Preprints of the 10th IFAC International Symposium on Dynamics and Control of Process Systems
The International Federation of Automatic Control
December 18-20, 2013. Mumbai, India

Robust MPC Based on Polyhedral Invariant Sets for LPV Systems

Soorathep Kheawhom * and **Pornchai Bumroongsri** **

* *Computational Process Engineering, Department of Chemical Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330 Thailand (e-mail: soorathep.k@chula.ac.th).*
** *Department of Chemical Engineering, Faculty of Engineering, Mahidol University, Nakhon Pathom 73170 Thailand (e-mail: pornchai.bum@mahidol.ac.th).*

Abstract: A robust model predictive control (RMPC) using polyhedral invariant sets for linear parameter varying (LPV) systems is presented in this work. A sequence of state feedback gains associated with a sequence of nested polyhedral invariant sets is constructed off-line in order to reduce the computational burdens. At each control iteration, when the measured state lies between any two adjacent polyhedral invariant sets constructed, a state feedback gain is determined by interpolation of two pre-computed state feedback gains incorporated with scheduling parameters. Three interpolation algorithms are proposed. In the first algorithm, the real-time state feedback gain is determined by maximizing the state feedback gain with subjected to a set of constraints associated with current invariant set. In the second algorithm, the real-time state feedback gain is calculated by minimizing the violation of the constraints of the adjacent inner invariant set with subjected to a set of constraints associated with current invariant set. In the last algorithm, the real-time state feedback gain is obtaned by minimizing the upper bound of infinite horizon worst case performance cost, which is estimated by Lyapunov function at current state, with subjected to a set of constraints associated with current invariant set. The controller design is illustrated with a case study of nonlinear two-tank system. The simulation results showed that the proposed RMPC with interpolation provides a better control performance while on-line computation is still tractable as compared to previously reported algorithms.

Keywords: linear parameter varying system; polyhedral invariant set; model predictive control; robust stability; stabilizable region.

1. INTRODUCTION

Model predictive control (MPC) is known as an effective control algorithm to deal with multiple input-multiple output processes. At each control iteration, MPC uses an explicit model to solve an optimal control problem, and implements the first element of the optimal input sequence computed. However, conventional MPC based on a linear model is often unsuitable for controlling nonlinear systems. The performance of linear MPC will deteriorate as the discrepancy between the real process and the model used increases (Morari and Lee, 1999).

Though, the behaviour of a nonlinear system is preferably captured by a nonlinear process model, MPC based on nonlinear model is computationally prohibitive in practical situations. To overcome the excessive computational cost of MPC application for large-scale nonlinear systems, representing the process model in a form of Linear Parameter Varying (LPV) systems has been recieving increasing attention(Paijmans et al., 2008). Thus, the synthesis of MPC for LPV system has been motivated(Lu and Arkun, 2000).

An on-line RMPC for LPV systems using parameter-dependent Lyapunov function was introduced by Wada et al. (2006). At each control iteration, the ellipsoidal invariant set containing the measured state is constructed in order to guarantee robust stability. However, the associated optimization problem must be solved on-line, the algorithm requires a relatively high computational effort.

Bumroongsri and Kheawhom (2012a) introduced an off-line RMPC for LPV systems. The sequences of state feedback gains corresponding to the sequences of ellipsoidal invariant sets are pre-computed off-line. At each control iteration, the smallest ellipsoid containing the state measured is determined. The corresponding real-time state feedback gain is obtained by linear interpolation between the pre-computed state feedback gains. The ellipsoidal invariant set computed at each control iteration is only an approximation. Thus, the algorithm trades off optimality in order to reduce on-line computational time.

Though the polyhedral invariant set has some advantages over the ellipsoidal invariant set such as better handling of asymmetric constraints and enlargement of stabilizable region (Pluymers et al., 2005), the ellipsoidal invariant set is usually used in robust model predictive control(RMPC) formulation due to its relatively low on-line computational complexity. In recent years, an off-line RMPC algorithm based on polyhedral invariant set has been developed by Bumroongsri and Kheawhom (2012b). A sequence of polyhedral invariant sets corresponding to a sequence of pre-computed state feedback gains is constructed off-line. At each control iteration, the smallest polyhedral invariant set containing the measured state is determined. The corresponding state feedback gain is then implemented to the process without interpolation of the pre-computed state feedback gains. Unfortunately, the conservativeness is obtained because the control law implemented at each control iteration is an approximation of the optimal control law. Moreover, the input discontinuities caused by a switching between state feedback control laws are occurred. Therefore, the algorithm requires constructing a large number of polyhedral invariant sets, hence large data storage, in order to improve the control performance and reduce the input discontinuities. Later, an interpolation technique for polyhedral invariant sets was introduced to off-line RMPC for polytopic uncertain systems in order to reduce conservativeness and improve the control performances (Kheawhom and Bumroongsri, 2013; Bumroongsri and Kheawhom, 2013).

In this paper, we present a robust model predictive control (RMPC) based on polyhedral invariant sets for LPV systems. The algorithm constructs off-line a sequence of nested polyhedral invariant sets corresponding to a sequence of state feedback gains. At each control iteration, when the state measured lies between any two adjacent polyhedral invariant sets constructed, a real-time state feedback gain is determined by interpolation of two pre-computed state feedback gains incorporated with scheduling parameters. Three interpolation algorithms are proposed. The algorithm proposed requires very small computation complexity. The paper is organized as follows. In section 2, the problem description is presented. In section 3, the RMPC with interpolation algorthms proposed are presented. In section 4, we illustrate the implementation of the algorithms proposed. Finally, in section 5, we conclude the paper.

Notation: For a matrix A, A^T denotes its transpose, A^{-1} denotes its inverse. I denotes the identity matrix. For a vector x, $x(k/k)$ denotes the state measured at real time k, $x(k+i/k)$ denotes the state at prediction time $k+i$ predicted at real time k. The symbol $*$ denotes the corresponding transpose of the lower block part of symmetric matrices.

2. PROBLEM DESCRIPTION

In this work, the discrete-time LPV system as shown in Eq. 1 is taken into accounted.

$$\begin{aligned} x(k+1) &= A(p(k))x(k) + B(p(k))u(k), \\ y(k) &= Cx(k), \end{aligned} \tag{1}$$

where $x(k) \in R^{n_x}$ is the state of the plant and $u(k) \in R^{n_u}$ is the control input. The scheduling parameter $p(k)$ is assumed to be on-line measurable at each control iteration k. In addition, the system matrix $A(p(k))$ and the control matrix $B(p(k))$ are assumed to be within a polytope Ω,

$$\Omega = Co\{[A_1, B_1], [A_2, B_2], ..., [A_L, B_L]\}. \tag{2}$$

Co denotes convex hull. $[A_j, B_j]$ is the vertex of the convex hull. Any $[A(p(k)), B(p(k))]$ being inside the polytope Ω is a convex combination of all vertices such that

$$[A(p(k)), B(p(k))] = \sum_{j=1}^{L} p_j(k)[A_j, B_j], \tag{3}$$

$$\sum_{j=1}^{L} p_j(k) = 1, 0 \leq p_j(k) \leq 1. \tag{4}$$

The objective is to find a state feedback control law

$$u(k+i/k) = Kx(k+i/k), \tag{5}$$

that stabilises the LPV system and achieves the minimum worst case performance cost.

$$\min_{u(k+i/k)} \max_{[A,B]\in\Omega} \sum_{i=0}^{\infty} \begin{bmatrix} x(k+i/k) \\ u(k+i/k) \end{bmatrix}^T \begin{bmatrix} \Theta & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} x(k+i/k) \\ u(k+i/k) \end{bmatrix}, \tag{6}$$

$$\text{s.t.} |u_h(k+i/k)| \leq u_{h,\max}, h = 1, 2, ..., n_u, \tag{7}$$

$$|y_r(k+i/k)| \leq y_{r,\max}, r = 1, 2, ..., n_y. \tag{8}$$

3. THE PROPOSED ALGORITHM

In this section, the RMPC based on polyhedral invariant set with interpolation algorithms proposed are described. The on-line computational time is reduced by solving off-line the optimization problem shown in Eqs. 9-12 in order to find a sequence of state feedback gains $K_i, i = 1, 2, ..., N$ associated with a sequence of polyhedral invariant sets. An approach to construct the polyhedral invariant set proposed by (Pluymers et al., 2005) is adopted here. At each control iteration, when the measured state lies between two adjacent polyhedral invariant sets, the real-time state feedback gain is calculated by solving optimization problem based on linear interpolation between two pre-computed state feedback gains.

Off-line:

(1) Choose a sequence of states $x_i, i = 1, 2, ..., N$. For each x_i, solve the optimization problem in Eqs. 9-12 by replacing $x(k/k)$ with x_i in order to obtain the corresponding state feedback gain $K_i = Y_i G_i^{-1}$,

$$\min_{\gamma_i, Y_i, Q_i} \gamma_i \tag{9}$$

$$\text{s.t.} \begin{bmatrix} 1 & * \\ x_i & Q_i \end{bmatrix} \geq 0, \tag{10}$$

$$\begin{bmatrix} Q_i & * & * & * \\ A_j Q_i + B_j Y_i & Q_i & * & * \\ \Theta^{\frac{1}{2}} Q_i & 0 \gamma_i I & * & \\ R^{\frac{1}{2}} Y_i & 0 & 0 & \gamma_i I \end{bmatrix} \geq 0,$$

$$\forall j = 1, 2, ..., L, \tag{11}$$

$$\begin{bmatrix} X & * \\ Y_i^T & Q_i \end{bmatrix} \geq 0, X_{hh} \leq u_{h,\max}^2, h = 1, 2, ..., n_u. \tag{12}$$

x_i is chosen such that $Q_{i+1}^{-1} \subset Q_i^{-1}$. Moreover, for each $i \neq N$, the following inequality must be satisfied

$Q_i^{-1} - (A_j + B_j K_{i+1})^T Q_i^{-1} (A_j + B_j K_{i+1}) \geq 0, \forall j = 1, 2, ..., L$ to assure robust stability satisfaction of a convex combination between K_i and K_{i+1}. The state feedback gains are derived based on the minimization of upper bound of infinite horizon worst-case performance proposed by (Kothare et al., 1996). However, the output constraints are not taken into account here in order to enlarge the stabilizable region. The ouput constraints are then properly handled in the next step.

(2) Given the state feedback gains $K_i = Y_i Q_i^{-1}, i = 1, 2, ..., N$ previously calculated from step 1. For each K_i, the corresponding polyhedral invariant set $S_i = \{x | M_i x \leq d_i\}$ is constructed by following these steps:
  (a) Set M_i=$[C^T, -C^T, K_i^T, -K_i^T]^T$, d_i=$[y_{\max}^T, y_{\min}^T, u_{\max}^T, u_{\min}^T]^T$ and $m = 1$.
  (b) Select row m from (M_i, d_i) and check $\forall j$ whether $M_{i,m}(A_j + B_j K_i)x \leq d_{i,m}$ by solving the following problem 13:

$$\max_x W_{i,m,j} \tag{13}$$

$$\text{s.t.} W_{i,m,j} = M_{i,m}(A_j + B_j K_i)x - d_{i,m}, \tag{14}$$

$$M_i x \leq d_i. \tag{15}$$

  If $W_{i,m,j} \geq 0$, the constraint $M_{i,m}(A_j + B_j K_i)x \leq d_{i,m}$ is non-redundant with respect to (M_i, d_i), then, add non-redundant constraints to (M_i, d_i) by assigning $M_i = [M_i^T, (M_{i,m}(A_j + B_j K_i))^T]^T$ and $d_i = [d_i^T, d_{i,m}^T]^T$.
  (c) Let $m = m + 1$ and return to step (b). If m is strictly larger than the number of rows in (M_i, d_i), the algorithm is stopped.

On-line: The real-time state feedback gain is calculated by linear interpolation between the pre-computed state feedback gains. Three interpolation algorithms are proposed.

Algorithm 1: In the first algorithm, the pre-computed state feedback gains $K_i = 1, 2, ..., N$ are interpolated in order to get the largest possible state feedback gain while robust stability is still guaranteed. At each control iteration, when $x(k) \in S_i$ and $x(k) \notin S_{i+1}, \forall i \leq N - 1$, the real-time state feedback gain $K(k) = \lambda(k) K_i + (1 - \lambda(k)) K_{i+1}$ can be obtained by solving the problem in Eqs. 16-20.

$$\min_{\lambda(k)} \lambda(k) \tag{16}$$

$$\text{s.t.} M_i \sum_{j=1}^{L} p_j(k)(A_j + B_j K(k))x(k) - d_i \leq 0, \tag{17}$$

$$|K(k)x(k)_h| \leq u_{h,\max}, h = 1, 2, ..., n_u, \tag{18}$$

$$K(k) = \lambda(k) K_i + (1 - \lambda(k)) K_{i+1}, \tag{19}$$

$$0 \leq \lambda \leq 1. \tag{20}$$

If $x(k) \in S_N$, the real-time state feedback gain is K_N.

K_{i+1} is always larger than K_i because input and output constraints impose less limit on the state feedback gain as i increases. Thus, the largest possible state feedback gain can be obtained by minimizing $\lambda(k)$, while robust stability is still guaranteed by Eq. 17. The input constraint is guaranteed by Eq. 18. The output constraint does not need to be incorporated into the problem formulation because the satisfaction of Eq. 17 also guarantees output constraint satisfaction. The optimization problem involved is formulated as a linear programming and the number of constraints is independent of the number of vertices of the polytope Ω.

Algorithm 2: The real-time state feedback gain is obtained by minimizing the violation of the contraints $(\gamma(k))$ of the adjacent inner invariant sets, so the real-time state feedback gain calculated has to regulate the state from the current invariant set to the adjacent inner invariant set as fast as possible. At each control iteration, when $x(k) \in S_i$ and $x(k) \notin S_{i+1}, \forall i \leq N - 1$, the real-time state feedback gain $K(k) = \lambda(k) K_i + (1 - \lambda(k)) K_{i+1}$ can be obtained by solving the optimization problem in Eqs. 21-26.

$$\min_{\lambda(k), \gamma(k)} \gamma(k) \tag{21}$$

$$\text{s.t.} M_i \sum_{j=1}^{L} p_j(k)(A_j + B_j K(k))x(k) - d_i \leq 0, \tag{22}$$

$$M_{i+1} \sum_{j=1}^{L} p_j(k)(A_j + B_j K(k))x(k) - d_{i+1} \leq \gamma(k), \tag{23}$$

$$|K(k)x(k)_h| \leq u_{h,\max}, h = 1, 2, ..., n_u, \tag{24}$$

$$K(k) = \lambda(k) K_i + (1 - \lambda(k)) K_{i+1}, \tag{25}$$

$$0 \leq \lambda \leq 1. \tag{26}$$

If $x(k) \in S_N$, the real-time state feedback gain is K_N.

By minimizing $\gamma(k)$, the real-time state feedback gain calculated has to regulate the state from the current invariant set to the adjacent inner invariant set as fast as possible. Robust stability as well as output constraint satisfaction are guaranteed by Eq. 22. The input constraint is guaranteed by Eq. 24. The optimization problem involved is formulated as a linear programming and the number of constraints is independent of the number of vertices of the polytope Ω. However, the number of constraints involved is larger than that of algorithm 1.

Algorithm 3: In the last algorithm, the real-time state feedback gain is obtaned by minimizing the upper bound of infinite horizon worst case performance cost, which is estimated by Lyapunov function at current state, with subjected to a set of constraints associated with current invariant set. At each control iteration, when $x(k) \in S_i$ and $x(k) \notin S_{i+1}, \forall i \leq N - 1$, the real-time state feedback gain $K(k) = \lambda(k) K_i + (1 - \lambda(k)) K_{i+1}$ can be obtained by solving the optimization problem in Eqs. 27-32.

$$\min_{\lambda(k), \gamma(k)} \gamma(k) \tag{27}$$

$$\text{s.t.} M_i \sum_{j=1}^{L} p_j(k)(A_j + B_j K(k))x(k) - d_i \leq 0, \tag{28}$$

$$\begin{bmatrix} \gamma(k) & x(k)^T \\ x(k) & Q_i \end{bmatrix} \geq 0, \tag{29}$$

$$|K(k)x(k)_h| \leq u_{h,\max}, h = 1, 2, ..., n_u, \tag{30}$$

$$K(k) = \lambda(k) K_i + (1 - \lambda(k)) K_{i+1}, \tag{31}$$

$$0 \leq \lambda \leq 1. \tag{32}$$

If $x(k) \in S_N$, the real-time state feedback gain is K_N.

By minimizing $\gamma(k)$, the real-time state feedback gain calculated has to regulate the system by using the minimum infinite horizon worst case performance cost. Robust stability and output constraint satisfaction are guaranteed by Eq. 28. The input constraint is guaranteed by Eq. 30. The optimization problem involved is formulated as a convex optimization involving linear matrix inequalities (LMIs) and the number of constraints is independent of the number of vertices of the polytope Ω.

4. CASE STUDY

In this section, we present an example that illustrates the implementation of the proposed robust MPC algorithms. The numerical simulations have been performed in 2.3 GHz Intel Core i-5 with 16 GB RAM, using SDPT3(Tütüncü et al., 2003), Gurobi(Gurobi Optimization, 2012) and YALMIP (Löfberg, 2004) within Matlab R2011b environment. We will consider the application of our approach to the nonlinear two-tank system (Angeli et al., 2000), which is described by Eqs. 33-34.

$$\rho s_1 \dot{h}_1 = -\rho a_1 \sqrt{2gh_1} + u, \tag{33}$$

$$\rho s_2 \dot{h}_2 = \rho a_1 \sqrt{2gh_1} - \rho a_2 \sqrt{2gh_2}. \tag{34}$$

Where h_1 is the water level in tank 1, h_2 is the water level in tank 2 and u is the water flowrate. The operating parameters are shown in table 1.

Table 1. The parameters of two-tank system

Parameters	Value
s_1	2500 cm^2
s_2	1600 cm^2
a_1	9 cm^2
a_2	4 cm^2
g	980 cm/s^2
ρ	0.001 kg/cm^3
$h_{1,eq}$	14 cm
$h_{2,eq}$	70 cm

Let $\bar{h_1} = h_1 - h_{1,eq}$, $\bar{h_2} = h_2 - h_{2,eq}$ and $\bar{u} = u - u_{eq}$. Subscript eq denotes the corresponding variable at equilibrium condition. The objective is to regulate $\bar{h_2}$ to the origin by manipulating $\bar{u}$. The input constraint are symmetic $\bar{u} \le 1.5$kg/s. In contrast, asymmetric output constraints $-13 \le \bar{h_1} \le 71$, and $-69 \le \bar{h_2} \le 29$ are considered.

By evaluating the Jacobian matrix of Eqs. 33 and 34 along the vertices of the constraints set, the solutions of Eqs. 33 and 34 are also the solution of the following differential inclusion

$$\begin{bmatrix} \rho s_1 \dot{\bar{h}}_1 \\ \rho s_2 \dot{\bar{h}}_2 \end{bmatrix} \in \sum_{j=1}^{4} p_j A_j \begin{bmatrix} \bar{h}_1 \\ \bar{h}_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \bar{u}, \tag{35}$$

where A_j, $j = 1, ..., 4$ are given by

$$A_1 = \begin{bmatrix} -\rho a_1 \sqrt{\dfrac{2g}{h_{1,\min}}} & 0 \\ \rho a_1 \sqrt{\dfrac{2g}{h_{1,\min}}} & -\rho a_2 \sqrt{\dfrac{2g}{h_{2,\min}}} \end{bmatrix},$$

$$A_2 = \begin{bmatrix} -\rho a_1 \sqrt{\dfrac{2g}{h_{1,\max}}} & 0 \\ \rho a_1 \sqrt{\dfrac{2g}{h_{1,\max}}} & -\rho a_2 \sqrt{\dfrac{2g}{h_{2,\min}}} \end{bmatrix},$$

$$A_3 = \begin{bmatrix} -\rho a_1 \sqrt{\dfrac{2g}{h_{1,\min}}} & 0 \\ \rho a_1 \sqrt{\dfrac{2g}{h_{1,\min}}} & -\rho a_2 \sqrt{\dfrac{2g}{h_{2,\max}}} \end{bmatrix},$$

$$A_4 = \begin{bmatrix} -\rho a_1 \sqrt{\dfrac{2g}{h_{1,\max}}} & 0 \\ \rho a_1 \sqrt{\dfrac{2g}{h_{1,\max}}} & -\rho a_2 \sqrt{\dfrac{2g}{h_{2,\max}}} \end{bmatrix}, \tag{36}$$

and p_j, $j = 1, ..., 4$ are given by

$$p_1 = [\frac{\frac{1}{\sqrt{h_{1,\max}}} - \frac{1}{\sqrt{h_1}}}{\frac{1}{\sqrt{h_{1,\max}}} - \frac{1}{\sqrt{h_{1,\min}}}}][\frac{\frac{1}{\sqrt{h_{2,\max}}} - \frac{1}{\sqrt{h_2}}}{\frac{1}{\sqrt{h_{2,\max}}} - \frac{1}{\sqrt{h_{2,\min}}}}],$$

$$p_2 = [\frac{\frac{1}{\sqrt{h_1}} - \frac{1}{\sqrt{h_{1,\min}}}}{\frac{1}{\sqrt{h_{1,\max}}} - \frac{1}{\sqrt{h_{1,\min}}}}][\frac{\frac{1}{\sqrt{h_{2,\max}}} - \frac{1}{\sqrt{h_2}}}{\frac{1}{\sqrt{h_{2,\max}}} - \frac{1}{\sqrt{h_{2,\min}}}}],$$

$$p_3 = [\frac{\frac{1}{\sqrt{h_{1,\max}}} - \frac{1}{\sqrt{h_1}}}{\frac{1}{\sqrt{h_{1,\max}}} - \frac{1}{\sqrt{h_{1,\min}}}}][\frac{\frac{1}{\sqrt{h_2}} - \frac{1}{\sqrt{h_{2,\min}}}}{\frac{1}{\sqrt{h_{2,\max}}} - \frac{1}{\sqrt{h_{2,\min}}}}],$$

$$p_4 = [\frac{\frac{1}{\sqrt{h_1}} - \frac{1}{\sqrt{h_{1,\min}}}}{\frac{1}{\sqrt{h_{1,\max}}} - \frac{1}{\sqrt{h_{1,\min}}}}][\frac{\frac{1}{\sqrt{h_2}} - \frac{1}{\sqrt{h_{2,\min}}}}{\frac{1}{\sqrt{h_{2,\max}}} - \frac{1}{\sqrt{h_{2,\min}}}}]. \tag{37}$$

The discrete-time model is obtained by discretization of Eq.35 using Euler first-order approximation with a sampling period of 0.1 s and it is omitted here for brevity. The proposed algorithm will be compared with an off-line RMPC algorithm based on polyhedral invariant set without interpolation(Bumroongsri and Kheawhom, 2012b). The tuning parameters are $\Theta = [\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}]$ and $R = 0.01$.

A sequence of four polyhedral invariant sets is constructed. Figure 1 shows the polyhedral invariant sets constructed. As the output constraints considered in this case are not symmetric. It affects the constructed polyhedral invariant sets of S_1 and S_2. Thus, these two invariant sets are also asymmetric.

Figure 2 shows the regulated output ($\bar{h_2}$). The RMPC without interpolation gives the slowest response, because the real-time state feedback gain used is an approximation of optimal state feedback gain. For instance, if we start from an initial state $x(k) \in S_i$ but $x(k) \notin S_{i+1}$, a state feedback gain K_i is implemented. The system is driven to $x(k+1)$, where $|x(k+1)| < |x(k)|$. If $x(k+1) \in S_i$ but $x(k+1) \notin S_{i+1}$, K_i is still used as a state feedback gain. We see that $|u(k+1)| < |u(k)|$, as $|x(k+1)| < |x(k)|$. In

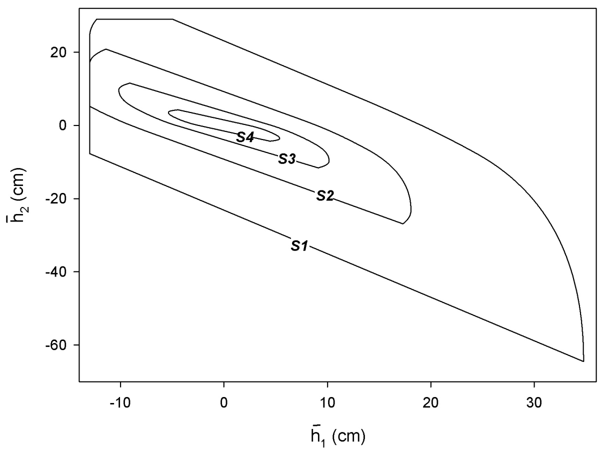

Fig. 1. The constructed polyhedral invariant sets.

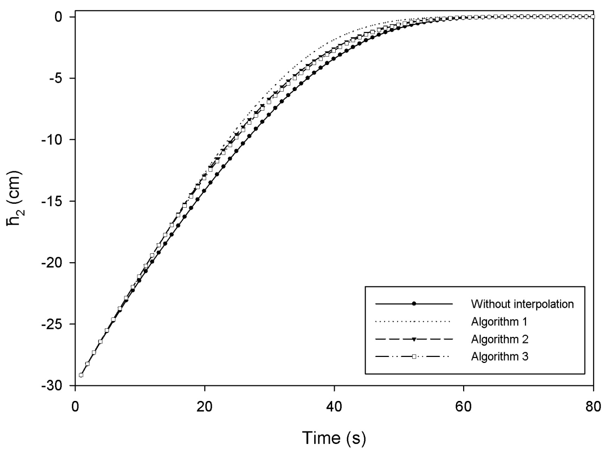

Fig. 2. Regulated output of the nonlinear two-tank system.

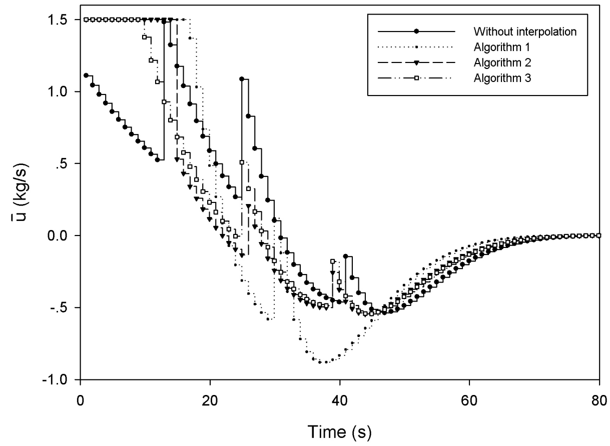

Fig. 3. Control input of the nonlinear two-tank system.

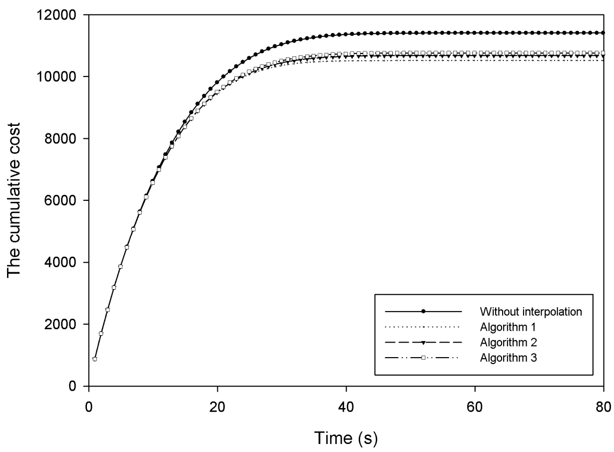

Fig. 4. The cumulative cost $\sum_{i=0}^{t} x(i)^T \Theta x(i) + u(i)^T R u(i)$.

other words, this algorithm implements the state feedback gain K_i for the whole region $x(k) \in S_i$ but $x(k) \notin S_{i+1}$.

By using interpolation, we can achieve better control performance. For each $x(k) \in S_i$ but $x(k) \notin S_{i+1}$, a state feedback gain $K(k)$ obtained by solving a simple optimization problem is implemented. We see that $K(k) \neq K_i$. Thus, a preferable control performance can be obtained.

Algorithm 1 yields the best control performance. In comparison, algorithms 2 and 3 give similar responses being slower than that of algorithm 1. In algorithm 1, the precomputed state feedback gains are interpolated to get the largest possible real-time state feedback gain, so algorithm 1 tends to produce fastest responses. In algorithm 2, the violation of the contraints of the adjacent inner invariant sets is minimized. Thus, a state feedback gain obtained from algorithm 2 leads to the shortest path to the inner adjacent invariant set. However, the shortest path to the inner adjacent invariant set does not guarantee the smallest worst case performance cost. Algorithm 3 minimizes the upper bound of infinite horizon worst case performance cost, which is estimated by Lyapunov function at current state. Unfortunately, Lyapunov function at each state is not determined on-line. Thus, Lyapunov function obtained off-line is used. That is for each $x(k) \in S_i$ but $x(k) \notin S_{i+1}$, Lyapunov function Q_i^{-1} is used for the whole region. Therefore, algorithm 3 becomes more conservative than algorithm 1.

Figure 3 shows the profiles of control input $\bar{u}$. The input discontinuities appeared in the response of the RMPC without interpolation are caused by the switching of feedback gains based on the distance between the state and the origin. In comparison, we can overcome this issue by using the interpolation algorithms proposed.

Figure 4 shows the cumulative performance cost. The cumulative performance costs of RMPC with interpolation are lower than the cumulative cost of the RMPC without interpolation. The lowest cumulative performance cost is obtained by using algorithm 1.

Table 2. The on-line computational burdens

Algorithm	On-line CPU time(s)/step
Without interpolation	< 0.0001
Algorithm 1	0.0001
Algorithm 2	0.0001
Algorithm 3	0.1800

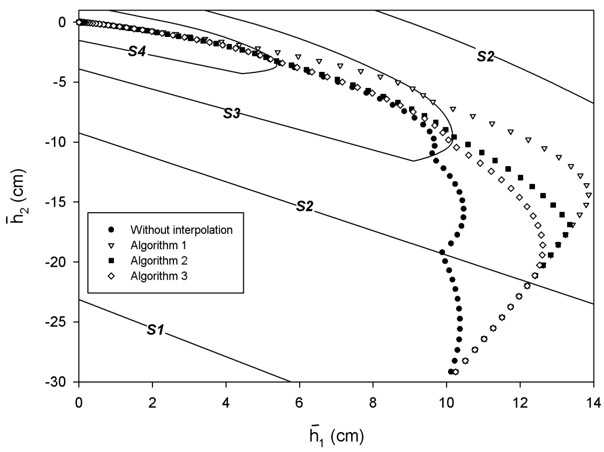

Fig. 5. State trajectories from initial condition of $(\bar{h_1}, \bar{h_2}) = (10, -30)$ to the origin.

Figure 5 shows state trajectories from initial condition of $(\bar{h_1}, \bar{h_2}) = (10, -30)$ to the origin. Algorithm 1 produces the trajectory with lowest control performance cost.

The on-line computational burdens are shown in table 2. For all algorithms, most of the computational burdens are moved off-line so the on-line computation is tractable. The optimization problem involved in each interpolation algorithm is independent of the number of vertices of the polytope Ω. Algorithms 1 and 2 use a linear programming. The number of constraints involved in algorithm 1 is lower than that of algorithm 2. In contrat, algorithm 3 uses a convex optimization involving LMIs. Thus, algorithm 3 requires higher computational time compared with other algorithms.

5. CONCLUSIONS

In this paper, we have presented an interpolation-based RMPC algorithms using polyhedral invariant sets for LPV systems. The proposed algorithms computes off-line a sequence of polyhedral invariant sets. The real-time control law is then calculated by interpolation between the two state feedback gains corresponding to two adjacent polyhedral invariant sets. Three interpolation algorithms are proposed. The controller design is illustrated with a case study of nonlinear two-tank system. The simulation results showed that the proposed RMPC with interpolation provides a better control performance while on-line computation is still tractable.

ACKNOWLEDGEMENTS

This work was supported by Rajadapisek Sompoj Fund, Chulalongkorn University Centenary Academic Development Project, and the Thailand Research Fund (TRF).

REFERENCES

Angeli, D., Casavola, A., and Mosca, E. (2000). Constrained predictive control of nonlinear plants via polytopic linear system embedding. *Int. J. Robust Nonlin.*, 10, 1091–1103.

Bumroongsri, P. and Kheawhom, S. (2012a). An ellipsoidal off-line model predictive control strategy for linear parameter varying systems with applications in chemical processes. *Syst. Control Lett.*, 61, 435–442.

Bumroongsri, P. and Kheawhom, S. (2012b). An off-line robust mpc algorithm for uncertain polytopic discrete-time systems using polyhedral invariant sets. *J. Process Contr.*, 22, 975–983.

Bumroongsri, P. and Kheawhom, S. (2013). An interpolation-based robust mpc algorithm using polyhedral invariant sets. In *Proceedings of the 2013 European Control Conference (ECC)*, 3161–3166.

Gurobi Optimization, I. (2012). Gurobi optimizer reference manual. URL `http://www.gurobi.com`.

Kheawhom, S. and Bumroongsri, P. (2013). Constrained robust model predictive control based on polyhedral invariant sets by off-line optimization. *Chemical Engineering Transactions*, 32, 1417–1442. doi: 10.3303/CET1332237.

Kothare, M., Balakrishnan, V., and Morari, M. (1996). Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32, 1361–1379.

Löfberg, J. (2004). Yalmip: A toolbox for modeling and optimization in matlab. In *Proc. IEEE international symposium on computer aided control systems design*, 284–289.

Lu, Y. and Arkun, Y. (2000). Quasi-min-max mpc algorithms for lpv systems. *Automatica*, 36, 527–540.

Morari, M. and Lee, J. (1999). Model predictive control: past, present and future. *Comput. Chem. Eng.*, 23, 667–682.

Paijmans, B., Symens, W., Brussel, H., and Swevers, J. (2008). Experimental identification of affine lpv models for mechatronic systems with one varying parameter. *Eur. J. Contr.*, 14, 16–29.

Pluymers, P., Rossiter, J., Suykens, J., and Moore, B. (2005). The efficient computation of polyhedral invariant sets for linear systems with polytopic uncertainty. In *Proc. of the American Control Conference*, 804–809.

Tütüncü, R., Toh, K., and Todd, M. (2003). Solving semidefinite-quadratic-linear programs using sdpt3. *Math. Prog. Series B*, 95, 189–217.

Wada, N., Saito, K., and Saeki, M. (2006). Model predictive control for linear parameter varying systems using parameter dependent lyapunov function. *IEEE T. Circuits Syst.*, 53, 1446–1450.