IFAC Journal of Systems and Control 12 (2020) 100084

Contents lists available at ScienceDirect

IFAC Journal of Systems and Control

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

Interpolation based predictive control by ellipsoidal invariant sets

Daniel Rubin[a,*], Pedro Mercader[b], Per-Olof Gutman[b], Hoai-Nam Nguyen[c], Alberto Bemporad[d]

[a] Technion Autonomous Systems Program (TASP), Technion – Israel Institute of Technology, Haifa, Israel
[b] Faculty of Civil and Environmental Engineering, Technion – Israel Institute of Technology, Haifa, Israel
[c] Control, Signal and System Department, IFP Energies Nouvelles, Solaize, France
[d] IMT School for Advanced Studies Lucca, Lucca, Italy

ARTICLE INFO

Article history:
Received 28 February 2019
Received in revised form 13 March 2020
Accepted 24 March 2020
Available online 9 April 2020

Keywords:
Predictive control
Constrained control
Uncertain systems
Linear time variant systems

ABSTRACT

A new interpolating control scheme for uncertain linear systems is developed. The interpolation is done between two saturated control laws, each given with its associated invariant set. A linear search is performed to compute a sub-optimal interpolation coefficient. Two algorithms are suggested — each incorporating a different optimization objective. The method guarantees robust stability and recursive feasibility, also in presence of persistent disturbances.

1. Introduction

The field of control of constrained discrete-time linear systems have attracted considerable attention in the control community. There are several control approaches able to deal with such systems, e.g., Model Predictive Control (MPC) (Borrelli, Bemporad, & Morari, 2017; Maciejowski, 2002), Vertex Control (Gutman & Cwikel, 1986) and Interpolation Control (Nguyen, 2014). Currently, the most popular approach is MPC that has become one of the few advanced control techniques employed in industry. The main potential of MPC is that it offers a systematic approach to control multivariate constrained systems, however it requires solving an optimization problem at each time step. This is an important limitation for systems with fast dynamics, but it may be circumvented with techniques like explicit MPC methods where an optimal control law at each state–space region is precomputed off-line (Bemporad, Morari, Dua, & Pistikopoulos, 2002), specialized optimization tools (Wang & Boyd, 2010), and tailored hardware (Jerez et al., 2014). Even though, for high dimensional systems, the computational load becomes impractically excessive. Another limitation of MPC is its feasibility and stability analysis, which relies on set invariance theory.

The aforementioned limitations of MPC becomes a major challenges when addressing the problem of uncertain systems. There exist robust MPC methods, such as the linear matrix inequalities (LMIs) based MPC in Kothare, Balakrishnan, and Morari (1996), the tube-based MPC in Langson, Chryssochoos, Raković, and Mayne (2004), and the min–max MPC approach in Bemporad, Borrelli, and Morari (2003). These may demand the on-line solution of large optimization problems, for the first two methods, or solving a point location problem in a large polyhedral partition, for the last mentioned method. Robust MPC methods thus are plagued by an excessive computational load, and therefore are impractical for implementation. In addition, feasibility and stability analysis of robust MPC designs are practically impossible to perform.

Numerous works are aimed at mitigating the computational load. Some works have succeeded to reduce the computational burden to some extent: see, e.g. Kouvaritakis, Rossiter, and Schuurmans (2000) with a more efficient LMI-MPC algorithm, and Kouvaritakis, Cannon, and Rossiter (2002) which avoids LMIs completely. Interpolation based control techniques were also suggested as robust MPC alternatives. By interpolation, simpler control algorithms may be used for the same class of problems addressed by MPC. Although optimality is only guaranteed in a local sense, these algorithms provide a good compromise between computational load, size of feasible region, and performance, and extends well to the uncertain case. A variety of interpolation based methods, closely related to MPC, are presented by Rossiter, Pluymers and co-workers in e.g. Pluymers, Rossiter, Suykens, and Moor (2005) and Rossiter and Ding (2010), which are recommended as background papers on robust invariant sets and their use in efficient robust MPC of albeit high dimensionality. The framework in Pluymers et al. (2005) and Rossiter and Ding (2010) facilitates the interpolation between two (or more) linear

* Corresponding author.
E-mail address: rubdaniel@campus.technion.ac.il (D. Rubin).

https://doi.org/10.1016/j.ifacsc.2020.100084

state-feedback controllers by solving an online optimization — typically a Quadratic Program (QP), with constraints given by the *polyhedral* invariant set associated with each control law.

An alternative framework based on an interpolation concept, known as Interpolating Control (IC), was recently developed (Nguyen, 2014; Nguyen, Gutman, Olaru, & Hovd, 2013). It interpolates between one (or more) local linear controllers and one global piece-wise-affine contractive control law, generated by e.g. Vertex Control (Gutman & Cwikel, 1986), or Minkowsky Functional Minimization Control (Nguyen & Gutman, 2018). The online implementation demands the solution of a Linear Program (LP).

Both the interpolation control techniques in Rossiter and Ding (2010) and IC demand explicit knowledge of the polyhedral invariant sets associated with the linear control laws. Unfortunately, the computation of such sets can become very computationally demanding for polytopic uncertain plants, rendering some problem impossible for solutions on modern personal computers (Rubin, Nguyen, & Gutman, 2018a, 2018b). Ellipsoidal invariant sets, on the other hand, are much easier to compute and store. However, they make the online solutions significantly more complex. The optimal solution for the resulting interpolating problem was developed in Nguyen, Olaru, Gutman, and Hovd (2011), and requires the online solution of an LMI problem at each time step. The sub-optimal QP based solutions in Rossiter and Ding (2010) would require the solution of quadratically-constrained QPs.

It is of-course useful to reduce online computation requirements to the minimum possible. A sub-optimal interpolation framework, which requires no on-line optimization, was introduced by the authors in Mercader, Rubin, Nguyen, Bemporad, and Gutman (2018). That, so called Simple Interpolating Control (SIC), is easy to implement, but closed loop stability is, in general, not guaranteed. Although the stability of the closed loop can be analyzed off-line, it is more 'idiot-proof' to have it systematically guaranteed. This work is aimed to extend the results from Mercader et al. (2018) by providing a computationally modest interpolating scheme which also guarantees closed-loop stability. Three new algorithms are proposed.

The reminder of this text includes some preliminary mathematical background in Section 2; main results are reported in Section 3; extension for plants with persistent disturbances is documented in Section 4; finally, numerical examples are presented in Section 5, and the conclusions are given in Section 6.

2. Preliminaries

Consider the discrete-time uncertain and possibly also time-varying linear system,

$$x(k+1) = A(k)x(k) + B(k)u(k) + D(k)w(k) \tag{1}$$

The state $x(k) \in \mathbb{R}^n$ is bounded by a symmetrical polytope, while the control and disturbance inputs $u(k) \in \mathbb{R}^m$, and $w(k) \in \mathbb{R}^d$, respectively are bounded as

$$x(k) \in \mathcal{X} = \{x \in \mathbb{R}^n : |Lx| \le \mathbf{1}\}, \tag{2a}$$

$$u(k) \in \mathcal{U} = \{u \in \mathbb{R}^m : |u| \le \bar{\mathbf{u}}\}, \tag{2b}$$

$$w(k) \in \mathcal{W} = \{w \in \mathbb{R}^d : \|w\|_2 \le 1\}. \tag{2c}$$

In (2), $\mathbf{1}$ is a column vector of h ones, $L \in \mathbb{R}^{h\times n}$ and $\bar{\mathbf{u}} \in \mathbb{R}^m$; the inequalities and absolute values in (2a), (2b) are to be interpreted element-wise. The matrices $A(k) \in \mathbb{R}^{n\times n}$ and $B(k) \in \mathbb{R}^{n\times m}$, and $D(k) \in \mathbb{R}^{n\times d}$ are given with polytopic uncertainty, without loss of generality (Nguyen, 2014), as follows,

$$A(k) = \sum_{i=1}^{s} \alpha_i(k)A_i, \tag{3a}$$

$$B(k) = \sum_{i=1}^{s} \alpha_i(k)B_i, \tag{3b}$$

$$D(k) = \sum_{i=1}^{s} \alpha_i(k)D_i, \tag{3c}$$

$$\sum_{i=1}^{s} \alpha_i(k) = 1,\ \ \alpha_i(k) \ge 0,\ \ \forall i = 1,\dots,s, \tag{3d}$$

where $\alpha_i(k)$ is unknown, and possibly also time-varying.

For a given positive definite matrix Q we define a corresponding ellipsoid,

$$\mathcal{E}(Q) = \{x : x^\top Q^{-1} x \le 1\}. \tag{4}$$

Definition 1. An ellipsoid $\mathcal{E}(Q) \subset \mathcal{X}$ is *robustly contractive* w.r.t system (1) subjected to the constraints (2), if there exist a control law $u(x) \in \mathcal{U}$ such that

$$x(k+1)^\top Q^{-1} x(k+1) - x(k)^\top Q^{-1} x(k) < 0 \tag{5}$$

for every k and for all $x(k) \in \mathcal{E}(Q) \setminus 0$ and $w(k) \in \mathcal{W}$.

Definition 2. An ellipsoid $\mathcal{E}(Q) \subset \mathcal{X}$ is *robustly invariant* w.r.t system (1) subjected to the constraints (2), and w.r.t a given control law $u(x) \in \mathcal{U}$, if

$$x(k+1)^\top Q^{-1} x(k+1) \le 1 \tag{6}$$

for every k and for all $x(k) \in \mathcal{E}(Q)$ and $w(k) \in \mathcal{W}$.

Remark 1. For persistent disturbances a contractive set as per Definition 1 cannot be found, as $\nexists Q$ such that inequality (5) holds in the proximity of the origin. A definition for this case can be formulated by replacing $x(k) \in \mathcal{E}(Q) \setminus 0$ with $x(k) \in \mathcal{E}(Q) \setminus \Omega_m$ where Ω_m is the minimal robustly positively invariant set.

The function $\mathrm{sat}(\cdot)$ denotes the standard vector valued saturation

$$\mathrm{sat}(u) = \begin{bmatrix} \mathrm{sign}(u_1)\cdot \min\{\bar{u}_1, |u_1|\} \\ \mathrm{sign}(u_2)\cdot \min\{\bar{u}_2, |u_2|\} \\ \vdots \\ \mathrm{sign}(u_m)\cdot \min\{\bar{u}_m, |u_m|\} \end{bmatrix}. \tag{7}$$

This nonlinear function can be represented by linear differential inclusions (LDI) (Boyd, Ghaoui, Feron, & Balakrishnan, 1994). Let $\{E_j\}, j = 1,\dots,2^m$ be the set of all diagonal matrices of dimension m with diagonal elements equaling 0 or 1. For example, for $m = 2$:

$$\{E_1, E_2, E_3, E_4\} = \left\{ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right\}.$$

Denote $E_j^- = I - E_j$. Following Hu and Lin (2001), $\mathrm{sat}(Kx)$ can be written as

$$\mathrm{sat}(Kx) = \sum_{j=1}^{2^m} \sigma_j \left(E_j K x + E_j^- H x\right),\ \ \sum_{j=1}^{2^m} \sigma_j = 1,\ \ \sigma_j \ge 0, \tag{8}$$

where $|Hx| \le \bar{\mathbf{u}}\ \forall x \in \mathcal{E}(Q)$. Note that the above LDI representation from Hu and Lin (2001) is used here for simplicity, and the results below may also be derived using the LDI models given in Alamo, Cepeda, Limon, and Camacho (2006) or in Molchanov and Pyatnitskiy (1989). For more details on the different LDI models see e.g. Tarbouriech, Garcia, Gomes da Silva, and Queinnec (2011).

Finally, the following version of *S-procedure* is used for the disturbed plant case in Section 4.

Lemma 1 (*S-procedure, Khlebnikov, Polyak, & Kuntsevich, 2011, pp. 2231-2231*). *Consider $A_i = A_i^\top \in \mathrm{R}^{n\times n}$. If there exist real numbers $\tau_i \geq 0$, $i = 1, \dots, m$, such that*

$$A_0 \leq \sum_{i=1}^{m} \tau_i A_i, \quad \alpha_0 \geq \sum_{i=1}^{m} \tau_i \alpha_i, \tag{9}$$

then the inequalities

$$x^\top A_i x \leq \alpha_i, \quad i = 1, \dots, m, \tag{10}$$

imply the inequality

$$x^\top A_0 x \leq \alpha_0. \tag{11}$$

3. Main results

For simplicity, the disturbance-free case, i.e., $w = 0$ in (1), is considered first. Let $u(k) = \mathrm{sat}(K_a x(k))$ be a robustly stabilizing saturated control law, designed to satisfy given performance requirements. A robustly contractive ellipsoid associated with this controller exists (Blanchini & Miani, 2008, p. 142), and is denoted as $\mathcal{E}(Q_a)$. To extend the feasible region of the control law to be admissible outside $\mathcal{E}(Q_a)$, a second, stabilizing, controller $u(k) = \mathrm{sat}(K_b x(k))$ is synthesized. To have the feasible region as large as possible, the gain K_b should be chosen such that the associated robustly contractive ellipsoid, $\mathcal{E}(Q_b)$, is maximized, see Lemma 3. This is however not a requirement, and K_b can be chosen in some different way, as long as it is robustly stabilizing. The only demand is that $\mathcal{E}(Q_a) \subset \mathcal{E}(Q_b)$. These two saturated control laws are denoted as *inner* and *outer*, respectively.

Inside $\mathcal{E}(Q_a)$ the inner control law is admissible and therefore should be used. In $\mathcal{E}(Q_b) \setminus \mathcal{E}(Q_a)$ the control law $u(k) = \mathrm{sat}(K_b x(k))$ can be used; however, the performance might be poor, and chattering might occur when switching between the two control laws. Therefore, we present an interpolation scheme to blend the controller actions in $\mathcal{E}(Q_b) \setminus \mathcal{E}(Q_a)$.

The control input at time k is decomposed as

$$u(k) = c(k)u_b(k) + (1 - c(k))u_a(k), \tag{12}$$

with $u_a(k) = \mathrm{sat}(K_a x(k))$ and $u_b(k) = \mathrm{sat}(K_b x(k))$. It is desirable to have $c(k)$ as small as possible, thus making the high-performance, inner, controller as dominant as possible. Such a minimization problem demands however the solution of an LMI problem at every time step (Nguyen et al., 2011). This is impractical for most real-time implementations, thus we resort to a suboptimal approach to reduce the computational burden. A simplistic approach, with very simple calculations, which uses geometrical relations to interpolate was presented by the authors in Mercader et al. (2018). Sufficient stability conditions are presented, and can be used to analyze the robust stability for a given plant and given inner and outer controllers. However, stability cannot, in general, be guaranteed *à priori*. The method presented in this paper aims to resolve that issue, with minimal computational burden.

The idea is to find $c(k)$ such that the applied control law at time k, for $x(k) \in \mathcal{E}(Q_b) \setminus \mathcal{E}(Q_a)$, guarantees that

$$x^\top(k+1)Q_b^{-1}x(k+1) - x^\top(k)Q_b^{-1}x(k) < 0, \tag{13}$$

i.e., that $\mathcal{E}(Q_b)$ is contractive.

To avoid complex optimization, a one-dimensional grid of possible choices of $c(k) \in [0, 1]$ is considered, i.e.,

$$\mathcal{C} = \begin{bmatrix} 0 & \delta_c & 2\delta_c & \cdots & 1 \end{bmatrix}, \tag{14}$$

where δ_c defines the grid distance. The number of grid points is $q = 1/\delta_c + 1$. Denote the q-dimensional row vector, $\mathbf{1}^\top = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}$. A 1-step prediction of all possible values of the future state is given as

$$\mathcal{R}_1 = \begin{bmatrix} r_1 & r_2 & \cdots & r_s \end{bmatrix}, \tag{15}$$

where $r_i \in \mathbb{R}^{n\times q}$ is the 1-step prediction for a given extreme case (A_i, B_i), for all q cases in $\mathcal{C}$,

$$\begin{aligned} r_i &= \begin{bmatrix} r_{i1} & \cdots & r_{iq} \end{bmatrix} \\ &= \begin{bmatrix} A_i x(k) + B_i\, u_a(x(k)) & \cdots & A_i x(k) + B_i\, u_b(x(k)) \end{bmatrix} \\ &= A_i x(k)\mathbf{1}^\top + B_i\, u_b\,(x(k))\,\mathcal{C} + B_i\, u_a\,(x(k))\,(\mathbf{1}^\top - \mathcal{C}). \end{aligned} \tag{16}$$

For every possible future state in (16), condition (13) can be checked. All the predictions that do not meet (13) are ruled out. The interpolation coefficient, $c(k)$, is chosen as the first instance of $\mathcal{C}$ for which (13) is satisfied for every pair (A_i, B_i). Note that such a $c(k)$ always exists since $c(k) = 1$ is feasible. By doing so, we guarantee that the state trajectory will always reach $\mathcal{E}(Q_a)$, and prioritize the inner control law. If the state is inside $\mathcal{E}(Q_a)$, the inner controller is admissible, and no interpolation is needed.

This new algorithm for one time step is summarized as Algorithm 1. Note that the algorithm always terminates with $c \in [0, 1]$. In the worst case, the number of points that need to be evaluated is sq. Hence, the number of grid elements q, and the number of uncertain extreme cases s, play a large part in determining the computational demand; for systems without uncertainty $i = s = 1$. Following are feasibility and robust stability proofs.

Algorithm 1 Interpolating Control

```
1:  procedure u(x)
2:      c = 0
3:      if x^T(k) Q_a^{-1} x(k) > 1 then
4:          for i ← 1 to s do
5:              while c ≤ 1 do
6:                  u_c ← c u_b(x) + (1 − c) u_a(x),
7:                  r_ic ← A_i x(k) + B_i u_c
8:                  if r_ic^T Q_b^{-1} r_ic − x^T(k) Q_b^{-1} x(k) ≥ 0 then
9:                      c ← c + δ_c
10:                     goto line 4
11:                 end if
12:             end while
13:         end for
14:     end if
15:     u ← c u_b(x) + (1 − c) u_a(x)
16: end procedure
```

Theorem 1 (*Feasibility*). *Consider the system (1)–(3), Algorithm 1 guarantees a feasible control law for all initial states $x(0) \in \mathcal{E}(Q_b)$.*

Proof. Algorithm 1 always terminates with $c \in [0, 1]$. Since the inner and outer control laws are saturated, $u_a \in \mathcal{U}$ and $u_b \in \mathcal{U}$. Therefore, by the properties of convex combinations, $u = c\, u_a + (1 - c)\, u_b \in \mathcal{U}$. $\square$

Theorem 2 (*Robust Stability*). *Consider the system (1)–(3), Algorithm 1 guarantees robust stability for initial states $x(0) \in \mathcal{E}(Q_b)$.*

Proof. Let $V(x) = x^\top Q_b^{-1} x \geq 0$ be a Lyapunov function candidate in $\mathcal{E}(Q_b) \setminus \mathcal{E}(Q_a)$. Since the outer control law, u_b, is robustly contractive in $\mathcal{E}(Q_b)$, we have that for $c(k) = 1$, V is decreasing. Algorithm 1 only returns $c < 1$ for $x \in \mathcal{E}(Q_b) \backslash \mathcal{E}(Q_a)$ if the resulting control input is contractive. From Theorem 1 the obtained control input is admissible for every $x(k) \in \mathcal{E}(Q_b)$. Hence, $V(x(k+1)) < V(x(k))\ \forall x(k)$, and the system is robustly stable in the Lyapunov sense for every $x \in \mathcal{E}(Q_b) \backslash \mathcal{E}(Q_a)$. The trajectories are thus destined to reach $\mathcal{E}(Q_a)$; once $x(k) \in \mathcal{E}(Q_a)$ the applied inner control law is robustly stabilizing. $\square$

3.1. A quick and dirty alternative

The proposed algorithm may become too computationally demanding for control systems involving limited computation capabilities, fast plant dynamics, or uncertainty. In this section we present another algorithm that, on an average for many cases, should be faster than Algorithm 1. This new approach is composed from Algorithm 1 and Mercader et al. (2018) as Algorithm 2.

In Mercader et al. (2018), the state $x(k) \in \mathcal{E}(Q_b) \setminus \mathcal{E}(Q_a)$ is decomposed as

$$x(k) = c(k)x_b(k) + (1 - c(k))x_a(k), \tag{17}$$

where $x_a(k) \in \mathcal{E}(Q_a)$, $x_b(k) \in \mathcal{E}(Q_b)$, and $0 \leq c(k) \leq 1$. Then, $x_a(k)$ and $x_b(k)$ are selected on the line from the origin through $x(k)$, i.e., $x_a(k) = x(k)/a(k)$ and $x_b(k) = x(k)/b(k)$. The interpolation coefficient is $c(k) = c_0(k)$, where

$$c_0(k) = \frac{b(k)(a(k) - 1)}{a(k) - b(k)}, \tag{18}$$

$a(k) = \sqrt{x(k)^\top Q_a^{-1} x(k)}$, and $b(k) = \sqrt{x(k)^\top Q_b^{-1} x(k)}$. The control input is then given as (12).

While this method is easy to implement, there are some cases in which the closed-loop exhibits limit cycles (Mercader et al., 2018). The following solution is proposed: use the method in Mercader et al. (2018) while checking at each step k if (13) holds for the computed $c(k)$ for every plant case. This means that instead of checking for every point in the grid $\mathcal{C}$, we only check for a single c. In case the inequality is false, we seek a new $c(k) \in (c_0(k) \;\; 1]$, This can be performed by a search over e.g the grid $\mathcal{C} = [c_0 + \delta_c \;\; \cdots \;\; 1 - \delta_c \;\; 1]$, like in Algorithm 1. However, a binary search ("bisection") can in general find a suitable c with fewer computations (Knuth, 2014, Sec. 6.2.1). This "quick and dirty" approach is summarized as Algorithm 2.

Algorithm 2 Quick n' Dirty

1: **procedure** $u(x)$
2: $\quad c = 0$
3: $\quad$ **if** $x^\top(k)Q_a^{-1}x(k) > 1$ **then**
4: $\quad\quad a \leftarrow \sqrt{x(k)^\top Q_a^{-1} x(k)}$, $b \leftarrow \sqrt{x(k)^\top Q_b^{-1} x(k)}$, $c \leftarrow b(a - 1)/(a - b)$
5: $\quad\quad$ **for** $i \leftarrow 1$ to s **do**
6: $\quad\quad\quad u_c \leftarrow c\, u_b(x) + (1 - c)\, u_a(x)$,
7: $\quad\quad\quad r_{ic} \leftarrow A_i x(k) + B_i u_c$
8: $\quad\quad\quad$ **if** $r_{ic}^\top Q_b^{-1} r_{ic} - x^\top(k) Q_b^{-1} x(k) \geq 0$ **then**
9: $\quad\quad\quad\quad c \leftarrow (1 - c)/2$
10: $\quad\quad\quad\quad$ **goto** line 5
11: $\quad\quad\quad$ **end if**
12: $\quad\quad$ **end for**
13: $\quad$ **end if**
14: $\quad u \leftarrow c\, u_b(x) + (1 - c)\, u_a(x)$
15: **end procedure**

Algorithm 2 is guaranteed to terminate at each time step with $c \in [0 \;\; 1]$. The binary search will find a suitable c in less operations than the linear search of Algorithm 1 provided that rounding in line 9 is done with accuracy δ_c. Feasibility and robust stability of the algorithm can be proven following the same lines as the proofs of Theorems 1 and 2, respectively.

4. Plants subject to disturbances

We now consider the disturbed case, i.e. the general case described in (1), with $w \in \mathcal{W} = \left\{ w \in \mathbb{R}^n : w^\top w \leq 1 \right\}$. Recall that from Remark 1, a contractive ellipsoid cannot be found in presence of persistent disturbances. However, we show below that a method, similar to Algorithms 1, 2, may be used with an invariant set taking the role of the (contractive) outer set. The computation of the required invariant set is newly developed in Section 4.1; the new control algorithm is given in Section 4.2 as Algorithm 3.

It may be argued that strict assumptions on the disturbance sizes cannot be made, although much research exists to control systems under such assumptions (see e.g. Blanchini & Miani, 2008; Borrelli et al., 2017; Shingin & Ohta, 2004; Tahir, 2010; Tarbouriech et al., 2011; Trodden, 2016). Here we provide some theoretical results. A control practitioner may however choose to neglect some disturbances.

4.1. Computing ellipsoidal invariant sets

The proposed scheme requires the knowledge of ellipsoidal invariant set for the two saturated control laws. Various methods for computation of ellipsoidal invariant sets can be found in the literature (Alamo et al., 2006; Boyd et al., 1994; Hu, Lin, & Chen, 2002; Nguyen et al., 2011; Shingin & Ohta, 2004), but, to the best of the authors' knowledge, not for saturated control laws in the presence of additive disturbances. Following is a method for the computation of the above required sets.

From (6), $\mathcal{E}(Q)$ is invariant w.r.t. system (1)–(3) under the control law $u = \mathrm{sat}(Kx)$, if and only if for every $x \in \mathcal{E}(Q)$ and for every $w \in \mathcal{W}$,

$$(A_i x + B_i \mathrm{sat}(Kx) + D_i w)^\top Q^{-1} (A_i x + B_i \mathrm{sat}(Kx) + D_i w) \leq 1, \tag{19}$$

for every $i = 1, \ldots, s$. Using (8), we can write

$$A_i x + B_i \mathrm{sat}(Kx) + D_i w = \sum_{j=1}^{2^m} \sigma_j \left(A_i x + B_i E_j K x + B_i E_j^- H x + D_i w \right). \tag{20}$$

Denote $\mathcal{A}_{ij} = A_i + B_i E_j K + B_i E_j^- H$. Substituting (20) into (19) and rearranging in matrix form yields, with stars ($\star$) denoting the corresponding transposed elements,

$$\begin{bmatrix} x \\ w \end{bmatrix}^\top \begin{bmatrix} \mathcal{A}_{ij}^\top Q^{-1} \mathcal{A}_{ij} & \star \\ D_i^\top Q^{-1} \mathcal{A}_{ij} & D_i^\top Q^{-1} D_i \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} \leq 1, \quad \forall\, i = 1, \ldots, s, \;\; j = 1, \ldots, 2^m, \tag{21}$$

where σ_j is subsumed into the inequality due to its definition in (8), making (21) a sufficient condition for (19) to hold. By using Eq. (11), (21) holds if $x^\top Q^{-1} x \leq 1$, $w^\top w \leq 1$ and there exist non-negative scalars τ_1 and τ_2 such that

$$\begin{bmatrix} \mathcal{A}_{ij}^\top Q^{-1} \mathcal{A}_{ij} & \star \\ D_i^\top Q^{-1} \mathcal{A}_{ij} & D_i^\top Q^{-1} D_i \end{bmatrix} \preceq \tau_1 \begin{bmatrix} Q^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \tau_2 \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix} \tag{22}$$

and $\tau_1 + \tau_2 \leq 1$ for every $i = 1, \ldots, s$ and $j = 1, \ldots, 2^m$. If (22) holds for $\tau_1 + \tau_2 = 1$, it holds for any $\tau_1 + \tau_2 < 1$, $\tau_1 \geq 0$, $\tau_2 \geq 0$. Therefore, we can set $\tau = \tau_1$, $\tau_2 = 1 - \tau$. Hence, with some algebraic manipulations, (22) yields

$$\begin{bmatrix} \mathcal{A}_{ij}^\top \\ D_i^\top \end{bmatrix} Q^{-1} \begin{bmatrix} \mathcal{A}_{ij} & D_i \end{bmatrix} - \begin{bmatrix} \tau Q^{-1} & 0 \\ 0 & (1 - \tau) I \end{bmatrix} \preceq 0, \quad \forall i = 1, \ldots, s, \;\; \forall j = 1, \ldots, 2^m. \tag{23}$$

By the Schur complement (Boyd et al., 1994, p. 28), we obtain

$$\begin{bmatrix} \tau Q & \star & \star \\ 0 & (1 - \tau) I & \star \\ (A_i + B_i E_j K + B_i E_j^- H) Q & D_i & Q \end{bmatrix} \succeq 0. \tag{24}$$

Introducing new variables: $Y = KQ$ and $Z = HQ$, we obtain the LMI

$$\begin{bmatrix} \tau Q & \star & \star \\ 0 & (1-\tau)I & \star \\ A_iQ+B_iE_jY+B_iE_j^-Z & D_i & Q \end{bmatrix} \succeq 0, \qquad (25)$$
$$\forall i = 1,\ldots,s,\ \forall j = 1,\ldots,2^m.$$

To satisfy the state and input constraints we use the known LMI conditions from e.g. Hu and Lin (2001). For state constraint satisfaction

$$\begin{bmatrix} 1 & l_iQ \\ Ql_i^\top & Q \end{bmatrix} \succeq 0, \quad \forall i = 1,\ldots,h, \qquad (26)$$

where l_i is the ith row in L. For inputs constraint

$$\begin{bmatrix} \bar{u}_i^2 & z_i \\ z_i^\top & Q \end{bmatrix} \succeq 0, \quad \forall i = 1,\ldots,m, \qquad (27)$$

where z_i is the ith row of Z.

Inequalities (25)–(27) can be readily used to solve various control problems. Perhaps the most basic of which is given in Lemma 2.

Lemma 2. *The smallest "size" invariant ellipsoid $\mathcal{E}(Q)$ w.r.t. system (1) under constraints (2) and a given saturated control $u(k) = \mathrm{sat}(Kx(k))$, can be computed by solving the semi-definite program (SDP),*

$$\min_{Q,Z}\ \mathrm{trace}(Q) \qquad (28)$$

subject to (25)–(27).

Often, it is beneficial to compute the control gain K which results in the maximal invariant ellipsoid. This can be done following Lemma 3.

Lemma 3. *The largest "size" invariant ellipsoid $\mathcal{E}(Q)$ w.r.t. system (1) under constraints (2) and any saturated control $u(k) = \mathrm{sat}(Kx(k))$, can be computed together with $K = YQ^{-1}$ by solving the SDP,*

$$\max_{Q,Y,Z}\ \mathrm{trace}(Q) \qquad (29)$$

subject to (25)–(27).

In the above Lemmas, each SDP is solved for a pre-chosen τ in (25). In practice one should conduct a linear search on $\tau \in (0, 1)$ to optimize the objective.

4.2. Algorithm for plants subject to disturbances

It is assumed that a performance driven inner control law $u_a = \mathrm{sat}(K_ax)$ is given, and that it robustly stabilizes the unconstrained system (1) in the presence of disturbances. The associated invariant set $\mathcal{E}(Q_a)$ can be computed by solving the SDP in Lemma 2 with objective given as $\max\{\mathrm{trace}(Q)\}$ instead of $\min\{\mathrm{trace}(Q)\}$. To achieve maximal feasibility of the interpolating control scheme we seek an outer control law $u(k) = \mathrm{sat}(K_bx)$ such that its associated invariant set is of maximal size; the above is facilitated by solving the SDP in Lemma 3, yielding $K_b = YQ$ and the corresponding *maximal* invariant ellipsoid $\mathcal{E}(Q_M)$. Solving the SDP from Lemma 2 with the obtained K_b, a *minimal* invariant ellipsoid $\mathcal{E}(Q_m)$ is computed. We show in Theorems 3, 4 below that the norm

$$V(x) = x^\top Q_m^{-1}x \geq 0, \qquad (30)$$

is suitable for a robustly stabilizing and recursively feasible interpolating control scheme.

With some conservativeness, $\mathcal{W}$ assumed to be a hypercube bounding $w^\top w < 1$. Hence, $\mathcal{W}$ can be written as the convex hull of 2^n vertices,

$$\mathcal{W} = \mathrm{conv}\left([W_1 \ldots W_{2^n}]\right), \qquad (31)$$

with $\mathrm{conv}(\cdot)$ denoting the convex hull of a given array of points in $\mathbb{R}^n$, and W_j, $j = 1,\ldots,2^n$, denotes a vertex of $\mathcal{W}$. The one-step prediction of all possible future states in (16) is modified to account for the additive disturbances as

$$r_i = (A_ix(k)\mathbf{1} + B_i\, u_b\,((k))\, \mathcal{C} + B_i\, u_a\,(x(k))\,(\mathbf{1} - \mathcal{C})) \oplus D_i\mathcal{W}. \qquad (32)$$

The Minkowski addition in (32) is carried out by adding each vertex of $\mathcal{W}$ to each disturbance-free prediction point. The number of points in $\mathcal{R}$ is thus 2^nsq.

Hence, modified versions of Algorithm 1 or Algorithm 2 can be used for the control in presence of disturbances. A modified version of Algorithm 1 is given as Algorithm 3.

Algorithm 3 Interpolating Control for Plants With Additive Disturbance

1: **procedure** $u(x)$
2: $\quad c = 0$
3: $\quad$**if** $x^\top(k)Q_a^{-1}x(k) > \mathbf{1}$ **then**
4: $\quad\quad$**for** $i \leftarrow 1$ to s **do**
5: $\quad\quad\quad$**while** $c \leq 1$ **do**
6: $\quad\quad\quad\quad u_c \leftarrow c\, u_b(x) + (1-c)\, u_a(x),$
7: $\quad\quad\quad\quad x_{ic} \leftarrow A_ix(k) + B_iu_c$
8: $\quad\quad\quad\quad$**for** $j \leftarrow 1$ to 2^n **do**
9: $\quad\quad\quad\quad\quad r_{ijc} \leftarrow x_{ic} + D_iW_j$
10: $\quad\quad\quad\quad\quad$**if** $r_{ijc}^\top\, Q_m^{-1}\, r_{ijc} - x^\top(k)Q_m^{-1}\, x(k) \geq 0$ **then**
11: $\quad\quad\quad\quad\quad\quad c \leftarrow c + \delta_c$
12: $\quad\quad\quad\quad\quad\quad$**goto** line 4
13: $\quad\quad\quad\quad\quad$**end if**
14: $\quad\quad\quad\quad$**end for**
15: $\quad\quad\quad$**end while**
16: $\quad\quad$**end for**
17: $\quad$**end if**
18: $\quad u \leftarrow c\, u_b(x) + (1-c)\, u_a(x)$
19: **end procedure**

Theorem 3 (*Feasibility with disturbances*)**.** *Consider the system (1)–(3), Algorithm 3 guarantees a feasible control law for all initial states $x(0) \in \mathcal{E}(Q_M)$ and $W \in \mathcal{W}$.*

Proof. The proof follows the lines of Theorem 1. □

Theorem 4 (*Robust Stability*)**.** *Consider the system (1)–(3), Algorithm 3 guarantees robust stability for initial states $x(0) \in \mathcal{E}(Q_M)$.*

Proof. The first part of the proof follows (Khlebnikov et al., 2011). Let $V(x) = x^\top Q_m^{-1}x \geq 0$ be a Lyapunov function candidate. It is required that

$$V(x(k+1)) < V(x(k)), \quad \forall x \in \left\{x : x^\top Q_m^{-1}x > 1\right\},$$
$$\forall w \in \left\{w : w^\top w \leq 1\right\}.$$

The above is equivalent to

$$V(x(k)) \leq 1, \quad \forall (x(k), w(k)) \ \text{ s.t. } \ V(x(k+1)) \geq V(x(k)),$$
$$w(k)^\top w(k) \leq 1.$$

By the S-procedure, the above is shown to be equivalent to LMI (25) with Q replaced by Q_m. Hence, $\mathcal{E}(Q_m)$ is attractive in $\mathcal{E}(Q_M) \setminus \mathcal{E}(Q_m)$ for the control law $u_b = \mathrm{sat}(K_bx)$. In other words, for $c = 1$, (30) is a Lyapunov function in $\mathcal{E}(Q_M) \setminus \mathcal{E}(Q_m)$.

Algorithm 3 only returns $c \in [0, 1]$ for $x \in \mathcal{E}(Q_M) \setminus \mathcal{E}(Q_m)$ if the resulting control input is contractive, and $c = 1$ is always

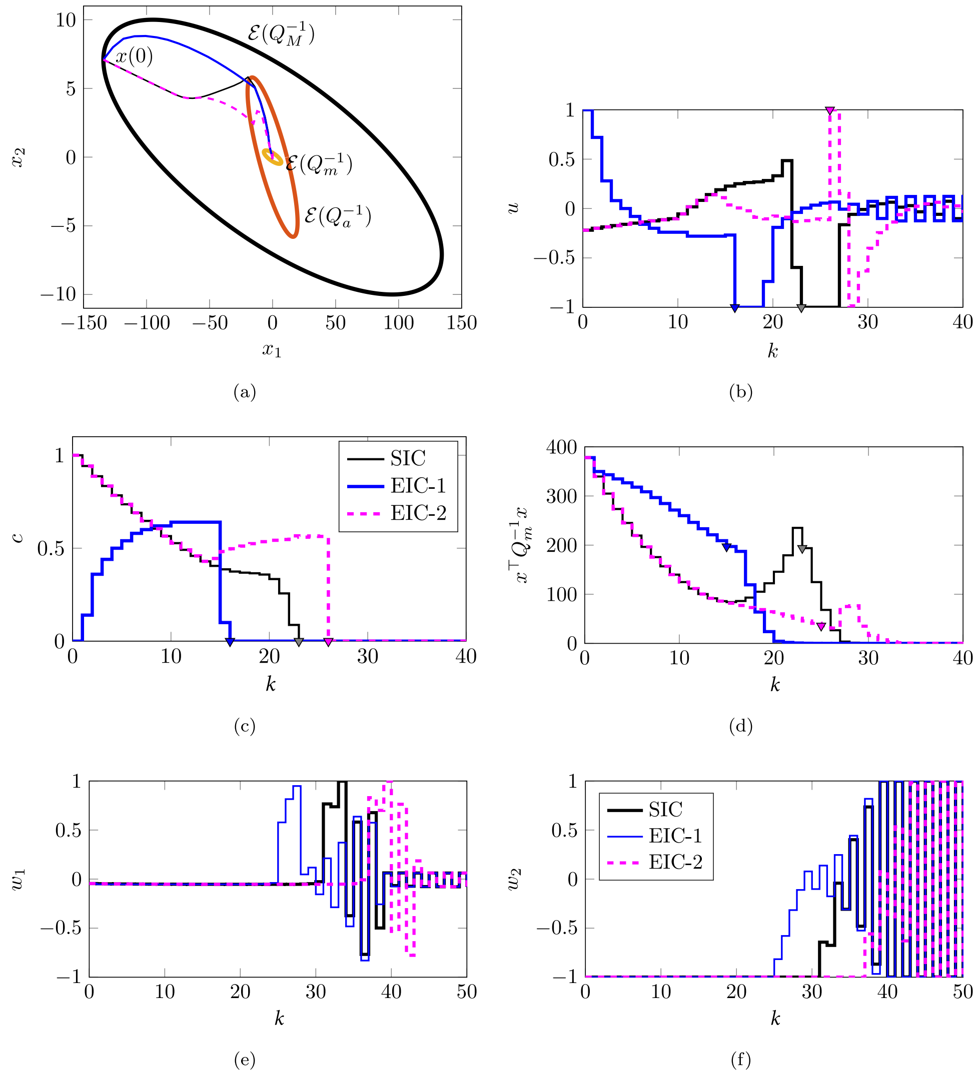

Fig. 1. Simulation results of example 1, for simple interpolating control (Mercader et al., 2018) (SIC), Algorithm 3 (EIC-1), and Algorithm 2 with disturbances (EIC-2). Points of entry into $\mathcal{E}(Q_a^{-1})$ are marked by ▽'s. Note that after entering $\mathcal{E}(Q_a)$ interpolation ceases and $x^\top Q_m^{-1} x$ may increase.

a feasible solution. By Theorem 3, the obtained control input is admissible for every $x(k) \in \mathcal{E}(Q_M)$. Hence, the system is robustly stable in the Lyapunov sense for every $x \in \mathcal{E}(Q_M) \setminus \mathcal{E}(Q_m)$. Assuming $\mathcal{E}(Q_m) \subseteq \mathcal{E}(Q_a)$, the trajectories are bound to enter $\mathcal{E}(Q_a)$; once $x(k) \in \mathcal{E}(Q_a)$ the applied inner control law is robustly stabilizing, guarantying convergence to a minimal invariant set whose size depends on the disturbances in (1). □

Remark 2. The assumption $\mathcal{E}(Q_m) \subseteq \mathcal{E}(Q_a)$ is non-trivial and indeed painful. In the case where no K_b which corresponds to an appropriate set $\mathcal{E}(Q_m)$ can be found, the inner controller u_a might have to be detuned to satisfy the assumption, by e.g., reducing its gain.

5. Examples

In this section some numerical examples are presented. The simulations have been performed on an Intel Xeon E3 V5 (2.8 GHz) with 32 GB RAM, running Matlab 2017a. The required SDPs were solved using the Yalmip (Löfberg, 2004) parser with the SDPT-3 (Toh, Todd, & Tütüncü, 1999) solver.

5.1. Example 1

Consider the constrained double integrator with disturbances adapted from Mayne, Seron, and Raković (2005),

$$x(k+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} u(k) + \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} w(k).$$

$$-10 \le x_2 \le 10, \quad -1 \le u \le 1, \quad w^\top w \le 1.$$

The inner and outer control gains are selected as

$$K_a = \begin{bmatrix} -0.3175 & -1.1664 \end{bmatrix}, \quad K_b = \begin{bmatrix} -0.0527 & -1.0280 \end{bmatrix},$$

to achieve required performance (H_∞ norm minimization (Boyd et al., 1994)) and to maximize the domain of attraction (by Lemma 2), respectively. The two saturated control law are associated with invariant ellipsoids given by

$$Q_a = \begin{bmatrix} 393.5250 & -93.8734 \\ -93.8734 & 33.6350 \end{bmatrix},$$

$$Q_M = \begin{bmatrix} 1.8024 & -0.0950 \\ -0.0950 & 0.0100 \end{bmatrix} 10^4,$$

$$Q_m = \begin{bmatrix} 47.7039 & -2.5731 \\ -2.5731 & 0.2391 \end{bmatrix}.$$

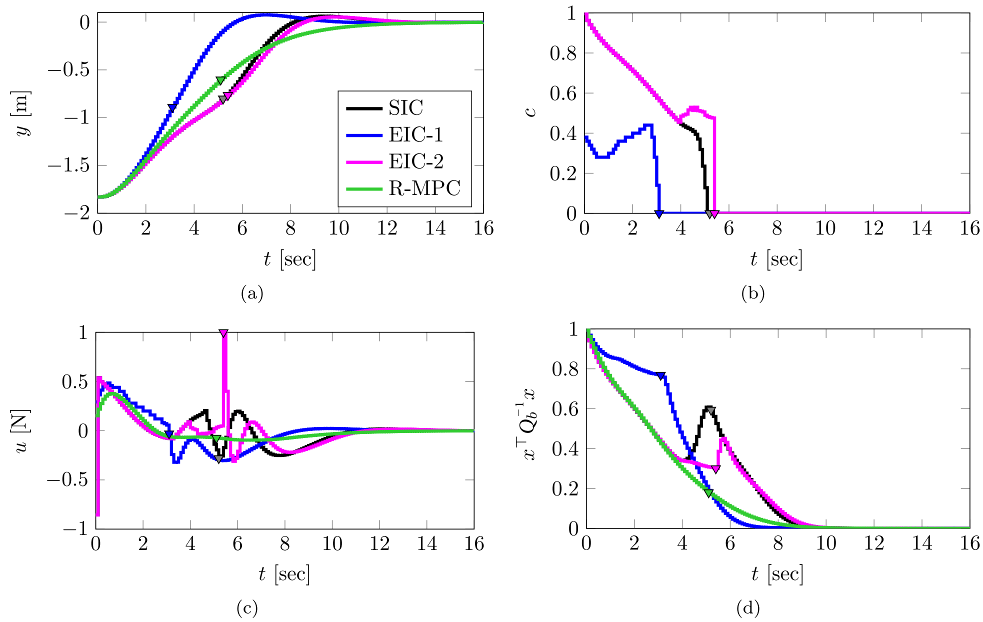

Fig. 2. Simulation results of example 2, for simple interpolating control (Mercader et al., 2018) (SIC), Algorithm 1 (EIC-1), Algorithm 2 (EIC-2), and the robust MPC scheme (Kothare et al., 1996) (R-MPC). Points of entrance into $\mathcal{E}(Q_a)$ are marked by ▿'s.

The offline computation times of the above ellipsoids are given as 0.37 s, 0.87 s, and 0.28 s, respectively. Simulation results for an initial condition $x(0) = [-134.2520 \quad 7.0957]^\top$ are given in Fig. 1. At each time step the worst case disturbance, in terms of maximizing the $V(x) = x^\top Q_m^{-1} x$ norm, is applied. The resulting $w(k)$ are presented in Figs. 1e and 1f. The algorithms used are Algorithm 3 – the extension of Algorithm 1 for the disturbed case (EIC-1), and a similar extension for Algorithm 2 (EIC-2). The SIC method (Mercader et al., 2018) was also simulated, and shown for reference.

For the given initial conditions SIC and Algorithm 2 obtained similar results until $t = 14$; for $t > 14$ Algorithm 2 achieved better performance w.r.t. the norm $V(x)$, but entered the set $\mathcal{E}(Q_a^{-1})$ later. EIC-1 yields the fastest convergence into $\mathcal{E}(Q_a^{-1})$ even though its $V(x)$ norm is larger. That is expected since EIC-1 does not aim at minimizing c. As expected, the SIC computation is fastest with 0.14 [ms] on average, followed by EIC-1 and EIC-2 with 0.41 [ms] and 0.44 [ms], respectively.

5.2. Example 2

Consider the two masses–spring system benchmark example from Kothare et al. (1996) and Reinelt (2000). It consists of two masses coupled by a linear spring. A discrete state–space model is given in Kothare et al. (1996) as

$$x(k+1) = \begin{bmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ -0.1K & 0.1K & 1 & 0 \\ 0.1K & -0.1K & 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} u(k)$$

$$y(k) = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} x(k)$$

corresponding to the masses being 1 kg each, and the sample time equal to 0.1 s. The states are, respectively, the position of the 1st mass, position of the 2nd mass, velocity of the 1st mass, and velocity of the 2nd mass. The spring coefficient $K \in [0.5 \; 1.5]$ N/m is constant yet unknown. We introduce the state and input constraints

$$-2 \le x_1 \le 2, \quad -2 \le x_2 \le 2, \quad -10 \le x_3 \le 10,$$
$$-10 \le x_4 \le 10, \quad -1 \le u \le 1.$$

The inner control law gain was designed as static H-infinity state-feedback (Cao, Lam, & Sun, 1998) with weighting $Q = R = 1$,

$$K_a = \begin{bmatrix} -13.4535 & 7.6441 & -6.1226 & -5.7299 \end{bmatrix}.$$

The outer control law was designed to assign largest invariant ellipsoid, and given as

$$K_b = \begin{bmatrix} -8.2475 & 7.6813 & -11.0494 & 3.5357 \end{bmatrix}.$$

These two control gains were used in interpolating control schemes given in Algorithm 1 and Algorithm 2, as well as in the simple interpolating scheme from Mercader et al. (2018). Simulation results for an initial condition $x(0) = [-1.60 \quad -1.83 \quad 0 \quad 0]^\top$ are given in Fig. 2. Simulation results of the robust MPC in Kothare et al. (1996) are also given as reference.

It is shown that Algorithm 1 brought the output faster to the origin (Fig. 2a) and to the inner invariant set (Fig. 2c). Algorithm 2 and SIC achieved similar results and approximately the same convergence time as the robust MPC. Algorithm 2 was also the slowest of the interpolating controllers with 0.1 ms mean computation time — rendering the "quick n' dirty" solution just dirty. SIC and Algorithm 1 had similar mean computation times: 0.067 ms and 0.075 ms, respectively. The robust MPC comes last with a 118 ms – just above the system sampling time – which means that a stronger computer has to be used if this robust MPC were to be implemented in real time.

6. Conclusion

A sub-optimal interpolating control scheme for uncertain linear systems using ellipsoidal invariant sets was presented. The method extends the SIC scheme of Mercader et al. (2018), and guarantees robust stability, and recursive feasibility. Two variants were presented: the first aims to minimize the interpolating coefficient using a sub-optimal linear search, whereas the second attempts the solution from Mercader et al. (2018) first. Both algorithms can be extended to deal with persistent disturbances as showed in Section 4. Numerical simulations show no particular advantage for the second aforementioned method (Algorithm 2) over SIC. The first method (Algorithm 1) provides fastest convergence and guaranteed stability, however it is more complicated for implementation compared to SIC. A comparison between SIC, polyhedral based IC, and MPC is found in Komarovsky and Haddad (2019). A comparison between the algorithms in this paper, polyhedral based IC, and robust MPC is under preparation. The extension of the proposed algorithms to handle reference tracking is due in future research.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Daniel Rubin: Conceptualization, Methodology, Software, Validation, Data curation, Writing - original draft, Visualization. **Pedro Mercader:** Conceptualization, Methodology, Software, Writing - review & editing. **Per-Olof Gutman:** Conceptualization, Supervision, Writing - review & editing. **Hoai-Nam Nguyen:** Conceptualization, Supervision, Methodology, Writing - review & editing. **Alberto Bemporad:** Conceptualization.

Acknowledgments

The work of the first author is supported at the Technion by grant 2022812 from General Motors – Technical Center Israel. The work of the second author is supported in part at the Technion by a fellowship from the Lady Davis Foundation, Israel.

References

Alamo, T., Cepeda, A., Limon, D., & Camacho, E. F. (2006). Estimation of the domain of attraction for saturated discrete-time systems. *International Journal of Systems Science*, *37*(8), 575–583. http://dx.doi.org/10.1080/00207720600784684.

Bemporad, A., Borrelli, F., & Morari, M. (2003). Min-max control of constrained uncertain discrete-time linear systems. *IEEE Transactions on Automatic Control*, *48*(9), 1600–1606.

Bemporad, A., Morari, M., Dua, V., & Pistikopoulos, E. N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, *38*(1), 3–20.

Blanchini, F., & Miani, S. (2008). *Systems & control: foundations & applications, Set-theoretic methods in control*. Boston: Birkhauser.

Borrelli, F., Bemporad, A., & Morari, M. (2017). *Predictive control for linear and hybrid systems*. Cambridge University Press.

Boyd, S., Ghaoui, L. E., Feron, E., & Balakrishnan, V. (1994). *SIAM studies in applied mathematics: Vol. 15, Linear matrix inequalities in system and control theory*. Society for Industrial and Applied Mathematics.

Cao, Y. -Y., Lam, J., & Sun, Y. -X. (1998). Static output feedback stabilization: An ILMI approach. *Automatica*, *34*, 1641–1645. http://dx.doi.org/10.1016/S0005-1098(98)80021-6.

Gutman, P. -O., & Cwikel, M. (1986). Admissible sets and feedback control for discrete-time linear dynamical systems with bounded controls and states. *IEEE Transactions on Automatic Control*, *31*(4), 373–376.

Hu, T., & Lin, Z. (2001). *Control systems with actuator saturation: analysis and design* (p. 392). Springer Science + Buisiness Media, http://dx.doi.org/10.1002/rnc.833.

Hu, T., Lin, Z., & Chen, B. M. (2002). Analysis and design for discrete-time linear systems subject to actuator saturation. *Systems & Control Letters*, *45*, 97–112.

Jerez, J. L., Goulart, P. J., Richter, S., Constantinides, G. A., Kerrigan, E. C., & Morari, M. (2014). Embedded online optimization for model predictive control at megahertz rates. *IEEE Transactions on Automatic Control*, *59*(12), 3238–3251.

Khlebnikov, M. V., Polyak, B. T., & Kuntsevich, V. M. (2011). Optimization of linear systems subject to bounded exogenous disturbances: The invariant ellipsoid technique. *Automation and Remote Control*, *72*(11), 2227–2275. http://dx.doi.org/10.1134/S0005117911110026.

Knuth, D. E. (2014). *Sorting and searching: Vol. 3, The art of computer programming* (2nd ed.). (p. 791). Addison–Wesley.

Komarovsky, S., & Haddad, J. (2019). Robust interpolating traffic signal control for uncertain road networks. In *2019 European control conference*. Naples.

Kothare, M. V., Balakrishnan, V., & Morari, M. (1996). Robust constrained model predictive control using linear matrix inequalities. *Automatica*, *32*(10), 1361–1379.

Kouvaritakis, B., Cannon, M., & Rossiter, J. (2002). Who needs QP for linear MPC anyway? *Automatica*, *38*, 879–884.

Kouvaritakis, B., Rossiter, J. A., & Schuurmans, J. (2000). Efficient robust predictive control. *IEEE Transactions on Automatic Control*, *45*(8).

Langson, W., Chryssochoos, I., Raković, S., & Mayne, D. Q. (2004). Robust model predictive control using tubes. *Automatica*, *40*(1), 125–133.

Löfberg, J. (2004). Yalmip: A toolbox for modeling and optimization in Matlab. In *2004 IEEE int. conf. robotics and automation* (pp. 285–289). Taipei, Taiwan.

Maciejowski, J. M. (2002). *Predictive control: with constraints*. Pearson Education.

Mayne, D., Seron, M. M., & Raković, S. (2005). Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, *41*, 219–224. http://dx.doi.org/10.1016/j.automatica.2004.08.019.

Mercader, P., Rubin, D., Nguyen, H. -N., Bemporad, A., & Gutman, P. -O. (2018). Simple interpolating control. *IFAC-PapersOnLine*, *51*(25), 42–47. http://dx.doi.org/10.1016/j.ifacol.2018.11.079, 9th IFAC Symposium on Robust Control Design ROCOND 2018.

Molchanov, A., & Pyatnitskiy, Y. (1989). Criteria of asymptotic stability of differential and difference inclusions encountered in control theory. *Systems & Control Letters*, *13*(1), 59–64. http://dx.doi.org/10.1016/0167-6911(89)90021-2.

Nguyen, H. -N. (2014). *Constrained control of uncertain, time-varying, discrete-time systems*. Springer.

Nguyen, H. -N., & Gutman, P. -O. (2018). On the modified Minkowski functional minimization controller for uncertain. *IFAC Journal of Systems and Control*, *4*, 17–24. http://dx.doi.org/10.1016/j.ifacsc.2018.04.001.

Nguyen, H. -N., Gutman, P. -O., Olaru, S., & Hovd, M. (2013). Implicit improved vertex control for uncertain, time-varying linear discrete-time systems with state and control constraints. *Automatica*, *49*(9), 2754–2759.

Nguyen, H. -N., Olaru, S., Gutman, P. -O., & Hovd, M. (2011). Constrained interpolation-based control for polytopic uncertain systems. In *IEEE conf. decision and control and european control conf.* (pp. 4961–4966). Orlando, FL: IEEE.

Pluymers, B., Rossiter, J., Suykens, J., & Moor, B. D. (2005). Interpolation based MPC for LPV systems using polyhedral invariant sets. In *2005 American control conference*. Portland, OR.

Reinelt, W. (2000). Robust control of a two-mass-spring system subject to its input constraints. In *Proceedings of the 2000 American control conference, Vol. 3* (pp. 1817–1821). http://dx.doi.org/10.1109/ACC.2000.879515.

Rossiter, J., & Ding, Y. (2010). Interpolation methods in model predictive control: an overview. *International Journal of Control*, *83*(2), 297–312.

Rubin, D., Nguyen, H. -N., & Gutman, P. -O. (2018a). Computation of polyhedral positive invariant sets via linear matrix inequalities. In *2018 European control conference* (pp. 2941–2946). Limassol, Cyprus.

Rubin, D., Nguyen, H. -N., & Gutman, P. -O. (2018b). Yet another algorithm for the computation of polyhedral positive invariant sets. In *2018 IEEE conference on control technology and applications* (pp. 698–703). Copenhagen, Denmark.

Shingin, H., & Ohta, Y. (2004). Optimal invariant sets for discrete-time systems: Approximation of reachable sets for bounded inputs. *IFAC Proceedings Volumes*, *37*(11), 389–394. http://dx.doi.org/10.1016/S1474-6670(17)31642-7.

Tahir, F. (2010). Efficient computation of robust positively invariant sets with linear state-feedback gain as a variable of optimization. In *Proceedings of the 2010 7th international conference electrical engineering computing science and automatic control* (pp. 199–204). IEEE.

Tarbouriech, S., Garcia, G., Gomes da Silva, J. M., Jr., & Queinnec, I. (2011). *Stability and stabilization of linear systems with saturating actuators* (p. 430). London: Springer-Verlag, http://dx.doi.org/10.1007/978-0-85729-941-3.

Toh, K. C., Todd, M. J., & Tütüncü, R. H. (1999). SDPT3 – A MATLAB software package for semidefinite programming. *Optimization Methods & Software*, *11*(1–4), 545–581. http://dx.doi.org/10.1080/10556789908805762.

Trodden, P. (2016). A one-step approach to computing a polytopic robust positively invariant set. *IEEE Transactions on Automatic Control*, *61*(12), 4100–4105. http://dx.doi.org/10.1109/TAC.2016.2541300.

Wang, Y., & Boyd, S. (2010). Fast model predictive control using online optimization. *IEEE Transactions on Control Systems Technology*, *18*(2), 267.