Robust Self-triggered Control for Time-varying and Uncertain Constrained Systems via Reachability Analysis ⋆

Yulong Gao [a], Pian Yu [a], Dimos V. Dimarogonas [a], Karl H. Johansson [a], Lihua Xie [b]

[a] *Division of Decision and Control Systems, KTH Royal Institute of Technology, SE-10044 Stockholm, Sweden*

[b] *School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore*

Abstract

This paper develops a robust self-triggered control algorithm for time-varying and uncertain systems with constraints based on reachability analysis. The resulting piecewise constant control inputs achieve communication reduction and guarantee constraint satisfactions. In the particular case when there is no uncertainty, we propose a control design with minimum number of samplings over finite time horizon. Furthermore, when the plant is linear and the constraints are polyhedral, we prove that the previous algorithms can be reformulated as computationally tractable mixed integer linear programs. The method is compared with the robust self-triggered model predictive control in a numerical example and applied to a robot motion planning problem with temporal constraints.

Keywords: Constrained systems, robust control, self-triggered control, reachability analysis

1 Introduction

Control for constrained systems has been extensively studied for over four decades in the literature, see e.g. [11,19]. It is well-known that model predictive control (MPC) [19] is a prime example of such a method. If one restricts the attention to networked control systems (NCS) subject to constraints, how to jointly design a controller and a communication protocol which can efficiently utilize network resources is a more recent challenge. In order to handle this problem, recent research efforts have been devoted to self-triggered control [13], in which the next sampling time is determined in advance by the controller according to the received information.

One intuitive idea is to incorporate the self-triggered control into an MPC framework. Many results have been obtained for deterministic constrained systems [3,14]. A self-triggered MPC scheme is proposed in [3] for constrained linear time-invariant systems and the inter-sampling time is maximized subject to the constraints on the cost function. For systems with uncertainty, some results of robust self-triggered MPC have been reported in [6,7]. In [7], the tube MPC method is utilized to guarantee constraint satisfactions despite the presence of a bounded additive disturbance and the principle of [3] is followed to obtain the next sampling instant. In addition, the effect of the uncertainty is made use of in the design of the self-triggered mechanism.

Despite recent developments, some fundamental issues remain for self-triggered MPC. The incorporation of the self-triggered scheme into MPC does not immediately preserve the conventional recursive feasibility and closed-loop stability of MPC. In order to guarantee these two properties, the price of more computational effort at the sampling instants is paid to satisfy the constraints on the cost function when maximizing the inter-sampling time.

Other than MPC, only a few work address the self-triggered control for constrained systems. For example, the focus of [16,24,25] is on the design of an event-triggered controller when the systems are subject to actuator saturation. One recent work [12] provides a contractive set-based approach to design self-triggered control for linear deterministic constrained systems.

⋆ The work of Yulong Gao, Pian Yu, Dimos V. Dimarogonas, and Karl H. Johansson was supported in part by the Knut and Alice Wallenberg Foundation, the Swedish Foundation for Strategic Research, and the Swedish Research Council. The work of Lihua Xie was supported by the National Natural Science Foundation of China under grant NSFC 61633014 and NSFC 61720106011.

Email addresses: yulongg@kth.se (Yulong Gao), piany@kth.se (Pian Yu), dimos@kth.se (Dimos V. Dimarogonas), kallej@kth.se (Karl H. Johansson), elhxie@ntu.edu.sg (Lihua Xie).

Different from the above results, this paper aims at proposing a robust self-triggered control framework for time-varying and uncertain systems with constraints. To the best of our knowledge, this topic has not been explored up to now and cannot be handled by the previous mentioned methods, such as self-triggered MPC. The main challenges are: (1) how to guarantee recursive feasibility in a time-varying setup by self-triggered control; (2) how to ensure constraint satisfaction for any disturbance realization. In this work, we make full use of reachability analysis to handle these issues. Although reachability has been widely studied [4,18,23], the incorporation of reachability into self-triggered control is novel. One recent work [1] uses reachability-based self-triggered control to design the variable sampling period for sampled-data linear systems. However, neither constraints nor uncertainties are considered in [1].

The use of reachability analysis in this paper provides a geometric interpretation for the self-triggered control from a set theoretical point of view. Available geometry software tools facilitate the implementation of our algorithms. Some practical applications of our algorithm include control of hybrid systems and robot motion planning (see Example 2). The main contributions are summarized below.

- We propose a novel robust self-triggered control algorithm (Algorithm 1) for time-varying and uncertain systems with constraints. Constraint satisfactions and recursive feasibility are shown to be guaranteed based on reachability analysis. We calculate the maximum inter-sampling time by solving the corresponding optimization problem ($\mathcal{P}^1_{[k_i,N]}(x_{k_i})$) only once at each sampling instant, which avoids the repetitive computation required in the self-triggered MPC. In the particular case when there is no uncertainty, we develop a control method with minimum number of samplings over a finite time horizon. This is achieved by solving the optimization problem ($\mathcal{P}^2_{[0,N]}(x_0)$) only once.
- When the plant is linear and the constraints are polyhedral, we prove that all the optimization problems ($\mathcal{P}^1_{[k_i,N]}(x_{k_i}$) and $\mathcal{P}^2_{[0,N]}(x_0)$) can be reformulated as mixed integer linear programming (MILP) problems, which are, in our cases, computationally tractable (Theorems 4.1 and 4.2). The numerical comparisons (Example 1) show that our algorithm achieves a better communication reduction and faster online computation than the robust self-triggered MPC in [6] without much loss in performance.

The remainder of the paper is organized as follows. The problem statement is given in Section 2. Section 3 presents the robust self-triggered control algorithm. The specialization to linear plants with polyhedral constraints is provided in Section 4. Two examples in Section 5 illustrate the effectiveness of our approach. Finally, Section 6 concludes this paper.

Notation. Let $\mathbb{N}$ be the set of nonnegative integers. For some $q, s \in \mathbb{N}$ and $q < s$, let $\mathbb{N}_{[q,s]}$ denote the set $\{r \in \mathbb{N} \mid q \le r \le s\}$, respectively. When $\le, \ge, <$, and $>$, are applied to vectors, they are interpreted element-wise. A column vector of ones with appropriate dimension is denoted by $\mathbf{1}$. The Minkowski sum of two sets is denoted by $\mathbb{A} \oplus \mathbb{B} = \{a + b \mid \forall a \in \mathbb{A}, \forall b \in \mathbb{B}\}$. The Minkowski difference of two sets is denoted by $\mathbb{A} \ominus \mathbb{B} = \{c \mid \forall b \in \mathbb{B}, c + b \in \mathbb{A}\}$. For a vector $x \in \mathbb{R}^n$, define $\|x\|_\infty = \max_i |x_i|$. For a polyhedron $\mathbb{P} = \{x \in \mathbb{R}^n \mid Px \le p\}$, define $\|\mathbb{P}\|_\infty = \max_{x\in\mathbb{P}}\{\|Px - p\|_\infty\}$. For $\mathbb{X} \subseteq \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$, define $A^{-1}\mathbb{X} = \{x \in \mathbb{R}^n \mid Ax \in \mathbb{X}\}$. For $x_l \in \mathbb{R}^n, l \in \mathbb{N}$, define $\sum_{l=k}^{j} x_l = \mathbf{0}$ if $k > j$. For $\mathbb{X}_l \subseteq \mathbb{R}^n$ and $A_l \in \mathbb{R}^{n\times n}$, $l \in \mathbb{N}$, define

$$\bigoplus_{l=k}^{j} \mathbb{X}_l = \begin{cases} \emptyset, \ k > j, \\ \mathbb{X}_k \oplus \mathbb{X}_{k+1} \oplus \dots \mathbb{X}_j, \ k \le j, \end{cases}$$
$$\prod_{l=k}^{j} A_l = \begin{cases} I, \ k > j, \\ A_j A_{j-1} \dots A_k, \ k \le j. \end{cases}$$

Given two sets $\mathbb{X}$ and $\tilde{\mathbb{X}}$, define

$$\mathbb{1}_{\mathbb{X}}(x) = \begin{cases} 1, \ x \in \mathbb{X}, \\ 0, \ x \notin \mathbb{X}, \end{cases} \quad \text{and } \mathbb{1}_{\mathbb{X}}(\tilde{\mathbb{X}}) = \begin{cases} 1, \ \tilde{\mathbb{X}} \subseteq \mathbb{X}, \\ 0, \ \tilde{\mathbb{X}} \not\subseteq \mathbb{X}. \end{cases}$$

2 Problem Statement

Consider a discrete-time dynamic control system

$$x_{k+1} = f_k(x_k, u_k) + w_k, \tag{1}$$

where $x_k \in \mathbb{R}^{n_x}$ and $u_k \in \mathbb{R}^{n_u}$, $w_k \in \mathbb{R}^{n_x}$, and $f_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$. The control input u_k at time k is constrained by a set $\mathbb{U}_k \subset \mathbb{R}^{n_u}$. The additive disturbance w_k at time instant k belongs to a compact set $\mathbb{W}_k \subset \mathbb{R}^{n_x}$. The initial state x_0 is contained in a given set $\mathbb{X}_0 \subset \mathbb{R}^{n_x}$. In addition, given a finite time horizon $N \in \mathbb{N}$, the system (1) is subject to a target tube, denoted by $\{(\mathbb{X}_k, k), k \in \mathbb{N}_{[1,N]}\}$, where $\mathbb{X}_k \subseteq \mathbb{R}^{n_x}$, $\forall k \in \mathbb{N}_{[1,N]}$. It is assumed that the function f_k and the disturbance set $\mathbb{W}_k$ are known for all $k \in \mathbb{N}_{[0,N-1]}$.

Assumption 2.1 *The function $f_k(x,u)$, $\forall k \in \mathbb{N}_{[0,N-1]}$, is continuous in x and u, respectively.*

Assumption 2.2 *The sets $\mathbb{U}_k$, $\forall k \in \mathbb{N}_{[0,N-1]}$, and $\mathbb{X}_k$, $\forall k \in \mathbb{N}_{[0,N]}$, are compact.*

The objective of this paper is to develop a self-triggered control algorithm for the system (1), thereby yielding a sequence of piecewise constant control inputs. More specifically, we aim to determine a sequence of sampling instants $\{k_0, k_1, \dots, k_T\}$ with $k_0 = 0$, $k_{i+1} = k_i + M_i$, and $k_T = N$ such that $u_j = u_l$, $\forall j, l \in \mathbb{N}_{[k_i,k_{i+1}-1]}$ and all the constraints are satisfied at each time instant

$k \in \mathbb{N}_{[0,N]}$. Here, $T+1$ is the total number of samplings within N time instants, which quantifies the communication consumption, and M_i denotes the inter-sampling time between k_i and k_{i+1}.

3 Self-triggered Control for Constrained Systems via Reachability Analysis

In this section, we will provide a reachability-based self-triggered control algorithm for the uncertain constrained system (1). Furthermore, a control method with minimum number of samplings will be designed when the system (1) is reduced to be deterministic, i.e., $\mathbb{W}_k = \{0\}$.

3.1 Robust self-triggered control

3.1.1 Computation of reachable sets

Definition 3.1 *The target tube $\{(\mathbb{X}_k, k), k \in \mathbb{N}_{[1,N]}\}$ of the system (1) is reachable from the initial state $x_0 \in \mathbb{X}_0$ if there exists a sequence of control inputs $u_k \in \mathbb{U}_k$, $\forall k \in \mathbb{N}_{[0,N-1]}$, such that the state $x_k \in \mathbb{X}_k$, $\forall k \in \mathbb{N}_{[1,N]}$, for all possible disturbance sequences $w_k \in \mathbb{W}_k$, $\forall k \in \mathbb{N}_{[0,N-1]}$.*

Let $\mathbb{X}_N^* = \mathbb{X}_N$. For $k \in \mathbb{N}_{[0,N-1]}$, the backward reachable set $\mathbb{X}_k^*$ for the system (1) is recursively computed by:

$$\mathbb{P}_k = \{z \in \mathbb{R}^{n_x} \mid \exists u_k \in \mathbb{U}_k, f_k(z, u_k) \oplus \mathbb{W}_k \subseteq \mathbb{X}_{k+1}^*\}, \tag{2a}$$

$$\mathbb{X}_k^* = \mathbb{P}_k \cap \mathbb{X}_k. \tag{2b}$$

Proposition 3.1 *[4] The target tube $\{(\mathbb{X}_j, j), j \in \mathbb{N}_{[k+1,N]}\}$ of the system (1) is reachable from x_k if and only if $x_k \in \mathbb{P}_k$. Furthermore, the target tube $\{(\mathbb{X}_k, k), k \in \mathbb{N}_{[1,N]}\}$ is reachable from the initial state $x_0 \in \mathbb{X}_0$ if and only if $x_0 \in \mathbb{X}_0^*$.*

According to [23], Assumptions 2.1 and 2.2 make the resulting reachable sets compact. Proposition 3.1 indicates that if the state $x_j \in \mathbb{X}_j^*$, the recursive feasibility and the constraint satisfactions can be guaranteed.

Remark 3.1 *There exist some methods to compute the reachable sets for a nonlinear system (1), e.g., [8,23]. In addition, there are results on the inner approximations of the reachable sets $\mathbb{X}_k^*$, e.g., [2,22]. Note that these inner approximations are applicable also for the following algorithms, since they provide constraint satisfaction and recursive feasibility guarantees.*

Remark 3.2 *Given the initial state x_0, one can choose the minimal horizon N such that $\{(\mathbb{X}_j, j), j \in \mathbb{N}_{[0,N]}\}$ is reachable from x_0.*

3.1.2 Algorithm

Define the self-triggered condition for the system (1) as

$$\begin{aligned} k_{i+1} = \max\{k \mid\ & k_i < k \leq N \text{ such that } u_j = u \in \mathbb{U}_j, \\ & j \in \mathbb{N}_{[k_i,k-1]}, \text{ and the target tube} \\ & \{(\mathbb{X}_j, j), j \in \mathbb{N}_{[k_i,N]}\} \text{ of (1) is reachable}\}. \end{aligned}$$

Recall that $k_{i+1} = k_i + M_i$. The following lemma provides the formulation to compute M_i.

Proposition 3.2 *Given the state $x_{k_i} \in \mathbb{X}_{k_i}^*$, $k_i \in \mathbb{N}_{[0,N-1]}$, the inter-sampling time M_i is obtained by solving the following optimization problem, denoted by $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$:*

$$\max_u \sum_{j=k_i+1}^{N} r_j \tag{3a}$$

subject to

$$\tilde{\mathbb{X}}_{[k_i,k_i]} = \{x_{k_i}\}, \tag{3b}$$

$$\forall j \in \mathbb{N}_{[k_i,N-1]} : \tilde{\mathbb{X}}_{[k_i,j+1]} = f_j(\tilde{\mathbb{X}}_{[k_i,j]}, u) \oplus \mathbb{W}_j, \tag{3c}$$

$$\forall j \in \mathbb{N}_{[k_i+1,N]} :$$

$$r_j = \begin{cases} \mathbb{1}_{\mathbb{X}^*_{k_i+1}}(\tilde{\mathbb{X}}_{[k_i,k_i+1]})\mathbb{1}_{\mathbb{U}_{k_i}}(u), & j = k_i+1, \\ r_{j-1}\mathbb{1}_{\mathbb{X}^*_j}(\tilde{\mathbb{X}}_{[k_i,j]})\mathbb{1}_{\mathbb{U}_{j-1}}(u), & j > k_i+1, \end{cases} \tag{3d}$$

where $f_j(\mathbb{X}, u) = \{z \in \mathbb{R}^{n_x} \mid z = f(x, u), \forall x \in \mathbb{X}\}$. That is, $M_i = \sum_{j=k_i+1}^{N} r_j^*$, where r_j^* corresponds to the optimal solution of $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$.

Proof. The definition of r_j characterizes the successive constraint satisfactions from time k_i for all possible disturbances $w_l \in \mathbb{W}_l, \forall l \in \mathbb{N}_{[k_i,j-1]}$. Then, the proof directly follows from Proposition 3.1 and the objective function of $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$.

The geometric interpretation of the optimization problem $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$ is to seek a fixed control input u such that starting from time k_i, the time length, during which the state constraints and the control input constraints are satisfied for all possible disturbances, is maximized.

We denote by u^* the optimal solution to the optimization problem $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$. The following Algorithm 1 presents the robust self-triggered control for the uncertain constrained system (1).

3.2 Control with minimum number of samplings

This subsection will provide a control method with minimum number of samplings, denoted by T^*+1, over a given finite horizon for the system (1). Without loss of generality, we assume that $N \geq 2$.

Algorithm 1 Robust self-triggered control

Offline:

Determine a sequence of backward reachable sets $\{(\mathbb{X}_j^*, j), j \in \mathbb{N}_{[0,N]}\}$ by (2).

Online:

1: Initialize $i = 0$. If $x_0 \in \mathbb{X}_0^*$, continue. Else, stop.
2: Sample the state x_{k_i}, solve $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$ to obtain u^* and M_i.
3: Set $k_{i+1} = k_i + M_i$. Implement $u_j = u^*$, $\forall j \in \mathbb{N}_{[k_i,k_{i+1}-1]}$ to system (1) for some realizations $w_j \in \mathbb{W}_j$, $j \in \mathbb{N}_{[k_i,k_{i+1}-1]}$.
4: Set $i = i + 1$.
5: If $k_i < N$, go to step 2. Else, stop.

Proposition 3.3 *The minimum number of samplings $T^* + 1$ is obtained by solving the following optimization problem, denoted by $\mathcal{P}^2_{[0,N]}(x_0)$,*

$$\min_{u_0,\Delta_j} \sum_{j=0}^{N-2} (1 - \mathbb{1}_{\{0\}}(\Delta_j)) \tag{4a}$$

subject to

$$\forall j \in \mathbb{N}_{[0,N-1]} : x_{j+1} = f_j(x_j, u_j), \tag{4b}$$

$$\forall j \in \mathbb{N}_{[0,N-1]} : u_j = \begin{cases} u_0, \ j = 0 \\ u_{j-1} + \Delta_{j-1}, \quad j \geq 1, \end{cases} \tag{4c}$$

$$\forall j \in \mathbb{N}_{[1,N]} : x_j \in \mathbb{X}_j^*, \tag{4d}$$

$$\forall j \in \mathbb{N}_{[0,N-1]} : u_j \in \mathbb{U}_j. \tag{4e}$$

That is, $T^* = \sum_{j=0}^{N-2}(1 - \mathbb{1}_{\{0\}}(\Delta_j^*))$, where Δ_j^* corresponds to the optimal solution of $\mathcal{P}^2_{[0,N]}(x_0)$.

Proof. In (4c), Δ_{j-1} denotes the difference between u_j and u_{j-1}. The objective function of $\mathcal{P}^2_{[0,N]}(x_0)$ aims at maximizing the number of zero (i.e., $\Delta_j = 0$) over the time interval $\mathbb{N}_{[0,N-1]}$. Thus, the optimal solution generates a sequence of piecewise constant control inputs with minimum number of switching times.

Note that Algorithm 1 is applicable for deterministic systems. In this case, the difference is that Algorithm 1 cannot guarantee the achievement of minimum number of samplings for deterministic systems.

Remark 3.3 *For uncertain constrained systems, it is difficult to design the control with minimum number of samplings since at each sampling instant k_i, the exact executions of the future disturbance w_j, $\forall j \in \mathbb{N}_{[k_i,N-1]}$, are unknown.*

4 Self-triggered Control for Linear Systems with Polyhedral Constraints

The development of geometry software allows us to compute the sets $\mathbb{X}_k^*$ exactly and efficiently if the system is linear and the constraint sets are polyhedral [15]. This section will specialize the systems (1) to be linear and the constraint sets to be polyhedral. We can reformulate the optimization problems $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$ and $\mathcal{P}^2_{[0,N]}(x_0)$ to be computationally tractable MILP problems.

If the model f_k is linear, the system (1) becomes

$$x_{k+1} = A_k x_k + B_k u_k + w_k. \tag{5}$$

Here A_k and B_k are deterministic real matrices with appropriate dimensions at each time $k \in \mathbb{N}_{[0,N-1]}$. The control input sets $\mathbb{U}_k$, $k \in \mathbb{N}_{[0,N-1]}$, are compact polyhedra. Each set $\mathbb{X}_k$ of the target tube $\{(\mathbb{X}_k, k), k \in \mathbb{N}_{[1,N]}\}$ is a compact polyhedron. The disturbance sets $\mathbb{W}_k$, $k \in \mathbb{N}_{[0,N-1]}$, are compact polyhedra.

Now, the computation of the sets $\mathbb{X}_k^*$ in (2) is given as follows. Note that the following equations involve only set operations and the corresponding sets can be well-defined even if the matrix A_k is not invertible. Hence, we do not impose any assumption on A_k.

Lemma 4.1 *[4] For the uncertain linear system (5) with polyhedral constraints, the set $\mathbb{X}_k^*$ in (2) evolves as*

$$\mathbb{Q}_k = \mathbb{X}_k^* \ominus \mathbb{W}_k, \tag{6a}$$

$$\mathbb{P}_k = A_k^{-1}(\mathbb{Q}_{k+1} \oplus (-B_k \mathbb{U}_k)), \tag{6b}$$

$$\mathbb{X}_k^* = \mathbb{P}_k \cap \mathbb{X}_k, \ \mathbb{X}_N^* = \mathbb{X}_N. \tag{6c}$$

Remark 4.1 *Since the sets $\mathbb{X}_k$, $k \in \mathbb{N}_{[0,N]}$, are compact, the sets $\mathbb{X}_k^*$, $k \in \mathbb{N}_{[0,N]}$, are also compact even when the matrices A_k are not invertible.*

The polyhedral sets $\mathbb{U}_k$ and $\mathbb{X}_k^*$ in (6) are written as

$$\mathbb{U}_k = \{z \in \mathbb{R}^{n_u} \mid E_k z \leq e_k\},$$
$$\mathbb{X}_k^* = \{z \in \mathbb{R}^{n_x} \mid F_k z \leq f_k\},$$

where E_k and F_k (or e_k and f_k) are matrices (or vectors) with appropriate dimensions.

4.1 Robust self-triggered control

Before providing the main result, we need some preliminary lemmas.

Lemma 4.2 *The set $\tilde{\mathbb{X}}_{[k_i,j]}$ in (3c) can be written as*

$$\tilde{\mathbb{X}}_{[k_i,j]} = (G_{[k_i,j]} x_k + H_{[k_i,j]} u) \oplus \mathbb{Z}_{[k_i,j]}, \ j \in \mathbb{N}_{[k_i,N]}, \tag{7}$$

where $G_{[k_i,j]} = \prod_{l=k_i}^{j-1} A_l$, $H_{[k_i,j]} = \sum_{m=k_i}^{j-1} \prod_{l=m+1}^{j-1} A_l B_m$, $\mathbb{Z}_{[k_i,j]} = \bigoplus_{m=k_i}^{j-1} \prod_{l=m+1}^{j-1} A_l \mathbb{W}_m$. Furthermore, the set $\mathbb{Z}_{[k_i,j]}$ is a closed polyhedron.

Proof. When $j = k_i$, (7) implies that $\tilde{\mathbb{X}}_{[k_i,k_i]} = \{x_{k_i}\}$. According to the definition of $\tilde{\mathbb{X}}_{[k_i,j]}$, $j \in \mathbb{N}_{[k_i+1,N]}$, in (3c), by induction, it follows

$$
\begin{aligned}
&\tilde{\mathbb{X}}_{[k_i,j+1]} \\
&= A_j\tilde{\mathbb{X}}_{[k_i,j]} \oplus B_j u \oplus \mathbb{W}_j \\
&= A_j(G_{[k_i,j]}x_{k_i} + H_{[k_i,j]}u) \oplus \mathbb{Z}_{[k_i,j]}) \oplus B_j u \oplus \mathbb{W}_j \\
&= (A_j \prod_{l=k_i}^{j-1} A_l x_{k_i} + (A_j \sum_{m=k_i}^{j-1} \prod_{l=m+1}^{j-1} A_l B_m + B_j)u) \\
&\quad \oplus (A_j \bigoplus_{m=k_i}^{j-1} \prod_{l=m+1}^{j-1} A_l \mathbb{W}_m \oplus \mathbb{W}_j) \\
&= (\prod_{l=k_i}^{j} A_l x_{k_i} + \sum_{m=k_i}^{j} \prod_{l=m+1}^{j} A_l B_m) \oplus (\bigoplus_{m=k_i}^{j} \prod_{l=m+1}^{j} A_l \mathbb{W}_m) \\
&= (G_{[k_i,j+1]}x_{k_i} + H_{[k_i,j+1]}u) \oplus \mathbb{Z}_{[k_i,j+1]}.
\end{aligned}
$$

Since the sets $\mathbb{W}_m$ are compact polyhedra, we have that the sets $\mathbb{Z}_{[k_i,j]}$ are closed polyhedra.

Lemma 4.3 *[5] Given two polyhedra $\mathbb{P} = \{z \in \mathbb{R}^n \mid Pz \le p\}$ and $\mathbb{Q} = \{z \in \mathbb{R}^n \mid Qz \le q\}$, $\mathbb{P} \subseteq \mathbb{Q}$ holds if and only if there exists a non-negative matrix S such that $SP = Q$ and $Sp \le q$.*

Assume now that $\mathbb{Z}_{[k_i,j]} = \{z \in \mathbb{R}^n \mid V_{[k_i,j]}z \le v_{[k_i,j]}\}$, $j \in \mathbb{N}_{[k_i+1,N]}$, where the matrix $V_{[k_i,j]}$ and vector $v_{[k_i,j]}$ can be computed offline according to $\mathbb{Z}_{[k_i,j]} = \bigoplus_{m=k_i}^{j-1} \prod_{l=m+1}^{j-1} A_l \mathbb{W}_m$. By Lemma 4.3, we derive the following result.

Lemma 4.4 *For the sets $\tilde{\mathbb{X}}_{[k_i,j]}$ and $\mathbb{X}_j^*$, $\tilde{\mathbb{X}}_{[k_i,j]} \subseteq \mathbb{X}_j^*$ holds if and only if there exists a non-negative matrix $S_{[k_i,j]}$ such that*

$$
S_{[k_i,j]}V_{[k_i,j]} = F_j, \tag{8}
$$
$$
S_{[k_i,j]}(v_{[k_i,j]} + V_{[k_i,j]}(G_{[k_i,j]}x_{k_i} + H_{[k_i,j]}u)) \le f_j. \tag{9}
$$

Proof. This directly follows from Lemmas 4.2–4.3.

Since u is the decision variable, the constraints (9) are nonlinear. To remedy this, we can calculate the nonnegative matrices $S_{[k_i,j]}$ offline to satisfy (8) by an LP [10]:

$$
(S_{[k_i,j]})_l = \underset{a^T}{\operatorname{argmin}}\{\mathbf{1}^T a \mid a^T V_{[k_i,j]} = (F_j)_l, a \ge \mathbf{0}\}, \tag{10}
$$

where a is a vector with appropriate dimension and $(S)_l$ denotes the lth row of the matrix S.

Remark 4.2 *The LP in (10) admits a nonnegative matrix solution with minimum infinity norm, which could lead to a larger feasible region for the optimization problem $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$ than other nonnegative solutions to (8).*

The next theorem shows that the robust self-triggered control for the system (5) with polyhedral constraints can be designed by solving a computationally tractable MILP.

Theorem 4.1 *For the uncertain linear system (5) with polyhedral constraints, the problem $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$ can be reformulated as an MILP, denoted by $\mathcal{P}^3_{[k_i,N]}(x_{k_i})$,*

$$
\max_{u,\delta_j} \sum_{j=k_i+1}^{N} (1-\delta_j) \tag{11a}
$$
subject to
$$
\begin{aligned}
&\forall j \in \mathbb{N}_{[k_i+1,N]}: \\
&\quad S_{[k_i,j]}(\tilde{G}_{[k_i,j]}x_{k_i} + \tilde{H}_{[k_i,j]}u) \le \tilde{f}_{[k_i,j]} + \delta_j\Gamma\mathbf{1}, &(11b)\\
&\forall j \in \mathbb{N}_{[k_i,N-1]}: E_j u \le e_j + \delta_j \Gamma \mathbf{1}, &(11c)\\
&\forall j \in \mathbb{N}_{[k_i+1,N-1]}: \delta_j \le \delta_{j+1}, &(11d)\\
&\forall j \in \mathbb{N}_{[k_i+1,N]}: \delta_j \in \{0,1\}, &(11e)
\end{aligned}
$$

where $\tilde{G}_{[k_i,j]} = V_{[k_i,j]}G_{[k_i,j]}$, $\tilde{H}_{[k_i,j]} = V_{[k_i,j]}H_{[k_i,j]}$, $\tilde{f}_{[k_i,j]} = f_j - S_{[k_i,j]}v_{[k_i,j]}$, and Γ is a positive constant satisfying

$$
\begin{aligned}
&\Gamma > \max\{\max_{j\in\mathbb{N}_{[k_i,N-1]}} \|\mathbb{U}_j\|_\infty, \\
&\max_{j\in\mathbb{N}_{[k_i+1,N]}} \max_{u\in\mathbb{U}_j} \|S_{[k_i,j]}(\tilde{G}_{[k_i,j]}x_{k_i} + \tilde{H}_{[k_i,j]}u) - \tilde{f}_{[k_i,j]}\|_\infty\}.
\end{aligned} \tag{12}
$$

Proof. Recall the definition of r_j in (3d). Let us introduce a sequence of 0-1 variables δ_j, $j \in \mathbb{N}_{[k_i+1,N]}$. By setting $r_j = 1-\delta_j$, we have

$$
\begin{cases}
\forall l \in \mathbb{N}_{[k_i+1,j]}: \\
\quad S_{[k_i,l]}(\tilde{G}_{[k_i,l]}x_{k_i} + \tilde{H}_{[k_i,l]}u) \le \tilde{f}_{[k_i,l]} + \delta_l\Gamma\mathbf{1}, \\
\forall l \in \mathbb{N}_{[k_i,j-1]}: E_l u \le e_l + \delta_l\Gamma\mathbf{1},
\end{cases}
$$

where Γ is a positive number satisfying (12). Furthermore, $\forall j \in \mathbb{N}_{[k_i+1,N-1]}$, $r_j \ge r_{j+1}$ can be rewritten as $\delta_j \le \delta_{j+1}$. Then, we get the problem $\mathcal{P}^2_{[k_i,N]}(x_{k_i})$.

Remark 4.3 *The computational complexity of the MILP (11) can be analyzed as follows. Considering the constraint (11d) on δ_j, it is easy to construct an enumeration tree with at most $N-k_i$ nodes, each of which is in the form of $\delta_j = 0, \forall j \in \mathbb{N}_{[k_i+1,k]}$, and $\delta_j = 1$, $\forall j \in \mathbb{N}_{[k+1,N]}$, for some $k \in \mathbb{N}_{[k_i+1,N]}$. This construction avoids the combinatorial explosion issue. By fixing the enumeration sequence $\{\delta_j\}_{k_i+1}^N$, the constraints (11b)–(11c) become linear in the decision variable u and in particular are redundant for $j \in \mathbb{N}_{[k+1,N]}$. Hence, the MILP (11) can be solved through several LPs by increasing k until the LP is infeasible. The computational complexity of the original MILP (11) is at most $N-k_i$ times the computational complexity of the LP. Note that the number of constraints in the LP is at most the total number of the inequalities in (11b)–(11c) over $j \in \mathbb{N}_{[0,N]}$, denoted by L. From [21], we conclude that the problem (11) can be solved in $\mathcal{O}(NL)$ time. In addition, several software tools have been developed to solve large MILPs in the past few years, e.g., [17], also allowing us to solve our problem online efficiently.*

Remark 4.4 *Following similar operations as for the linear case, the previous optimization problem* $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$ *can be reformulated as an integer program with a constraint like (11d). The resulting integer program can be iteratively cast as a classic constrained robust nonlinear control problem.*

Remark 4.5 *In general,* Γ *can be arbitrarily chosen to be a sufficiently large positive constant. The lower bound on* Γ *defined in (12) aims at quantifying how large* Γ *should be, which can be calculated by solving a linear program (LP) since the sets* $\mathbb{U}_j$ *are compact polyhedra and the norm is the* $\ell_\infty-$*norm.*

4.2 Control with minimum number of samplings

When the disturbance set $\mathbb{W}_k = \{0\}$, $\forall k \in \mathbb{N}_{0,N-1}$, we can also reformulate the optimization problem $\mathcal{P}^2_{[0,N]}(x_0)$ as a computationally tractable MILP.

Theorem 4.2 *For the deterministic linear system (5) with polyhedral constraints, the problem* $\mathcal{P}^2_{[0,N]}(x_0)$ *can be reformulated as an MILP, denoted by* $\mathcal{P}^4_{[0,N]}(x_0)$*,*

$$\max_{u_0,\Delta_j,\delta_j,c_j} J_{[0,N]}(x_0) = \sum_{j=0}^{N-2}(1-\delta_j)$$

subject to

$$\forall j \in \mathbb{N}_{[0,N-1]} : x_{j+1} = A_j x_j + B_j u_j,$$
$$\forall j \in \mathbb{N}_{[0,N-1]} : u_j = \begin{cases} u_0, \; j=0 \\ u_{j-1} + \Delta_{j-1}, \quad j \geq 1, \end{cases}$$
$$\forall j \in \mathbb{N}_{[1,N]} : F_j x_j \leq f_j,$$
$$\forall j \in \mathbb{N}_{[0,N-1]} : E_j u_j \leq e_j,$$
$$\forall j \in \mathbb{N}_{[0,N-2]} : \begin{cases} \Delta_j \leq \Gamma \delta_j \mathbf{1}, \\ \Delta_j \geq -\gamma \delta_j \mathbf{1}, \\ \Delta_j \leq c_j + \gamma(1-\delta_j)\mathbf{1}, \\ \Delta_j \geq c_j - \Gamma(1-\delta_j)\mathbf{1}, \end{cases}$$
$$\forall j \in \mathbb{N}_{[0,N-2]} : -\gamma \mathbf{1} \leq c_j \leq \Gamma \mathbf{1},$$
$$\forall j \in \mathbb{N}_{[0,N-2]} : \delta_j \in \{0,1\},$$

where γ *and* Γ *are two large positive constants satisfying*

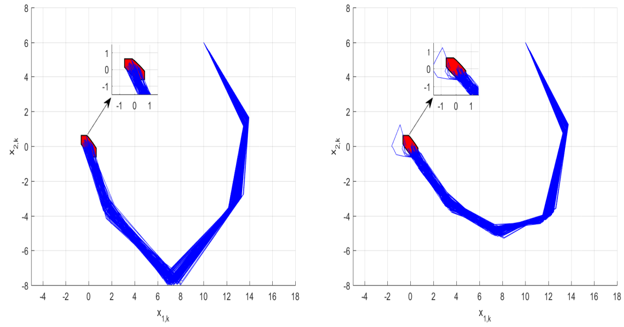
$$\gamma, \Gamma > \max_{j \in \mathbb{N}_{[0,N-2]}} \max_{u \in \mathbb{U}_j, v \in \mathbb{U}_{j+1}} \|u - v\|_\infty. \tag{13}$$

Proof. By introducing new variables $c_j \in \mathbb{R}^{n_u}$ and 0-1 variables δ_j, $j \in \mathbb{N}_{[k+2,N]}$, we define $\Delta_j = \delta_j c_j$, i.e., $\Delta_j = 0$ if $\delta_j = 0$ and $\Delta_j = c_j$ if $\delta_j = 1$. And it follows that $\forall j \in \mathbb{N}_{[k+1,N]}$:,

$$\Delta_j = \delta_j c_j \Leftrightarrow \begin{cases} \Delta_j \leq \delta_j \Gamma \mathbf{1}, \\ \Delta_j \geq \delta_j \gamma \mathbf{1}, \\ \Delta_j \leq c_j + (1-\delta_j)\gamma \mathbf{1}, \\ \Delta_j \geq c_j - (1-\delta_j)\Gamma \mathbf{1}, \\ -\gamma \mathbf{1} \leq c_j \leq \Gamma \mathbf{1}, \end{cases}$$

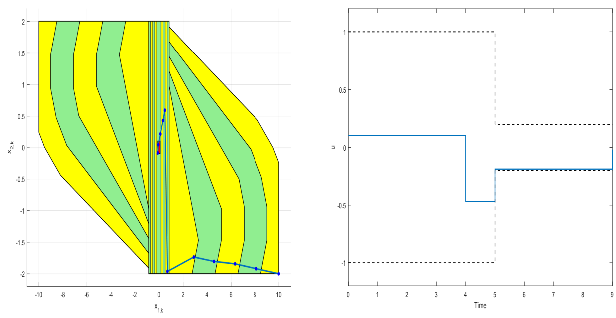
where Γ and γ are two positive numbers satisfying (13). Then, we get the optimization problem $\mathcal{P}^4_{[0,N]}(x_0)$.

Remark 4.6 *The statements in Remark 4.5 also apply with* γ *and* Γ *in (13). In addition, the optimization problem* $\mathcal{P}^4_{[0,N]}(x_0)$ *is equivalent to* $\mathcal{P}^2_{[0,N]}(x_0)$*, while the optimal solution to* $\mathcal{P}^3_{[k_i,N]}(x_{k_i})$ *is a suboptimal solution to* $\mathcal{P}^1_{[k_i,N]}(x_{k_i})$*.*



(a) (b)

Fig. 1. State trajectories under Algorithm 1 and robust self-triggered MPC [6] for 100 realizations of the uncertainty sequence. (a) Algorithm 1; (b) Robust self-triggered MPC. The algorithms terminate when the state enters the robust invariant set (the red region).



(a) State (b) Control input

Fig. 2. State trajectory and control input trajectory under Algorithm 1 for one realization of the uncertainty sequence.

5 Examples

This section provides three examples to illustrate the effectiveness of our proposed algorithms. The following numerical experiments were run in Matlab R2016a with MPT toolbox [15] on a Dell laptop with Window 7, Intel i7-6600U CPU 2.80GHz and 16.0 GB RAM.

Example 1: Compare the proposed robust self-triggered algorithm with the robust self-triggered MPC in [6]. Consider the same model as in [6], where $A = [1\ 1; 0\ 1]$, $B = [0.5\ 0]^T$. The constraint sets are $\mathbb{X} = \{z \in \mathbb{R}^2 \mid [-20\ \ -8]^T \leq z \leq [20\ 8]^T\}$, $\mathbb{U} = \{z \in \mathbb{R} \mid -4.5 \leq z \leq 4.5\}$, $\mathbb{W} = \{z \in \mathbb{R}^2 \mid [-0.25\ \ -0.25]^T \leq z \leq [0.25\ 0.25]^T\}$. The initial state is $[10\ 6]^T$.

Let $Q = [1\ 0; 0\ 1]$ and $R = 0.1$ be the weight matrices in the objective function. By solving the discrete-time algebraic Riccati equation, we obtain the matrix $P = [2.0599\ 0.5916; 0.5916\ 1.4228]$ and the corresponding optimal feedback gain $K = [-0.6167\ \ -1.2703]$. The control objective is to steer the state to the robust invariant set, denoted by Ω (the red region in Fig. 1), which is computed by the method in [20]. The implementation will stop if the state enters the robust invariant set.

For the robust self-triggered control in this paper, the terminal set of the target tube is Ω. For the robust self-triggered MPC in [6], we choose the maximal inter-sampling time $M_{\max} = 4$. The state trajectories for 100 realizations of the uncertainty sequence are depicted in Fig. 1. We compare the two different methods for several indexes, of which the average is taken over 500 realizations.

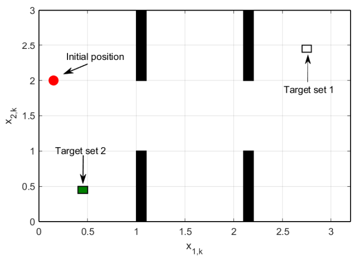


Fig. 3. Scenario

- *Average inter-sampling time*: The average inter-sampling time is $\bar{M} = 1.2045$ under the robust self-triggered MPC of [6] while it is $\bar{M} = 1.3333$ under the self-triggered scheme of this paper. Thus, our control scheme achieves an average communication reduction by 9.66% more than that of the self-triggered MPC.
- *Average online computation time*: The average online computation time at each sampling instant is $0.5758s$ for the robust self-triggered MPC while it is $0.1964s$ for our control method despite the presence of integer variables.
- *Average performance*: The performance measure is defined by $J = \sum_{k=0}^{T_{run}} (\|x_k\|_Q^2 + \|u_k\|_R^2)$ where T_{run} is the time when the state enters the robust invariant set. The robust self-triggered MPC achieves a slightly better average performance than our scheme. The performance measure is 591.9191 for MPC while it is 621.4552 for our scheme.

Example 2: Consider a time-varying linear system (5), where $A_k = [1\ 1; 0\ 1]$, $B_k = [1; 0.5]$, $k \in \mathbb{N}_{[0,5]}$ and $A_k = [1\ 0; 1\ 0]$, $B_k = [1; 0.5]$, $k \in \mathbb{N}_{[6,10]}$. Note that A_k, $k \in \mathbb{N}_{[6,10]}$, are singular. Let $N = 10$. The input constraints are $|u_k| \le 1$, $k \in \mathbb{N}_{[0,5]}$, and $|u_k| \le 0.2$, $k \in \mathbb{N}_{[6,9]}$. The terminal constraint set is $\mathbb{X}_N = \{z \in \mathbb{R}^2 \mid [-0.01; -0.01] \le z \le [0.01; 0.01]\}$. And the state constraint set is $\mathbb{X}_k = \{z \in \mathbb{R}^2 \mid [-10; -2] \le z \le [10; 2]\}$, $k \in \mathbb{N}_{[0,9]}$. The disturbance set is $\mathbb{W}_k = \{z \in \mathbb{R}^2 \mid [-0.03; -0.03] \le z \le [0.03; 0.03]\}$, $k \in \mathbb{N}_{[0,10]}$.

Algorithm 1 is implemented to the system with the initial state $[10\ \ -2]^T$. Fig. 2 depicts the state trajectory and the control input trajectory under Algorithm 2 for one realization of the disturbance sequence. In subfigure (a), the red square is the target set and the yellow and lightgreen regions are the sets $\mathbb{X}_k^*$, $k \in \mathbb{N}_{[0,9]}$. The black dotted lines in subfigure (b) are the constraint bounds on the control input. From subfigure (b), 3 updates for control inputs are needed to tolerate the disturbances as well as guarantee the constraints satisfaction.

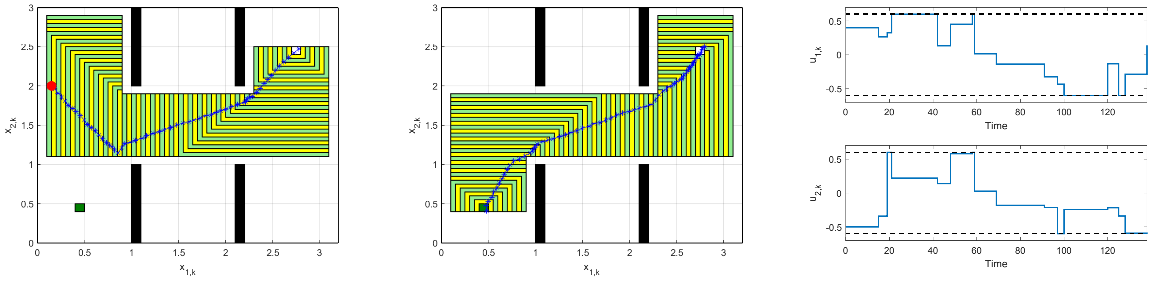
Example 3: Consider a mobile robot with dynamics (5), where $A_k = [1\ 0; 0\ 1]$, $B_k = [0.1\ 0; 0\ 0.1]$, $\forall k \in \mathbb{N}$. The input constraint set is $\mathbb{U}_k = \{z \in \mathbb{R}^2 \mid [-0.6\ \ -0.6]^T \le z \le [0.6\ 0.6]^T\}$, $k \in \mathbb{N}$. The robot moves in a closed workspace, as shown in Fig. 4, in which there are some obstacles (the black rectangles). The robot should achieve collision avoidance with the obstacles and the boundaries of the workspace. We set the safe distance as 0.1. In addition, the robot can exchange the information (the position and the control input) with the control center via a bandwidth-limited communication network. At each time, the robot can only receive one control input from the control center. The initial position is $[0.15\ \ 2]^T$. The target set 1 is $\{z \in \mathbb{R}^2 \mid [2.7\ 2.4]^T \le z \le [2.8\ 2.5]^T\}$ and the target set 2 is $\{z \in \mathbb{R}^2 \mid [0.4\ 0.4]^T \le z \le [0.5\ 0.5]^T\}$. A sequence of temporal constrained tasks for the mobile robot are

- stage 1: the robot should arrive at the target set 1 before $k = 60$;
- stage 2: the robot should stay in target set 1 for at least 10 time steps after arrival.
- stage 3: the robot should arrive at the target set 2 before $k = 140$.

To save the communication recourses, our self-triggered control strategies are implemented. We choose the convex inner approximations of the backward reachable sets (which are the intersection between the predecessor sets and the safe regions). As mentioned in Remark 3.1, these approximations still respect the feasibility and the constraint satisfactions.

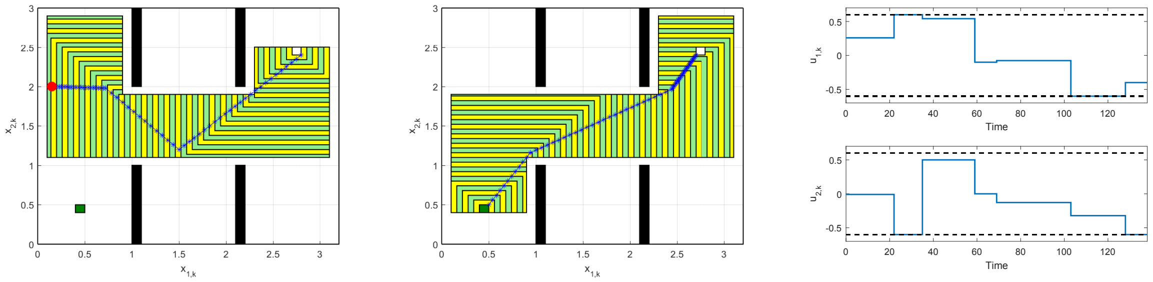
In the first case, assume that the disturbance set is $\mathbb{W}_k = \{z \in \mathbb{R}^2 \mid [-0.01\ \ -0.01]^T \le z \le [0.01\ 0.01]^T\}$. Subfigures (a)–(b) of Fig. 4 depict the state trajectories for stage 1 and 3 under Algorithm 1. The yellow and lightgreen regions are the sets $\mathbb{X}_k^*$. The control input trajectory is shown in subfigure (c) of Fig. 4 with the times of control update being 15.

In the second case, assume that $\mathbb{W}_k = [0\ 0]^T$, $\forall k \in \mathbb{N}$. Subfigures (a)–(b) of Fig. 5 depict the state trajectories for stage 1 and 3 by using the control with minimum number of samplings. The yellow and lightgreen regions



(a) State of stage 1 (b) State of stage 3 (c) Control input

Fig. 4. State trajectories of stage 1 and 3 and control input trajectories under Algorithm 1.



(a) State of stage 1 (b) State of stage 3 (c) Control input

Fig. 5. State trajectories of stage 1 and 3 and control input trajectories by using control with minimum number of samplings.

are the convex approximations of the sets $\mathbb{X}_k^*$. The control input trajectory is shown in subfigure (c) of Fig. 5 with the times of control update being 6.

6 Conclusion

In this paper, we proposed a robust self-triggered control algorithm for time-varying uncertain systems with constraints. By using reachability analysis, the constraint satisfactions and recursive feasibility were guaranteed. The proposed algorithm provided us a geometric interpretation for self-triggered control. The problem of control with minimum number of samplings was investigated for deterministic constrained systems. For linear systems with polyhedral constraints, the proposed methods were reformulated as computationally tractable MILP problems. In simulations, the results were compared with robust self-triggered MPC and applied to the robot motion planning.

References

[1] Al Khatib, M., Girard, A., & Dang, T. (2017). Self-triggered control for sampled-data systems using reachability analysis. In *Proceedings of 20th IFAC World Congress* (pp. 7881-7886).

[2] Althoff, M., & Krogh, B. H. (2014). Reachability analysis of nonlinear differential-algebraic systems. *IEEE Transactions on Automatic Control*, 59(2), 371-383.

[3] Barradas Berglind, J. D. J., Gommans, T. M. P., & Heemels, W. P. M. H. (2012). Self-triggered MPC for constrained linear systems and quadratic costs. In *Proceedings of IFAC Conference on Nonlinear Model Predictive Control* (pp. 342-348).

[4] Bertsekas, D.P., & Rhodes, I.B. (1971). On the minimax reachability of target sets and target tubes. *Automatica*, 7(2), 233-247.

[5] Blanchini, F., & Miani, S. (2007). *Set-theoretic methods in control*. Boston: Springer.

[6] Brunner, F. D., Heemels, W. P. M. H., & Allgöwer, F. (2014). Robust self-triggered MPC for constrained linear systems. In *Proceedings of European Control Conference* (pp. 472-477).

[7] Brunner, F. D., Heemels, W. P. M. H., & Allgöwer, F. (2016). Robust self-triggered MPC for constrained linear systems: A tube-based approach. *Automatica*, 72, 73-83.

[8] Chen, M., Herbert, S. L., Vashishtha, M., Bansal, S., & Tomlin, C. J. (2018). Decomposition of reachable sets and tubes for a class of nonlinear systems. *IEEE Transactions on Automatic Control*, to appear.

[9] Dai, L., Gao, Y. L., Xie, L. H., Johansson, K. H., & Xia, Y. Q. (2018) Stochastic self-triggered model predictive control for linear systems with probabilistic constraints. *Automatica*, 92, 9-17.

[10] Fleming, J., Kouvaritakis, B., & Cannon, M. (2013). Regions of attraction and recursive feasibility in robust MPC. In *Proceedings of the 21st Mediterranean Conference on Control and Automation* (pp. 801-806).

[11] Gutman, P.O., & Hagander, P. (1985). A new design of constrained controllers for linear systems. *IEEE Transactions on Automatic Control*, 30(1), 22-33.

[12] Hashimoto, K., Adachi, S., & Dimarogonas, D.V. (2018). Aperiodic sampled-data control via explicit transmission mapping: a set invariance approach. *IEEE Transactions on Automatic Control*, to appear.

[13] Heemels, W. P. M. H., Johansson, K. H., & Tabuada, P. (2012). An introduction to event-triggered and self-triggered control. In *Proceedings of IEEE Conference on Decision and Control* (pp. 3270-3285).

[14] Henriksson, E., Quevedo, D. E., Sandberg, H., & Johansson, K. H. (2012). Self-triggered model predictive control for network scheduling and control. In *Proceedings of IFAC Symposium on Advanced Control of Chemical Processes* (pp. 432-438).

[15] Herceg, M., Kvasnica, M., Jones, C. N., & M. Morari. (2013). Multi-parametric toolbox 3.0. In *Proceedings of European Control Conference* (pp. 502-510).

[16] Lehmann, D., Kiener, G.A., & Johansson, K.H. (2012). Event-triggered PI control: Saturating actuators and anti-windup compensation. In *Proceedings of IEEE Conference on Decision and Control* (pp. 6566-6571).

[17] Linderoth, J. T., & Lodi, A. (2010). MILP software. Wiley Encyclopedia of Operations Research and Management Science.

[18] Lygeros, J., Tomlin, C., & Sastry, S. (1999). Controllers for reachability specifications for hybrid systems. *Automatica*, 35(3), 349-370.

[19] Mayne, D.Q., Rawlings, J.B., Rao, C.V., & Scokaert, P.O. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36(6), 789-814.

[20] Mayne, D. Q., Seron, M. M., & Raković, S.V. (2005). Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2), 219-224.

[21] Megiddo, N. (1984). Linear programming in linear time when the dimension is fixed. *Journal of the ACM*, 31(1), 114-127.

[22] Mitchell, I. M. (2011). Scalable calculation of reach sets and tubes for nonlinear systems with terminal integrators: a mixed implicit explicit formulation. In *Proceedings of the 14th ACM International Conference on Hybrid Systems: Computation and Control* (pp. 103-112).

[23] Raković, S.V., Kerrigan, E.C., Mayne, D.Q. & Lygeros, J. (2006). Reachability analysis of discrete-time systems with disturbances. *IEEE Transactions on Automatic Control*, 51(4), 546-561.

[24] Seuret, A., Prieur, C., Tarbouriech, S., & Zaccarian, L. (2013). Event-triggered control with LQ optimality guarantees for saturated linear systems. In *Proceedings of IFAC Symposium on Nonlinear Control Systems* (pp. 341-346).

[25] Wu, W., Reimann, S., & Liu, S. (2014). Event-triggered control for linear systems subject to actuator saturation. In *Proceedings of IFAC World Congress* (pp. 9492-9497).