ISA Transactions 55 (2015) 1–12

Contents lists available at ScienceDirect

ISA Transactions

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

Distributed model predictive control for constrained nonlinear systems with decoupled local dynamics

Meng Zhao, Baocang Ding*

College of Automation, Chongqing University, Chongqing 400044, China

ARTICLE INFO

Article history:
Received 14 February 2014
Received in revised form
17 June 2014
Accepted 15 July 2014
Available online 12 November 2014
This paper was recommended for publication by Dr. Rickey Dubay

Keywords:
Distributed control
Model predictive control (MPC)
Compatibility condition
Exponentially stability

ABSTRACT

This paper considers the distributed model predictive control (MPC) of nonlinear large-scale systems with dynamically decoupled subsystems. According to the coupled state in the overall cost function of centralized MPC, the neighbors are confirmed and fixed for each subsystem, and the overall objective function is disassembled into each local optimization. In order to guarantee the closed-loop stability of distributed MPC algorithm, the overall compatibility constraint for centralized MPC algorithm is decomposed into each local controller. The communication between each subsystem and its neighbors is relatively low, only the current states before optimization and the optimized input variables after optimization are being transferred. For each local controller, the quasi-infinite horizon MPC algorithm is adopted, and the global closed-loop system is proven to be exponentially stable.

1. Introduction

Model predictive control (MPC) has received much attention in recent years due to its capability to control the constrained systems [1,2]. The MPC algorithms are optimization-based control strategies and generally in centralized form [3,4]. For centralized MPC, the computation requirement [5] is the main barrier to enlarge the scopes of applications, especially for nonlinear large-scale systems. A viable way to reduce the computation time is by utilizing the distributed MPC. Compared to the centralized MPC, the optimization in distributed MPC is totally decentralized into a number of small-scale optimizations. Since the communication bandwidth is often limited, each subsystem cannot get the full information of other subsystems when solving its optimization. As a result, the control performance may be worse than centralized MPC, but the advantages outweigh its disadvantages. Up to now, there are many works on distributed MPC [6–8]. In [9], an overall introduction to all existing important works on distributed MPC is given. In [8], the distributed MPC algorithms are divided into 2 kinds: the cooperative distributed MPC [10–13] and non-cooperative distributed MPC [14]. In a cooperative distributed MPC, each local controller optimizes a global objective function, while in a non-cooperative MPC, a local one. In this paper, we consider distributed MPC of a set of dynamically decoupled subsystems with a separable common objective function. The common objective function, in which the states and inputs of subsystems are coupled, is separated into each local controller. In order to guarantee stability, the overall compatibility constraints are also disassembled. Since the common objective function is totally divided, the proposed distributed MPC is the non-cooperative one.

Distributed MPC has been widely applied to control systems, such as multi-agent systems [35], four-tank systems [15,16], building temperature regulation systems [17], and supply chain management systems [18]. Most of these systems are physically distributed and can be divided into 2 categories: system with coupled dynamic [15,16,19] and that with decoupled one [35,17,20,21]. In this paper, the latter system is considered.

Since the subsystems are spatially distributed in most cases, the communication between them is required to share the information. Nevertheless, the communication burden will be huge if the amount of subsystems is large and all the local information is required to exchange. In [22], a method which requires no communication is proposed, and the algorithm utilized is called the decentralized MPC. In [23], the information of each subsystem is transmitted iteratively between samplings. However, in [32,34–36], the information is transmitted in a non-iterative way. In [24], the communication delay in the

* Corresponding author. Tel.: +86 23 6534 1593.
E-mail addresses: meng.zh@outlook.com (M. Zhao), baocang.ding@gmail.com (B. Ding).

http://dx.doi.org/10.1016/j.isatra.2014.07.012
0019-0578/

information network is considered. In [25], the data losses is considered. In the present paper, the communication between each subsystem and its neighbors is required. By comparison with procedures for finding neighbors in [35,36], each subsystem in our proposed distributed MPC has less neighbors because the relations are non-mutual.

In order to guarantee closed-loop stability, a special form of compatibility constraint [34,32] is adopted in this paper, which is a sufficient condition for stability. Unlike the compatibility constraint utilized in [35,36,19], the proposed compatibility constraint makes the controlled system exponentially stable. Other techniques include the Lyapunov-based approach [26–28] and the game approach [29,30].

The main contribution of this paper is that the distributed MPC for nonlinear systems with decoupled dynamics is considered. The local controller in each subsystem is designed by using the quasi-infinite horizon MPC algorithm. The proposed distributed MPC decomposes a centralized objective function into each local controller. In order to guarantee the overall stability, the compatibility constraint is deduced and disassembled. Moreover, new procedures for finding neighbors are presented. Unlike any other methods, the proposed method is available not only for the isomorphic subsystems, but also for the heterogeneous subsystems. Previous results of this paper are given in [34] for linear nominal systems and in [32] for uncertain linear systems.

This paper is organized as follows. Section 2 describes the nonlinear system with dynamically decoupled subsystems, and the corresponding centralized MPC. Section 3 introduces the naive distributed MPC, and the compatibility condition for stability. In Section 4, by decoupling the compatibility condition, the synthesis approach of distributed MPC is presented. In Section 5, a multi-vehicle formation control example and the corresponding simulation results are given. In the Appendix, new procedures for finding neighbors are proposed.

Notations: I is the identity matrix with appropriate dimension. I_n is the identity matrix of n-th order. For a vector x and positive-definite matrix W, $\|x\|_W^2 = x^{\mathrm{T}}Wx$. $x(j|k)$ is the value of vector x at a future time $k+j$ predicted at time k. For any integer N_a, $\mathbb{N}_a$:=$\{1,2,\ldots,N_a\}$. For any integer $N>0$, $\mathbb{N}$:=$\{0,1,\ldots,N-1\}$, $\mathbb{N}_1$:=$\{0,1,\ldots,N-2\}$ and $\mathbb{N}_2$:=$\{1,2,\ldots,N-1\}$. A variable with $*$ as superscript indicates that it is the optimal solution of the optimization problem. The time-dependence of the MPC decision variables is often omitted for simplicity.

2. Problem statement

Let us consider the following N_a physically distributed local systems:

$$x_i(k+1) = f_i(x_i(k), u_i(k)), \quad i \in \mathbb{N}_a \tag{1}$$

where $x_i \in \Re^{n_i}$ and $u_i \in \Re^{m_i}$ are measurable state and input, respectively. The state and control are confined by

$$x_i \in \mathcal{X}_i \subseteq \Re^{n_i}, \quad x_i \in \mathcal{U}_i \subseteq \Re^{m_i} \tag{2}$$

Assumption 1. The functions f_i's are twice continuously differentiable, with $f_i(0,0)=0$.

Assumption 2. $\mathcal{X}_i$ is a closed set and $\mathcal{U}_i$ a compact, convex set, both of them containing the origin as interior point.

Define the local linearization of (1) at origin

$$A_i = \frac{\partial f_i}{\partial x_i}(0,0), \quad B_i = \frac{\partial f_i}{\partial u_i}(0,0) \tag{3}$$

Assumption 3. (A_i, B_i) is stabilizable.

At each time k, the control objective is to minimize

$$J(k) = \sum_{j=0}^{\infty} \left[\|x(j|k)\|_Q^2 + \|u(j|k)\|_R^2 \right] \tag{4}$$

with respect to $u(j|k)$, $j \ge 0$, where $x = \left[x_1^{\mathrm{T}}, x_2^{\mathrm{T}}, \ldots, x_{N_a}^{\mathrm{T}}\right]^{\mathrm{T}}$, $u = \left[u_1^{\mathrm{T}}, u_2^{\mathrm{T}}, \ldots, u_{N_a}^{\mathrm{T}}\right]^{\mathrm{T}}$; $x_i(j+1|k) = f_i(x_i(j|k), u_i(j|k))$, $x_i(0|k) = x_i(k)$; $Q, R>0$ are symmetric weighting matrices. The input and state constraints, under Assumption 2, are supposed to have the following form in the minimization:

$$\mathcal{X}_i{:=}\{x_i| -\underline{\psi}_i \le \Psi_i x_i(j+1|k) \le \overline{\psi}_i\}, \quad \mathcal{U}_i{:=}\{u_i| -\underline{u}_i \le u_i(j|k) \le \overline{u}_i\}, \quad j \ge 0 \tag{5}$$

where $i \in \mathbb{N}_a$, $\Psi_i \in \Re^{q_i \times n_i}$, q_i is the number of rows in matrix Ψ_i, $\underline{\psi}_i{:=}\left[\underline{\psi}_1^i, \underline{\psi}_2^i, \ldots, \underline{\psi}_{q_i}^i\right]^{\mathrm{T}}$ and $\overline{\psi}_i{:=}\left[\overline{\psi}_1^i, \overline{\psi}_2^i, \ldots, \overline{\psi}_{q_i}^i\right]^{\mathrm{T}}$ with $\underline{\psi}_l^i, \overline{\psi}_l^i > 0$, $l \in \{1, \ldots, q_i\}$, $\underline{u}_i{:=}\left[\underline{u}_1^i, \underline{u}_2^i, \ldots, \underline{u}_{m_i}^i\right]^{\mathrm{T}}$ and $\overline{u}_i{:=}\left[\overline{u}_1^i, \overline{u}_2^i, \ldots, \overline{u}_{m_i}^i\right]^{\mathrm{T}}$ with $\underline{u}_l^i, \overline{u}_l^i > 0$, $l \in \{1, \ldots, m_i\}$.

In order to implement the control in a distributed manner, $J(k)$ is divided as $J_i(k)$'s with

$$J_i(k) = \sum_{j=0}^{\infty} \left[\|z_i(j|k)\|_{\overline{Q}_i}^2 + \|v_i(j|k)\|_{\overline{R}_i}^2 \right], \quad J(k) = \sum_{i=1}^{N_a} J_i(k), \tag{6}$$

where $z_i = \left[x_i^{\mathrm{T}}, x_{-i}^{\mathrm{T}}\right]^T$, $v_i = \left[u_i^{\mathrm{T}}, u_{-i}^{\mathrm{T}}\right]^T$; $x_{-i}(u_{-i})$ includes the states (inputs) of the neighbors of i; $\overline{Q}_i \ge 0$ and $\overline{R}_i \ge 0$ are symmetric weighting matrices,

$$\overline{Q}_i = \begin{bmatrix} \overline{Q}_1^i & \overline{Q}_{12}^i \\ \overline{Q}_{12}^{i\mathrm{T}} & \overline{Q}_3^i \end{bmatrix}, \quad \overline{R}_i = \begin{bmatrix} \overline{R}_1^i & \overline{R}_{12}^i \\ \overline{R}_{12}^{i\mathrm{T}} & \overline{R}_3^i \end{bmatrix}. \tag{7}$$

In this paper, the neighbors of i are defined as those local systems that are related to i via $J_i(k)$.

Denote

$$x_{-i} = \left[x_{\nu_2^i}^T, x_{\nu_3^i}^T, \ldots, x_{\nu_{\mathcal{N}_i}^i}^T\right]^T, \quad \nu_1^i = i, \quad \left\{\nu_1^i, \nu_2^i, \ldots, \nu_{\mathcal{N}_i}^i\right\} \subseteq \mathbb{N}_a$$

where $\mathcal{N}_i - 1 \ge 0$ is the number of neighbors of local system i, and $\nu_j^i, j \in \{2, 3, \ldots, \mathcal{N}_i\}$ is the sequence number of the $(j-1)$-th neighbor of i. For each local system, the control objective is to minimize $J_i(k)$ satisfying its input and state constraints.

Remark 1. Given Q, R, a procedure is proposed in [35] (see its Section 4.3) for finding $\overline{Q}_i$, $\overline{R}_i$. With the procedure in [35] applied, $J(k) = \sum_{i=1}^{N_a} J_i(k)$. A procedure different from that in [35] is given in Appendices A and B. $J(k) = \sum_{i=1}^{N_a} J_i(k)$ if and only if $\|x(\cdot)\|_Q = \sum_{i=1}^{N_a} \|z_i(\cdot)\|_{\overline{Q}_i}$ and $\|u(\cdot)\|_R = \sum_{i=1}^{N_a} \|v_i(\cdot)\|_{\overline{R}_i}$. If $\dim\{x_i\} \ne \dim\{z_i\}$ or $\dim\{u_i\} \ne \dim\{v_i\}$, then i has neighbors. The neighbors of i include any local system whose state is a part of z_i, or whose input is a part of v_i. Since the neighbors of i are defined through $J_i(k)$, which is equivalent to through $\overline{Q}_i$ and $\overline{R}_i$, finding the neighbors of i clearly depends on Q, R.

3. Stability of the naive distributed MPC

We will discuss the stability of distributed MPC by simply applying, in each local controller, the procedure as in the centralized MPC.