Received: 20 September 2016 | Revised: 16 May 2017 | Accepted: 9 June 2017
DOI: 10.1002/rnc.3881

RESEARCH ARTICLE

WILEY

Distributed nonlinear model predictive control based on contraction theory

Yushen Long[1] | Shuai Liu[1] | Lihua Xie[1] | Karl Henrik Johansson[2]

[1]School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore
[2]ACCESS Linnaeus Center, School of Electrical Engineering, Royal Institute of Technology, Stockholm SE-100 44, Sweden

Correspondence
Lihua Xie, School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore.
Email: elhxie@ntu.edu.sg

Summary

A novel distributed model predictive control algorithm for continuous-time nonlinear systems is proposed in this paper. Contraction theory is used to estimate the prediction error in the algorithm, leading to new feasibility and stability conditions. Compared to existing analysis based on Lipschitz continuity, the proposed approach gives a distributed model predictive control algorithm under less conservative conditions, allowing stronger couplings between subsystems and a larger sampling interval when the subsystems satisfy the specified contraction conditions. A numerical example is given to illustrate the effectiveness and advantage of the proposed approach.

KEYWORDS

contraction theory, distributed control, model predictive control, nonlinear systems

1 | INTRODUCTION

Model predictive control (MPC) has attracted intensive research efforts because of its ability in handling control input and state constraints while optimizing some performance index.[1] In centralized MPC, at every sampling time instant, the controller needs to solve a finite-horizon constrained optimal control problem. However, for large-scale systems such as traffic networks[2] and building systems,[3] centralized implementation is often impractical. Firstly, it is usually not possible to collect full information of the whole system to realize a centralized controller. Secondly, as the dimension of the system grows, the corresponding optimal control problem becomes intractable and cannot be solved within a reasonable period. To overcome these difficulties, distributed MPC schemes have been proposed. Compared with the centralized MPC, distributed MPC solves subproblems of smaller size in parallel based only on local information. Therefore, the communication burden and computational complexity can be largely reduced.

Distributed MPC has been studied extensively in recent literature. For dynamically coupled linear discrete-time systems, inspired by the approach in Mayne et al,[4] a non-iterative tube-based MPC algorithm is proposed in Farina and Scattolini.[5] In Giselsson and Rantzer,[6] an iterative approach based on relaxed dynamic programming[7] is designed. It requires controllers to exchange information iteratively with their neighbours in every sampling time interval. An iterative MPC algorithm for dynamically coupled nonlinear discrete-time systems is proposed in Stewart et al.[8] For continuous-time systems, a dynamically decoupled nonlinear model is considered in Li and Shi[9] where robustness constraints are used to bound the future behaviour of each subsystem. In Farina et al,[10] a continuous-time algorithm is proposed for dynamically coupled linear systems. A Lyapunov-based algorithm is given in Liu et al[11] for nonlinear dynamically coupled systems. This algorithm requires controllers to know the model of the whole plant and a feasible Lyapnunov-based controller must be provided before the design of the MPC algorithm. A continuous-time distributed algorithm for nonlinear dynamically coupled systems based on partial state information is developed in Dunbar,[12] where each controller informs neighbours about its predicted future behaviour and a consistent constraint is used to bound the difference between the evolution of the real state and the predicted one. More recently, an algorithm based on partial state information is provided in Liu et al,[13] which assumes that the coupling is weak enough so that the error caused by ignoring the coupling can be compensated by properly designed constraints.

The aim of this paper is to develop a novel distributed MPC algorithm for continuous-time nonlinear systems based on partial state information with reduced conservativeness of feasibility and stability conditions. In Dunbar[12] and Liu et al,[13] the recursive feasibility and stability conditions are rather conservative. Instead of using Lipschitz continuity argument as in Dunbar[12] and Liu et al,[13] in this paper, contraction theory[14] is adopted to estimate the state prediction error.

The main contributions of this paper are summarized as follows: (1) Compared with the robust MPC using contraction theory,[15] our MPC algorithm is distributed and asymptotic stability of the whole system can be achieved; (2) Compared with the algorithms proposed in Dunbar[12] and Liu et al,[13] our result is less conservative; (3) The online optimization problem in our algorithm does not require a terminal cost function. Indeed, the objective function in our algorithm can be chosen freely. Two constraints are instead used to guarantee feasibility and stability. In particular, the objective function is not used as a Lyapunov functional candidate in our approach as in conventional MPC settings.[1] Therefore, we have more flexibility on the choice of objective function. It should be noted that also in Liu et al,[11] the terminal conditions are not required, but in that work, the controllers are assumed to know the model of the whole plant while we assume each controller only knows the model of the corresponding subsystem. In Li and Shi,[9] another type of contraction constraint is used in the online optimization problem, but the objective function still needs to be designed carefully. In Magni and Scattolini,[16] the contraction constraint proposed in Kothare and Morari[17] is used to ensure stability, but couplings are considered as bounded disturbances, while in our work, we allow information exchange among subsystems so that the dynamical couplings can be handled in a more efficient way.

The remainder of this paper is organized as follows. In Section 2, the system model and control objective are stated together with some necessary assumptions and preliminaries. In Section 3, the new distributed MPC algorithm is proposed. In Section 4, recursive feasibility and stability of the proposed algorithm are analysed. In Section 5, a numerical example is given to illustrate the effectiveness and advantages of the proposed algorithm. In Section 6, conclusions are drawn.

Some remarks on notation are as follows. $\mathbb{R}$ represents the set of real numbers and $\mathbb{R}^n$ represents the n-dimensional Euclidean space. $\mathbf{0}$ is used to denote the zero matrix of appropriate dimension. For a matrix P, P^T denotes its transpose. For a vector x, $\|x\|$ denotes its 2-norm and $\|x\|_P$ the P-norm, ie, $\|x\|_P^2 = x^T P x$, where P is a positive definite matrix. $\lambda_{\max}(P)$ and $\lambda_{\min}(P)$ are the largest and the smallest eigenvalue of $P = P^T$, respectively.

2 | PRELIMINARIES AND PROBLEM FORMULATION

In this section, some preliminaries on graph theory and contraction analysis will be introduced. Then, the system model will be given, and the control problem under investigation will be formulated.

2.1 | Preliminaries

Consider a digraph defined as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}_\mathcal{G}\}$, where $\mathcal{V} = \{1, \dots, N\}$ is the set of nodes, each of which represents a subsystem. $\mathcal{E}_\mathcal{G} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. If pair $(i, j) \in \mathcal{E}_\mathcal{G}$, the state of subsystem i is in the dynamics of subsystem j and subsystem i is referred to as an upstream neighbour of subsystem j while j is referred to as a downstream neighbour of i. The sets $\mathcal{N}_i^u = \{j \in \mathcal{V} | j$ is an upstream neighbour of $i\}$ and $\mathcal{N}_i^d = \{j \in \mathcal{V} | j$ is a downstream neighbour of $i\}$ collect all upstream neighbours and downstream neighbours of subsystem i, respectively.

Contraction analysis is a powerful tool to analyze the stability of a class of nonlinear systems. A brief introduction to contraction analysis will be given in the following. More detailed discussions can be found in Lohmiller and Slotine,[14] and Forni and Sepulchre,[18] and references therein. Consider an autonomous system $\dot{x}(t) = f(x(t), t)$ where f is a $n \times 1$ nonlinear vector function continuously differentiable with respect to x and continuous in t, and $x \in \mathbb{R}^n$ is the state vector. Suppose that we have 2 state trajectories starting from different initial conditions. The virtual displacement δx is an infinitesimal displacement at a fixed time and defines a linear tangent differential form differentiable with respect to time. The virtual velocity $\delta \dot{x}$ can be written as $\delta \dot{x} = \frac{\partial f}{\partial x}(x, t)\delta x$. The virtual displacement can also be expressed under the differential coordinate transformation

$$\delta z = \Theta(x, t)\delta x,$$

where $\Theta(x, t)$ is a square matrix. The squared length is

$$\delta z^T \delta z = \delta x^T M(x, t) \delta x,$$

where $M(x,t) = \Theta^T(x,t)\Theta(x,t)$. Hence,

$$\frac{d}{dt}(\delta z^T \delta z) = 2\delta z^T F \delta z, \tag{1}$$

where $F = (\dot{\Theta} + \Theta\frac{\partial f}{\partial x})\Theta^{-1}$ is the generalized Jacobian, which represents the co-variant derivative of f in δz coordinates. We can also rewrite (1) in δx coordinates as

$$\frac{d}{dt}(\delta x^T M \delta x) = \delta x^T \left(\frac{\partial f}{\partial x}^T M + \dot{M} + M\frac{\partial f}{\partial x} \right) \delta x.$$

It is easy to see that the exponential convergence of $\delta z^T\delta z$ or $\delta x^T M \delta x$ can be concluded in regions where $\frac{\partial f}{\partial x}^T M + \dot{M} + M\frac{\partial f}{\partial x} \leq -\beta_M M$ holds with β_M a strictly positive constant. The definition of contraction region is then consequently given as follows:

Definition 1. Lohmiller and Slotine[19] Given the system equation $\dot{x} = f(x,t)$, a region of the state space is called a contraction region with respect to a uniformly positive definite metric $M(x,t) = \Theta^T(x,t)\Theta(x,t)$, if in that region F in (1) is uniformly negative definite or $\frac{\partial f}{\partial x}^T M + \dot{M} + M\frac{\partial f}{\partial x} \leq -\beta_M M$ for some $\beta_M > 0$.

Convergence in the contraction region is summarized as follows:

Theorem 1. *Lohmiller and Slotine[19] Given the system $\dot{x} = f(x,t)$, any trajectory $x(t)$, which starts in a ball of constant radius with respect to the metric $M(x,t)$, centered at a given trajectory and contained at all times in a contraction region with respect to $M(x,t)$, remains in that ball and converges exponentially to this trajectory. Furthermore, global exponential convergence to the given trajectory is guaranteed if the whole state space is a contraction region with respect to the metric $M(x,t)$.*

In the rest part of this paper, we assume that $M(x,t) = M$ is a constant matrix.

2.2 | Problem formulation

Without loss of generality, we assume that the states as well as control inputs of all the subsystems have the same dimension. Subsystem i has the following dynamics:

$$\dot{x}_i(t) = f_i(x_i(t), x_{-i}(t), u_i(t)), t \geq t_0, \tag{2}$$

where $x_i \in \mathbb{R}^n$ is its state, $u_i \in \mathbb{U}_i \subseteq \mathbb{R}^m$ the control input, and $x_{-i} = \{x_j,\ j \in \mathcal{N}_i^u\}$ denotes the concatenated vector of the states of the upstream neighbours of subsystem i. Denote $x = (x_1^T, \ldots, x_N^T)^T$, $u = (u_1^T, \ldots, u_N^T)^T$ so that the whole system can be written as

$$\dot{x}(t) = f(x(t), u(t)), t \geq t_0, \tag{3}$$

where $f(x,u) = (f_1^T(x_1, x_{-1}, u_1), \ldots, f_N^T(x_N, x_{-N}, u_N))^T$. Our objective is to design a distributed control algorithm to stabilize system (3) while enforcing the input constraints $u_i \in \mathbb{U}_i, \forall i \in \mathcal{V}$.

To facilitate the controller design, we need the following assumptions.[12]

Assumption 1. (**a**) The function $f : \mathbb{R}^{nN} \times \mathbb{R}^{mN} \to \mathbb{R}^{nN}$ is twice continuously differentiable and satisfies $f(\mathbf{0}, \mathbf{0}) = \mathbf{0}$; (**b**) System (3) has a unique, absolutely continuous solution for any initial condition $x(t_0)$ and any piecewise right-continuous control $u : [t_0, \infty) \to \mathbb{U} \triangleq \mathbb{U}_1 \times \ldots \times \mathbb{U}_N$; (**c**) Sets $\mathbb{U}_i, \forall i \in \mathcal{V}$ are compact subsets of $\mathbb{R}^m$ with the origin as their interior.

Consider the linearization of (2) around the origin as

$$\dot{x}_i(t) = A_{ii}x_i(t) + \sum_{j \in \mathcal{N}_i^u} A_{ij}x_j(t) + B_i u_i(t), \tag{4}$$

where $A_{ij} = \frac{\partial f_i}{\partial x_j}\Big|_{(\mathbf{0},\mathbf{0})}$ and $B_i = \frac{\partial f_i}{\partial u_i}\Big|_{(\mathbf{0},\mathbf{0})}$ and let the decentralized model be

$$\dot{x}_i^d(t) = f_i\left(x_i^d(t), \mathbf{0}, u_i(t)\right), t \geq t_0.$$

We need the following decentralized stabilizability assumption:

Assumption 2. There exists a decentralized linear feedback controller gain $K = \mathrm{diag}(K_1, \ldots, K_N)$ such that (4) is stabilized with $u_i(t) = K_i x_i(t)$ and $A_{ii} + B_i K_i$ is Hurwitz, $\forall i \in \mathcal{V}$.

Assumption 2 is used to construct a terminal controller that is critical for proving recursive feasibility and stability. It is a standard approach, see, eg, Dunbar,[12] Farina et al,[10] and Farina and Scattolini.[5] Based on this assumption, we have the following lemma.

Lemma 1. *Chen and Allgöwer,[20] Dunbar[12] Under Assumption 2, there exist positive constants* ϵ*, r and positive definite matrix* $M = \mathrm{diag}(M_1, \dots, M_N)$ *such that*

$$\Omega_\epsilon \triangleq \left\{x \in \mathbb{R}^{nN} \,|\, \|x\|_M \le \epsilon\right\}$$

is a positively invariant set of $\dot{x} = f(x, Kx)$, $\|x(t)\|_M$ *and* $\|x_i^d(t)\|_{M_i}$ *exponentially decay to zero with decay rate not less than r, and* $Kx \in \mathbb{U}$ *for all* $x \in \Omega_\epsilon$.

We also need Lipschitz continuity of f_i:

Assumption 3. The functions $f_i, \forall i \in \mathcal{V}$ satisfy

$$\|f_i(x_i, x_{-i}, u_i) - f_i(x_i, x'_{-i}, u_i)\| \le \sum_{j \in \mathcal{N}_i^u} L_{ij} \left\|x_j - x'_j\right\|,$$

where $x'_{-i} = \{x'_j, j \in \mathcal{N}_i^u\}$ and L_{ij} are positive constants.

Remark 1. In most of the nonlinear MPC, the Lipschitz constant of f_i with respect to x_i is used. This approach creates conservativeness. In our work, however, we use only the Lipschitz constants of the couplings, which give less conservative results.

3 | CONTROLLER DESIGN AND IMPLEMENTATION

3.1 | Offline design

In the offline phase, we assume that a feasible control input can be found. This control input should satisfy Assumption 4 below such that the corresponding state trajectory has a contraction property:

Assumption 4. A feasible open-loop control input $u^0(t) = (u_1^{0T}(t), \dots, u_N^{0T}(t))^T \in \mathbb{U}, t \in [t_0, t_0 + T]$ exists, where $T > 0$ is the prediction horizon, ie, the corresponding state trajectory of system (3) $x^0(t) = (x_1^{0T}(t), \dots, x_N^{0T}(t))^T$ satisfies

$$\begin{aligned} & x^0(t_0) = x_0, \\ & \frac{\partial f_i}{\partial x_i}^T M_i + M_i \frac{\partial f_i}{\partial x_i} \le -\beta_i M_i, \forall x_i \in \Theta_i, \\ & x_i^0(t_0 + T) \in \Omega_i(\alpha\epsilon) \triangleq \left\{ x_i | \|x_i\|_{M_i} \le \frac{\alpha\epsilon}{\sqrt{N}} \right\}, \\ & i = 1, \dots, N, \end{aligned} \tag{5}$$

where

$$\Theta_i \triangleq \left\{ x_i | \inf_{z_i \in \Gamma_i} \|x_i - z_i\|_{M_i} \le \bar{l}_i \right\}, \tag{6}$$

$\beta_i > 0, \bar{l}_i > 0, 0 < \alpha < 1$ and $\Gamma_i \triangleq \{x_i^0(t), t \in [t_0, t_0 + T]\} \cup \Omega_i(\alpha\epsilon)$.

Inequality (5) replaces the commonly used Lipschitz continuity condition. It guarantees that 2 trajectories starting from different initial conditions with the same control input converge to each other exponentially. This condition is also used in process control,[19] networked system,[21] Kalman filter,[22] and so on.

Suppose that for each subsystem, the same prediction horizon T and sampling period δ are used. Denote $c_i \triangleq 2\sum_{j \in \mathcal{N}_i^u} \frac{\lambda_{\max}(M_i^{\frac{1}{2}}) L_{ij} l_j}{\lambda_{\min}(M_j^{\frac{1}{2}})}$, $l_i < \bar{l}_i$, $d_i = 2\sum_{j \in \mathcal{N}_i^u} \frac{L_{ij}\alpha\epsilon\lambda_{\max}(M_i^{\frac{1}{2}})}{\sqrt{N}\beta_i\lambda_{\min}(M_j^{\frac{1}{2}})}$. We need to choose parameters ξ_i, T, δ, l_i, and γ_i such that the following inequalities hold, which will be used later to prove stability of the proposed algorithm:

$$\frac{c_i}{\beta_i} + d_i - l_i \le 0, \tag{7}$$

$$\gamma \triangleq \sum_{i=1}^N \gamma_i < 1, \tag{8}$$

$$\xi_i T + \frac{2}{\beta_i}\left(l_i - d_i - \frac{c_i}{\beta_i}\right)\left(1 - e^{-\frac{1}{2}\beta_i\delta}\right) < \left(\gamma_i\epsilon - d_i - \frac{c_i}{\beta_i}\right)\delta. \tag{9}$$

These parameters can be found numerically as follows. Since M_i, L_{ij}, d_i are constants determined by the system parameters, we only need to determine l_i of inequality (7). Therefore, the first step is to choose the smallest l_i such that (7) holds. Then we fix $\gamma_i = \frac{1}{N+1}$ and choose small enough ξ_i and T and large enough δ such that (9) holds.

In the offline design phase, we need to find $u^0, x^0, M_i, \beta_i, \bar{l}_i, T, \delta, \gamma_i$, which should satisfy Assumption 4 and (7) to (9). In particular, x^0 serves as reference trajectory for the online optimization, M_i and β_i are determined by the contraction property of the corresponding subsystem, and $\bar{l}_i$ is a parameter that provides a margin of freedom. In general, u^0 can be found by solving a constrained problem satisfy Assumption 4. However, due to the dynamical coupling and the nonlinearity, how to solve this problem in a distributed way is still under study. In this paper, we therefore assume that these parameters are known. Indeed, to have an initial feasible solution is necessary for most optimization problems and a lot of centralized implementations of MPC also assume that an initial feasible solution is available.[1,23] This assumption can also be found in Trodden and Richards,[24] Richards and How,[25] and Dunbar[12] for the distributed case.

3.2 | Initialization

Given initial state x_0, at time t_0, and initial feasible control input $u^0(t), t \in [t_0, t_0 + T]$, for $t > t_0 + T$, $x_i^0(t)$ and $u_i^0(t)$, $i = 1, \dots, N$ are defined by the following subsystem:

$$\begin{aligned} \dot{x}_i^0(t) &= f_i\left(x_i^0(t), \mathbf{0}, K_i x_i^0(t)\right), \\ u_i^0(t) &= K_i x_i^0(t), t \in (t_0 + T, \infty), \end{aligned} \tag{10}$$

where K_i satisfies Assumption 2.

Denote $t_k = t_0 + k\delta, k = 1, 2, \dots$. Then the control algorithm is stated as follows:

At initial time t_0, given initial state x_0, if $x_0 \in \Omega_\epsilon$, apply $u = u^0 = Kx^0$ where $K = \text{diag}(K_1, \dots, K_N)$; otherwise send $x_i^0(s), s \in [t_0, t_0 + T]$ to subsystem i and its downstream neighbours.

For any subsystem i over time interval $[t_0, t_1]$,

1. apply $u_i^0(s), s \in [t_0, t_1]$;
2. compute $x_i^0(s), s \in [t_0 + T, t_1 + T]$ according to Equation 10;
3. transmit $x_i^0(s), s \in [t_0 + T, t_1 + T]$ to its downstream neighbours;
4. receive $x_j^0(s), s \in [t_0 + T, t_1 + T]$ from its upstream neighbours.

After introducing the offline design phase, we next describe the online implementation of our distributed MPC algorithm.

3.3 | Online implementation

For any subsystem i at time instant t_k, $k \geq 1$, if $x(t_k) \in \Omega_\epsilon$, apply $u_i = K_i x_i$; otherwise solve the following problem.

Subproblem i

$$\min_{\hat{u}_i(\cdot;t_k) \in \mathbb{U}_i} J_i(x_i(t_k), \hat{u}_i(\cdot; t_k))$$

subject to

$$\begin{aligned} &\hat{x}_i(t_k; t_k) = x_i(t_k), \\ &\dot{\hat{x}}_i(s; t_k) = f_i\left(\hat{x}_i(s; t_k), x_{-i}^0(s), \hat{u}_i(s; t_k)\right), &(11) \\ &\left\|\hat{x}_i(s; t_k) - x_i^0(s)\right\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}\left(1 - e^{-\frac{1}{2}\beta_i(s - t_k)}\right), &(12) \\ &\|\hat{x}_i(s; t_k) - \tilde{x}_i(s; t_k)\|_{M_i} \leq \xi_i, &(13) \\ &s \in [t_k, t_k + T], \end{aligned}$$

where $\tilde{x}_i(s; t_k)$ is defined as $\tilde{x}_i(t_k; t_k) = x_i(t_k)$, $\dot{\tilde{x}}_i(s; t_k) = f_i(\tilde{x}_i(s; t_k), x_{-i}^0(s), \tilde{u}_i(s; t_k))$, $\tilde{u}_i(s; t_k) \triangleq u_i^*(s; t_{k-1})$ when $s \in [t_k, t_{k-1} + T)$, $\tilde{u}_i(s; t_k) \triangleq K_i x_i^0(s)$ when $s \in [t_{k-1} + T, t_k + T]$ and $u_i^*(s; t_{k-1})$ is the optimal solution of Subproblem i obtained at t_{k-1} and with $u_i^*(s; t_0) = u_i^0(s)$, $\xi_i > 0$.

Denote $x_i^*(s; t_k), s \in [t_k, t_k + T]$ as the state trajectory of (11) with initial condition $x_i^*(t_k; t_k) = x_i(t_k)$ and control input $u_i^*(s; t_k)$.

For any subsystem i over time interval $[t_k, t_{k+1}]$, $k \geq 1$,

1. apply $u_i^*(s; t_k), s \in [t_k, t_{k+1}]$;

2. compute $x_i^0(s), s \in [t_k + T, t_{k+1} + T]$ according to Equation 10;
3. transmit $x_i^0(s), s \in [t_k + T, t_{k+1} + T]$ to downstream neighbours;
4. receive $x_j^0(s), s \in [t_k + T, t_{k+1} + T]$ from upstream neighbours.

In Subproblem i, continuous-time system models, constraints, and control inputs are considered. In practice, however, the MPC control algorithm is usually to be implemented in discrete time. In the numerical example in Section 5, approximate discrete-time problems are solved with integration time much smaller than the sampling interval δ. See Magni and Scattolini[26] for detailed discussion.

In this proposed control algorithm, a dual mode strategy,[23] which uses MPC algorithm when the state is outside of the terminal region and uses terminal controller after the state enters the terminal region, is applied, since if $x \in \Omega_\epsilon$, according to Lemma 1, the linear feedback controller $u = Kx$ stabilizes the system to the origin. However, this requires that every subsystem knows the full state information $x(t_k)$. In the simulation given in Section 5, the MPC algorithm is used even after $x(t)$ enters Ω_ϵ but the system is still stabilized to the origin.

In Subproblem i, constraints (12) and (13) are used to guarantee recursive feasibility and stability. In particular, constraint (12) is used to make sure that the real trajectory is not far away from x^0 so that the nominal model (11) is a reasonable approximation. Constraint (13) is used to guarantee that the solution of the current time instant is not far away from that obtained at the previous time instant, which is critical for stability. Therefore, the objective function J_i can be chosen as any performance index and the commonly used assumptions on the objective function in Mayne et al[1] are not required in our approach.

In Dunbar,[12] each subsystem i needs to transmit the whole predicted state trajectory $x_i^*(s;t_k), s \in [t_k, t_k + T]$ to its downstream neighbours. In our algorithm, however, only the incremental part $x_i^0(t), t \in [t_k + T, t_{k+1} + T]$ needs to be transmitted.

4 | ANALYSIS

In this section, recursive feasibility and stability of the system under the proposed MPC algorithm are established.

4.1 | Feasibility

Since the receding horizon algorithm is optimization based, one may wonder if there exists some future time instant, say t_k, when Subproblem i becomes infeasible. Recursive feasibility is an important property to be established. It means that given an initial feasible solution at time t_0, for all $t_k, k = 1, \dots,$ every Subproblem i is feasible.

In **Subproblem** i, x_{-i}^0 is used to predict the future behaviour of subsystem i. There is a discrepancy between $x_i^*(s;t_k), s \in [t_k, t_k + T]$ and the real trajectory $x_i(s), s \in [t_k, t_k + T]$. We introduce the following lemmas to estimate the discrepancy. The first lemma is similar to Theorem 5 in Russo et al.[27]

Lemma 2. *Consider 2 systems $\dot{x}_0 = f(x_0, \mathbf{0}, t)$ and $\dot{x}_1 = f(x_1, w, t)$, where $w \in \mathbb{W}$ is a bounded disturbance. Suppose that $\frac{\partial f}{\partial x}^T M + M\frac{\partial f}{\partial x} \leq -\beta M$ holds for some positive number β and positive definite matrix M in some region C where the 2 systems evolve. Then $V(t) \leq \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})} + \left(V(0) - \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})}\right)e^{-\frac{1}{2}\beta t}$, where $V(t) = \|x_0(t) - x_1(t)\|_M$, L is a positive constant that satisfies $\|f(x,y,z) - f(x,y',z)\| \leq L\|y - y'\|$, $d = \sup_{w\in\mathbb{W}}\|w\|_M$.*

Proof. Similar to the proof of Theorem 5 in Russo et al,[27] pick $\gamma(r) = x_0(0) + r(x_1(0) - x_0(0))$. Consider an auxiliary trajectory $\dot{x}_r = f(x_r, rw, t)$ that starts from $x_r(0) = \gamma(r), r \in [0,1]$. Define $p(t,r) = \frac{\partial x_r}{\partial r}(t,r)$. Then it follows that $\frac{\partial p}{\partial t}(t,r) = \frac{\partial}{\partial t}(\frac{\partial x_r}{\partial r}) = \frac{\partial}{\partial r}(\frac{\partial x_r}{\partial t}) = \frac{\partial}{\partial r}f(x_r, rw, t) = \frac{\partial f}{\partial x_r}p + \frac{\partial f}{\partial rw}w$.

Consider $\Phi(t,r) = p(t,r)^T M p(t,r)$. Then by the contraction property, we have $\frac{\partial}{\partial t}\Phi(t,r) \leq -\beta\Phi(t,r) + 2\sqrt{\Phi(t,r)}L\frac{d\lambda_{\max}(M^{\frac{1}{2}})}{\lambda_{\min}(M^{\frac{1}{2}})}$.

Note that $\sqrt{\Phi(0,r)} = V(0)$ and by comparison principle

$$\sqrt{\Phi(t,r)} \leq \mu + (V(0) - \mu)e^{-\frac{1}{2}\beta t}, \tag{14}$$

where $\mu = \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})}$, $\forall x \in C, t \geq 0$, and $\forall r \in [0,1]$.

According to the fundamental theorem of calculus, one can write that $x_1(t) - x_0(t) = \int_0^1 p(t,r)dr$ and by inequality (14), $V(t) \leq \int_0^1 \|p(t,r)\|_M dr = \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})} + (V(0) - \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})})e^{-\frac{1}{2}\beta t}$. □

The following lemma provides an estimate of the discrepancy.

Lemma 3. *If over time interval $[t_k, t_k + T]$, the real state trajectory $x_i(s)$ and the predicted state trajectory $x_i^*(s)$ are both in Θ_i defined in (6), $i = 1, \dots, N$, then the inequality $\|x_i(s) - x_i^*(s; t_k)\|_{M_i} \leq \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_k)}), s \in [t_k; t_k + T]$ holds for $i = 1, \dots, N$.*

Proof. Given the initial condition $x_i(t_k)$, the real state trajectory is determined by

$$\dot{x}_i(s) = f_i\left(x_i(s), x_{-i}(s), u_i^*(s; t_k)\right), s \in [t_k; t_k + T].$$

The predicted state trajectory is given by $\dot{x}_i^*(s; t_k) = f_i(x_i^*(s; t_k), x_{-i}^0(s), u_i^*(s; t_k)), s \in [t_k; t_k + T]$.

Since $x_i(s)$ is in Θ_i, $\|x_i(s) - x_i^0(s)\|_{M_i} \leq l_i$. Define $V_i(s) = \|x_i(s) - x_i^*(s; t_k)\|_{M_i}$. By considering $w = x_{-i}(s) - x_{-i}^0(s)$ and noting that $V_i(t_k) = 0$, the result directly follows from Lemma 2. □

On the basis of the above lemmas, we now establish the recursive feasibility of the proposed control algorithm.

Theorem 2. *Suppose that Assumptions 1 to 4 hold, initial feasible control $u^0(t), t \in [t_0, t_0 + T]$ is available at time instant t_0 and Subproblem i is feasible for $i = 1, \dots, N$ at time instant t_k. Then Subproblem i, $i = 1, \dots, N$, are feasible at time instant t_{k+1} with the feasible solution $\tilde{u}_i(s; t_{k+1})$ defined as*

$$\begin{cases} u_i^*(s; t_k), s \in [t_{k+1}, t_k + T) \\ K_i x_i^0(s), s \in [t_k + T, t_{k+1} + T] \end{cases}$$

provided that Equation 7 holds.

Proof. Firstly, we show that $\tilde{u}_i(s; t_{k+1}) \in \mathbb{U}_i$, $s \in [t_{k+1}, t_{k+1} + T]$. When $s \in [t_{k+1}, t_k + T)$, $\tilde{u}_i(s; t_{k+1}) = u_i^*(s; t_k) \in \mathbb{U}_i$ since $u_i^*(s; t_k)$ is the optimal solution of Subproblem i at time instant t_k. When $s \in (t_k + T, t_{k+1} + T]$, by Lemma 1 and Assumption 4, we know that $K_i x_i^0(s) \in \mathbb{U}_i$. Therefore, $\tilde{u}_i(s; t_{k+1}) \in \mathbb{U}_i$, $s \in [t_{k+1}, t_{k+1} + T]$.

In the following of this proof, it will be shown that the state trajectory given by

$$\dot{\tilde{x}}_i(s; t_{k+1}) = f_i\left(\tilde{x}_i(s; t_{k+1}), x_{-i}^0(s), \tilde{u}_i(s; t_{k+1})\right)$$

with initial condition $\tilde{x}_i(t_{k+1}, t_{k+1}) = x_i(t_{k+1})$ satisfies the constraints in Subproblem i. The optimal state trajectory computed at time instant t_k over time interval $[t_{k+1}, t_k + T)$ is given by $\dot{x}_i^*(s; t_k) = f_i(x_i^*(s; t_k), x_{-i}^0(s), u_i^*(s; t_k))$ with initial condition $x_i^*(t_{k+1}; t_k)$.

Consider $V_i(s) = \|x_i^*(s; t_k) - \tilde{x}_i(s; t_{k+1})\|_{M_i}$. According to Lemma 3, we know that $V_i(t_{k+1}) = \|x_i(t_{k+1}) - x_i^*(t_{k+1}; t_k)\|_{M_i} \leq \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i\delta})$. By constraint (12), $\|x^*(t_{k+1}; t_k) - x_i^0(t_{k+1})\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i\delta})$. Therefore, we obtain that $\|\tilde{x}_i(t_{k+1}; t_{k+1}) - x_i^0(t_{k+1})\|_{M_i} \leq \|x_i(t_{k+1}) - x_i^*(t_{k+1}; t_k)\|_{M_i} + \|x^*(t_{k+1}; t_k) - x_i^0(t_{k+1})\|_{M_i} \leq l_i$ that implies that at time instant t_{k+1}, the feasible state trajectory is in the contraction region Θ_i and the contraction property holds. Then by Lemma 2, one can obtain that $\|\tilde{x}_i(s; t_{k+1}) - x_i^*(s; t_k)\|_{M_i} \leq \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i\delta})e^{-\frac{1}{2}\beta_i(s-t_{k+1})}$.

According to constraint (12), $\|x_i^*(s; t_k) - x_i^0(s)\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_k)})$, which leads to

$$\left\|\tilde{x}_i(s; t_{k+1}) - x_i^0(s)\right\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}\left(1 - e^{-\frac{1}{2}\beta_i(s-t_k)}\right) + \frac{c_i}{\beta_i}\left(1 - e^{-\frac{1}{2}\beta_i\delta}\right)e^{-\frac{1}{2}\beta_i(s-t_{k+1})} \triangleq C_i.$$

By some simple calculation, we have $C_i = l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_{k+1})}) + \frac{c_i}{\beta_i}e^{-\frac{1}{2}\beta_i(s-t_k)} - \frac{c_i}{\beta_i}e^{-\frac{1}{2}\beta_i\delta - \frac{1}{2}\beta_i(s-t_{k+1})} = l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_{k+1})})$, which implies that $\|\tilde{x}_i(s; t_{k+1}) - x_i^0(s)\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_{k+1})})$ when $s \in [t_{k+1}, t_k + T]$.

When $s \in [t_k + T, t_{k+1} + T]$, consider the state trajectory given by

$$\dot{\tilde{x}}_i(s; t_{k+1}) = f_i\left(\tilde{x}_i(s; t_{k+1}), x_{-i}^0(s), K_i x_i^0(s)\right),$$

where $x_i^0(s)$ is defined by (10).

Define $E_i(s) = \|\tilde{x}_i(s; t_{k+1}) - x_i^0(s)\|_{M_i}$ with initial condition $E_i(t_k + T) = l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(T-\delta)})$ and $s \in [t_k + T, t_{k+1} + T]$. It follows from Lemma 2 that $E_i(s) \le d_i + (E_i(t_k + T) - d_i)e^{-\frac{1}{2}\beta_i(s-t_k-T)}$. By letting $d_i + (E_i(t_k + T) - d_i)e^{-\frac{1}{2}\beta_i(s-t_k-T)} \le l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_{k+1})})$ and rearranging this inequality, we obtain that

$$\left[(E_i(t_k + T) - d_i)e^{\frac{1}{2}\beta_i T} - \frac{c_i}{\beta_i}e^{\frac{1}{2}\beta_i\delta}\right] e^{-\frac{1}{2}\beta(s-t_k)} \le l_i - \frac{c_i}{\beta_i} - d_i. \tag{15}$$

Note that the equality is achieved when $s = t_k+T$. Therefore, (15) holds $\forall s \in (t_k+T, t_{k+1}+T]$ if $(E_i(t_k+T)-d_i)e^{\frac{1}{2}\beta_i T}-\frac{c_i}{\beta_i}e^{\frac{1}{2}\beta_i\delta} \ge 0$, which is true if $l_i \ge \frac{c_i}{\beta_i} + d_i$.

Finally, constraint (13) trivially holds. Therefore, $\tilde{x}_i(s; t_{k+1}), s \in [t_{k+1}, t_{k+1} + T]$ is a feasible trajectory. □

4.2 | Stability

In this subsection, stability of system (3) under the proposed control algorithm is analysed.

Theorem 3. *Suppose that Assumptions 1 to 4 hold, initial feasible control $u^0(t), t \in [t_0, t_0 + T]$ is available at time instant t_0 and inequalities (7) to (9) are satisfied. Then system (3) is asymptotically stable under the proposed MPC algorithm.*

Proof. Since by assumption, after x enters the terminal region Ω_ϵ, the terminal controller stabilizes the system, we only need to show that x will enter the terminal region in finite time.

Consider nonnegative function $V_k = \int_{t_k}^{t_k+T} \|x^*(s; t_k) - x^0(s)\|_M ds$. Then according to constraint (13), we know that $V_k \le \tilde{V}_k + \int_{t_k}^{t_k+T} \|x^*(s; t_k) - \tilde{x}(s; t_k)\|_M ds \le \tilde{V}_k + \sum_{i=1}^N \xi_i T$, where $\tilde{V}_k \triangleq \int_{t_k}^{t_k+T} \|\tilde{x}(s; t_k) - x^0(s)\|_M ds$. Therefore, we have $V_{k+1} - V_k \le \tilde{V}_{k+1} + \sum_{i=1}^N \xi_i T - V_k$.

By using triangle inequality and noticing that

$$\int_{t_{k+1}}^{t_k+T} \|x^*(s; t_k) - x^0(s)\|_M ds - V_k = -\int_{t_k}^{t_{k+1}} \|x^*(s; t_k) - x^0(s)\|_M ds,$$

it can be obtained that

$$\begin{aligned}\tilde{V}_{k+1} - V_k \le & -\int_{t_k}^{t_{k+1}} \|x^*(s; t_k) - x^0(s)\|_M ds + \int_{t_{k+1}}^{t_k+T} \|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\|_M ds \\ & + \int_{t_k+T}^{t_{k+1}+T} \|\tilde{x}(s; t_{k+1}) - x^0(s)\|_M ds.\end{aligned} \tag{16}$$

In the proof of Theorem 2, we have shown that $\|\tilde{x}_i(s; t_{k+1}) - x_i^*(s; t_k)\|_{M_i} \le \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i\delta})e^{-\frac{1}{2}\beta_i(s-t_{k+1})}$.

From

$$\int_{t_{k+1}}^{t_k+T} \|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\|_M ds \le \sum_{i=1}^N \int_{t_{k+1}}^{t_k+T} \|\tilde{x}_i(s; t_{k+1}) - x_i^*(s; t_k)\|_{M_i} ds$$

and some simple calculation, one can write

$$\begin{aligned}\int_{t_{k+1}}^{t_k+T} \|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\|_M ds &\le \sum_{i=1}^N \int_{t_{k+1}}^{t_k+T} \|\tilde{x}_i(s; t_{k+1}) - x_i^*(s; t_k)\|_{M_i} ds \\ &\le \sum_{i=1}^N \int_{t_{k+1}}^{t_k+T} \frac{c_i}{\beta_i}\left(1 - e^{-\frac{1}{2}\beta_i\delta}\right) e^{-\frac{1}{2}\beta_i(s-t_{k+1})} ds \\ &= \sum_{i=1}^N \frac{2c_i}{\beta_i^2}\left(1 - e^{-\frac{1}{2}\beta_i\delta}\right)\left(1 - e^{-\frac{1}{2}\beta_i(T-\delta)}\right)\end{aligned}$$

and similarly $\int_{t_k+T}^{t_{k+1}+T} \|\tilde{x}(s; t_{k+1}) - x^0(s)\|_M ds \le \sum_{i=1}^N d_i\delta + \frac{2}{\beta_i}(l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(T-\delta)}) - d_i)(1 - e^{-\frac{1}{2}\beta_i\delta})$.

Suppose that over time interval $[t_k, t_{k+1}]$, $\|x(s) - x^0(s)\|_M \ge \gamma\epsilon$. Then by Lemma 3, over this time interval, $\|x^*(s; t_k) - x^0(s)\|_M \ge \|x(s) - x^0(s)\|_M - \|x(s) - x^*(s; t_k)\|_M \ge \gamma\epsilon - \sum_{i=1}^N \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_k)})$.

Substituting the above inequalities into (16) leads to $\tilde{V}_{k+1} - V_k \leq -\gamma\epsilon\delta + \sum_{i=1}^{N}(\frac{c_i}{\beta_i} + d_i)\delta + \sum_{i=1}^{N} \frac{2}{\beta_i}(l_i - d_i - \frac{c_i}{\beta_i})(1 - e^{-\frac{1}{2}\beta_i\delta})$ and then it follows from inequality (9) that there exists a positive constant ρ_i such that $\xi_i T + \frac{2}{\beta_i}(l_i - d_i - \frac{c_i}{\beta_i})(1 - e^{-\frac{1}{2}\beta_i\delta}) \leq \gamma_i\epsilon\delta - (d_i + \frac{c_i}{\beta_i})\delta - \rho_i$, which implies that $V_{k+1} - V_k \leq \sum_{i=1}^{N} \xi_i T + \tilde{V}_{k+1} - V_k \leq -\sum_{i=1}^{N} \rho_i$.

Similar to Theorem 2 in Dunbar,[12] we can now claim that $x(s)$ converges to $\Omega_{x^0(s)}(\gamma\epsilon) \triangleq \{x \mid \|x - x^0(s)\|_M \leq \gamma\epsilon\}$ in finite time. On the other hand, by (10) and Lemma 1, $x^0(s)$ asymptotically converges to the origin. Therefore, $x^0(s)$ converges to $\Omega_{(1-r)\epsilon}$ in finite time and stays in it thereafter, which implies that $x(s)$ converges to Ω_ϵ in finite time. This completes the proof. □

5 | NUMERICAL EXAMPLE

In this section, the proposed algorithm is applied to a building heating ventilation and air-conditioning system with 4 neighbouring rooms, see Ma et al[3] and Betti et al.[28] The system dynamics is given by

$$\dot{T}_1 = \frac{60}{C}\left(\dot{m}_s^1 c_p(T_s - T_1) + \frac{T_2 - T_1}{R} + \frac{T_3 - T_1}{R} + \frac{T_e - T_1}{R_e}\right)$$
$$\dot{T}_2 = \frac{60}{C}\left(\dot{m}_s^2 c_p(T_s - T_2) + \frac{T_1 - T_2}{R} + \frac{T_4 - T_2}{R} + \frac{T_e - T_2}{R_e}\right)$$
$$\dot{T}_3 = \frac{60}{C}\left(\dot{m}_s^3 c_p(T_s - T_3) + \frac{T_1 - T_3}{R} + \frac{T_4 - T_3}{R} + \frac{T_e - T_3}{R_e}\right)$$
$$\dot{T}_4 = \frac{60}{C}\left(\dot{m}_s^4 c_p(T_s - T_4) + \frac{T_2 - T_4}{R} + \frac{T_3 - T_4}{R} + \frac{T_e - T_4}{R_e}\right)$$

where T_j is the temperature of room j, $C = 9163$ kJ/K is the lumped mass of each room, $c_p = 1012$ J/kg·K is the heat capacity of air, $R = 1$ K/kW is the heat resistance between neighbouring rooms, $R_e = 30$ K/kW is the heat resistance between rooms and the outside environment, $T_e = 30°C$ is the outside temperature, $T_s = 16°C$ is the temperature of the supply air, $\dot{m}_s^j$ is the air flow rate of room j. Our aim is to regulate the temperature of each room to the desired set point $T_d = 22$°C by adjusting air flow rate $\dot{m}_s^j \in [0.005\ 3]$ kg/s. The partitions of the states and inputs are $x_1 = [T_1 - T_d\ \ T_2 - T_d]^T, x_2 = [T_3 - T_d\ \ T_4 - T_d]^T, u_1 = [\dot{m}_s^1 - \bar{m}_s\ \dot{m}_s^2 - \bar{m}_s]^T$ and $u_2 = [\dot{m}_s^3 - \bar{m}_s\ \dot{m}_s^4 - \bar{m}_s]^T$ where $\bar{m}_s = \frac{T_e - T_d}{(T_d - T_s)R_e c_p}$. Then the whole system can be rewritten as

$$\dot{x}_1 = f_1(x_1, x_2, u_1)$$
$$\dot{x}_2 = f_2(x_2, x_1, u_2).$$

The matrices M_1 and M_2 are chosen as identity matrix. Since the system is stable around the equilibrium, the terminal controller is chosen as $K = \mathbf{0}$. The parameters ϵ and r in Lemma 1 are, respectively, calculated as 0.5 and 0.4 according to Lemma 1 in Dunbar.[12] β_i is calculated as 0.0267. The design parameters are chosen as: $\alpha = 0.3, l_i = 3, \delta = 5, T = 30, \xi_i = 0.5, \gamma_i = 0.49, i = 1, 2$ to satisfy conditions (7) to (9). The objective functions are chosen as $J_i(x_i(t_k), \hat{u}_i(\cdot, t_k)) = \int_{t_k}^{t_k+T}(\|\hat{x}_i(s;t_k)\|^2 + 0.01\|\hat{u}_i(s;t_k) + \bar{m}_s\|^3)ds, i = 1, 2$, where the quadratic term represents the tracking error while the cubic term represents the power consumed by the supply fan.[29] The initial condition is given by $x_1(0) = (4, 3), x_2(0) = (2, 1)$. Note that under these given parameters, a Lipschitz constant of f_i with respect to x_i is 0.0332, and no feasible parameters can be found to satisfy those sufficient conditions in Dunbar[12] and Liu and Shi.[13] However, by using the proposed method, we are able to obtain an optimal solution. Therefore, compared with the proposed algorithms in Dunbar[12] and Liu and Shi,[13] our algorithm is less conservative. In the case that both Dunbar[12] and Liu and Shi[13] and the proposed method are feasible, it is not clear which one will give better control performance. However, our method allows larger sampling interval. Since the communications among agents and calculations of new control input are executed at each sampling time instant, the communication and computational burden could then be reduced.

The initial feasible state trajectory and control input are shown in Figure 1. The state responses and control inputs under the proposed distributed and centralized algorithms are shown in Figure 2, where the solid line denotes the state response and control input under the distributed control while the dashed line represents those under centralized control. It can be observed that the temperature response under the distributed algorithm is close to that under centralized control. The fan power $\int_0^{50} \sum_{i=1}^{4} (\dot{m}_s^i(\tau))^3 d\tau$ are 11.25×10^4, 9.15×10^4, and 9.08×10^4 for the initial feasible guess and the distributed and the centralized algorithms, respectively. Therefore, by using the proposed algorithm, around 19% energy saving can be achieved compared to the initial feasible control.

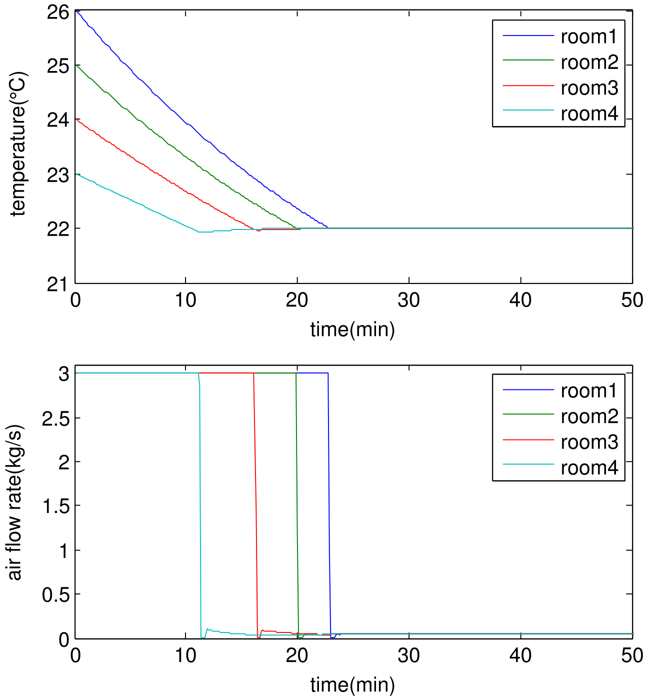

FIGURE 1 Initial feasible state trajectory and control profile [Colour figure can be viewed at wileyonlinelibrary.com]

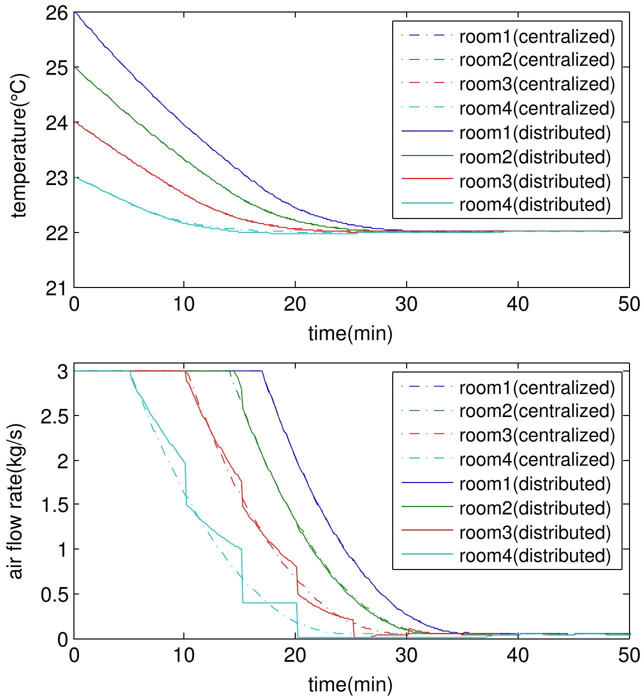

FIGURE 2 Comparison of the state trajectories and control profiles under centralized and distributed algorithms [Colour figure can be viewed at wileyonlinelibrary.com]

Finally, the average computational time with respect to different discretization steps and algorithms is compared in Table 1. The algorithms used in this section are coded in Matlab and run on a PC with Intel Core Duo i5-2400 CPU 3.10 GHz. The optimization problems are described using ICLOCS[30] and solved by IPOPT.[31]

TABLE 1 Average computational time of distributed and centralized algorithm

Steps	120	240	360	480	600
Distributed, s	1.37	1.80	2.14	2.67	3.02
Centralized, s	5.63	8.09	11.62	14.62	18.29

6 | CONCLUSION

A novel distributed nonlinear continuous-time MPC algorithm has been proposed. In this algorithm, the traditional Lipschitz continuity condition is replaced by a contraction condition, which implies that there is a contractive tube around the nominal state trajectory. By leveraging the contraction condition, sufficient conditions were derived to guarantee feasibility and asymptotic stability of the closed-loop control system. A numerical example showed that the proposed MPC algorithm is less conservative than those proposed algorithms in the literature.

REFERENCES

1. Mayne DQ, Rawlings JB, Rao CV, Scokaert POM. Constrained model predictive control: stability and optimality. *Automatica*. 2000;36(6): 789-814.
2. Tettamanti T, Varga I, Kulcsar B, Bokor J. Model predictive control in urban traffic network management. Paper presented at: 16th Mediterranean Conference on Control and Automation, Athens, Greece; 2008.
3. Ma Y, Anderson G, Borrelli F. A distributed predictive control approach to building temperature regulation. Paper presented at: American Control Conference (ACC), San Francisco, California, USA; 2011.
4. Mayne DQ, Seron MM, Rakovi SV. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*. 2005;41(2):219-224.
5. Farina M, Scattolini R. Distributed predictive control: a non-cooperative algorithm with neighbor-to-neighbor communication for linear systems. *Automatica*. 2012;48(6):1088-1096.
6. Giselsson P, Rantzer A. On feasibility, stability and performance in distributed model predictive control. *IEEE Trans Autom Control*. 2014;59(4):1031-1036.
7. Lincoln B, Rantzer A. Relaxing dynamic programming. *IEEE Trans Autom Control*. 2006;51(8):1249-1260.
8. Stewart BT, Wright SJ, Rawlings JB. Cooperative distributed model predictive control for nonlinear systems. *J Process Control*. 2011;21(5):698-704.
9. Li H, Shi Y. Robust distributed model predictive control of constrained continuous-time nonlinear systems: a robustness constraint approach. *IEEE Trans Autom Control*. 2014;59(6):1673-1678.
10. Farina M, Betti G, Scattolini R. Distributed predictive control of continuous-time systems. *Syst Control Lett*. 2014;74:32-40.
11. Liu J, Chen X, Muoz de la Pea D, Christofides PD. Sequential and iterative architectures for distributed model predictive control of nonlinear process systems. *AIChE J*. 2010;56(8):2137-2149.
12. Dunbar WB. Distributed receding horizon control of dynamically coupled nonlinear systems. *IEEE Trans Autom Control*. 2007;52(7):1249-1263.
13. Liu X, Shi Y, Constantinescu D. Distributed model predictive control of constrained weakly coupled nonlinear systems. *Syst Control Lett*. 2014;74:41-49.
14. Lohmiller W, Slotine JJE. On contraction analysis for non-linear systems. *Automatica*. 1998;34(6):683-696.
15. Liu X, Shi Y, Constantinescu D. Robust constrained model predictive control using contraction theory. Paper presented at: IEEE 53rd Annual Conference on Decision and Control (CDC), Los Angeles, California, USA; 2014.
16. Magni L, Scattolini R. Stabilizing decentralized model predictive control of nonlinear systems. *Automatica*. 2006;42(7):1231-1236.
17. Kothare SLdO, Morari M. Contractive model predictive control for constrained nonlinear systems. *IEEE Trans Autom Control*. 2000;45(6):1053-1071.
18. Forni F, Sepulchre R. A differential Lyapunov framework for contraction analysis. *IEEE Trans Autom Control*. 2014;59(3):614-628. https://doi.org/10.1109/TAC.2013.2285771
19. Lohmiller W, Slotine JJE. Nonlinear process control using contraction theory. *AIChE J*. 2000;46(3):588-596.
20. Chen H, Allgöwer F. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*. 1998;34(10):1205-1217.
21. Russo G, di Bernardo M, Sontag ED. A contraction approach to the hierarchical analysis and design of networked systems. *IEEE Trans Autom Control*. 2013;58(5):1328-1331.
22. Bonnabel S, Slotine JJE. A contraction theory-based analysis of the stability of the deterministic extended Kalman filter. *IEEE Trans Autom Control*. 2015;60(2):565-569.
23. Michalska H, Mayne DQ. Robust receding horizon control of constrained nonlinear systems. *IEEE Trans Autom Control*. 1993;38(11):1623-1633.
24. Trodden P, Richards A. Cooperative distributed MPC of linear systems with coupled constraints. *Automatica*. 2013;49(2):479-487.

25. Richards A, How JP. Robust distributed model predictive control. *Int J Control*. 2007;80(9):1517-1531.

26. Magni L, Scattolini R. Model predictive control of continuous-time nonlinear systems with piecewise constant control. *IEEE Trans Autom Control*. 2004;49(6):900-906.

27. Russo G, di Bernardo M, Sontag ED. Global entrainment of transcriptional systems to periodic inputs. *PLoS Comput Biol*. 2010;6(4):e1000 739.

28. Betti G, Farina M, Scattolini R. Realization issues, tuning, and testing of a distributed predictive control algorithm. *J Process Control*. 2014;24(4):424-434.

29. Kelman A, Yudong M, Borrelli F. Analysis of local optima in predictive control for energy efficient buildings. Paper presented at: 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), Orlando, Florida, USA; 2011.

30. P Falugi ECK, Wyk EV. *Imperial College London Optimal Control Software User Guide (ICLOCS)*. London, UK: Department of Electrical Engineering, Imperial College London; 2014.

31. Wächter A, Biegler LT. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math Program*. 2006;106(1):25-57.

How to cite this article: Long Y, Liu S, Xie L, Johansson KH. Distributed nonlinear model predictive control based on contraction theory. *Int J Robust Nonlinear Control*. 2018;28:492–503. https://doi.org/10.1002/rnc.3881