ROBUST MODEL PREDICTIVE CONTROL FOR INPUT SATURATED AND SOFTENED STATE CONSTRAINTS

Vu Trieu Minh and Nitin Afzulpurkar

ABSTRACT

This paper starts with a brief review of robust model predictive control (RMPC) algorithms for uncertain systems using linear matrix inequalities (LMIs) subject to input and/or output saturated constraints. However when RMPC has both input and state constraints, a difficulty will arise due to the inability of the optimizer to satisfy the state constraints due to the constraints on inputs. Therefore, a novel RMPC scheme is presented that softens the state constraints as penalty terms are added to its objective function. These terms maintain state violation at low values until a constrained solution is returned. The state violation can be regulated by changing the value of the weighting factor. A novel robust predictive controller for input saturated and softened state constraints for linear time varying (LTV) systems with polytopic model uncertainties is presented.

KeyWords: Model predictive control, robust model predictive control, min-max control law, uncertain systems, soften state constraints.

I. INTRODUCTION

Model predictive control (MPC) is one of advanced control theories that has been studied extensively by the research community. MPC works out the optimal open-loop manipulated input trajectory that minimizes the difference between the predicted plant behavior and the desired plant behavior. It differs from other control theories in that the optimal control problem is solved on-line for the current state of the plant, rather than off-line as a feedback policy. MPC has been widely applied in the petrochemical and related industries because of its ability to handle input and output constraints in the optimal control problem. However, MPC is not designed to explicitly handle plant-model uncertainty, and such a system might perform very poorly when implemented on a physical system which is not exactly described by the model [1].

In this paper, we study robust model predictive control (RMPC) that guarantees stability in the presence of model uncertainty. RMPC is a class of predictive control schemes that counts for the modeling errors in the controller design. Instead of forecasting the system behavior by using one process model as in MPC, RMPC forecasts the system behavior for all possible models in the uncertainty set. The optimal actions are determined through min-max optimization, which minimizes the deviation of the forecasted behavior from the desired behavior for the model with the largest deviation.

Modeling of the uncertainty has been studied by a number of researchers. The two common types of uncertainties are parametric uncertainty and bounded disturbance regions. Magni et al. [2] and Rami and Zhou [3] proposed using multiplicative input and state uncertainties. Researchers have proved robust stability for MPC by using the closed loop stability criteria, i.e., terminal cost and terminal stability constraints. The simplest way to enforce stability with a finite prediction horizon is to add a so-called zero-terminal equality at the end of the prediction horizon [4], i.e., to add the zero constraint for the terminal prediction state: $x(K+N) = 0$. Mayne and Schorerder [5], and Scokaert and Mayne [6] proposed a min-max control law that steered the state into an invariant region in which the state feedback law guaranteed convergence to the origin for
all states. However the main disadvantage of min-max control theories is the computational complexity. These optimization problems are expensive to solve on-line.

Recent developments in the theory and application of optimization involving linear matrix inequalities (LMIs) to MPC have resulted in a new class of methodologies for MPC because it is now possible to recast much of existing robust control theory in the framework of LMIs and because optimization problems can now be solved and implemented on-line [7,8].

Kothare et al. [9] proposed using an LMI-based optimizer to find robustly stabilizing state feedback gains that minimize the worst performance objective for a given model uncertainty description subject to input and state constraints. Casavola et al. [10] extended this concept to uncertain systems with input saturation constraints. Robust stability is guaranteed if there is a common Lyapunov function for all the models and feedback gains.

Our paper is organized as follows: Section 2 gives a description of the uncertain system used by the authors. Section 3 reviews the RMPC tree trajectory and regulator. Section 4 introduces an infinite horizon RMPC algorithm using LMIs. Then in section 5, the infinite horizon RMPC algorithm is extended by including a finite free terms prediction. In section 6 our novel RMPC algorithm with softened state constraints is presented. Finally, in section 7, conclusions are drawn and some directions of future research are discussed.

II. DESCRIPTION OF AN UNCERTAIN SYSTEM

In this RMPC study, instead of relying on parameterization of the process, our uncertain system is described by the following discrete state-space model, which is similar to the multiple models described by Kothare et al. [9]:

\[ x(k + 1) = A_i(k) x(k) + B_i(k) u(k) , \]
\[ y(k) = C x(k) , \]  

(1)

in which \( A_i(k) \) and \( B_i(k) \) are the time-varying state-space model matrices, \( i = 1, 2, \ldots, L \); \( C \in \mathbb{R}^{p,n} \) describes the relationship between the output and the state without any uncertainty; \( x(k) \in \mathbb{R}^{n} \), \( u(k) \in \mathbb{R}^{m} \) and \( y(k) \in \mathbb{R}^{p} \) are the state, input and output vectors, respectively. The state, input and output are subject to the following constraints:

\[ x(k) \in \mathcal{X} \text{, } u(k) \in \mathcal{U} \text{ and } y(k) \in \mathcal{Y} , \]

(2)

in which \( \mathcal{X}, \mathcal{U} \) and \( \mathcal{Y} \) are convex and closed subsets of \( \mathbb{R}^{n} \), \( \mathbb{R}^{m} \) and \( \mathbb{R}^{p} \) respectively.

When model uncertainty is present, the exact plant model \([A(k) B(k)]\) is unknown. The model uncertainty region is described by \( \Omega: [A(k) B(k)] \in \Omega = \text{Co} \{ [A_1 B_1], \ldots, [A_L B_L] \} \), a convex hull of \([A(k) B(k)] = \sum_{i=1}^{L} \alpha_i(k) [A_i B_i] \), where \( \alpha_i(k) \geq 0 \) and \( \sum_{i=1}^{L} \alpha_i(k) = 1 \).

III. RMPC TREE TRAJECTORY AND REGULATOR

The regulator determines the optimal manipulated input trajectory based on the system behavior predictions of the models. We use a scheme similar to the method introduced by Seokaert and Mayne [6] that produces a tree trajectory for all possible time-varying combinations of the models. Figure 1 illustrates all possible state trajectories of all models with \( L = 2 \) over the prediction horizon length, \( N = 3 \). \( X_{k+i}^h \) and \( U_{k+i}^h \) denote the possible predicted state and input vectors \( h \) at time \( k + i, i = 0, 1, \ldots, N \).

The total number of nodes \( \eta \) depends on the total number of possible state trajectories of all models \( L \) and the prediction horizon length,

\[ N; \eta = \begin{cases} N+1 & \text{if } L = 1 \\ \frac{L^{N+1}-1}{L-1} & \text{if } L > 1 \end{cases} \]

The total number of branches \( \beta \) is dependent on the number of models \( L \) and the prediction horizon length, \( N; \beta = L^N \).

The objective of the control problem is to find the control actions that, once implemented, cause all branches in the tree trajectory to converge to a robust control invariant set in which a state feedback law guarantees convergence to the origin for all states.
IV. INFINITE HORIZON RMPC USING LMIS

In nominal infinite horizon MPC, the regulator computes the optimal input that minimizes the objective function in quadratic form at each sampling time $k$:

$$J_u(k) = \sum_{i=0}^{\infty} \left[ x(k+i|k) Q x(k+i|k) + u(k+i|k) R u(k+i|k) \right].$$

It does so by means of a feedback control law $u(k+i|k) = K(k) x(k+i|k)$, in which $Q$ and $R$ are symmetric weighting matrices, where $Q > 0$ and $R > 0$. $x(k+i|k)$ is the state of the system at time $k+i$ predicted at time $k$, $u(k+i|k)$ is the state of the system measured at time $k$, and $u(k+i|k)$ is the control move at time $k+i$, computed at time $k$, and $u(k)$ is the control move implemented at time $k$. Kothare et al. [9] proposed using an LMI-based optimizer to find robustly stabilizing state feedback gains that minimize the worst performance objective for the given model uncertainty description subject to input and state constraints. The optimizer is described below.

Suppose there exists a Lyapunov function $\Phi$ with $\Phi(0) = 0$ and $\Phi(x(k)) = x(k)' P_k x(k) \geq J_u(k)$ with $P_k > 0$. For the objective function to be finite, $x(\infty|k) = 0$, and hence, $\Phi(x(\infty|k)) = 0$, and the system will be stable if the Lyapunov function is decreasing, that is, $\Phi(x(k+1)) - \Phi(x(k)) \leq 0$. In addition, suppose

$$\Phi(x(k+i+1|k)) - \Phi(x(k+i|k)) = x(k+i+1|k)' P_k x(k+i+1|k) - x(k+i|k)' P_k x(k+i|k) - u(k+i|k)' R u(k+i|k) \leq -x(k+i|k)' Q x(k+i|k) - u(k+i|k)' R u(k+i|k)$$

for all $k$, $i = 0, 1, \ldots, \infty$.

For robust unconstrained infinite horizon MPC, we can find the state feedback gain $K$ in the control law for all model uncertainties $u_i(k) = K x_i(k)$ and $x_i(k+1) = A_i x_i(k) + B_i u_i(k)$ with $i = \{1, 2, \ldots, L\}$, which guarantees decreasing robust performance. Then,

$$P_k = (A_i + B_i K)' P_k (A_i + B_i K) - Q - K' R K \geq 0.$$

Set $K = Y P^{-1}$ and $P_k = \gamma P^{-1}$ (scalar $\gamma > 0$, $P > 0$); the above equation can be transformed as

$$-P_k (A_i + B_i Y)' P_k (A_i + B_i Y) - (Q^{1/2} P)' \gamma^{-1} (Q^{1/2} P) - (R^{1/2} Y)' \gamma^{-1} (R^{1/2} Y) \geq 0.$$  \hspace{1cm} (3)

Equation (3) is equivalent to an LMI (using the Schur complement):

$$\begin{bmatrix}
P & PA_i' + Y' B_i' & PQ^{1/2} & Y' R^{1/2} \\
A_i P + B_i Y & P & 0 & 0 \\
Q^{1/2} P & 0 & \gamma I & 0 \\
R^{1/2} Y & 0 & 0 & \gamma I
\end{bmatrix} \geq 0$$

$$i = \{1, 2, \ldots, L\}.$$  \hspace{1cm} (4)

For robust constrained infinite horizon MPC, we incorporate both input and output constraints into the optimization problem. Then, the receding horizon state feedback gain $K$, which at the sampling time $k$ minimizes the upper bound $\Phi(x(k|k))$ on $J_u(k)$ and satisfies the specified input and output constraints, is given by $K = Y P^{-1}$, where $P > 0$ and $Y$ are the solutions to the following LMIs:

$$\min_{P > 0, \gamma > 0} \gamma \text{ subject to Eq. (4) and}$$

$$\begin{bmatrix}
1 & x(k|k)' \\
x(k|k) & P
\end{bmatrix} \geq 0,$$

$$\begin{bmatrix}
X & Y \\
Y' & P
\end{bmatrix} \geq 0 \text{ with min } X_{ii} \leq u_{\max}^{2} \text{ for input constraints}$$

$$u(k) \in \mathbb{R}^{m} \leq u_{\max}, \forall k = 1, 2, \ldots, \infty, i \in \{1, 2, \ldots, n\},$$

where $X_{ii}$ denotes the $i^{th}$ entry of matrix $X = X^* \geq 0$ and/or

$$\begin{bmatrix}
X & CA_i P + CB_i Y \\
PA_i C' + Y' B_i' C' & P
\end{bmatrix} \geq 0$$

with min $X_{jj} \leq y_{\max}^{2}$ for output constraints $y(k) \in \mathbb{R}^{n} \leq y_{\max}, \forall k = 1, 2, \ldots, \infty, j \in \{1, 2, \ldots, m\}, X_{jj}$ denotes the $j^{th}$ entry of matrix $X = X^* \geq 0$.  \hspace{1cm} (8)

At each time instant and based on the current state $x(k)$, this algorithm generates a feasible state feedback $K(k)$.

V. FINITE HORIZON PREDICTION OF RMPC WITH INPUT-SATURATED CONSTRAINTS

Based on the above infinite horizon RMPC scheme, Casavola et al. [10] formulated a finite receding horizon control sequence with $N$ free control moves and the current state with the following structure:

$$u(.|k) = \begin{cases}
\bar{u}(k+i|k) & \text{for } i = 0, 1, \ldots, N-1, \\
u(k+i|k) = K(k)x(k+i|k) & \text{for } i \geq N.
\end{cases}$$

\hspace{1cm} (9)
In Eq. (9), \( u^*(k+i|k) \) corresponds to \( N \) initial free moves, while \( K(k) \) and suitable \( P_i(k) \) and \( \gamma(k) \) are assumed to temporarily exist and jointly satisfy the LMIs of infinite horizon MPC with \( x = x(k) \). Then the state-feedback gain \( K(k) \) is used to ideally define the input sequence after the first \( N \) moves, which are selected by minimizing the following quadratic function:

\[
\begin{align*}
\min_{P_i(k)\in\mathcal{P}_i(k)} & \Phi(x(k), P_i(k), u^*(.| k)) \\
= & \max_{z(\in\{z(i)\in K_i^N(k)\})} z^TP_i(k)z \\
& + \sum_{i=0}^{N-1} \max_{(i)\in\{z(i)\in K_i^N(k)\}} (i)^TQ(i)z(i) \\
& + u^*(k+i|k)'Ru^*(k+i|k)
\end{align*}
\]

subject to the constraints that \( u^*(k+i|k) \in \Omega_z \), \( \forall i \in \{0, 1, ..., N-1\} \) and \( \text{vert} \{x(k+i|k)\} \subset \zeta(P_i(k), \gamma(k)) \), where \( \text{vert} \{.\} \) denotes vertices and \( \zeta(.) \) is the ellipsoidal set

\[
\zeta(P_i(k), \gamma(k)) = \left\{ x(k+i|k) \in \mathbb{R}^n : x(k+i|k)P_i(k)x(k+i|k) \leq \gamma(k) \right\}.
\]

The set \( x(0|k) \subset \mathbb{R}^n \) denotes the convex hull of all \( i \)-step ahead state predictions from \( x(k) \) at time \( k \) through application of the input sequence \( u^*(k+i|k) \) \( i=0 \). A finite RMPC algorithm to select \( u^*(k+i|k) \) subject to input saturation can be set up as follows.

- **Initialization:**
  0.1 Given \( x(0) \), find \( [P(0), Y(0), \gamma(0), u^0(0|0)] = \arg \min_{p(0), y(0), \gamma(0), u^0(0|0)} \gamma \), subject to
    
    (i) Eq. (4),
    (ii) Eq. (7),
    (iii) \( \begin{bmatrix} 1 & z' \end{bmatrix} P \begin{bmatrix} z \end{bmatrix} \geq 0 \), \( \forall z \in \text{vert} \left\{ x(0|0) \right\} \) and
    (iv) \( u^*(i|0) \in \Omega_z \), \( \forall i \in \{0, 1, ..., N-1\} \).

- **Generic step:**
  1. For any \( k \geq 0 \) given \( x(k), P(k), \) and \( \gamma(k) \), find
    
    \[
    u^0(\cdot | k) = \arg \min_{J_i, u^0(\cdot | k)} \sum_{i=0}^{N-1} J_i, \text{ subject to }
    \]
    
    (i) \( \begin{bmatrix} J_i & z' \end{bmatrix} \geq 0 \), where \( J_i \) is a variable for each \( \forall z \in \text{vert} \left\{ x(0|k) \right\} \),

    (ii) \( u^*(k+i|k) \in \Omega_z \), \( \forall i \in \{0, 1, ..., N-1\} \), and

    (iii) \( \begin{bmatrix} 1 & z'(i)Q^{1/2} & u^*(k+i|k)R^{1/2} \end{bmatrix} \geq 0 \),

    (iv) Eq. (7),

    (vii) \( \begin{bmatrix} 1 & z'(A_i+B_i K(k))' \end{bmatrix} \geq 0 \),

    where \( i \in \{0, 1, ..., L\} \).

- **Feed the plant by means of the input \( u(k) = u^0(k|k) \).**

- **2.** For any \( k \geq 0 \), given \( x(k), K(k) \) and \( u^0(\cdot | k) \), find

    \[
    [P(k+1), Y(k+1), \gamma(k+1)] = \arg \min_{P>0, Y>0, \gamma>0} \gamma, \text{ subject to }
    \]

    (iv) Eq. (4),

    (v) Eq. (7) and

    (vii) \( \begin{bmatrix} 1 & z'(A_i+B_i K(k))' \end{bmatrix} \geq 0 \),

    where \( i \in \{0, 1, ..., L\} \).

4. \( k \leftarrow k+1 \) and go to step 1.

We will demonstrate this RMPC algorithm with an example. Then we will present a modified algorithm for RMPC in section 6.

**Example 1.** We considered the following set of uncertain models \( \Pi = \{(A_i, B_i), (A_j, B_j)\} \) with

- **model 1:** \( x(k+1) = \begin{bmatrix} 0.90 & -0.80 \\ 0.30 & 0.80 \end{bmatrix} x(k) + \begin{bmatrix} 0.20 \\ 0.20 \end{bmatrix} u(k) \); and
- **model 2:** \( x(k+1) = \begin{bmatrix} 0.85 & -0.75 \\ 0.35 & 0.75 \end{bmatrix} x(k) + \begin{bmatrix} 0.10 \\ 0.10 \end{bmatrix} u(k) \).

A simulation was conducted with weighting matrices \( Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \) and \( R = [1] \). We assumed that the input constraint was \( |u| \leq u_{\text{max}} = 1 \); that the real model \( [A(k) B(k)] = \sum_{i=1}^{2} \alpha_i(k) [A_i B_i] \) had \( \alpha_1 = \alpha_2 = 0.5 \); and that the initial state was \( x_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \).

Figure 2 shows the input and state of the RMPC performance with a receding horizon forecast with \( N = 0, 1, \) and 2.
Fig. 2. RMPC performance for model uncertainty with $N = \{0, 1, 2\}$.

Results similar to those by Casavola et al. [10] are obtained, but a significant reduction in conservativeness is achieved by simply letting one free control move be added as the first input commands are closer to the constraint boundary ($-1$). We obtained $u_0 = -0.8841$, $-0.9622$ and $-0.9982$ for $N = 0, 1$ and $2$, respectively.

VI. RMPC SCHEME WITH SOFTENED STATE CONSTRAINTS

The state response results shown in Fig. 2 motivate us to examine the minimum values of state $x_1$ since all of them are less than minus one ($-1$). If we impose a hard state constraint on $|x_1| \leq 1$ or in the general form

$$\begin{bmatrix} 1 & z_i' \\ z_i & X + \mu \epsilon, I \end{bmatrix} \geq 0,$$

with $\min_j X_{ji} \leq x_{\text{max}}^2$ for hard state saturation, that is,

$$\forall z_i \in \text{vert} \{\chi_{x_1(k)}(x(k))\} \text{ and } \forall i \in \{1, ..., N\},$$

then this RMPC algorithm will become infeasible due to the constraints on the input, which will lead to state constraint violations.

In reality, the processes might have both input and state constraints, so a difficulty will arise due to the inability to satisfy the state constraints. Since the RMPC regulator is designed for on-line implementation, any infeasible solution of the optimization problems cannot be tolerated. Input constraints are based on the physical limits of the process and can be considered as hard constraints. If the state constraints are not strictly imposed and can be violated somewhat during evolution of the system, they can be considered as softened constraints. To guarantee system stability once the state space violate constraints, we have developed an RMPC algorithm with softened state constraints in LMI form:

$$\begin{bmatrix} 1 & z_i' \\ z_i & X + \mu \epsilon, I \end{bmatrix} \geq 0,$$

$$\min_j X_{ji} \leq x_{\text{max}}^2$$

$$\forall z_i \in \text{vert} \{\chi_{x_1(k)}(x(k))\}, \forall i \in \{1, ..., N\}$$

(17)

where $\mu$ is a weighting factor ($\mu > 0$, usually a small value) and $\epsilon_i$ represents state penalty terms ($\epsilon_i \geq 0$ added to the RMPC objective function. From Eq. (17), $\epsilon_i$ depends on the current state $x(k)$, application of the input sequence $\{u^*(k+i|k)\}_{i=0}^{N-1}$ and the weighting factor $\mu$. These terms represented by $\epsilon_i$ keep the state violations at low values until the constrained solution is returned. Hence, a novel RMPC algorithm for input saturation and softened state constraints for selecting $u^*(k+i|k)$ can be set up as shown below.

- **Initialization:**

0.1 Given $x(0)$, find $[K(0), P(0), \gamma(0)] = \arg \min_{x(0), P(0), \gamma(0)} \sum_{i=1}^{N} x_i$, subject to

(0.i) Eq. (4),

(0.ii) Eq. (7),

(0.iii) Eq. (17),

(0.iv) Eq. (11) and

(0.v) Eq. (12).

- **Generic step:**

1. For any $k \geq 0$, given $x(k)$, $P(k)$ and $\gamma(k)$, find $[u^0(.|k)] = \arg \min_{J, \gamma > 0} \sum_{i=1}^{N} x_i$, subject to

(i) Eq. (13),

(ii) Eq. (14),

(iii) Eq. (17) and

(iv) Eq. (15).

2. Feed the plant by means of the input $u(k) = u^0(k|k)$

3. For any $k \geq 0$, given $x(k)$, $K(k)$ and $u^0(.|k)$, find

$$[P(k+1), Y(k+1), \gamma(k+1)] = \arg \min_{P > 0, \gamma > 0} \gamma$$

subject to

(v) Eq. (4),

(vi) Eq. (7) and

(vii) Eq. (16).

4. $k \leftarrow k + 1$ and go to step 1.

We present the following example in which the states are subject to softened constraints so as to highlight the advantages of the novel RMPC algorithm.

**Example 2.** We considered the same uncertain models in the example 1 and simulated the softened state RMPC scheme with the addition of softened state constraints $|x_i| \leq ...
1. Figure 3 compares RMPC with only the hard input constraint and RMPC with both the hard input and softened state constraints for the same receding horizon forecast, \(N = 1\). The weighting factor was set at \(\mu = 1/10\).

Similar reduction in conservativeness was achieved by adding softened state penalty terms to the objective function. The input was closer to the constraint boundary. Hence, its minimum state value was also closer to the state constraint boundary (−1.1058 compared to −1.1525, respectively).

Once the state constraints are violated, the RMPC algorithm in Eq. (17) will calculate the softened state penalty terms \(\varepsilon_i\) based on the current state \(x(k)\) and the weighting factor \(\mu\). For a receding horizon forecast \(N = 2\) and an uncertain model \(L = 2\), with \(\mu = 1\), the maximum values of the state penalty terms can be calculated for the total state trajectories \(X^{1+1}_{k+1}, X^{1+2}_{k+2}, X^{2+1}_{k+1}\) and \(X^{2+2}_{k+2}\) (see Fig. 1), leading to \(\varepsilon_i = [0.48 \ 0.38 \ 0.52 \ 0.37 \ 0.45 \ 0.26]\), respectively. When we decrease the weighting values \(\mu\), the values of the state penalty terms increase. For \(\mu = 1/100\), we get maximum values of \(\varepsilon_i = [33.82 \ 29.16 \ 41.40 \ 28.21 \ 37.79 \ 21.31]\). Softened state penalty terms help RMPC to find a solution when the state values are violated constraints and to keep the state violations at low values until a constrained solution is returned. When the state constraints are satisfied, the RMPC algorithm in Eq. (17) will set all softened state penalty terms, \(\varepsilon_i = 0\).

Finally in Fig. 4, we show the results of controlling the state violation by changing the value of \(\mu\). For the values \(\mu = 1, 1/10, \text{ and } 1/100\), we obtained minimum values of \(x_1 = -1.1773, -1.1058, \text{ and } -1.0301\), respectively. By decreasing the weighting values \(\mu\), we could significantly reduce the state constraint violations (the minimum state values are closer to the state constraint boundary).

However, when we decrease the values of \(\mu\) further, as shown by this example with \(\mu < 1/100\), we cannot further reduce the state constraint violations since we have reached a control “limit.” If we assign a weighting value \(\mu\) of only \(\mu = 1/10000\), we will have large maximum values of the state penalty terms, \(\varepsilon_i = [3398.3 \ 2919.3 \ 4161.0 \ 2814.3 \ 3781.6 \ 2136.3]\) be compensated in Eq. (17) so that the minimum state value of \(x_1\) is still unchanged at −1.03.

**VII. CONCLUSION**

In this paper, an RMPC algorithm with softened state constraints has been presented. It is assumed that all the states of the MPC models are directly measured, and that there are no disturbances. This novel RMPC scheme is able to guarantee stability when the initial conditions lead to violations of the state constraints, a situation in which other hard constrained RMPC schemes become infeasible. For the same prediction horizon length, softened state constrained RMPC becomes less conservative than RMPC with only hard input constraints. Another advantage of our novel RMPC is its ability to regulate the state violation by changing the weighting factor \(\mu\).

However, additional analysis is needed to determine the effectiveness of our novel scheme with respect to the achievable performance and the corresponding elapsed CPU time because of additional penalty terms added to the objective function. For a long prediction length \(N\) or a large system, on-line min-max optimization becomes more computationally expensive. More simulations and experimental applications are also needed to verify the effectiveness of this RMPC algorithm approach.
REFERENCES


