A Heuristic Parameter-dependent Open-loop Model Predictive Control

XIN ZAN¹, LU JIN², and JUN WANG².

¹School of Automation Science and Engineering, Faculty of Electrical and Information Engineering, Xi’an Jiaotong University, Xi’an, 710049, China
²College of Automation, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China

Corresponding author: Jun Wang (e-mail: hellowangjun@yeah.net).

This work is supported by NSFC (Natural Science Foundation of China)-Zhejiang Joint Fund for the Integration of Industrialization and Information under grant U1809207.

ABSTRACT For the uncertain linear systems described by the linear parameter varying (LPV) model, a parameter-dependent open-loop model predictive control (MPC) is proposed. The controller applies a tree trajectory to generate the vertices of uncertainty state predictions. Based on the state prediction tree, the future free control moves are parameter-dependent, whose vertices correspond to those of state predictions. The cost function penalizes the deviations of all the vertices of state/input from their steady-state target values. It is shown that the offset-free property is achieved by this method. A simulation example is given to demonstrate effectiveness of the approach.

INDEX TERMS Model predictive control, linear parameter varying model, heuristic algorithm

I. INTRODUCTION

Model predictive control (MPC) has been widely applied as a representative advanced process control (APC) algorithm in industrial circle since 1978 [1]. Nowadays, MPC expands to both theoretical and industrial fields to obtain insightful results (e.g., referring to [2]–[10] for several good algorithms). The main feature of MPC is its ability to handle physical constraints and multiple variables in a systematic manner (i.e., pose into an receding-horizon optimization problem). At each control interval, MPC optimizes a cost function associated with the future state/output/input predictions based on an explicit model of the system satisfying some physical constraints, then yields a sequence of control moves. However, only the first control move among this sequence is implemented, and the refreshed optimization is performed at the next control interval. Since the future predictions for state/output/input are needed from the system model, the accuracy of this model is crucial for the the future prediction, which as a result, can influence the control performance.

An efficient system model should be representative for a wide class of systems. In MPC studies, there is a widely accepted system model called linear parameter varying (LPV) model. LPV model can include the dynamics of both nonlinear and uncertain systems by the utilizing the dynamic vector for a family of linear models [11]. Usually, thanks to the convexity property of the LPV model, applying LPV model in MPC can yield the convex optimization which can be solved very efficiently using the current optimization method such as interior point method. Another advantage for LPV model is the local linearity for each sub-linear model, which allows for the application of powerful linear design tools. MPC for LPV models, with guaranteed recursive feasibility (of optimization problem) and stability, has been referred to as the synthesis approach. Usually, a synthesis approach is designed based the state-or output-feedback laws (see e.g., [12]–[16]). Some free perturbation items have been added to the state-feedback law in order to enlarge regions of attraction [17]. The class of linearly parameter-dependent Lyapunov functions are proposed for MPC of LPV model in [18], which gives rise to less conservative stability conditions than those arising from classical quadratic Lyapunov functions in e.g. [19]. In [20], the authors presented a class of nonlinearly parameterized Lyapunov functions instrumental to the achievement of more efficient relaxed stability conditions. In [21], an efficient algorithm is given which constructs the maximal admissible set for LPVs. Considering the high-speed control for constrained LPV models, some explicit MPCs have been developed [22], [23]. MPC for LPV models with bounded parameter variations have been investigated in [24]–[26]. In [27], the output feedback MPC is proposed for LPV models based on the quasi-min-max algorithm. In [28], the authors considered the robust MPC for LPV models, where the scheduling parameter of LPV model is known online (advantageous for feedback).
Although there are excellent theoretical results in pursuing the synthesis approaches of MPC, they have not been reported for wide applications in real industrial systems. Instead, the widely applied MPC is the heuristic approach without guaranteeing the stability and recursive feasibility. For practical applications, one of the major drawbacks with synthesis MPC is its high computational burden, as compared with the heuristic one. Another disadvantage of MPC synthesis approach is its conservativeness since a min-max optimization is usually formulated which take into account all possible realization of the system model. Hence, the linear model is typically addressed for heuristic MPC. However, this can be risky for a intrinsic uncertain systems. This paper aims at LPV model, with consideration of state and input constraints, and adopts a parameter-dependent open-loop MPC scheme based on the heuristic MPC. The proposed controller uses a tree trajectory to forecast the vertices of future state predictions, which is inspired by [29]. The optimization problem is properly formulated as a classic quadratic programming (QP) with the cost function involving all vertices of state predictions, input predictions and steady-state targets. By solving this QP, it yields the vertex control moves. By applying the scheme, the computational burden is less than the synthesis MPC and the offset-free control is achieved.

**Notation:** \( \mathbb{R}^n \) is the \( n \)-dimensional Euclidean space, and \( \mathbb{R}^{m \times n} \) the \( m \times n \)-dimensional real matrix space. For any matrix \( A \), \( A^T \) denotes its transpose. For the variable \( x, x(i|k) \) denotes the value at the future time \( k + i \), predicted at \( k \). The symbol \( \star \) implies that the element can be deduced from the symmetry of the matrix. A variable with \( \star \) as superscript indicates that it is the optimal solution of the optimization problem. For the column vectors \( x \) and \( y \), \( [x; y] = [x^T, y^T]^T \).

The time-dependence of MPC decision variables is often omitted for simplicity.

**II. PROBLEM STATEMENT**

Consider the discrete-time LPV model, i.e.,

\[
x(k + 1) = A(k)x(k) + B(k)u(k),
\]

\[
y(k) = Cx(k),
\]

where \( x(k) \in \mathbb{R}^n, u(k) \in \mathbb{R}^m \) are measurable state and input, respectively. We assume that \( [A(k)|B(k)] \in \Omega \), for all \( k \geq 0 \), where

\[
\Omega = \text{Co} \{ [A_1|B_1], [A_2|B_2], \ldots, [A_L|B_L] \}
\]

i.e., there exist \( l \) time-varying nonnegative combining parameter \( \omega_l(k), l \in \{1, \ldots, L\} \) such that

\[
\sum_{l=1}^{L} \omega_l(k) = 1, \quad [A(k)|B(k)] = \sum_{l=1}^{L} \omega_l(k) [A_l|B_l],
\]

where \( [A_l|B_l] \) are known vertices of the polytope.

The input and state constraints are

\[
u(k) \in U = \{ u \in \mathbb{R}^m | -u \leq u \leq \bar{u} \},
\]

\[
x(k + 1) \in X = \{ x \in \mathbb{R}^n | -\bar{x} \leq \Psi x \leq \bar{x} \},
\]

where \( u = [u_1, u_2, \ldots, u_m], \bar{u} = [\bar{u}_1, \bar{u}_2, \ldots, \bar{u}_m]; \psi = [\psi_1, \psi_2, \ldots, \psi_q]; \bar{u}_i > 0, \psi_i > 0, i = 1 \ldots m; \bar{x}_j > 0, \psi_j > 0, j = 1 \ldots q; \Psi \in \mathbb{R}^{q \times n} \).

In practice, (1) is short for

\[
\nabla x(k + 1) = A(k)\nabla x(k) + B(k)\nabla u(k).
\]

Namely, (1) neglects the symbol \( \nabla \) in (4), where \( \nabla x = x - x_{eq} \) and \( \nabla u = u - u_{eq} \). \( x_{eq} \) and \( u_{eq} \) denote the steady-state operating points (equilibrium) of the system.

The system is said to be at steady state at time \( T \) if

\[
x_{ss} = x(k) = x(k + 1), \forall k \geq T
\]

\[
u_{ss} = u(k) = u(k + 1), \forall k \geq T
\]

\[
y_{ss} = y(k) = y(k + 1), \forall k \geq T
\]

where \( y_{ss}, u_{ss}, x_{ss} \) are the steady-state targets (setpoints) of \( y, u \) and \( x \), respectively. Since there is no uncertainty in matrix \( C \), we have, at the steady state,

\[
y_{ss} = Cx_{ss}
\]

**III. THE OPEN-LOOP CONTROL PROBLEM**

In this section, in order to effectively counteract the time-varying uncertainty, an effective method is designed which calculates the vertex control moves for all corners of the uncertainty evolution.

Define the vertex control moves

\[
\pi_u = \{ u(0|k), u^1(1|k), \ldots, u^{l_{N-2}\ldots0}(N - 1|k) \},
\]

\[
l_j = 1 \ldots L, j = 0 \ldots N - 2,
\]

where \( N \) is the control horizon. Note that the future control moves are based on the vertices of the uncertain polytope. With the increase of \( N \), the number of vertices are increased dramatically. Then, the corresponding vertex state predictions are

\[
x^{l_0}(1|k) = A_{l_0}x(k) + B_{l_0}u(0|k),
\]

\[
x^{l_i\ldots0}(i + 1|k) = A_{l_i}x^{l_i\ldots0}(i|k) + B_{l_i}u^{l_i\ldots0}(i|k),
\]

\[
i = 1 \ldots N - 1, l_j = 1 \ldots L, j = 0 \ldots N - 1,
\]

where \( \pi_x = \{ x^{l_0}(1|k), x^{l_1\ldots0}(2|k), \ldots, x^{l_{N-1}\ldots0}(N|k) \} \) are vertex state predictions.

The true control move \( u(i|k) \) for \( i > 0 \) is defined by

\[
u(i|k) = \sum_{l_0\ldots l_{i-1} = 1}^{L} \left( \prod_{h=0}^{i-1} \omega_{l_h}(k + h) \right) u^{l_1\ldots l_{i-1}0}(i|k),
\]

\[
\sum_{l_0\ldots l_{i-1} = 1}^{L} \left( \prod_{h=0}^{i-1} \omega_{l_h}(k + h) \right) = 1,
\]

\[
i = 1 \ldots N - 1
\]
where \( u(i|k) \) is parameter-dependent, i.e., each \( u(i|k) \) is a convex combination through the parameters \( \omega_{l_h}(k + h) \). According to (1) and (7), the future state predictions are found as

\[
x(1|k) = A(k)x(0|k) + B(k)u(0|k)
\]

\[
= \sum_{l_0=1}^{L} \omega_{l_0}(k) [A_{l_0}x(k) + B_{l_0}u(0|k)]
\]

\[
= \sum_{l_0=1}^{L} \omega_{l_0}(k)x^{l_0}(1|k)
\]

\[
x(2|k) = A(k + 1)x(1|k) + B(k + 1)u(1|k)
\]

\[
= \sum_{l_1=1}^{L} \omega_{l_1}(k + 1) [A_{l_1}x(1|k) + B_{l_1}u(1|k)]
\]

\[
= \sum_{l_1=1}^{L} \omega_{l_1}(k + 1) \left[ A_{l_1} \sum_{l_0=1}^{L} \omega_{l_0}(k)x^{l_0}(1|k) + B_{l_1} \sum_{l_0=1}^{L} \omega_{l_0}(k)u^{l_0}(1|k) \right]
\]

\[
= \sum_{l_1=1}^{L} \sum_{l_0=1}^{L} \omega_{l_1}(k + 1) \omega_{l_0}(k)x^{l_1l_0}(2|k)
\]

Apparently, \( x(i|k) \) belongs to polytope, i.e.,

\[
\begin{bmatrix}
  x(1|k) \\
  x(2|k) \\
  \vdots \\
  x(N|k)
\end{bmatrix}
= \sum_{l_0=1}^{L} \prod_{h=0}^{N-1} \omega_{l_h}(k + h) \begin{bmatrix}
  x^{l_0}(1|k) \\
  x^{l_1l_0}(2|k) \\
  \vdots \\
  x^{l_{N-1}l_1l_0}(N|k)
\end{bmatrix}
\]

\[
, \sum_{l_0 \cdots l_{i-1}=1}^{L} \prod_{h=0}^{i-1} \omega_{l_h}(k + h) = 1, \ i = 1 \ldots N,
\]

\[
\begin{bmatrix}
  x^{l_0}(1|k) \\
  x^{l_1l_0}(2|k) \\
  \vdots \\
  x^{l_{N-1}l_1l_0}(N|k)
\end{bmatrix}
= \begin{bmatrix}
  A_{l_0} \\
  A_{l_1}A_{l_0} \\
  \vdots \\
  \prod_{i=0}^{N-1} A_{l_{i-1}}
\end{bmatrix} x(k)
\]

\[
+ \begin{bmatrix}
  B_{l_0} \\
  A_{l_1}B_{l_0} \\
  \vdots \\
  \prod_{i=0}^{N-2} A_{l_{i-1}}B_{l_0}
\end{bmatrix} \begin{bmatrix}
  u(0|k) \\
  u^{l_0}(1|k) \\
  \vdots \\
  u^{l_{N-2}l_1l_0}(N - 1|k)
\end{bmatrix}
\]
Note that since the parameters \( w(k) \) is completely unknown, so the accurate value of state is unknown. Hence, we utilize the vertex state predictions for MPC controller design, where the accurate value of state is allowed to vary in the polytope.

We define the following positive definite quadratic function with respect to vertices \( \pi_x \) and \( \pi_u \) as (10). In (10), \( \mathcal{P}_1, \cdots, \mathcal{P}_N \) are nonnegative weighing matrices. It is unnecessary that the steady-state targets are always equal to the equilibrium.

The objective of the control problem is to find the control actions that, once implemented, drive all branches (vertices) in the tree trajectory to converge to \( x_{ss} \) and \( u_{ss} \). Accordingly, let vertices \( \pi_x \) and \( \pi_u \) be the decision variables. The optimization problem at each control interval \( k \) is formulated as a QP problem

\[
\begin{align*}
\min_{\pi_x, \pi_u} & \quad J_0^N(k), \\
\text{s.t.} & \quad (9), (11b), (11c) \\
& \quad -u \leq u(0|k) \leq \bar{u}, \quad -u \leq u_{l_1 \cdots l_N}(i|k) \leq \bar{u}, \\
& \quad i = 1 \cdots N - 1, \quad l_i - 1 = 1 \cdots L.
\end{align*}
\]

(11a)

(11b)

(11c)

After the optimization problem is solved, only \( u(0|k) \) is implemented on the plant. Each diagonal block of \( \tilde{\Psi} \) is \( \Psi \). The approach based on the optimization problem (11) is called the open-loop model predictive heuristic control (MPHC).

Remark 1: Vertex control moves, vertex state predictions and the cost function as \( \tilde{J}_0^N(k) \) are found in [29]. This open-loop MPHC has some features, i.e., i) the computation load is less than the synthesis MPC; ii) stability of the system cannot be proved theoretically; iii) suppose that weighting matrices \( \mathcal{P} \) and \( \mathcal{S} \) are positive definite, when \( y_{ss} \neq 0 \) and \( u_{ss} \neq 0 \), it is not easy to achieve \( J_0^N(\infty) = 0 \), i.e., there may exist the offset. This is because usually, \( C_2x_{l_1 \cdots l_N}(N - 1|k) = y_{ss} \neq 0 \) cannot hold for all \( i = 1, 2, \cdots, N \) and \( l_i - 1 \cdots l_N \) even if the closed-loop system is stable.

IV. AN IMPROVED OPEN-LOOP MPHC

Since we cannot give deterministic \([A(k)|B(k)]\), usually, \( y_{ss} \) and \( u_{ss} \) do not always satisfy the following equations:

\[
\begin{align*}
x_{ss} &= \hat{A}_{ss}x_{ss} + \hat{B}_{ss}u_{ss} \\
y_{ss} &= \hat{C}x_{ss}
\end{align*}
\]

(12)

However, there is an exception, i.e., when \( k \to \infty \), \([A(k)|B(k)]\) can converge to the fixed \([\hat{A}_{ss}|\hat{B}_{ss}]\). In this case, \( y_{ss} \) and \( u_{ss} \) in the optimization problem (11a) must satisfy (12).

In general, in order to achieve offset-free control, we can assume that \( x_{ss} \) and \( u_{ss} \) satisfy the steady-state nonlinear equation

\[
g(x_{ss} + x_{eq}, u_{ss} + u_{eq}) = 0,
\]

(13)

where \( g(\cdot) \) is assumed to be Lipschitz continuous and differentiable with respect to \( x \) and \( u \) in \( \mathbb{X} \times \mathbb{U} \), with \( g(0,0) = 0 \). When we obtain \( x_{ss} \) and \( u_{ss} \) from (13), we calculate

\[
\begin{align*}
\begin{bmatrix}
x_{ss}^0 \\
x_{ss}^{l_1l_0} \\
\vdots \\
x_{ss}^{l_{N-1}l_1l_0}
\end{bmatrix} = &\begin{bmatrix}
A_{l_0} & 0 \\
A_{l_1} & A_{l_0} \\
\vdots \\
\prod_{i=0}^{N-2} A_{l_i} & A_{l_{N-1}} \cdot \cdots \cdot \cdot \cdot \cdot A_{l_1} \\
\prod_{i=0}^{N-3} A_{l_i} & A_{l_{N-1}} \cdot B_{l_0} \\
\prod_{i=0}^{N-4} A_{l_i} & A_{l_{N-1}} \cdot B_{l_1} \\
\vdots \\
\prod_{i=0}^{N-2} A_{l_i} & A_{l_{N-1}} \cdot B_{l_{N-1}} \\
\prod_{i=0}^{N-3} A_{l_i} & A_{l_{N-1}} \cdot B_{l_{N-1} \cdot \cdots \cdot \cdot \cdot \cdot A_{l_1} \\
\end{bmatrix} \begin{bmatrix}
B_{l_0} \\
B_{l_1} \\
\vdots \\
B_{l_{N-1}} \\
\end{bmatrix} & \begin{bmatrix}
x_{ss} \\
u_{ss} \\
\end{bmatrix}
\end{align*}
\]

(14)
Then, the cost function \( J_0^N(k) \) is replaced by

\[
J_0^N(k) = \sum_{l_0=1}^{L} \| x^{l_0}(1|k) - x^{l_0}_{ss} \|_{2,1,l_0}^2 + \| u(0|k) - u_{ss} \|_{2,1,0}^2 + \sum_{l_1=1}^{L} \sum_{l_0=1}^{L} \| x^{l_1,l_0}(2|k) - x^{l_1,l_0}_{ss} \|_{2,1,l_0}^2 + \sum_{l_0=1}^{L} \| u^{l_0}(1|k) - u_{ss} \|_{2,1,l_0}^2 + \cdots + \sum_{l_{N-1}=1}^{L} \sum_{l_1=1}^{L} \| x^{l_{N-1},l_1,l_0}(N|k) - x^{l_{N-1},l_1,l_0}_{ss} \|_{2,1,l_0}^2 + \sum_{l_{N-2}=1}^{L} \cdots \sum_{l_1=1}^{L} \| x^{l_{N-2},l_1,l_{N-1},l_0}(N-1|k) - u_{ss} \|_{2,1,l_0}^2
\]

The optimization problem (11) is updated to

\[
\min_{\pi_u, \pi_x} J_0^N(k), \quad \text{s.t.} \ (9), \ (11b), \ (11c)
\]

Remark 2: In [29], if \( J_0^N(k) \) is utilized, the offset-free control is achieved because \( x_{ss} \) and \( u_{ss} \) are obtained by a special procedure. Otherwise the proof of Theorem 5.2 in [29] must take advantage of the cost function \( J_0^N(k) \) in this paper. The open-loop MPHC algorithm is summarized as follows.

Algorithm 1.

Off-line Stage:

i) Find appropriate weighing matrices \( \mathcal{Q}_{1,l_0}, \mathcal{Q}_{2,l_1,l_0}, \cdots, \mathcal{Q}_{N,l_{N-1},l_1,l_0} \).

ii) Find the steady-state equations \( \bar{g}(\cdot) \) according to a given nonlinear system. Calculate the steady-state \( x_{ss} \) and \( u_{ss} \) in (13).

iii) Calculate \( x^{l_0}_{ss}, x^{l_1,l_0}_{ss}, \cdots, x^{l_{N-1},l_1,l_0}_{ss}, u^{l_0}_{ss}, u^{l_1,l_0}_{ss}, \cdots, u^{l_{N-1},l_1,l_{N-2},l_0}_{ss} \) according to (14).

On-line Stage:

i) Solve optimization problem (15) and obtain optimal control sequence \( \pi_u \) and predictive state sequence \( \pi_x \).

ii) Implement \( u^*(0|k) \) into the plant.

iii) Set \( k = k + 1 \), go to i).

V. NUMERICAL EXAMPLE

A nonlinear model of a continuous stirred tank reactor (CSTR) is adopted in this simulation (see Figure 1). With constant volume, the CSTR for an exothermic, irreversible reaction \( A \rightarrow B \) is described by

\[
\begin{align*}
\dot{C}_A(t) &= \frac{q}{V} (C_A^{eq} - C_A(t)) - k_0 \exp \left( -\frac{E}{RT(t)} \right) C_A(t), \\
\dot{T}(t) &= \frac{q}{V} (T_c - T(t)) + \frac{(-\Delta H)}{\rho C_p} k_0 \exp \left( -\frac{E}{RT(t)} \right) C_A(t) + \frac{UA}{\rho C_p} (T_c(t) - T(t)),
\end{align*}
\]

where \( C_A \) is the concentration of material \( A \) in the reactor, \( T \) the reactor temperature, \( T_c \) the coolant stream temperature, \( V \) and \( UA \) denote the volume of the reactor and the rate of heat input, respectively. \( k_0, E, \) and \( \Delta H \) denote the preexponential constant, the activation energy, and the enthalpy of the reaction, respectively. \( C_p \) and \( \rho \) stand for the heat capacity and density of the fluid in the reactor, respectively. The objective is to regulate \( T \) by manipulating \( T_c \) satisfying \( 328 \text{ K} \leq T_c(t) \leq 348 \text{ K} \).

Denote the non-zero equilibrium as \( \{ C_A^{eq}, T_c^{eq} \} \). Choose \( C_A^{eq} = 0.5 \text{ mol/l} \), \( T_c^{eq} = 350 \text{ K} \), \( 340 \text{ K} \leq T_c \leq 360 \text{ K} \), \( 0 \leq C_A \leq 1 \text{ mol/l} \), \( q = 100 \text{ l/min} \), \( C_A^{} = 0.9 \text{ mol/l}, T_i = 350 \text{ K}, V = 100 \text{ l}, \rho = 1000 \text{ g/l}, C_p = 0.239 \text{ J/(g K)}, \Delta H = -2.5 \times 10^4 \text{ J/mol}, E/R = 8750 \text{ K}, k_0 = 3.456 \times 10^{10} \text{ min}^{-1}, \) \( UA = 5 \times 10^4 \text{ J/(min K)} \).

VOLUME 4, 2016

FIGURE 1: continuous stirred-tank reactors
Define
\[
\varphi_1(x_2) = k_0 \exp \left( -\frac{E/R}{x_2 + T_{\text{eq}}} \right),
\]
\[
\varphi_2(x_2) = k_0 \left[ \exp \left( -\frac{E/R}{x_2 + T_{\text{eq}}} \right) - \exp \left( -\frac{E/R}{T_{\text{eq}}} \right) \right] \frac{C_{\Delta A}}{x_2},
\]
\[
\varphi_1^0 = \frac{\varphi_1(x_2) + \varphi_1(x_2)/2}{2}, \quad \varphi_2^0 = \frac{\varphi_2(x_2) + \varphi_2(x_2)/2}{2}, \quad g_1(x_2) = \varphi_1(x_2) - \varphi_1^0, \quad g_2(x_2) = \varphi_2(x_2) - \varphi_2^0,
\]
\[
h_1 = \frac{1}{2} g_1(x_2) - g_1(x_2), \quad h_2 = \frac{1}{2} g_1(x_2) - g_1(x_2),
\]
\[
h_3 = \frac{1}{2} g_2(x_2) - g_2(x_2), \quad h_4 = \frac{1}{2} g_2(x_2) - g_2(x_2).
\]
Then (16) can be exactly represented by
\[
x(t) = A(t)x(t) + B(t)u(t),
\]
\[
\dot{A}(t) = \sum_{l=1}^{4} \omega_l(t)A_l, \quad \dot{B}(t) = \sum_{l=1}^{4} \omega_l(t)B_l,
\]
where
\[
A_1 = \begin{bmatrix}
-\frac{\varphi_1}{\rho C_p} - \frac{g_2}{\rho C_p} & -\frac{\varphi_1^0}{\rho C_p} & -\frac{g_1}{\rho C_p} - \frac{\varphi_1^0}{\rho C_p} & -\frac{g_1}{\rho C_p} - \frac{\varphi_1^0}{\rho C_p} \\
\end{bmatrix},
\]
\[
A_2 = \begin{bmatrix}
-\frac{\varphi_1}{\rho C_p} - \frac{g_2}{\rho C_p} & -\frac{\varphi_1^0}{\rho C_p} & -\frac{g_1}{\rho C_p} - \frac{\varphi_1^0}{\rho C_p} & -\frac{g_1}{\rho C_p} - \frac{\varphi_1^0}{\rho C_p} \\
\end{bmatrix},
\]
\[
A_3 = \begin{bmatrix}
-\frac{\varphi_1}{\rho C_p} - \frac{g_2}{\rho C_p} & -\frac{\varphi_1^0}{\rho C_p} & -\frac{g_1}{\rho C_p} - \frac{\varphi_1^0}{\rho C_p} & -\frac{g_1}{\rho C_p} - \frac{\varphi_1^0}{\rho C_p} \\
\end{bmatrix},
\]
\[
A_4 = \begin{bmatrix}
-\frac{\varphi_1}{\rho C_p} - \frac{g_2}{\rho C_p} & -\frac{\varphi_1^0}{\rho C_p} & -\frac{g_1}{\rho C_p} - \frac{\varphi_1^0}{\rho C_p} & -\frac{g_1}{\rho C_p} - \frac{\varphi_1^0}{\rho C_p} \\
\end{bmatrix},
\]
\[
B_1 = B_2 = B_3 = B_4 = \begin{bmatrix}
0 \\
\frac{UA}{\rho C_p} \\
\frac{UA}{\rho C_p} \\
\frac{UA}{\rho C_p}
\end{bmatrix}.
\]
By discretizing the continuous system (17) with sampling period \( T_s = 0.05 \text{min} \), we obtain the discrete-time LPV model, i.e.,
\[
A_1 = \begin{bmatrix}
0.8837 & -0.0011 \\
6.9390 & 0.9645
\end{bmatrix}, \quad A_2 = \begin{bmatrix}
0.9568 & -0.0011 \\
-0.7085 & 0.96459
\end{bmatrix},
\]
\[
A_3 = \begin{bmatrix}
0.9202 & -0.0018 \\
3.1152 & 1.0368
\end{bmatrix}, \quad A_4 = \begin{bmatrix}
0.9202 & -0.0005 \\
3.1152 & 0.8922
\end{bmatrix},
\]
\[
B_1 = B_2 = B_3 = B_4 = \begin{bmatrix}
0 \\
0.1046
\end{bmatrix}.
\]
Base on (16), letting \( \dot{C}_{\Delta A} = 0 \) and \( \dot{T} = 0 \), we obtain the steady-state model \( g(x_{ss}, u_{ss}) \) of (16). Choose the input steady-state setpoint \( g(x_{ss}, u_{ss}) = (0,0) \). From (13), we have \( C_{\Delta A}^s = 0.693, T_{ss}^s = 343.465 \). Thus, we have \( x_{ss} = [0.135; -9.982], u_{ss} = -8 \). Choose the initial state
\[
x(0) = [0.3; 3], N = 5.
\]
To illustrate the effectiveness of the proposed approach, we take the synthesis MPC in [30] which also uses the tree trajectory approach for comparison. The simulation results are shown in Figures 2 and 3. From the figures we find that the values of the input and state finally converge to the steady-state setpoints, while the deviations between the actual steady-state values and the steady-state setpoints are zero. This reveals that the offset-free property is achieved. However, since [30] needs to guarantee the stability and considers all possible realization of the system model in the robust worst-case manner, optimization inevitably suffers from high computational burden (see Table 1).

![FIGURE 2: State responses](image)

![FIGURE 3: Control input](image)

| Approach in this paper | Approach in [30] |
|------------------------|------------------|
| Total time for 100 simulation steps (seconds) | 230.4209 | 884.2202 |

**TABLE 1: Computational time**

VI. CONCLUSIONS

In this paper, a parameter-dependent open-loop MPC for LPV model is investigated. The controller utilizes a tree trajectory to predict the vertices of future state predictions. Based on these predictions, the optimization problem is transformed into a QP involving all vertices of state predictions, input predictions and steady-state targets. Under this scheme, the offset-free control is achieved with a low computational burden.
REFERENCES

[1] J. Richalet, A. Rault, J.L. Testud, and J. Papon, Model predictive heuristic control: Applications to industrial processes, Automatica, 14(5): 413–428, 1978.

[2] D.Q. Mayne, Model predictive control: Recent developments and future promise, Automatica, 50: 2967–2986, 2014.

[3] P.D. Christofides, R. Scattolini, D.M.D.L. Pena, and J. Liu, Distributed model predictive control: A tutorial review and future research directions, Computers & Chemical Engineering, 51(14): 21–41, 2013.

[4] M. Ellis, H. Durand, and P.D. Christofides, A tutorial review of economic model predictive control methods, Journal of Process Control, 24(8): 1156–1178, 2014.

[5] R. Scattolini, Architectures for distributed and hierarchical Model Predictive Control A review, Journal of Process Control, 19(5): 723–731, 2009.

[6] H. Li, W. Yan, Receding horizon control based consensus scheme in general linear multi-agent systems, Automatica, 56: 12–18, 2015.

[7] J. Zhao, Y. Zhu, R. Patwardhan, Identification of k-step-ahead prediction error model and MPC control, Journal of Process Control, 24:48–56, 2015.

[8] M. Fan, Z. Chen, H. Zhang, Semi-global consensus of nonlinear second-order multijagent systems with measurement output feedback, IEEE Transactions on Automatic Control, 59(8): 2222–2227, 2014.

[9] X. Liu, D. Jiang, and K. Y. Lee, Quasi-min-max fuzzy MPC of UTSG water level based on off-line invariant set, IEEE Transactions on Nuclear Science, 62(5): 2266-2272, 2015.

[10] Y. Zheng, S. Li, and H. Qiu, Distributed model predictive control for large-scale systems, IEEE Transactions on Control Systems Technology, 21(3): 991-998, 2013.

[11] J. Belikov, Ülle Kotta, and M. Tönso, Comparison of LPV and nonlinear system theory: A realization problem , Systems & Control Letters, 64(1): 72–78, 2014.

[12] D. He, H. Huang, Q. Chen, Quasi-min-max MPC for constrained nonlinear systems with guaranteed input-to-state stability, Journal of the Franklin Institute, 351(6): 3405–3423, 2014.

[13] T. Zou, S. Li, Stabilization via extended nonquadratic boundedness for constrained nonlinear systems in Takagi-Sugeno’s form, Journal of the Franklin Institute, 348(10): 2849–2862, 2011.

[14] W. Yang, G. Feng, T. Zhang, Robust Model Predictive Control for Discrete-Time Takagi-Sugeno Fuzzy Systems With Structured Uncertainties and Persistent Disturbances, IEEE Transactions on Fuzzy Systems, 22(5): 1213–1228, 2011.

[15] Y. Zou, J. Lam, Y. Niu, D. Li, Constrained predictive control synthesis for quantized systems with Markovian data loss, Automatica, 55: 217–225, 2015.

[16] J. Hu, B. Ding, An efficient offline implementation for output feedback min-max MPC, International Journal of Robust and Nonlinear Control, 29(2): 492–506, 2019.

[17] B. Kouvaritakis, J. A. Rossiter, and J. Schuurmans, Efficient robust predictive control, IEEE Transactions on Automatic Control, 45(8): 1454–1459, 2000.

[18] W. J. Mao, Robust stabilization of uncertain time-varying discrete systems and comments on an improved approach for constrained robust model predictive control,” Automatica, 39(6): 1109–1112, 2003.

[19] Y. Lu, Y. Arkun, Quasi-min-max MPC algorithms for LPV systems,” Automatica, 36(4): 527–540, 2000.

[20] E. Garone, and A. Casavola, Receding horizon control strategies for constrained LPV systems based on a class of nonlinearly parameterized Lyapunov functions,” IEEE Transactions on Automatic Control, 31(1): 2354–2360, 2012.

[21] B. Pluymers, J. Rossiter, and B. De Moor, The efficient computation of polyhedral invariant sets for linear systems with polytopic uncertainty,” In American control conference: 804–809, 2005.

[22] T. Besselmann, J. Löfberg, and M. Morari, Explicit MPC for LPV systems: stability and optimality,” IEEE Transactions on Automatic Control, 53(3): 2322–2332, 2012.

[23] J. Zhang, X. Xiu, K-d tree based approach for point location problem in explicit model predictive control,” Journal of the Franklin Institute, 355(3): 5431–5451, 2018.

[24] D. Li, Y. Xi, The feedback robust MPC for LPV systems with bounded rates of parameter changes,” IEEE Transactions on Automatic Control, 55(2): 503–507, 2010.

[25] P. Park, S. C. Jeong, Constrained RHC for LPV systems with bounded rates of parameter variations,” Automatica, 40(5): 865–872, 2004.

[26] P. Zheng, D. Li, Y. Xi, and J. Zhang, Improved model prediction and RMPC design for LPV systems with bounded parameter changes,” Automatica, 49(12): 3695–3699, 2013.

[27] J. H. Park, T. H. Kim, and T. Sugie, Output feedback model predictive control for LPV systems based on quasi-max-min algorithm, Automatica, 47(9): 2052–2058, 2011.

[28] S. Yu, C. Böhm, H. Chen, and F. Allgöwer, Model predictive control of constrained LPV systems, International Journal of Control, 85(6): 671–683, 2012.

[29] Y. J. Wang, J. B. Rawlings, A new robust model predictive control method I: theory and computation, Journal of Process Control, 14(3): 231–247, 2004.

[30] B. Ding, Properties of parameter-dependent open-loop MPC for uncertain systems with polytopic description, Asian Journal of Control,12(1): 58–70, 2010.