\Big[\big([*]Q_t x_t(k+i|k,s) - [*]Q_t\tilde{x}_t(k+i|k+1,s)\big)\\
&\quad - \big([*]Q_s x_s(k+i|k-1,s) - [*]Q_s x_s(k+i|k,s)\big)\Big],\\
\epsilon_3 &= [*]Q_t x_t(k+N|k+1,s) + [*]Q_s x_s(k+N|k,s)\\
&\quad - [*]P_t x_t(k+N|k,s) + [*]P_t x_t(k+1+N|k+1,s)\\
&\quad + [*]RKx(k+N|k,s). \qquad (22)
\end{aligned}$$

A sufficient condition for the recursive feasibility and stability of the closed-loop system is given as follows.

Theorem 3. *Given a feasible solution to the problem (8) at time instant k, there exist a feasible control trajectory (19) and a feasible state trajectory (20) at time instant $k+1$, which guarantee recursive feasibility and asymptotic stability of the closed-loop system, if following conditions are satisfied:*

(i) $Kx(k+N|k,s) \in \mathbb{U}^s$;
(ii) $x_t(k+N+1|k+1,s) \in \mathbb{X}_f^s$;
(iii) $\epsilon < 0$.

Proof. Condition (i) above ensures that (19) satisfies the control constraints, whereas condition (ii) requires that the feasible terminal state enters the invariant set where the terminal control law takes over and drives the state trajectory to the origin. Condition (iii) imposes a consistency constraint between the assumed and the actual spatial information to establish asymptotic stability of the closed-loop system. In case of no discrepancy (no coupling among the subsystems), $\epsilon_2 = 0$ holds, and $\epsilon_3 < 0$ due to (11b).

3.4. Online implementation

Next, we present the online implementation of the distributed MPC problem (8). Let the weighting matrix Q be decomposed as $Q = \mathrm{diag}\{Q_t, Q_s\}$ with $Q_t \in \mathbb{R}^{n_t}$ and $Q_s \in \mathbb{R}^{n_s}$. The cost function (8a) can be rewritten as

$$\begin{aligned}
J(k,s) &= [*]Qx(k,s) + [*]\mathcal{Q}_t x_t^p(k,s) + [*]\mathcal{Q}_s\hat{x}_s(k,s)\\
&\quad + [*]\mathcal{R}\mathbf{u}(k,s), \qquad (23)
\end{aligned}$$

with $\mathcal{Q}_t = \mathrm{diag}\{I_{N-1}\otimes Q_t, P_t\}$, $\mathcal{Q}_s = I_{N-1}\otimes Q_s$ and $\mathcal{R} = I_N \otimes R$.

The minimization of (8a) subject to a linear constraint (8b) and a quadratic constraint (8c) in the form of a quadratic inequality (16) leads to a quadratic optimization problem. Nevertheless, the minimization of $J(k,s)$ is equivalent to minimizing a scalar β with $J(k,s) < \beta$ ($\beta > 0$). With the application of Schur complement, the quadratic cost function (8a), as well as the quadratic constraint (8c), can be formulated as LMIs, such that (8) can be solved using LMI solvers:

$$\min_{\beta,\mathbf{u}(k,s)} \quad \beta, \qquad (24a)$$

$$\text{s.t.} \begin{bmatrix} \mathcal{Q}_t^{-1} & 0 & 0 & x_t^p(k,s)\\ 0 & \mathcal{Q}_s^{-1} & 0 & \hat{x}_s(k,s)\\ 0 & 0 & \mathcal{R}^{-1} & \mathbf{u}(k,s)\\ * & * & * & \beta \end{bmatrix} > 0, \qquad (24b)$$

$$-\bar{u} \le u(k+i|k,s) \le \bar{u}, \qquad (24c)$$

$$\begin{bmatrix} P_t^{-1} & x_t(k+N|k,s)\\ * & \alpha_m^2 \end{bmatrix} > 0, \qquad (24d)$$

with $i = 0, \ldots, N-1$, where P_t and α_m have already been obtained in Sections 3.1 and 3.2, respectively. Note that symmetric bounds on the inputs are assumed here due to the formulation of the MPC problem in the form of LMIs.

The implementation of the proposed distributed MPC approach is summarized in Algorithm 1 . Steps 4–12 are implemented in parallel at all N_s subsystems.

Algorithm 1 Distributed MPC Implementation

1: compute P_t, K, and α_m offline
2: initialization ($k = 0$): find a feasible $\tilde{\mathbf{u}}(0,s)$, receive spatial variables from neighbors
3: **Repeat**
4: construct $\tilde{\mathbf{u}}(k,s)$ and compute $\tilde{\mathbf{x}}_t(k,s)$ using (20)
5: **if** (24) is feasible **then**
6: apply $\mathbf{u}(k,s)$
7: **else**
8: **if** conditions (i)-(iii) are satisfied by $\tilde{\mathbf{u}}(k,s)$ and $\tilde{x}_t(k,s)$ **then**
9: apply $\tilde{\mathbf{u}}(k,s)$
10: **end if**
11: **end if**
12: transmit $x_t^p(k,s)$ to neighbors and receive spatial variables from neighbors

4. Simulation results

In this section, we evaluate the performance of the proposed distributed MPC design approach using a heat example of one spatial dimension, whose governing PDE is written as

$$\frac{\partial y(t,s)}{\partial t} - \kappa\frac{\partial^2 y(t,s)}{\partial x^2} = \frac{A}{C}u(t,s), \qquad (25)$$

where κ is a positive constant that denotes the thermal diffusivity, $y(t,s)$ is the temperature, A is the cross sectional area, C is the heat capacity and $u(t,s)$ is the heat flux.

Let the sampling time and the sampling space be chosen as $T_t = 0.01$ s and $T_s = 0.25$ m, respectively, and $\kappa = 1\ \frac{\mathrm{m}^2}{\mathrm{s}}$. After applying the finite difference method to discretize (25) both in time and space, the resulting subsystems take the two-dimensional I–O form (1) with $n_a = m_a = n_b = m_b = 1$. The polynomial coefficient vectors in (3) are expressed as

$$\tilde{\mathcal{A}} = \begin{bmatrix}1 & a_{(1,0)} & a_{(1,1)} & a_{(1,-1)}\end{bmatrix}, \quad \tilde{\mathcal{B}} = \begin{bmatrix}0 & b_{(1,0)} & 0 & 0\end{bmatrix},$$

with $a_{(1,0)} = -1 + 2\gamma$, $a_{(1,1)} = a_{(1,-1)} = -\gamma$, $b_{(1,0)} = \frac{AT_t}{C}$, and $\gamma = \frac{T_t}{T_s^2}$. To scale the outputs down, we let $b_{(1,0)} = 1$ for the following simulation.

The overall system consists of $N_s = 21$ LTSI subsystems. We assume that the boundary effects are negligible and the finite interconnection of the subsystems can be approximated by an LTSI model. Considered here is the free-free boundary condition (where no restriction is imposed on the boundary subsystems at both ends). Each subsystem is subject to input constraint $-2 \le u(k,s) \le 2$ for all $s = 1, \ldots, N_s$. Let the prediction horizon be chosen as $N = 5$. The weighting matrices are selected as $Q = 10I_{n_x}$ and $R = 1$ to penalize the states and the control input, respectively.

For the purpose of comparison, two centralized MPC schemes – one using a polytopic terminal set, the other one using an ellipsoidal terminal set – have been designed by solving centralized optimization problems. The state vector of a centralized model of (25), $x(k) \in \mathbb{R}^{n_t N_s}$, is constructed by stacking the temporal states of the subsystems in one vector, i.e., $x(k) = \begin{bmatrix}x_t^{\mathrm{T}}(k,1) & \ldots & x_t^{\mathrm{T}}(k,N_s)\end{bmatrix}^{\mathrm{T}}$. The resulting centralized state space realization is by definition also non-minimal.

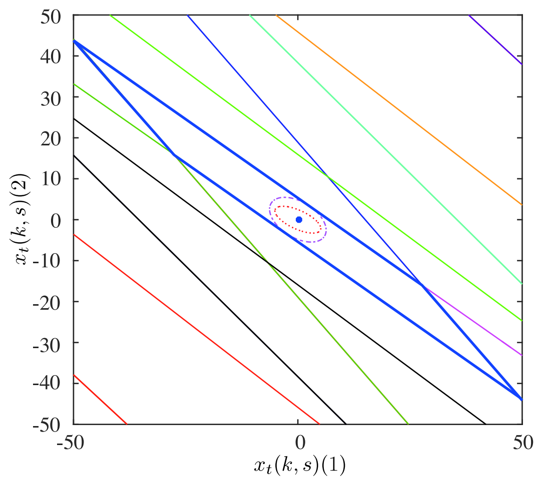


Fig. 2. Comparison of the distributed and centralized terminal sets: Elliptical terminal set of the distributed MPC (red dotted ellipse), and the centralized polytopic (blue bold polygon) and ellipsoidal (magenta dash-dotted ellipse) terminal sets projected onto the subsystem temporal state space.

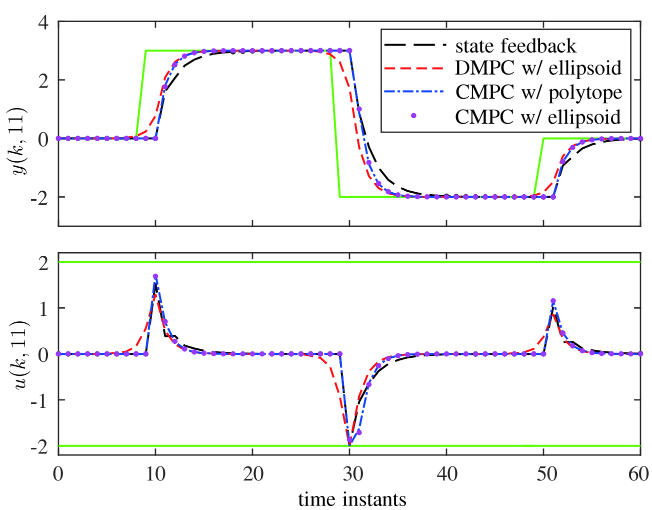


Fig. 3. Comparison of the outputs and control inputs at subsystem 11: State feedback control without MPC (black dashed line), distributed MPC (DMPC) with ellipsoidal terminal sets (red dashed line), centralized MPC (CMPC) with a polytopic set (blue dash-dotted line) and with an ellipsoidal set (magenta dotted line). The given reference (upper) and the control limits (lower) are shown with green solid lines.

The centralized Lyapunov matrix and state feedback gain can be computed by solving a discrete LQR problem with the use of the same weights $Q = 10I_{n_t N_s}$ and $R = I_{N_s}$. Fig. 2 shows a comparison of the distributed and the centralized terminal sets, where the centralized terminal sets are projected onto the subsystem temporal state space; the resulting polygon is the intersection of $2(\nu+1)$ (here $\nu = 16$) halfspaces (Kouvaritakis & Cannon, 2016). A larger terminal set indicates a larger region of attraction. It is apparent that the distributed MPC is more conservative compared to the centralized ones.

Performance of the proposed distributed MPC design approach is evaluated and compared with the centralized MPC designs in terms of reference tracking at subsystems 9 to 12 from time instants 10 to 30 and 30 to 50, i.e., $w(10:30, 9:12) = 3$ and $w(30:50, 9:12) = -2$. Fig. 3 compares the step responses of subsystem 11 with and without MPC. Without MPC, the plant is controlled by the state feedback control law obtained in Section 3.1 with the control input subject to saturation. It can be observed that the conservatism with the use of a distributed MPC does not result in a significant performance degradation within the feasible region; both distributed and centralized MPC schemes stabilize the overall system and achieve a satisfactory performance. It is also worth mentioning that the online implementation of the centralized MPC is more computationally demanding compared to the distributed MPC. The centralized MPC solves an optimization problem of a system order 42, whereas the distributed MPC handles the subsystems of order 6. Furthermore, the terminal constraint of the centralized MPC with a polytopic terminal set involves $2(\nu+1)$ linear inequalities (Kouvaritakis & Cannon, 2016), whereas the distributed MPC needs to fulfill only one quadratic terminal constraint (24d), due to that the distributed MPC scheme is scalable.

5. Conclusion

This paper proposes a distributed MPC design approach for distributed systems governed by PDEs. We have shown how a non-minimal state space realization of a two-dimensional I–O model can lead to a terminal controller design solved as an LMI problem. Sufficient conditions for stability and recursive feasibility of the closed-loop system are derived when the online optimization problem is implemented in parallel at the subsystem level and in the presence of the one-step delayed exchanging information. The proposed approach has been evaluated on a discretized heat equation and demonstrates its capability of ensuring stability and performance of the closed-loop system. Moreover, it results in a much less computational complexity compared to the centralized MPC schemes.

References

D'Andrea, R., & Dullerud, G. E. (2003). Distributed control design for spatially interconnected systems. *IEEE Transactions on Automatic Control*, *48*(9), 1478–1495.

De Schutter, B., Hellendoorn, H., Hegyi, A., van den Berg, M., & Zegeye, S. K. (2010). Model-based control of intelligent traffic networks. In R. R. Negenborn, H. Hellendoorn, & Z. Lukszo (Eds.), *Intelligent systems, control and automation: Science and engineering*: *vol. 42. Intelligent infrastructures* (pp. 277–310). Netherlands: Springer.

Dunbar, W. B. (2007). Distributed receding horizon control of dynamically coupled nonlinear systems. *IEEE Transactions on Automatic Control*, *52*(7), 1249–1263.

El Ghaoui, L., & Niculescu, S. L. (Eds.). (2000). *Advances in linear matrix inequality methods in control*. Philadelphia, PA: SIAM.

Keviczky, T., Borrelli, F., & Balas, G. J. (2006). Decentralized receding horizon control for large scale dynamically decoupled systems. *Automatica*, *42*(12), 2105–2115.

Kouvaritakis, B., & Cannon, M. (2016). Model predictive control: Classical, robust and stochastic. *Advanced textbooks in control and signal processing*. Switzerland: Springer International Publishing.

Langbort, C., & D'Andrea, R. (2005). Distributed control of spatially reversible interconnected systems with boundary conditions. *SIAM Journal on Control and Optimization*, *44*(1), 1–28.

Liu, Q., Abbas, H. S., Mohammadpour, J., Wollnack, S., & Werner, H. (2016). Distributed model predictive control of constrained spatially-invariant interconnected systems in input-output form. In *Proceedings of american control conference* (pp. 3600–3605).

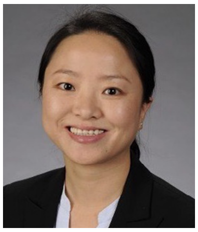
Liu, Q., & Werner, H. (2013). Experimental identification of spatially-interconnected parameter-invariant and LPV models for actuated beams. In *Proceedings of 52nd IEEE conference on decision control* (pp. 4122–4127).

Liu, Q., & Werner, H. (2016). Distributed identification and control of spatially interconnected systems with application to an actuated beam. *Control Engineering Practice*, *54*, 104–116.

Mayne, D. Q., Rawlings, J. B., Rao, C. V., & Scokaert, P. O. M. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, *36*(6), 789–814.

Negenborn, R. R., & De Schutter, B. (2008). A distributed model predictive control approach for the control of irrigation canals. In *First international conference on infrastructure systems and services: Building networks for a brighter future* (pp. 1–6).

Wu, F. (2003). Distributed control for interconnected linear parameter-dependent systems. *IEE Proceedings - Control Theory and Applications*, *150*(5), 518–527.



Qin Liu received her B.Eng. and M.Eng. degrees in Mechanical Engineering from Nanjing University of Science and Technology, Nanjing, China, in 2005 and 2007, respectively. She obtained her M.Sc. in Mechatronics and MBA from Hamburg University of Technology, Hamburg, Germany, in 2009. She completed her Ph.D. degree in Control Engineering at Hamburg University of Technology, Germany, in 2015. She was a postdoctoral fellow at University of Georgia, Athens, USA, from 2015 to 2016. She is currently working at Hilti Germany. Her research interests include linear parameter varying systems, identification and control of distributed systems, and model predictive control.

Hossam Seddik Abbas is an Associate Professor in the Electrical Engineering Department, Faculty of Engineering, Assiut University, Egypt, where he has been an assistant professor from 2010 to 2015. He completed his B.Sc. and M.Sc. degrees at Assiut University in 1997 and 2001, respectively, and his Ph.D. at Hamburg University of Technology, Germany, in 2010. He was a research fellow in Hamburg University of Technology, and Eindhoven University of Technology The Netherlands, in 2011 and 2013, respectively, and currently he is with the Institute for Electrical Engineering in Medicine, Universitaet zu Luebeck, Germany, as a senior researcher (Humboldt fellow). His research interests lie in the area of control systems engineering and its applications. His focus includes linear parameter varying systems, optimal and robust control, model predicative control, distributed control and system identification.

Javad Mohammadpour Velni received BS and MS degrees in electrical engineering from Sharif University of Technology and University of Tehran, Iran, respectively, and PhD degree in mechanical engineering from University of Houston, TX. He joined the University of Georgia as an assistant professor of electrical engineering in Aug. 2012. Prior to that, he was with the University of Michigan, where he worked in the naval architecture & marine engineering dept. from Oct. 2011 to Jul. 2012. He was also a Research Assistant Professor of mechanical engineering at University of Houston from Oct. 2008 to Sep. 2011 and a Research Associate at the same institution from Jan. 2008 to Sep. 2008. He has published over 100 articles in international journals and conference proceedings, served in the editorial boards of ASME and IEEE conferences on control systems and edited two books on control of large-scale systems (published in 2010) and LPV systems modeling, control and applications (published in 2012). His current research interests are in secure control of cyber–physical systems (and in particular, smart grids), coverage control of heterogeneous multi-agent systems, and data-driven approaches for model learning and control of complex distributed systems.