

Fast Self-Triggered MPC for Constrained Linear Systems with Additive Disturbances

Li Dai, Mark Cannon, Fuwen Yang, *Senior Member, IEEE*, Shuhao Yan

Abstract—This paper proposes a robust self-triggered model predictive control (MPC) algorithm for a class of constrained linear systems subject to bounded additive disturbances, in which the inter-sampling time is determined by a fast convergence self-triggered mechanism. The main idea of the self-triggered mechanism is to select a sampling interval so that a rapid decrease in the predicted costs associated with optimal predicted control inputs is guaranteed. This allows for a reduction in the required computation without compromising performance. By using a constraint tightening technique and exploring the nature of the open-loop control between sampling instants, a set of minimally conservative constraints is imposed on nominal states to ensure robust constraint satisfaction. A multi-step open-loop MPC optimization problem is formulated, which ensures recursive feasibility for all possible realisations of the disturbance. The closed-loop system is guaranteed to satisfy a mean-square stability condition. To further reduce the computational load, when states reach a predetermined neighbourhood of the origin, the control law of the robust self-triggered MPC algorithm switches to a self-triggered local controller. A compact set in the state space is shown to be robustly asymptotically stabilized. Numerical comparisons are provided to demonstrate the effectiveness of the proposed strategies.

Index Terms—Model predictive control, self-triggered control, fast convergence, robustness.

I. INTRODUCTION

MODEL predictive control (MPC) has been developed and successfully applied to a large number of applications with slow dynamics, such as chemical engineering [1], financial engineering [2] and supply chain management [3], due to its ability to effectively handle constraints and to provide desirable performance. However, as advanced control strategies for constrained optimal control problems, MPC algorithms, for computational reasons, are in general not suitable for fast dynamic systems such as aircraft and electromechanical processes. This is because an MPC optimization problem with constraints on inputs and states has to be solved online at each time step, which makes it difficult to meet real-time computing requirements for systems with small sampling intervals. Hence the computational burden of the standard periodically-triggered MPC strategy should be reduced to cater for such systems. A considerable reduction of overall computation can be achieved by sampling and

updating control inputs in an aperiodically-triggered fashion, while the computational complexity of the optimization problems formulated at sampling instants is similar to that of standard MPC problems. Aperiodic implementation implies that systems will be controlled in open loop between sampling instants. As a result, the trade off between the reduction of the number of control updates and desirable control performance has become a new challenge for MPC of fast dynamic systems, especially if the system to be controlled is subject to unknown disturbances.

Two main types of aperiodically-triggered schemes, event-triggered and self-triggered control, have attracted much attention in recent decades (see [4], [5]). The difference between them is that the former requires continuous monitoring of system states to decide when a control task should be triggered, while in the latter the next sampling instant is predetermined at the current time based on only the current state measurement. Self-triggered MPC strategies have been investigated in [6], [7] for unconstrained and undisturbed systems, in [8] for undisturbed systems with constraints, and in [9], [10] for disturbed systems without constraints on states or inputs.

In this paper, we focus on discrete-time linear systems where bounded additive disturbances and constraints on states and inputs are taken into account. The aim is to achieve a reduction in the time-average computation, while guaranteeing real-time operation and rapid convergence. To this end, we propose a self-triggered control strategy based on robust MPC (RMPC). Other results on RMPC-based self-triggered control schemes can for example be found in [11], [12], [13], [14], [15]. In [11], a robust tube MPC approach [16] is used to prevent constraint violations, despite the effect of additive disturbances. The inter-sampling time is maximized subject to constraints on the cost functions of a set of optimization problems parameterized by the length of the inter-sampling time. The feasible optimization problem with the longest inter-sampling time is selected, and control inputs are obtained by solving it. Based on homothetic Tube MPC [17], an alternative self-triggered MPC scheme is devised in [12] to stabilize systems with disturbances. Instead of the fixed form of cost functions in [11], the inter-sampling time is included in the cost function as a decision variable. At each sampling instant, the maximal permissible inter-sampling time and the optimal control input sequence are obtained by solving multiple optimization problems until a feasibility condition ceases to hold. Similarly, the authors of [13] also treat the inter-sampling time as a part of the cost function. The self-triggered MPC algorithm in [13] for

L. Dai is with School of Automation, Beijing Institute of Technology, Beijing 100081, P. R. China. E-mail: li.dai@bit.edu.cn. M. Cannon and S. Yan are with Department of Engineering Science, University of Oxford, Parks Road, Oxford, OX1 3PJ, United Kingdom. E-mail: mark.cannon@eng.ox.ac.uk, shuhao.yan@eng.ox.ac.uk. F. Yang is with School of Engineering, Griffith University, Gold Coast, QLD 4222, Australia. E-mail: fuwen.yang@griffith.edu.au.

disturbed systems combines the self-triggered mechanism in [12] with the constraint tightening technique for robustness in [18]. Although the ideas to determine the maximal length of the inter-sampling time in [11], [12], [13] are appealing, one has to note that these control schemes require much more computation at each sampling instant than standard MPC with periodic sampling. Therefore, they are impracticable for fast dynamic systems.

The main contribution of this paper is to address the issue of excessive computation at sampling instants in [11], [12], [13] by proposing a novel robust MPC algorithm with a fast convergence self-triggered mechanism. Inspired by [14], [15], the sampling interval is determined as the maximum period over which a bound on the predicted cost corresponding to the optimal predicted control inputs is decreasing. However, unlike [14], [15], we minimize this bound (i.e. maximize the reduction in cost) to determine an inter-sampling time. Although this may lead to a shorter inter-sampling time compared with [14], [15], the algorithm in this paper can provide a reduction in the number of time steps for states to converge, and therefore achieve a reduction in the time-average number of optimization problems to be solved. More importantly, to reduce the potential closed-loop performance loss due to open-loop control between sampling instants, an infinite-horizon problem is considered at each sampling instant instead of a finite-horizon one as in [14], [15]. Moreover, we apply a closed-loop control paradigm to self-triggered MPC, which has the advantage of reducing the conservativeness in constraint handling over the open-loop MPC employed in [14], [15]. The setup of this paper addresses new challenges in the formulation of tractable infinite-horizon MPC optimization problems with recursive feasibility. In particular, decisions relating to the length of the inter-sampling period affect the predicted control strategies that could be employed at a future time step. To address this problem, we analyse the propagation of the effects of disturbances along predicted input/state trajectories by considering the fact that measurements will be obtained at an unknown future sampling instant. The paper's approach constructs a set of minimally conservative constraints that ensure robust constraint satisfaction and recursive feasibility for any allowable inter-sampling time. After states converge to a terminal region, a local state feedback controller is applied in [14] to stabilize the system. In contrast to [14], we propose a local self-triggered feedback controller and a corresponding self-triggered mechanism such that system states converge to a robustly positively invariant (RPI) set. The RPI set is parameterized in terms of a scalar parameter which enables us to balance the size of this set with the average inter-sampling time.

Compared with previous work on RMPC-based self-triggered control [11], [12], [13], [14], [15], three key features of the self-triggered MPC algorithms proposed in this paper are that (i) the optimization problem formulated at sampling instants has similar computational complexity to that of a periodically-triggered MPC algorithm; (ii) recursive feasibility over an infinite horizon can be ensured in the self-triggered framework; (iii) the new algorithms have a relatively low susceptibility to adverse effects of delays

in computation. These features make the algorithms more suitable for constrained systems with limited computational resources and short sampling intervals, and they can therefore be used as an enabler of MPC on fast dynamic systems. We show that the proposed algorithms constitute a significant advance in self-triggered MPC algorithms by providing an order of magnitude reduction in computation without an appreciable affect on performance.

The remainder of this paper is organized as follows. The problem formulation is given in Section II. In Section III, a multi-step open-loop MPC optimization problem with a finite number of deterministic constraints is formulated. In Section IV, a self-triggered mechanism is designed and a robust self-triggered MPC algorithm is proposed. The properties of this algorithm are developed in Section V. In Section VI, a robust self-triggered MPC with a local controller is designed to further reduce computation. Section VII contains some discussions. A numerical example is provided in Section VIII to highlight the merits of the proposed algorithms in comparison with a periodically-triggered RMPC and [13]. We conclude in Section IX.

Notation: Let \mathbb{N} , \mathbb{R}^n , and $\mathbb{R}^{m \times n}$ be the set of non-negative integers, n -dimensional Euclidean space, and the set of $m \times n$ real matrices, respectively. Let I be the identity matrix with appropriate dimensions and $\mathbf{0}$ be a vector or a matrix with zero entries. For $q, s \in \mathbb{N}$ and $q < s$, let $\mathbb{N}_{< q}$, $\mathbb{N}_{\leq q}$, $\mathbb{N}_{> q}$, $\mathbb{N}_{\geq q}$, and $\mathbb{N}_{[q, s]}$ be the sets $\{r \in \mathbb{N} \mid r < q\}$, $\{r \in \mathbb{N} \mid r \leq q\}$, $\{r \in \mathbb{N} \mid r > q\}$, $\{r \in \mathbb{N} \mid r \geq q\}$, and $\{r \in \mathbb{N} \mid q \leq r \leq s\}$, respectively. Given two sets $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}^n$ and a matrix $A \in \mathbb{R}^{m \times n}$, $A\mathcal{X} := \{Ax \mid x \in \mathcal{X}\}$. The Minkowski set addition is defined by $\mathcal{X} \oplus \mathcal{Y} := \{x + y \mid x \in \mathcal{X}, y \in \mathcal{Y}\}$ and the Minkowski (Pontryagin) set difference by $\mathcal{X} \ominus \mathcal{Y} := \{z \in \mathbb{R}^n \mid \{z\} \oplus \mathcal{Y} \subseteq \mathcal{X}\}$. Given a sequence of sets $\mathcal{X}_i \subseteq \mathbb{R}^n$ for $i \in \mathbb{N}_{[a, b]}$ with $a, b \in \mathbb{N}$, define $\bigoplus_{i=a}^b \mathcal{X}_i := \left\{ \sum_{i=a}^b x_i \mid x_i \in \mathcal{X}_i \right\}$. By convention, the set is $\{\mathbf{0}\}$ if $a > b$. For a matrix $W \in \mathbb{R}^{n \times n}$, let $W \succ 0$ denote that W is a positive definite matrix and $\|W\|$ denote its 2-norm. Given a matrix $W \succ 0$ and a vector $x \in \mathbb{R}^n$, $\|x\|$ denotes the Euclidean norm and we define $\|x\|_W^2 := x^T W x$. The operators $\lambda_{\min}(\cdot)$ and $\lambda_{\max}(\cdot)$ denote the smallest and the largest eigenvalues of a matrix, and $x(t_k + i|t_k)$ denotes the i -step-ahead predicted value of x at time t_k .

II. PROBLEM SETUP

Consider the following discrete-time linear time-invariant system with additive disturbances

$$x(t+1) = Ax(t) + Bu(t) + w(t), \quad t \in \mathbb{N}, \quad (1)$$

where $x(t) \in \mathbb{R}^{N_x}$, $u(t) \in \mathbb{R}^{N_u}$ are the system state and the control input respectively. The disturbance input $w(t) \in \mathcal{W} \subset \mathbb{R}^{N_x}$ is unknown but bounded. The set \mathcal{W} is a compact convex polytope containing the origin in its interior and the disturbances $w(t)$ for $t \in \mathbb{N}$ are independent. The pair (A, B) is assumed to be stabilizable. The system (1) is subject to constraints

$$(x(t), u(t)) \in \Xi, \quad t \in \mathbb{N}, \quad (2)$$

where Ξ is a compact convex polytope. For convenience of illustration, only one of the constraints defining (2) is considered in the sequel, which is given by

$$Fx(t) + Gu(t) \leq \ell, \quad t \in \mathbb{N}, \quad (3)$$

where $F \in \mathbb{R}^{1 \times N_x}$, $G \in \mathbb{R}^{1 \times N_u}$ are row vectors and $\ell \in \mathbb{R}$ is a constant. For the other constraints in (2), the same constraint handling can be performed separately.

Our aim is to construct an MPC algorithm, which can robustly stabilize the system without violating constraints, while achieving a low frequency of control updates without much loss in convergence rate of the system state. To achieve such objectives, we combine an RMPC framework with a fast convergence self-triggered condition. Under this condition, system states can be driven into a neighbourhood of the origin in a short time, without the need to solve an MPC optimization problem at every time step.

In particular, let $t_k, k \in \mathbb{N}$, be the sampling instant when an MPC optimisation problem is to be solved, and let

$$t_{k+1} = t_k + M(x(t_k)), \quad k \in \mathbb{N},$$

where $t_0 = 0$ and $M(\cdot) : \mathbb{R}^{N_x} \rightarrow \mathbb{N}_{[1,N]}$ is a state-dependent sampling function with $N \in \mathbb{N}_{\geq 1}$. For simplicity, we use M_k instead of $M(x(t_k))$ in the sequel. We decompose the predicted actual state $x(t_k + i|t_k)$ into a nominal part and an uncertain part, denoted by $z(t_k + i|t_k)$ and $e(t_k + i|t_k)$ respectively, that is $x(t_k + i|t_k) = z(t_k + i|t_k) + e(t_k + i|t_k)$. The predicted control input sequence at time t_k , $\{u(t_k + i|t_k), i \in \mathbb{N}\}$, is formulated as

$$u(t_k + i|t_k) = \begin{cases} Kz(t_k + i|t_k) + c(t_k + i|t_k), & i \in \mathbb{N}_{\leq N-1} \\ Kx(t_k + i|t_k), & i \in \mathbb{N}_{\geq N} \end{cases} \quad (4)$$

where $c(t_k + i|t_k)$ is the decision variable, feedback gain K is fixed and designed as the unconstrained LQ-optimal for the quadratic cost defined in Section III-D, and the predicted nominal state $z(t_k + i|t_k)$ evolves as

$$z(t_k + i + 1|t_k) = \Phi z(t_k + i|t_k) + Bc(t_k + i|t_k), \quad i \in \mathbb{N}, \quad (5)$$

with the initial condition $z(t_k|t_k) = z(t_k) = x(t_k)$, $\Phi := A + BK$, and $c(t_k + i|t_k) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$. At any sampling instant t_k , for a given system state $x(t_k)$, the inter-sampling time M_k and a sequence of control inputs $\{u(t_k + i|t_k), i \in \mathbb{N}_{\leq N-1}\}$ are determined online. The first M_k elements of the control sequence are then applied at the time steps $t_k, t_k + 1, \dots, t_k + M_k - 1$. At the next sampling instant t_{k+1} , this process is repeated using the knowledge of the system state $x(t_{k+1})$.

Remark 2.1: The input parameterisation $u(t) = Kx(t) + c(t)$ is widely used in RMPC problems and it is convenient to construct an augmented autonomous MPC formulation [19]. A requirement is that the system state is measurable at every time step. However, in the self-triggered framework, the first few elements of a predicted control input sequence are applied in an open-loop fashion and during that time interval state measurements may not be available. To make the

control inputs deterministic and implementable in the open-loop phase, the predicted actual states $x(t_k + i|t_k)$ are replaced in the predicted control law (4) by their predicted nominal counterparts $z(t_k + i|t_k)$ for $i \in \mathbb{N}_{\leq N-1}$. For all prediction time steps $i \in \mathbb{N}_{\geq N}$, we set $c(t_k + i|t_k) = \mathbf{0}$ so that the fixed feedback law $u(t_k + i|t_k) = Kx(t_k + i|t_k)$ is adopted and hence the number of decision variables is finite.

III. MULTIPLE-STEP OPEN-LOOP MPC OPTIMIZATION FORMULATION

In this section, an infinite-horizon MPC optimization problem is developed to define the control input sequence that is applied to the system (1) between sampling instants. The main challenge when formulating the MPC optimization in the self-triggered setup is that the predicted control inputs applied in an open-loop fashion are required to ensure robust constraint satisfaction for the closed-loop system. Furthermore, since MPC optimizations are solved in an aperiodic way, another challenge is to make the constraints in the MPC optimization recursively feasible for all possible future inter-sampling times. To remedy these, in Section III-A, by applying elements of the input sequence obtained at each sampling instant in an open-loop fashion, we give conditions to ensure constraint satisfaction over an infinite prediction horizon. In Section III-B, these constraints are further modified to guarantee recursive feasibility by considering that feedback is present at a certain time in the future. A finite number of constraints, which is equivalent to the constraints derived in Section III-B, is obtained in Section III-C and the multi-step open-loop optimization used in the self-triggered MPC scheme is formulated in Section III-D.

A. Constraint Satisfaction by Predictions

For a given system state $x(t_k)$ at sampling instant t_k , the constraints on predicted states and control inputs are given by

$$Fx(t_k + i|t_k) + Gu(t_k + i|t_k) \leq \ell, \quad i \in \mathbb{N}. \quad (6)$$

The predicted dynamics (5) can be described equivalently by the augmented autonomous state space model

$$p(t_k + i + 1|t_k) = \Psi p(t_k + i|t_k), \quad i \in \mathbb{N},$$

where

$$p(t_k|t_k) := \begin{bmatrix} x(t_k) \\ \mathbf{c}(t_k) \end{bmatrix}, \quad p(t_k + i|t_k) := \begin{bmatrix} z(t_k + i|t_k) \\ T^i \mathbf{c}(t_k) \end{bmatrix},$$

$$\mathbf{c}(t_k) := [c^T(t_k|t_k) \ c^T(t_k + 1|t_k) \ \dots \ c^T(t_k + N - 1|t_k)]^T,$$

$$\Psi := \begin{bmatrix} \Phi & BE \\ \mathbf{0} & T \end{bmatrix}, \quad E := [I \ \mathbf{0} \ \dots \ \mathbf{0}], \quad T := \begin{bmatrix} \mathbf{0} & I & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & I \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

so that $T\mathbf{c}(t_k) := [c^T(t_k + 1|t_k) \ \dots \ c^T(t_k + N - 1|t_k) \ \mathbf{0}]^T$. In this description, $z(t_k + i|t_k) = [I \ \mathbf{0}] \Psi^i p(t_k|t_k)$ and (6) is then transformed into another form

$$H\Psi^i p(t_k|t_k) \leq \begin{cases} \ell - Fe(t_k + i|t_k), & i \in \mathbb{N}_{\leq N-1} \\ \ell - (F + GK)e(t_k + i|t_k), & i \in \mathbb{N}_{\geq N} \end{cases} \quad (7)$$

with

$$H := [F \quad G] \begin{bmatrix} I & \mathbf{0} \\ K & E \end{bmatrix}.$$

The uncertain component $e(t_k + i|t_k)$ in (7) evolves for $i \in \mathbb{N}$ according to

$$e(t_k + i + 1|t_k) = \begin{cases} Ae(t_k + i|t_k) + w(t_k + i|t_k), & i \in \mathbb{N}_{\leq N-1} \\ \Phi e(t_k + i|t_k) + w(t_k + i|t_k), & i \in \mathbb{N}_{\geq N} \end{cases} \quad (8)$$

with $e(t_k|t_k) = e(t_k) = \mathbf{0}$. Let the set of all possible realizations of $e(t_k + i|t_k)$ be

$$\mathcal{R}_i := \begin{cases} A\mathcal{R}_{i-1} \oplus \mathcal{W}, & i \in \mathbb{N}_{[1,N]} \\ \Phi\mathcal{R}_{i-1} \oplus \mathcal{W}, & i \in \mathbb{N}_{\geq N+1} \end{cases}, \quad \mathcal{R}_0 := \{\mathbf{0}\},$$

which can be rewritten as

$$\mathcal{R}_i = \begin{cases} \bigoplus_{j=0}^{i-1} A^j \mathcal{W}, & i \in \mathbb{N}_{[1,N]} \\ \bigoplus_{j=0}^{i-N-1} \Phi^j \mathcal{W} \oplus \Phi^{i-N} \mathcal{R}_N, & i \in \mathbb{N}_{\geq N+1} \end{cases}, \quad \mathcal{R}_0 = \{\mathbf{0}\}.$$

Conditions on $p(t_k|t_k)$ to ensure robust satisfaction of (6) under the predicted control law (4) are stated as follows.

Lemma 3.1: At any sampling instant t_k , $k \in \mathbb{N}$, and for a given state $x(t_k)$, the constraint (6) is satisfied for all possible disturbances $w \in \mathcal{W}$ if and only if decision variable $\mathbf{c}(t_k)$ satisfies the constraint

$$H\Psi^i p(t_k|t_k) \leq \ell - h_i, \quad i \in \mathbb{N}, \quad (9)$$

where the sequence $\{h_i, i \in \mathbb{N}\}$ is defined as

$$h_i := \begin{cases} \max_{e \in \mathcal{R}_i} Fe, & i \in \mathbb{N}_{\leq N-1} \\ \max_{e \in \mathcal{R}_i} (F + GK)e, & i \in \mathbb{N}_{\geq N} \end{cases}$$

$$= \begin{cases} \sum_{j=0}^{i-1} \max_{w \in \mathcal{W}} F A^j w, & i \in \mathbb{N}_{\leq N-1} \\ \sum_{j=0}^{i-N-1} \max_{w \in \mathcal{W}} (F + GK) \Phi^j w \\ + \sum_{j=0}^{N-1} \max_{w \in \mathcal{W}} (F + GK) \Phi^{i-N} A^j w, & i \in \mathbb{N}_{\geq N}. \end{cases}$$

Proof: From (7), by considering $e(t_k + i|t_k) \in \mathcal{R}_i$, necessary and sufficient conditions for the satisfaction of (6) for all possible future disturbance sequences are given by the tightened constraint (9) on $p(t_k|t_k)$. ■

The maximization defining h_i is performed by solving a set of linear programming (LP) problems. These LP problems are independent of the information available at time t_k , and hence the values of h_i can be computed offline.

B. Recursively Feasible Constraints

Although the constraint in Lemma 3.1 is derived by considering worst-case values of the uncertain components of $Fx(t_k + i|t_k) + Gu(t_k + i|t_k)$ over all admissible disturbances, it cannot ensure the existence of a feasible $\mathbf{c}(t_{k+1})$ at time t_{k+1} . This is because two different controllers are applied in different stages of the infinite prediction horizon as shown in Fig. 1. Specifically, constraint (9) ensures the satisfaction

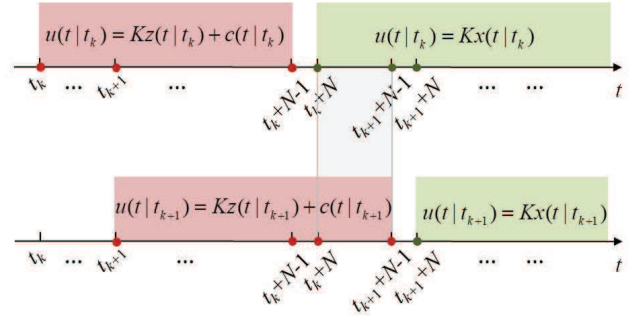


Fig. 1. Illustration of predicted input sequences at time t_k (above) and t_{k+1} (below). The red and green shading illustrates time steps at which different predicted control strategies are employed.

of $Fx(t_k + i|t_k) + Gu(t_k + i|t_k) \leq \ell$ at time t_k for $i \in \mathbb{N}_{[N, M_k+N-1]}$ under the controller $u(t_k + i|t_k) = Kx(t_k + i|t_k)$, whereas for the same prediction time steps, the constraint $Fx(t_k + i|t_{k+1}) + Gu(t_k + i|t_{k+1}) \leq \ell$ at time t_{k+1} is required to be satisfied under the controller $u(t_k + i|t_{k+1}) = Kz(t_k + i|t_{k+1}) + c(t_k + i|t_{k+1})$. To ensure that the ‘tail’ of $\mathbf{c}(t_k)$, namely $T^{M_k} \mathbf{c}(t_k)$, is a feasible choice for $\mathbf{c}(t_{k+1})$, constraint (9) needs to be further modified.

Theorem 3.1: At sampling instant t_k , if the constraint (9) is replaced with

$$H\Psi^i p(t_k|t_k) \leq \ell - g_i, \quad i \in \mathbb{N}, \quad (10)$$

where the sequence $\{g_i, i \in \mathbb{N}\}$ is defined by $g_0 := 0$ and for $i \in \mathbb{N}_{\geq 1}$

$$g_i := \begin{cases} \max \left\{ h_i, \max_{m \in \mathbb{N}_{[1,i]}} \left\{ \max_{e \in \mathcal{R}_m} (F + GK) \Phi^{i-m} e + g_{i-m} \right\} \right\}, & i \in \mathbb{N}_{[1, N-1]} \\ \max_{m \in \mathbb{N}_{[1, N]}} \left\{ \max_{e \in \mathcal{R}_m} (F + GK) \Phi^{i-m} e + g_{i-m} \right\}, & i \in \mathbb{N}_{\geq N} \end{cases}$$

then:

- (i) the satisfaction of (6) is ensured at sampling instant t_k ;
- (ii) constraint (10) remains feasible at all sampling instants t_{k+1}, t_{k+2}, \dots

Proof: From Lemma 3.1, the result in (i) can be guaranteed if $g_i \geq h_i$ for all $i \in \mathbb{N}$. For $i \in \mathbb{N}_{\leq N-1}$, this is obvious, whereas for $i \in \mathbb{N}_{\geq N}$ we have

$$g_i \geq \max_{e \in \mathcal{R}_N} (F + GK) \Phi^{i-N} e + g_{i-N} \geq h_i, \quad (11)$$

where the last inequality follows from the second equation of the definition of h_i , for $i \in \mathbb{N}_{\geq N}$,

$$h_i = \sum_{j=0}^{i-N-1} \max_{w \in \mathcal{W}} (F + GK) \Phi^j w + \max_{e \in \mathcal{R}_N} (F + GK) \Phi^{i-N} e,$$

and from

$$g_j \geq \max_{w \in \mathcal{W}} (F + GK) \Phi^{j-1} w + g_{j-1} \text{ for all } j \in \mathbb{N}_{\geq 1}. \quad (12)$$

Summing both sides of (12) from $j = 1$ to $j = i - N$ gives, for $i \in \mathbb{N}_{\geq N}$,

$$g_{i-N} \geq \sum_{j=0}^{i-N-1} \max_{w \in \mathcal{W}} (F + GK)\Phi^j w.$$

To show (ii), we introduce a candidate solution $\bar{\mathbf{c}}(t_{k+1}) := T^{M_k} \mathbf{c}(t_k)$ at time t_{k+1} . We will prove that $\bar{\mathbf{c}}(t_{k+1})$ is a feasible solution for (10) at sampling instant t_{k+1} . Denote by $\bar{p}(t_{k+1}) = [x^T(t_{k+1}) \ \bar{\mathbf{c}}^T(t_{k+1})]^T$ the augmented state corresponding to $\bar{\mathbf{c}}(t_{k+1})$. Since, by definition, $x(t_{k+1}) = z(t_{k+1}|t_k) + e(t_{k+1}|t_k)$ for any $M_k \in \mathbb{N}_{[1,N]}$, we have

$$\begin{aligned} \bar{p}(t_{k+1}) &= \begin{bmatrix} z(t_{k+1}|t_k) \\ T^{M_k} \mathbf{c}(t_k) \end{bmatrix} + \begin{bmatrix} e(t_{k+1}|t_k) \\ \mathbf{0} \end{bmatrix} \\ &= p(t_{k+1}|t_k) + \begin{bmatrix} e(t_{k+1}|t_k) \\ \mathbf{0} \end{bmatrix}. \end{aligned}$$

Hence, proving the satisfaction of $H\Psi^i \bar{p}(t_{k+1}) \leq \ell - g_i$, $i \in \mathbb{N}$, is equivalent to proving the satisfaction of the constraint

$$H \left(\Psi^{M_k+i} p(t_k|t_k) + \begin{bmatrix} \Phi^i e(t_{k+1}|t_k) \\ \mathbf{0} \end{bmatrix} \right) \leq \ell - g_i, \quad i \in \mathbb{N}, \quad (13)$$

by taking into account $p(t_{k+1}|t_k) = \Psi^{M_k} p(t_k|t_k)$. Furthermore, due to $e(t_{k+1}|t_k) \in \mathcal{R}_{M_k}$, constraint (13) can be ensured if the following condition is satisfied

$$H\Psi^{M_k+i} p(t_k|t_k) \leq \ell - \max_{e \in \mathcal{R}_{M_k}} (F + GK)\Phi^i e - g_i, \quad i \in \mathbb{N}. \quad (14)$$

From the definition of the sequence $\{g_i, i \in \mathbb{N}\}$, it follows that for $i \in \mathbb{N}$

$$\begin{aligned} g_{M_k+i} &\geq \begin{cases} \max_{m \in \mathbb{N}_{[1, i+M_k]}} \left\{ \max_{e \in \mathcal{R}_m} (F + GK)\Phi^{i+M_k-m} e + g_{i+M_k-m} \right\}, & i \in \mathbb{N}_{\leq N-M_k} \\ \max_{m \in \mathbb{N}_{[1, N]}} \left\{ \max_{e \in \mathcal{R}_m} (F + GK)\Phi^{i+M_k-m} e + g_{i+M_k-m} \right\}, & i \in \mathbb{N}_{\geq N-M_k+1} \end{cases} \\ &\geq \max_{e \in \mathcal{R}_{M_k}} (F + GK)\Phi^i e + g_i. \end{aligned}$$

Therefore the satisfaction of the constraint

$$H\Psi^{M_k+i} p(t_k|t_k) \leq \ell - g_{M_k+i}, \quad i \in \mathbb{N},$$

at sampling instant t_k ensures that (14) is satisfied. Hence there exists at least one feasible solution at sampling instant t_{k+1} such that (10) is satisfied. This implies the feasibility of (10) at all subsequent sampling instants if (10) is feasible at sampling instant t_k . ■

Note that the sequence $\{g_i, i \in \mathbb{N}\}$ is monotonically non-decreasing for $i \in \mathbb{N}_{\leq N-1}$ while not necessarily monotonic for $i \in \mathbb{N}_{\geq N}$. Also, as $i \rightarrow \infty$, there exists a limit for this sequence due to the strict stability of Φ .

C. A Finite Set of Constraints

Although the control input sequence (4) depends on a finite-dimensional decision variable, the number of constraints in (10) is infinite because of the infinite prediction horizon. To reduce this infinite set of constraints to a finite one, we extend the approach proposed in [20] to the current context of aperiodically-triggered MPC. The following result provides a method of determining the minimum number of linear constraints that define a feasible set identical to that of (10).

Theorem 3.2: If $g_i < \ell$ for all $i \in \mathbb{N}$, then the constraints in (10) over the infinite prediction horizon $i \in \mathbb{N}$ are equivalent to a finite number of constraints

$$H\Psi^i p(t_k|t_k) \leq \ell - g_i, \quad i \in \mathbb{N}_{\leq \hat{N}}, \quad (15)$$

where $\hat{N} \in \mathbb{N}_{\geq N}$ is such that $H\Psi^{\hat{N}+i} p(t_k|t_k) \leq \ell - g_{\hat{N}+i}$ holds for all $i \in \mathbb{N}_{[1, N]}$ and all $p(t_k|t_k)$ satisfying (15).

Proof: Let $\mathcal{X}_j := \cap_{i=0}^j \{p(t_k|t_k) \mid H\Psi^i p(t_k|t_k) \leq \ell - g_i\}$, $j \in \mathbb{N}_{\geq N}$. Then $\mathcal{X}_\infty := \lim_{j \rightarrow \infty} \mathcal{X}_j$ denotes the set of all $p(t_k|t_k)$ such that (10) is satisfied for all $i \in \mathbb{N}$. By definition we have $\mathcal{X}_N \supseteq \mathcal{X}_{N+1} \supseteq \mathcal{X}_{N+2} \supseteq \dots \supseteq \mathcal{X}_\infty$. Therefore, the condition of the theorem that $p(t_k|t_k) \in \mathcal{X}_{\hat{N}+i}$ holds for all $i \in \mathbb{N}_{[1, N]}$ if $p(t_k|t_k) \in \mathcal{X}_{\hat{N}}$, is equivalent to $\mathcal{X}_{\hat{N}} = \mathcal{X}_{\hat{N}+N}$. This implies $\mathcal{X}_{\hat{N}} = \mathcal{X}_\infty$, as we demonstrate next. First note that if $p(t_k|t_k) \in \mathcal{X}_{\hat{N}}$ and $\mathcal{X}_{\hat{N}} = \mathcal{X}_{\hat{N}+N}$, then we have $p(t_k|t_k) \in \mathcal{X}_{\hat{N}+N}$ and hence $H\Psi^i p(t_k|t_k) \leq \ell - g_i$ for all $i \in \mathbb{N}_{\leq \hat{N}+N}$. Therefore, for all $m \in \mathbb{N}_{[1, N]}$, it holds that

$$\begin{aligned} H\Psi^j \Psi^m p(t_k|t_k) &\leq \ell - g_{j+m} \\ &\leq \ell - g_j - \max_{e \in \mathcal{R}_m} (F + GK)\Phi^j e \end{aligned}$$

for all $j \in \mathbb{N}_{\leq \hat{N}+N-m}$. Then we have, for all $m \in \mathbb{N}_{[1, N]}$,

$$\begin{aligned} H\Psi^j \left(\Psi^m p(t_k|t_k) + \begin{bmatrix} e \\ \mathbf{0} \end{bmatrix} \right) &\leq \ell - g_j, \\ \forall j \in \mathbb{N}_{\leq \hat{N}+N-m}, \quad \forall e \in \mathcal{R}_m, \end{aligned}$$

and hence

$$\Psi^m p(t_k|t_k) + \begin{bmatrix} e \\ \mathbf{0} \end{bmatrix} \in \mathcal{X}_{\hat{N}+N-m}, \quad \forall e \in \mathcal{R}_m, \quad m \in \mathbb{N}_{[1, N]}. \quad (16)$$

From (16) and the observation that $\mathcal{X}_{\hat{N}} = \mathcal{X}_{\hat{N}+1} = \dots = \mathcal{X}_{\hat{N}+N}$ implies $\mathcal{X}_{\hat{N}+N-m} = \mathcal{X}_{\hat{N}+N-m+1}$ for all $m \in \mathbb{N}_{[1, N]}$, we obtain

$$\begin{aligned} &H\Psi^{\hat{N}+N+1} p(t_k|t_k) \\ &\leq \ell - \max_{m \in \mathbb{N}_{[1, N]}} \left\{ \max_{e \in \mathcal{R}_m} (F + GK)\Phi^{\hat{N}+N+1-m} e + g_{\hat{N}+N+1-m} \right\} \\ &= \ell - g_{\hat{N}+N+1}. \end{aligned}$$

It follows that, if $p(t_k|t_k) \in \mathcal{X}_{\hat{N}}$ and $\mathcal{X}_{\hat{N}} = \mathcal{X}_{\hat{N}+N}$, then $p(t_k|t_k) \in \mathcal{X}_{\hat{N}+N+1}$ and hence $\mathcal{X}_{\hat{N}} = \mathcal{X}_{\hat{N}+N+1}$. This argument can be repeated to show that $\mathcal{X}_{\hat{N}} = \mathcal{X}_{\hat{N}+N+j}$ for all $j \in \mathbb{N}_{\geq 1}$ and hence $\mathcal{X}_{\hat{N}} = \mathcal{X}_\infty$ holds. This implies that the constraints in (10) for all $i \in \mathbb{N}$ are satisfied if and only if (15) is satisfied, with \hat{N} selected as stated in the theorem. ■

Remark 3.1: The smallest value for $\hat{N} \in \mathbb{N}_{\geq N}$ satisfying the conditions of Theorem 3.2 can be computed by solving at most $(\hat{N} - N + 1)N$ LP problems, namely for $n \in \mathbb{N}_{[N, \hat{N}]}$

$$\begin{aligned} & \max_p H\Psi^{n+i}p + g_{n+i} \\ & \text{subject to } H\Psi^j p + g_j \leq \ell, \quad \forall j \in \mathbb{N}_{\leq n}, \end{aligned}$$

for $i \in \mathbb{N}_{[1, N]}$. Clearly the value of \hat{N} does not depend on the system state, and this procedure can therefore be performed offline. Furthermore, from Theorem 4.1 in [20], \hat{N} is necessarily finite if Ψ is strictly stable and the pair (Ψ, H) is observable.

D. Optimization Problem Formulation

The infinite horizon cost function to be minimized is defined as

$$J(x(t_k), \mathbf{c}(t_k)) := \sum_{i=0}^{\infty} \left(\|z(t_k + i|t_k)\|_Q^2 + \|Kz(t_k + i|t_k) + c(t_k + i|t_k)\|_R^2 \right), \quad (17)$$

where $Q \succ 0$, $R \succ 0$ are weighting matrices. From Theorem 2.10 in [19], the cost function (17), with K being the unconstrained LQ-optimal, can be transformed into a quadratic form of the current state $x(t_k)$ and the decision variable $\mathbf{c}(t_k)$

$$J(x(t_k), \mathbf{c}(t_k)) = \|x(t_k)\|_{W_x}^2 + \|\mathbf{c}(t_k)\|_{W_c}^2, \quad (18)$$

where $W_x \succ 0$ is the solution of the algebraic Riccati equation

$$W_x = A^T W_x A + Q - A^T W_x B (B^T W_x B + R)^{-1} B^T W_x A, \quad (19)$$

and

$$W_c = \begin{bmatrix} B^T W_x B + R & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & B^T W_x B + R & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & B^T W_x B + R \end{bmatrix}. \quad (20)$$

Hence, for a given state $x(t_k)$ at sampling instant t_k , the minimization of the cost (17) is equivalent to the minimization of $\|\mathbf{c}(t_k)\|_{W_c}^2$.

Using the results above, at any sampling instant t_k , the optimization problem $\mathbb{P}(x(t_k))$ given a state $x(t_k)$ is then summarized as

$$\min_{\mathbf{c}(t_k)} J(x(t_k), \mathbf{c}(t_k)) = \|x(t_k)\|_{W_x}^2 + \|\mathbf{c}(t_k)\|_{W_c}^2 \quad (21a)$$

subject to

$$p(t_k|t_k) = \begin{bmatrix} z(t_k|t_k) \\ \mathbf{c}(t_k) \end{bmatrix}, \quad z(t_k|t_k) = x(t_k), \quad (21b)$$

$$p(t_k + i|t_k) = \Psi p(t_k + i - 1|t_k), \quad \forall i \in \mathbb{N}_{[1, \hat{N}]}, \quad (21c)$$

$$Hp(t_k + i|t_k) \leq \ell - g_i, \quad \forall i \in \mathbb{N}_{\leq \hat{N}}. \quad (21d)$$

The optimal solution for $\mathbb{P}(x(t_k))$ is denoted by $\mathbf{c}^*(t_k)$, and $z^*(t_k + i|t_k)$ and $J(x(t_k), \mathbf{c}^*(t_k))$ denote the corresponding predicted nominal state trajectory and the cost respectively.

IV. FAST SELF-TRIGGERED MPC ALGORITHM

A fast convergence self-triggered mechanism is first derived in this section and a self-triggered MPC algorithm is then proposed to achieve the control goal described in Section II.

After solving (21) at a sampling instant t_k , the system is controlled under the optimal control input sequence $u^*(t_k + i) = Kz^*(t_k + i|t_k) + c^*(t_k + i|t_k)$ in an open-loop fashion until the next sampling instant t_{k+1} . Hence no computation is required in the time interval $\mathbb{N}_{[t_k+1, t_k+M_k-1]}$. However, it is worth noting that the reduction in computation is achieved by sacrificing the closed loop control performance.

To deal with this issue, we propose a fast convergence self-triggered mechanism to determine the inter-sampling time M_k through consideration of the convergence rate and stability analysis. This makes it possible to achieve fast convergence of the closed-loop system, while maintaining the advantages of the self-triggered controller. The main idea is at each sampling instant t_k to compute $M_k \in \mathbb{N}_{[1, N]}$ such that $J(x(t_k + M_k), \mathbf{c}^*(t_k + M_k)) - J(x(t_k), \mathbf{c}^*(t_k))$ reaches the minimum under the optimal predicted control input sequence. The detailed expression for $J(x(t_k + M_k), \mathbf{c}^*(t_k + M_k)) - J(x(t_k), \mathbf{c}^*(t_k))$ is expressed in terms of M_k in Theorem 4.1.

Theorem 4.1: If the optimal control inputs $u^*(t_k + i) = Kz^*(t_k + i|t_k) + c^*(t_k + i|t_k)$ are applied to (1) for $i \in \mathbb{N}_{\leq M_k-1}$ in an open-loop fashion, then the optimal cost $J(x(t_{k+1}), \mathbf{c}^*(t_{k+1}))$ obtained by solving $\mathbb{P}(x(t_{k+1}))$ based on the new state $x(t_{k+1})$ satisfies

$$\begin{aligned} & J(x(t_{k+1}), \mathbf{c}^*(t_{k+1})) - J(x(t_k), \mathbf{c}^*(t_k)) \\ & \leq f(M_k, x(t_k), \mathbf{c}^*(t_k)), \end{aligned} \quad (22)$$

where

$$\begin{aligned} f(M_k, x(t_k), \mathbf{c}^*(t_k)) &:= - \sum_{i=0}^{M_k-1} \|ET^i \mathbf{c}^*(t_k)\|_S^2 \\ &+ [\|z^*(t_k + M_k|t_k)\|_{W_x} - \|x(t_k)\|_{W_x} + \gamma(M_k)]^2 \\ &+ 2\|x(t_k)\|_{W_x} [\|z^*(t_k + M_k|t_k)\|_{W_x} - \|x(t_k)\|_{W_x} + \gamma(M_k)] \end{aligned} \quad (23)$$

with $\gamma(M_k) := \max_{e \in \mathcal{R}_{M_k}} \|e\|_{W_x}$ and $S := B^T W_x B + R \succ 0$.

Proof: By the reasoning in the proof of Theorem 3.1, specifically considering that $\bar{\mathbf{c}}(t_{k+1}) = T^{M_k} \mathbf{c}^*(t_k)$ is a feasible solution at sampling instant t_{k+1} , we have

$$\begin{aligned} & \mathbf{c}^{*T}(t_{k+1}) W_c \mathbf{c}^*(t_{k+1}) \\ &= \sum_{i=0}^{N-1} \mathbf{c}^{*T}(t_{k+1} + i|t_{k+1}) S \mathbf{c}^*(t_{k+1} + i|t_{k+1}) \\ &\leq \sum_{i=0}^{N-1} \bar{\mathbf{c}}^T(t_{k+1} + i|t_{k+1}) S \bar{\mathbf{c}}(t_{k+1} + i|t_{k+1}) \\ &= \sum_{i=M_k}^{N-1} \mathbf{c}^{*T}(t_k + i|t_k) S \mathbf{c}^*(t_k + i|t_k) \\ &= \mathbf{c}^{*T}(t_k) W_c \mathbf{c}^*(t_k) - \sum_{i=0}^{M_k-1} \mathbf{c}^{*T}(t_k + i|t_k) S \mathbf{c}^*(t_k + i|t_k) \\ &= \mathbf{c}^{*T}(t_k) W_c \mathbf{c}^*(t_k) - \sum_{i=0}^{M_k-1} (ET^i \mathbf{c}^*(t_k))^T S (ET^i \mathbf{c}^*(t_k)) \end{aligned} \quad (24)$$

where the first inequality follows from the optimality of $\mathbf{c}^*(t_{k+1})$ at sampling instant t_{k+1} . Then, the difference of the optimal costs between any two successive sampling instants satisfies

$$\begin{aligned} & J(x(t_{k+1}), \mathbf{c}^*(t_{k+1})) - J(x(t_k), \mathbf{c}^*(t_k)) \\ & \leq \|x(t_{k+1})\|_{W_x}^2 - \|x(t_k)\|_{W_x}^2 - \sum_{i=0}^{M_k-1} \|ET^i \mathbf{c}^*(t_k)\|_S^2. \end{aligned} \quad (25)$$

Further, by the definition of $\gamma(M_k)$, it holds that

$$\begin{aligned} & \|x(t_{k+1})\|_{W_x}^2 - \|x(t_k)\|_{W_x}^2 \\ & = [\|x(t_{k+1})\|_{W_x} - \|x(t_k)\|_{W_x}]^2 + 2\|x(t_k)\|_{W_x} [\|x(t_{k+1})\|_{W_x} - \|x(t_k)\|_{W_x}] \\ & \leq [\|z^*(t_k + M_k|t_k)\|_{W_x} - \|x(t_k)\|_{W_x} + \gamma(M_k)]^2 \\ & \quad + 2\|x(t_k)\|_{W_x} [\|z^*(t_k + M_k|t_k)\|_{W_x} - \|x(t_k)\|_{W_x} + \gamma(M_k)]. \end{aligned} \quad (26)$$

Hence, bounding (25) with (26) yields (22). ■

By Theorem 4.1, one may select the interval M_k such that the bound on $J(x(t_{k+1}), \mathbf{c}^*(t_{k+1})) - J(x(t_k), \mathbf{c}^*(t_k))$ is minimized. Therefore, the fast convergence self-triggered mechanism is formulated as follows for $k \in \mathbb{N}$

$$M_k := \underset{M \in \mathbb{N}_{[1, N]}}{\operatorname{argmin}} f(M, x(t_k), \mathbf{c}^*(t_k)). \quad (27)$$

We now summarize the fast self-triggered MPC algorithm in Algorithm 1.

Algorithm 1 Fast Self-triggered MPC

Offline:

Choose weighting matrices Q , R , and prediction horizon N . Determine \hat{N} satisfying the condition of Theorem 3.2, compute constraint parameter g_i for $i \in \mathbb{N}_{\leq \hat{N}}$. Calculate W_x and W_c by (19)–(20) and design K as the unconstrained LQ-optimal gain.

Online:

- 1: Initialize $t_k = 0$ and $k = 0$.
 - 2: At time t_k , measure the current state $x(t_k)$ of system (1).
 - 3: Solve the optimization problem $\mathbb{P}(x(t_k))$ in (21) to obtain $\mathbf{c}^*(t_k)$.
 - 4: Choose the inter-sampling time M_k according to (27).
 - 5: Apply $u^*(t_k + i) = Kz^*(t_k + i|t_k) + c^*(t_k + i|t_k)$ to the system (1) for $i \in \mathbb{N}_{\leq M_k-1}$.
 - 6: At time $t_k + M_k$, set $t_{k+1} = t_k + M_k$, $k \leftarrow k + 1$.
 - 7: Go to step 2.
-

By implementing Algorithm 1, the resulting closed-loop system is

$$x(t+1) = Ax(t) + Bu^*(t) + w(t), \quad (28)$$

$$u^*(t) = Kz^*(t|t_k) + c^*(t|t_k), \quad (29)$$

for $t \in \mathbb{N}_{[t_k, t_k + M_k - 1]}$ and $k \in \mathbb{N}$. The optimal perturbation sequence $\mathbf{c}^*(t_k)$ and the inter-sampling time M_k are only calculated at sampling instants $\{t_k, k \in \mathbb{N}\}$. At other time steps, no computation of the MPC optimization problem takes place.

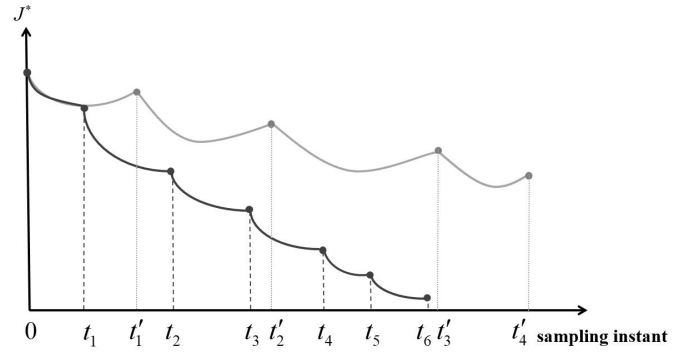


Fig. 2. Illustrative sketch of two self-triggered mechanisms inspired by the reduction in optimal costs. Black line is the optimal cost trajectory under (27), in which the sampling instant t_k is determined by maximizing the difference $J(x(t_{k-1}), \mathbf{c}^*(t_{k-1})) - J(x(t_k), \mathbf{c}^*(t_k))$. Grey line is the optimal cost trajectory under the self-triggered mechanism of [14], in which the sampling instant t'_k is determined when $J(x(t'_k), \mathbf{c}^*(t'_k)) - J(x(t'_{k-1}), \mathbf{c}^*(t'_{k-1})) < 0$ is violated.

Remark 4.1: Some comparisons may be made with [14], which is a self-triggered MPC scheme for continuous systems without disturbances and state constraints. The self-triggered mechanism in [14] is also inspired by the reduction in optimal costs, and the inter-sampling time M is determined using the criterion that the optimal cost-to-go after M time-steps of open-loop control should not exceed the optimal cost at the current time. In contrast to [14], in this paper, we determine the next sampling instant by selecting the inter-sampling time M such that the optimal cost-to-go reaches a minimum after M time-steps of open-loop control. The basic ideas of these two self-triggered mechanisms are sketched in Fig. 2. It is to be expected that the inter-sampling time under the self-triggered mechanism in [14] is larger than the one obtained under Algorithm 1. However, intuitively, the time-average number of optimization problems to be solved under Algorithm 1 might be similar to that under [14], since Algorithm 1 is likely to provide a faster convergence of system states to the origin.

V. PROPERTIES OF FAST SELF-TRIGGERED MPC ALGORITHM

In this section, we derive the control theoretic properties of Algorithm 1, including recursive feasibility, constraint satisfaction and mean-square stability.

For a given state $x(t_k)$, let $\mathbb{D}(x(t_k)) := \{c(t_k) \in \mathbb{R}^{N_u N} \mid (21b) - (21d) \text{ in } \mathbb{P}(x(t_k)) \text{ hold}\}$ denote the set of all feasible solutions of $\mathbb{P}(x(t_k))$. If $\mathbb{D}(x(t_k)) \neq \emptyset$, the optimization problem $\mathbb{P}(x(t_k))$ is feasible. Theorem 5.1 shows that the multi-step open-loop MPC optimization problem (21) is recursively feasible and the constraint (3) is satisfied along closed-loop trajectories.

Theorem 5.1: (Recursive feasibility and constraint satisfaction) Provided that the optimization problem $\mathbb{P}(x(t_0))$ is feasible at time t_0 , i.e. $\mathbb{D}(x(t_0)) \neq \emptyset$, then for the closed-loop system (28)–(29) under Algorithm 1, it holds that

- (i) $\mathbb{D}(x(t_k)) \neq \emptyset$ for all t_k and all $k \in \mathbb{N}$;
- (ii) $Fx(t) + Gu(t) \leq \ell$ for all possible realizations of $w(t) \in \mathcal{W}$, $t \in \mathbb{N}$.

Proof: By Theorem 3.1 (ii), the assumption that $\mathbb{P}(x(t_0))$ is feasible at the initial time implies that all subsequent optimization problems $\mathbb{P}(x(t_k))$ are feasible, which proves (i). The second part of the statement follows directly from Theorem 3.1 (i) and the recursive feasibility implied by (i). ■

The following lemma, which is derived from l_2 stability theory, will be required to establish mean-square stability in Theorem 5.2.

Lemma 5.1: Consider a closed-loop system

$$x(t+1) = \Phi x(t) + Bc(t) + Dw(t), \quad t \in \mathbb{N}. \quad (30)$$

If Φ is strictly stable, then the states of (30) satisfy the bound

$$\sum_{t=0}^{\infty} \|x(t)\|^2 \leq \|x(0)\|_P^2 + \alpha_1 \sum_{t=0}^{\infty} \|c(t)\|^2 + \alpha_2 \sum_{t=0}^{\infty} \|w(t)\|^2,$$

for some matrix $P \succ 0$ and some scalars $\alpha_1, \alpha_2 > 0$.

Proof: See [19], page 85. ■

The following theorem establishes the mean-square stability of the closed-loop system under Algorithm 1.

Theorem 5.2: (Mean-square stability) Provided $\mathbb{P}(x(t_0))$ is feasible at time t_0 , the closed-loop system (28)–(29) under Algorithm 1 satisfies the following properties for all possible realisations of $w(t) \in \mathcal{W}$, $t \in \mathbb{N}$,

- (i) $c(t) \rightarrow \mathbf{0}$ as $t \rightarrow \infty$;
- (ii) there exists an $\alpha > 0$ satisfying the mean-square stability condition

$$\lim_{m \rightarrow \infty} \frac{1}{m} \sum_{t=0}^m \|x(t)\|^2 \leq \alpha \tilde{w}, \quad (31)$$

where $\tilde{w} := \max_{\hat{w} \in \mathcal{T}} \|\hat{w}\|^2$ and $\mathcal{T} := \mathcal{W} \oplus (-BK \oplus_{j=0}^{N-2} A^j \mathcal{W})$.

Proof: We first show the convergence of the perturbation sequence $\{c(t), t \in \mathbb{N}\}$. By the reasoning in the proof of Theorem 4.1, particularly by (24), it holds that

$$\begin{aligned} \mathbf{c}^{*T}(t_{k+1}) W_c \mathbf{c}^*(t_{k+1}) &\leq \mathbf{c}^{*T}(t_k) W_c \mathbf{c}^*(t_k) \\ &\quad - \sum_{i=0}^{M_k-1} (E\hat{\mathbf{c}}(t_k+i))^T S (E\hat{\mathbf{c}}(t_k+i)) \end{aligned} \quad (32)$$

with $\hat{\mathbf{c}}(t_k+i) := T^i \mathbf{c}^*(t_k)$ for $i \in \mathbb{N}_{\leq M_k-1}$. Summing both sides of (32) over $k = 0, 1, \dots$ gives

$$\begin{aligned} \sum_{k=0}^{\infty} \sum_{i=0}^{M_k-1} (E\hat{\mathbf{c}}(t_k+i))^T S (E\hat{\mathbf{c}}(t_k+i)) &\leq \mathbf{c}^{*T}(0) W_c \mathbf{c}^*(0) \\ &\quad - \lim_{k \rightarrow \infty} \mathbf{c}^{*T}(t_k) W_c \mathbf{c}^*(t_k), \end{aligned}$$

which implies

$$\sum_{t=0}^{\infty} c^T(t) S c(t) \leq \mathbf{c}^{*T}(0) W_c \mathbf{c}^*(0), \quad (33)$$

where $c(t)$ is the actual perturbation signal used to define the control input at time step t . Hence, we have that $c(t) \rightarrow \mathbf{0}$ as $t \rightarrow \infty$ by taking into account the facts that $\mathbf{c}^{*T}(0) W_c \mathbf{c}^*(0)$ is finite and $S \succ 0$.

We will next prove that the closed-loop system satisfies the mean-square stability condition. Denote $\hat{w}(t) := w(t) -$

$BK e(t)$, and $e(t)$ for $t \in \mathbb{N}_{[t_k, t_{k+1}-1]}$ as the deviation between the actual state $x(t)$ and the optimal nominal state $z^*(t|t_k)$ in the open loop interval $[t_k, t]$. Then, for all $t \in \mathbb{N}_{[t_k, t_{k+1}-1]}$ and $k \in \mathbb{N}$, the closed-loop system (28)–(29) can be rewritten as

$$\begin{aligned} x(t+1) &= Ax(t) + BK z^*(t|t_k) + Bc^*(t|t_k) + w(t) \\ &= \Phi x(t) + BE\hat{\mathbf{c}}(t) + \hat{w}(t). \end{aligned} \quad (34)$$

Since Φ is strictly stable due to the assumption on K , by Lemma 5.1, it follows that there exists some matrix $P \succ 0$ and positive scalars δ, α satisfying

$$\begin{aligned} \sum_{t=0}^{\infty} \|x(t)\|^2 &\leq \|x(0)\|_P^2 + \delta \sum_{t=0}^{\infty} \|E\hat{\mathbf{c}}(t)\|^2 + \alpha \sum_{t=0}^{\infty} \|\hat{w}(t)\|^2 \\ &\leq \|x(0)\|_P^2 + \frac{\delta}{\lambda_{\min}(S)} \sum_{t=0}^{\infty} \|E\hat{\mathbf{c}}(t)\|_S^2 + \alpha \sum_{t=0}^{\infty} \|\hat{w}(t)\|^2 \\ &\leq \|x(0)\|_P^2 + \frac{\delta}{\lambda_{\min}(S)} \mathbf{c}^{*T}(0) W_c \mathbf{c}^*(0) + \alpha \sum_{t=0}^{\infty} \|\hat{w}(t)\|^2, \end{aligned} \quad (35)$$

where the last line follows from (33). As the maximal length of the open-loop phase is $N-1$, it holds that $e(t) \in \mathcal{R}_{N-1}$, and hence the polytopic set $\mathcal{T} = \mathcal{W} \oplus (-BK \oplus_{j=0}^{N-2} A^j \mathcal{W})$ contains all possible realizations of $\hat{w}(t)$ for any $t \in \mathbb{N}$. Let \tilde{w} denote the maximum of $\|\hat{w}(t)\|^2$ over \mathcal{T} . Then, the satisfaction of mean-square stability condition (31) follows from (35) and the fact that $\mathbf{c}^{*T}(0) W_c \mathbf{c}^*(0)$ is finite. ■

VI. FAST SELF-TRIGGERED MPC WITH A LOCAL CONTROLLER

From Theorem 5.2 (i), we show that as $t \rightarrow \infty$, the MPC law (29) converges asymptotically to the control law given by

$$u^*(t) = K z^*(t|t_k), \quad (36)$$

and hence the closed-loop system of (34) becomes

$$x(t+1) = \Phi x(t) + \hat{w}(t). \quad (37)$$

In order to further reduce the required computation, one alternative control strategy is to switch the MPC law to (36) after states reach a terminal set \mathcal{X} . In particular, we define a terminal set

$$\mathcal{X} := \{x \mid (F + GK)\Phi^i x \leq \ell - g_i, \quad i \in \mathbb{N}_{\leq \hat{N}}\}, \quad (38)$$

which is derived from (15) by letting $\mathbf{c}(t_k) = \mathbf{0}$. Theorem 6.1 states that, if a state trajectory enters \mathcal{X} , it will stay in the set thereafter and satisfy constraint (3) under the control law (36) without needing to solve the MPC optimization problem.

Theorem 6.1: (Robust positive invariance of \mathcal{X}) The terminal set \mathcal{X} is RPI for the system (37) and constraint (3).

Proof: Assume that the state trajectory $x(t)$ first reaches \mathcal{X} at time $t_{k'}$. From Theorem 3.2 and the assumption $x(t_{k'}) \in \mathcal{X}$, it follows that for $i \in \mathbb{N}_{\leq \hat{N}}$ and $m \in \mathbb{N}_{[1, \hat{N}]}$

$$\begin{aligned} (F + GK)\Phi^{i+m} x(t_{k'}) &= (F + GK)\Phi^i z(t_{k'} + m|t_{k'}) \\ &\leq \ell - g_{i+m} \leq \ell - g_i - (F + GK)\Phi^i e(t_{k'} + m|t_{k'}), \end{aligned}$$

and therefore $(F + GK)\Phi^i x(t_{k'} + m) \leq \ell - g_i$. Hence, $x(t) \in \mathcal{X}$ for all $t \in \mathbb{N}_{[t_{k'}+1, t_{k'}+1]}$. By induction, we have $x(t) \in \mathcal{X}$ for all $t \in \mathbb{N}_{\geq t_{k'}}$. The satisfaction of constraint (3) in \mathcal{X} follows directly from Theorem 3.1. ■

Now we reconsider the self-triggering criterion inside the terminal set \mathcal{X} . For $x(t_k) \in \mathcal{X}$, and in particular if $x(t_k)$ is close to the origin, the value of $f(M, x(t_k), \mathbf{0})$ is dominated by the value of $\gamma(M)$. Since $\gamma(M)$ is increasing with M , the self-triggering condition (27) may therefore result in a small sampling interval M_k close to 1. To obtain a larger sampling interval, a different policy for deciding M_k is employed in a neighborhood of the origin $\mathcal{X} \cap \mathcal{X}(\varepsilon)$, where $\mathcal{X}(\varepsilon)$ is a level set of the cost $J(x(t_k), \mathbf{0})$ defined by

$$\mathcal{X}(\varepsilon) := \{x \mid \|x\|_{W_x}^2 \leq \varepsilon\}, \quad \varepsilon \geq \varepsilon_0, \quad (39)$$

$$\varepsilon_0 := \frac{\max_{w \in \mathcal{W}} \|w\|_{W_x}^2}{\left[1 - \sqrt{\lambda_{\max}\left(W_x^{-\frac{1}{2}} \Phi^T W_x \Phi W_x^{-\frac{1}{2}}\right)}\right]^2}.$$

Specifically, for $x(t_k)$ outside the set $\mathcal{X} \cap \mathcal{X}(\varepsilon)$, the triggering mechanism (27) is employed, but once $x(t_k)$ reaches $\mathcal{X} \cap \mathcal{X}(\varepsilon)$, to avoid a high sampling rate, the triggering mechanism switches to

$$M_k := \max \{M \in \mathbb{N}_{[1, N]} \mid J(x(t_k), \mathbf{0}) + f(M, x(t_k), \mathbf{0}) \leq \varepsilon\}. \quad (40)$$

The proposed fast self-triggered MPC with a local controller is summarized in Algorithm 2.

Algorithm 2 Fast Self-triggered MPC with a Local Controller

Offline:

Perform the offline step of Algorithm 1 and construct sets \mathcal{X} and $\mathcal{X}(\varepsilon)$ according to (38)–(39).

Online:

Perform the online steps of Algorithm 1 except for steps 3 and 4, which are modified to

- 3: If $x(t_k) \in \mathcal{X}$, then set $\mathbf{c}^*(t_k) = \mathbf{0}$. Otherwise, solve the optimization problem $\mathbb{P}(x(t_k))$ in (21) to obtain $\mathbf{c}^*(t_k)$.
 - 4: If $x(t_k) \in \mathcal{X} \cap \mathcal{X}(\varepsilon)$, then choose the inter-sampling time M_k according to (40). Otherwise, choose M_k according to (27).
-

Lemma 6.1 demonstrates the robust positive invariance of $\mathcal{X}(\varepsilon)$ under a static periodic LQ-optimal control law.

Lemma 6.1: If $\varepsilon \geq \varepsilon_0$ holds, then the set $\mathcal{X}(\varepsilon)$ in (39) is RPI for the system $x(t+1) = \Phi x(t) + w(t)$ with $w(t) \in \mathcal{W}$.

Proof: This can be shown if it holds that

$$\max_{x \in \mathcal{X}(\varepsilon), w \in \mathcal{W}} \|\Phi x + w\|_{W_x}^2 \leq \varepsilon. \quad (41)$$

Specifically, by the triangle inequality, we have

$$\max_{x \in \mathcal{X}(\varepsilon), w \in \mathcal{W}} \|\Phi x + w\|_{W_x} \leq \max_{x \in \mathcal{X}(\varepsilon)} \|\Phi x\|_{W_x} + \max_{w \in \mathcal{W}} \|w\|_{W_x},$$

where

$$\max_{x \in \mathcal{X}(\varepsilon)} \|\Phi x\|_{W_x} = \varepsilon^{\frac{1}{2}} \sqrt{\lambda_{\max}\left(W_x^{-\frac{1}{2}} \Phi^T W_x \Phi W_x^{-\frac{1}{2}}\right)}.$$

Therefore, condition (41) is satisfied if $\varepsilon \geq \varepsilon_0$. ■

Remark 6.1: If the disturbance set \mathcal{W} is ellipsoidal, e.g. $\mathcal{W} = \{w \mid \|w\|_V^2 \leq 1\}$ with $V \succ 0$, then $\varepsilon \geq \varepsilon_0$ is sufficient as well as necessary for $\mathcal{X}(\varepsilon)$ to be RPI for $x(t+1) = \Phi x(t) + w(t)$, $w(t) \in \mathcal{W}$. Moreover, for this case, the quantity $\max_{w \in \mathcal{W}} \|w\|_{W_x}^2$ in ε_0 becomes

$$\max_{w \in \mathcal{W}} \|w\|_{W_x}^2 = \lambda_{\max}(V^{-1}W_x).$$

In order to derive closed-loop stability of Algorithm 2, some technical results are established.

Lemma 6.2:

- (i) $f(1, x, \mathbf{0}) \leq 0$ if $\|x\|_{W_x}^2 \geq \varepsilon_0$.
- (ii) $J(x, \mathbf{0}) + f(1, x, \mathbf{0}) \leq \varepsilon$ for all $x \in \mathcal{X}(\varepsilon)$ if and only if $\varepsilon \geq \varepsilon_0$.

Proof: From the proof of Lemma 6.1 and the fact that $\gamma(1) = \max_{w \in \mathcal{W}} \|w\|_{W_x}$, we have $\max_{x \in \mathcal{X}(\varepsilon)} \|\Phi x\|_{W_x} + \gamma(1) \leq \varepsilon^{\frac{1}{2}}$ if and only if $\varepsilon \geq \varepsilon_0$. Hence, if $\|x\|_{W_x}^2 \geq \varepsilon_0$, then we have $\|\Phi x\|_{W_x} + \gamma(1) \leq \|x\|_{W_x}$, which is equivalent to $f(1, x, \mathbf{0}) = (\|\Phi x\|_{W_x} + \gamma(1))^2 - \|x\|_{W_x}^2 \leq 0$.

For (ii), it follows from $J(x, \mathbf{0}) = \|x\|_{W_x}^2$ that $J(x, \mathbf{0}) + f(1, x, \mathbf{0}) \leq \varepsilon$ if and only if $\|\Phi x\|_{W_x} + \gamma(1) \leq \varepsilon^{\frac{1}{2}}$, which is true for all $x \in \mathcal{X}(\varepsilon)$ if and only if $\varepsilon \geq \varepsilon_0$. ■

Lemma 6.2 (ii) provides a guarantee that the set on the right-hand side of (40) is non-empty whenever $x(t_k) \in \mathcal{X} \cap \mathcal{X}(\varepsilon)$. Also $\varepsilon = \varepsilon_0$ is the minimum value of ε for which this guarantee holds.

Theorem 6.2: (Ultimate boundedness) Under Algorithm 2, the state $x(t)$ of the closed-loop system asymptotically converges to $\mathcal{X} \cap \mathcal{X}(\varepsilon)$ as $t \rightarrow \infty$ for any disturbances with $w(t) \in \mathcal{W}$, $t \in \mathbb{N}$.

Proof: Suppose that the current state $x(t_k)$ is outside \mathcal{X} . Theorem 5.2 shows that $x(t_k)$ asymptotically converges to the RPI set \mathcal{X} as $t_k \rightarrow \infty$. Assume that at a certain time $t_{k'}$ with $t_{k'} > t_k$, the state $x(t_{k'})$ enters $\mathcal{X} \setminus \mathcal{X}(\varepsilon)$, i.e., $\|x(t_{k'})\|_{W_x}^2 > \varepsilon \geq \varepsilon_0$ and $\mathbf{c}(t_{k'}) = \mathbf{0}$. Then from Lemma 6.2 (i) and self-triggered mechanism (27) adopted in $\mathcal{X} \setminus \mathcal{X}(\varepsilon)$, it follows that

$$f(M_{k'}, x(t_{k'}), \mathbf{0}) \leq f(1, x(t_{k'}), \mathbf{0}) < 0,$$

and further that

$$\begin{aligned} \|x(t_{k'+1})\|_{W_x}^2 &\leq \|x(t_{k'})\|_{W_x}^2 + f(M_{k'}, x(t_{k'}), \mathbf{0}) \\ &< \|x(t_{k'})\|_{W_x}^2. \end{aligned}$$

This implies the convergence of the state trajectory $x(t_k)$ to $\mathcal{X}(\varepsilon)$. Therefore, we have $x(t_k) \rightarrow \mathcal{X} \cap \mathcal{X}(\varepsilon)$ as $t_k \rightarrow \infty$. Once the state $x(t_k)$ reaches $\mathcal{X} \cap \mathcal{X}(\varepsilon)$, M_k is chosen according to (40) such that $\|x(t_{k+1})\|_{W_x}^2 \leq J(x(t_k), \mathbf{0}) + f(M_k, x(t_k), \mathbf{0}) \leq \varepsilon$, which, together with Theorem 6.1, gives the robust positive invariance of $\mathcal{X} \cap \mathcal{X}(\varepsilon)$ under Algorithm 2. Therefore, it can be concluded that $x(t) \rightarrow \mathcal{X} \cap \mathcal{X}(\varepsilon)$ as $t \rightarrow \infty$. ■

Theorem 6.2 implies that the state trajectory converges to the intersection of sets \mathcal{X} and $\mathcal{X}(\varepsilon)$, where \mathcal{X} has the property that constraint (3) is satisfied under (36) and $\mathcal{X}(\varepsilon)$ is a level set of the MPC cost. The parameter ε defining $\mathcal{X}(\varepsilon)$ is a tuning knob of Algorithm 2, which allows a balance between the asymptotic bound on the MPC cost function and the average sampling frequency.

VII. DISCUSSION

In this section, some beneficial features, applications, drawbacks, and triggering behavior of the proposed self-triggered MPC strategy are discussed in detail.

A. Computational complexity

The computation of $\gamma(M)$ for $M \in \mathbb{N}_{[1,N]}$ in (23) requires the solution of a sequence of Quadratic Programming (QP) problems. Each of these QPs requires the maximization of a convex quadratic function subject to linear constraints, and hence is nonconvex; however, such QP problems can be efficiently and reliably solved using commercially available software [21], [22], [23]. We also note that the computation of $\gamma(M)$, $M \in \mathbb{N}_{[1,N]}$, is independent of the decision variable $\mathbf{c}^*(t_k)$ and the actual state $x(t_k)$, and can therefore be performed offline.

The inter-sampling time M_k is calculated online by (27), but once $\mathbf{c}^*(t_k)$ is obtained, $f(M, x(t_k), \mathbf{c}^*(t_k))$ is then a function with only one variable, M , taking values in $\{1, 2, \dots, N\}$. Solving (27) requires significantly less computational load to determine the inter-sampling time than solving up to N optimization problems at each sampling instant as in [11], [12], [13]. Hence, the self-triggered scheme (27) is more practical from a computational perspective.

B. Multiple constraints handling

The constraint set considered in this paper, that is $(x, u) \in \Xi$, with Ξ being a compact convex polytope, can be treated equivalently by combining the conditions required by each of the constraints separately. Specifically, in general $(x, u) \in \Xi$ if and only if $\mathbf{F}x + \mathbf{G}u \leq \mathbf{l}$, where $\mathbf{F} \in \mathbb{R}^{N_c \times N_x}$, $\mathbf{G} \in \mathbb{R}^{N_c \times N_u}$, $\mathbf{l} \in \mathbb{R}^{N_c}$ is a vector with elements equal to ℓ , and the inequality applies elementwise. Following the approach of Section III-A, the constraints $\mathbf{F}x(t_k + i|t_k) + \mathbf{G}u(t_k + i|t_k) \leq \mathbf{l}$, $i \in \mathbb{N}$, are satisfied for all possible disturbances $w \in \mathcal{W}$ if and only if decision variable $\mathbf{c}(t_k)$ satisfies the constraints

$$\mathbf{H}\Psi^i p(t_k|t_k) \leq \mathbf{l} - \mathbf{h}_i, \quad i \in \mathbb{N}, \quad (42)$$

where the sequence $\{\mathbf{h}_i, i \in \mathbb{N}\}$ is defined as

$$\mathbf{h}_i := \begin{cases} \max_{e \in \mathcal{R}_i} \mathbf{F}e, & i \in \mathbb{N}_{\leq N-1}, \\ \max_{e \in \mathcal{R}_i} (\mathbf{F} + \mathbf{G}K)e, & i \in \mathbb{N}_{\geq N}. \end{cases}$$

Here the maximization of a vector-valued function is to be performed elementwise, so that \mathbf{h}_i denotes the vector in \mathbb{R}^{N_c} with j th element $h_i^{(j)}$ equal to $\max_{e \in \mathcal{R}_i} F^{(j)}e$ for $i \in \mathbb{N}_{\leq N-1}$ and $\max_{e \in \mathcal{R}_i} (F^{(j)} + G^{(j)}K)e$ for $i \in \mathbb{N}_{\geq N}$, where $F^{(j)}$ and $G^{(j)}$ are the j th rows of the matrices \mathbf{F} and \mathbf{G} respectively. Therefore, the constraints in (42) can be written equivalently as

$$H^{(j)}\Psi^i p(t_k|t_k) \leq \ell - h_i^{(j)}, \quad i \in \mathbb{N}, \quad j \in \mathbb{N}_{[1,N_c]},$$

with $H^{(j)} := [F^{(j)} \quad G^{(j)}] \begin{bmatrix} I & \mathbf{0} \\ K & E \end{bmatrix}$. Thus the feasible set for (42) is the intersection of the feasible sets for the constraints $F^{(j)}x(t_k + i|t_k) + G^{(j)}u(t_k + i|t_k) \leq \ell$, $i \in \mathbb{N}$, $j \in \mathbb{N}_{[1,N_c]}$.

C. Applications in fast dynamic systems

The self-triggered MPC algorithms in [11], [12], [13] require N optimization problems to be executed within one time step, while this requirement is not necessary for Algorithms 1 and 2. In Algorithms 1 and 2, instantaneous optimizations are desirable but not vital in terms of constraint satisfaction. Due to the feasibility guarantee provided in Section III-B, constraints will be satisfied for any length of a sampling interval between 1 and N . Therefore, under the circumstance that instantaneous optimizations are not possible at a sampling instant t_k , the algorithms in this paper could be modified to allow the whole interval $[t_k, t_{k-1} + N]$ rather than just one time step for computation if computational delays are bounded by a certain level; and then, during this time interval, the optimal predicted inputs $u^*(t_k|t_{k-1}), \dots, u^*(t_{k-1} + N - 1|t_{k-1})$ calculated at sampling instant t_{k-1} can still be applied to the system to satisfy the constraints, while compromising some of the performance. However, for algorithms in [11], [12], [13], no readily available inputs can be applied at time t_k without instantaneous optimizations. Consequently, the algorithms proposed in this paper offer a clear benefit compared with [11], [12], [13], when computational delays are present, which make them suitable for fast dynamic systems.

D. Conservativeness in constraint handling

Clearly the algorithms proposed here have constraints that are at least as conservative as [11], [12], [13] because: (a) we require that a single predicted control sequence can ensure constraint satisfaction for all M_k between 1 and N ; (b) we impose additional constraints to enforce recursive feasibility in Theorem 3.1. If the constraint tightening parameters satisfy $g_i = h_i$, which would seem likely when the eigenvalues of Φ are close to zero, then the recursive feasibility constraints do not introduce any extra conservativeness. On the other hand, we do not seem to be able to avoid the conservativeness of (a), which is the price to pay for requiring the solution of fewer optimization problems per time-step (see Section VII-A) and allowing for uncertainty in the time required to perform one full optimization cycle (see Section VII-C). The conservativeness has the drawback of imposing an upper bound on the maximal prediction horizon N . In the practical implementation, the largest possible value for N is determined offline such that the feasible set for the MPC optimization (21) is non-empty. This can be ensured only if $\ell > g_i$ for all $i \in \mathbb{N}_{\leq \bar{N}}$ or if $\ell > \bar{g} := \sup\{g_i, i \in \mathbb{N}\}$. From the expression for g_i in Theorem 3.1, where Φ is by assumption a strictly stable matrix, it follows that $\max_{e \in \mathcal{R}_m} (F + GK)\Phi^{i-m}e$ converges to 0 as $i \rightarrow \infty$, thereby ensuring the convergence of g_i ($g_\infty := \lim_{i \rightarrow \infty} g_i$) and the existence of \bar{g} . Note that, for $i \in \mathbb{N}_{[1,N-1]}$, the sequence of g_i is monotonically non-decreasing and, for $i \in \mathbb{N}_{\geq N}$, g_i is the maximum over $m \in \mathbb{N}_{[1,N]}$. Therefore, a smaller N implies a smaller \bar{g} , which is more likely to satisfy $\ell > \bar{g}$. One computationally tractable way of determining N is given in Algorithm 3, which is performed offline.

Algorithm 3 Determining Prediction Horizon N

- 1: Set $N = 1$.
- 2: Compute the sequence $\{g_i, i \in \mathbb{N}_{[1,T]}\}$ with prediction horizon N , where T is sufficiently large such that differences of consecutive elements at the end of this sequence are very small (for example, smaller than $\epsilon = 10^{-12}$).
- 3: Compute $\tilde{g} := \max\{g_i, i \in \mathbb{N}_{[1,T]}\}$, which is almost equivalent to \bar{g} as defined above.
- 4: If $\tilde{g} < \ell$, then increase N by 1 and go back to step 2; otherwise the current value of $N-1$ is the largest one can use.

E. Dependence of the triggering behavior on states

While the exact dependence of the triggering behavior on system parameters is an open question, it seems that lower sampling frequencies coincide with system states away from the origin. More specifically, at each sampling instant, after sampling $x(t_k)$ and solving (21) for $\mathbf{c}^*(t_k)$, $f(M, x(t_k), \mathbf{c}^*(t_k))$ in (27) can be viewed as the sum of a quadratic functional and a monotonically non-increasing function in M denoted $c(M) := -\sum_{i=0}^{M-1} \|ET^i \mathbf{c}^*(t_k)\|_S^2 < 0$. We denote the quadratic functional by $\bar{f}(s) := s^2(M) + 2bs(M)$, where $s(M) := \|z^*(t_k + M|t_k)\|_{W_x} - \|x(t_k)\|_{W_x} + \gamma(M)$ and $b := \|x(t_k)\|_{W_x}$. Due to the non-negativity of $\|z^*(t_k + M|t_k)\|_{W_x} + \gamma(M)$, we can see that $\bar{f}(s)$ is monotonically increasing in s .

On one hand, when $x(t_k)$ is close to the origin, elements of $\mathbf{c}^*(t_k)$ and $z^*(t_k + M|t_k)$ would be quite small. Then it is likely that the variations of $f(M, x(t_k), \mathbf{c}^*(t_k))$ and $s(M)$ with M are dominated by $\bar{f}(s)$ and by $\gamma(M)$ respectively. This would imply that $f(M, x(t_k), \mathbf{c}^*(t_k))$ is increasing with M , and hence that M_k tends to be close to 1. On the other hand, when $x(t_k)$ is relatively large, since the decrease rate of $\|z^*(t_k + M|t_k)\|_{W_x}$ with M is probably higher than the increase rate of $\gamma(M)$, larger M generally implies smaller $s(M)$, which leads to the decrease of $f(M, x(t_k), \mathbf{c}^*(t_k))$, and then M_k is more likely to be close to N .

VIII. NUMERICAL EXAMPLES

To show the performance of the proposed algorithms, comparisons are made with the traditional time-triggered MPC and self-triggered tube MPC (STTMPC) in [13]. Consider a system as in [13] and [24]

$$x(t+1) = \begin{bmatrix} 1.1 & 1 \\ 0 & 1.2 \end{bmatrix} x(t) + \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} u(t) + w(t), \quad t \in \mathbb{N},$$

where the disturbance sequence $\{w(t), t \in \mathbb{N}\}$ is independent with identically uniformly distributed elements $w(t) \in \mathcal{W} = [-0.25, 0.25]^2$ (generated using the Matlab `rand` function). The state and input constraints are defined by $|x(t)| \leq 115$ and $|u(t)| \leq 20$ (here the inequality sign \leq and absolute value $|\cdot|$ apply elementwise to vectors). The cost weights are chosen as $Q = \begin{bmatrix} 0.5 & 0 \\ 0 & 1 \end{bmatrix}$ and $R = 0.1$, and K in (4) is computed as the corresponding unconstrained optimal feedback gain $K = [-0.6023, -1.3667]$. Furthermore, by (19)–(20), we have

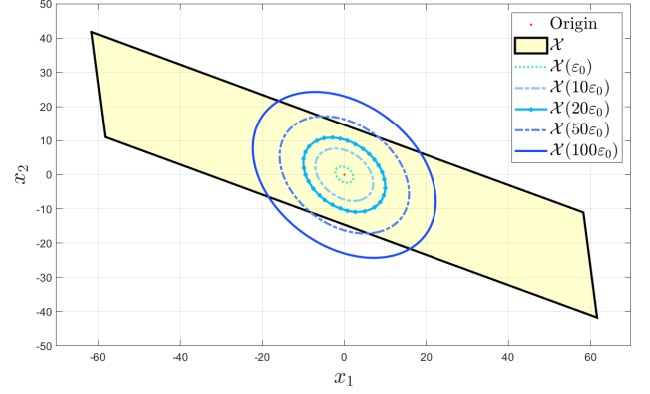


Fig. 3. The sets $\mathcal{X}(c_1\epsilon_0)$ for $c_1 = 1, 10, 20, 50, 100$ and the terminal set \mathcal{X} .

$W_x = \begin{bmatrix} 1.5810 & 0.4654 \\ 0.4654 & 1.3332 \end{bmatrix}$ and $W_c = 2.2939I$. We choose $N = 8$ satisfying $g_i < \ell$ for all $i \in \mathbb{N}$ and calculate $\hat{N} = 25$ by Theorem 3.2. The set $\mathcal{X}(\epsilon)$ is calculated as $\mathcal{X}(\epsilon) = \mathcal{X}(c_1\epsilon_0)$ with $\epsilon_0 = 7.0775$ and different values of c_1 . By Lemma 6.1, the set $\mathcal{X}(c_1\epsilon_0)$ is RPI for the system $x(t+1) = \Phi x(t) + w(t)$ with $w(t) \in \mathcal{W}$, if $c_1 \geq 1$. Fig. 3 shows the sets $\mathcal{X}(c_1\epsilon_0)$ for $c_1 = 1, 10, 20, 50, 100$ and the terminal set (38), and $c_1 = 20$ is chosen in subsequent simulations. The simulations are implemented in Matlab R2017a with CVX [25] and Gurobi solver [23].

Stability and constraint satisfaction: In the time-triggered algorithm, the inter-sampling time is fixed to $M_k = 1$ and the predicted control input sequence $u(t+i|t) = Kx(t+i|t) + c(t+i|t)$ with $c(t+i|t) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$ is adopted. In [13], two additional parameters – the size of terminal set and a tuning parameter β in the self-triggered mechanism – also have effects on closed-loop performance and average inter-sampling time. For comparison, the same terminal sets $\mathcal{X} \cap \mathcal{X}(20\epsilon_0)$ are used in Algorithm 2 and STTMPC of [13] with $\beta = 1.75$ (as chosen in [13]). For the same initial condition $x(0) = [-140 \ 30]^T$ and the same disturbance sequence, Algorithm 1, Algorithm 2, the corresponding time-triggered algorithm, and STTMPC of [13] are each implemented for a simulation length of $T_{\text{run}} = 20$ steps. The resulting state-space trajectories are shown in Fig. 4–Fig. 7 with the dashed lines being state constraint bounds and the ellipses being the sets $\mathcal{X} \cap \mathcal{X}(20\epsilon_0)$. As shown in Fig. 5, the triggering mechanism switches from (27) to (40) at time step $t = 8$, which is the first time that the input sequence is updated after the state has entered the set $\mathcal{X} \cap \mathcal{X}(20\epsilon_0)$. In Fig. 8, the evolution of each element of the system state is depicted. From these figures, it is clear that the state trajectories under these four schemes satisfy state constraints at all time steps and converge to a neighborhood of the origin with a comparable convergence speed. Fig. 9 plots the input trajectories of four schemes and shows the satisfaction of input constraints by aperiodically executing control tasks.

Average inter-sampling time: In Fig. 4–Fig. 7, we highlight the sampling instants by red markers and enlarge the region $\mathcal{X} \cap \mathcal{X}(20\epsilon_0)$ to show the sampling reduction. The average time between sampling instants is $\bar{M} = 1.58$ for Algorithm

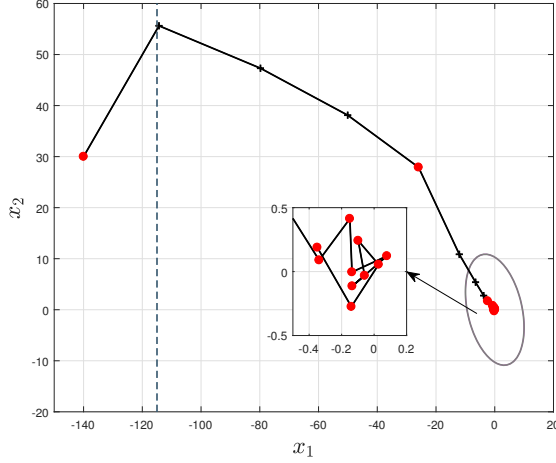


Fig. 4. State trajectory of the closed loop system under Algorithm 1 and the set $\mathcal{X} \cap \mathcal{X}(20\varepsilon_0)$.

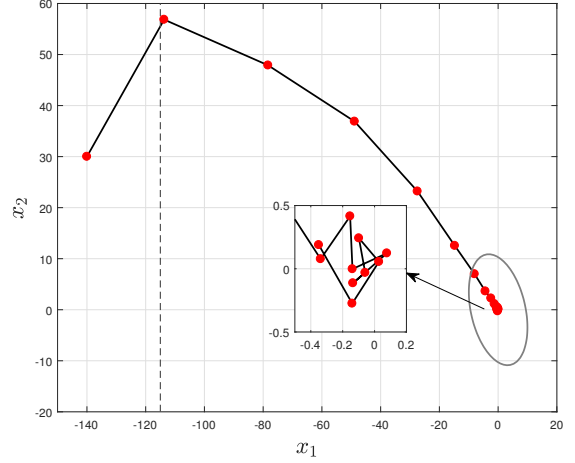


Fig. 6. State trajectory of the closed loop system under the time-triggered RMPC algorithm and the set $\mathcal{X} \cap \mathcal{X}(20\varepsilon_0)$.

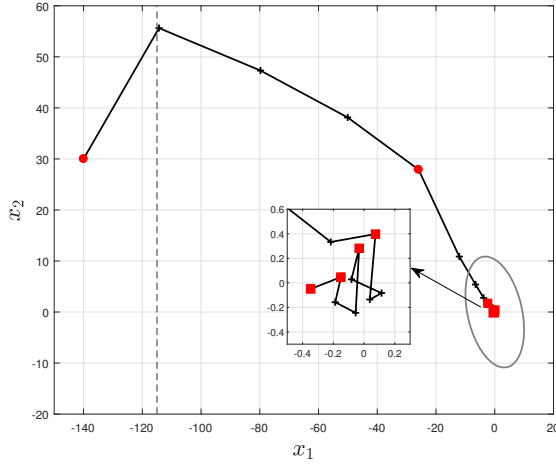


Fig. 5. State trajectory of the closed loop system under Algorithm 2 and the set $\mathcal{X} \cap \mathcal{X}(20\varepsilon_0)$. Red solid circles represent the sampling instants under (27) and red solid squares represent the sampling instants under (40).

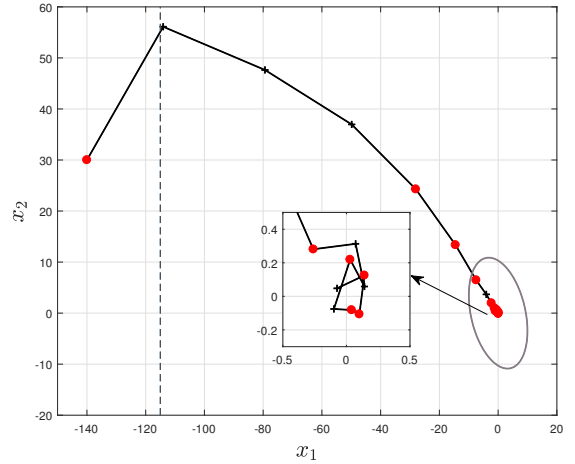


Fig. 7. State trajectory of the closed loop system under STTMPC algorithm of [13] and the set $\mathcal{X} \cap \mathcal{X}(20\varepsilon_0)$.

1, $\bar{M} = 3.8$ for Algorithm 2, and $\bar{M} = 1.54$ for STTMPC, which lead to 36.7%, 73.7% and 35.1% fewer control updates respectively when compared with the time-triggered RMPC algorithm with $\bar{M} = 1$. It can also be observed in Fig. 4 that using Algorithm 1, the inter-sampling time is always equal to one when the system is close to the origin. The reason for the frequent triggering is related to the fact that $f(M, x(t_k), c^*(t_k))$ in (27) is dominated by the increasing function $\gamma(M)$ when $x(t_k)$ is close to zero, but as shown in Fig. 5, this problem is addressed under Algorithm 2 by introducing a triggering condition with a tuning parameter ε in a neighborhood of the origin.

Performance and computational load: Consider the cost averaged over 20 time steps as a control performance index

$$J_{perf} = \frac{1}{T_{run}} \sum_{t=0}^{T_{run}-1} [\|x(t)\|_Q^2 + \|u(t)\|_R^2].$$

The performance costs are compared as follows: $J_{perf} = 1501.9$ for Algorithm 1, $J_{perf} = 1502.8$ for Algorithm 2, $J_{perf} = 1490.7$ for time-triggered RMPC, and $J_{perf} = 1497.2$ for STTMPC. For this example, Algorithm 1, Algorithm 2, and STTMPC reduce the required number of control updates without significant loss in performance when compared with the time-triggered scheme, and the performance cost is around 0.3% less for STTMPC than for Algorithm 1 and Algorithm 2. However STTMPC requires much longer online computation times at sampling instants than when Algorithm 1 or Algorithm 2 is employed (Table I). The average computation time per time-step before $t = 8$ (after which control law (36) is applied in Algorithm 2 without solving optimization problems) when STTMPC is used is almost 15 times longer than that when Algorithm 1 or Algorithm 2 is employed.

Effect of the parameter ε in Algorithm 2: To analyze the effect of $\varepsilon = c_1\varepsilon_0$, we implement Algorithm 2 with initial

TABLE I
COMPUTATION TIMES AT EACH SAMPLING INSTANT BEFORE $t = 8$

Algorithms	Computation times (sec)								Average
	$t = 0$	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$t = 5$	$t = 6$	$t = 7$	
Algorithms 1 and 2	0.58	0	0	0	0.53	0	0	0	0.14
Time-triggered MPC	0.61	0.59	0.58	0.57	0.55	0.59	0.68	0.55	0.59
STTMPC of [13]	4.91	0	0	0	4.14	3.51	3.79	0	2.04

TABLE II
RELATIONSHIP BETWEEN AVERAGE INTER-SAMPLING TIME AND THE SIZE OF $\mathcal{X}(c_1 \varepsilon_0)$

c_1	1.0	2.0	5.0	10.0	15.0	20.0	50.0	70.0	80.0	100.0
Average inter-sampling time (steps)	2.11	3.00	3.96	4.95	4.95	5.82	6.60	7.62	7.62	7.62

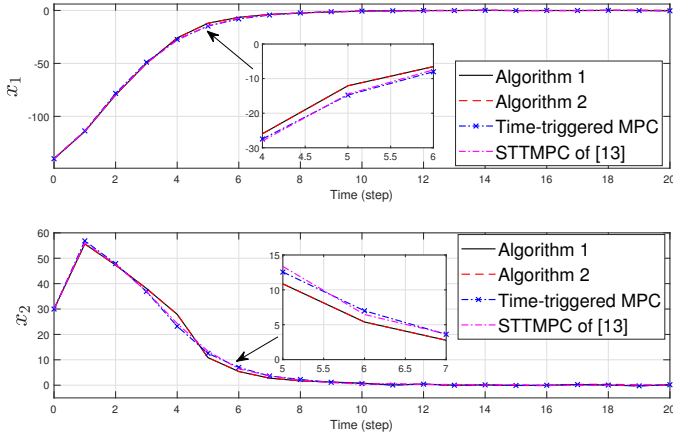


Fig. 8. Evolution of the state trajectories in time under Algorithm 1, Algorithm 2, the time-triggered RMPC algorithm and STTMPC algorithm of [13].

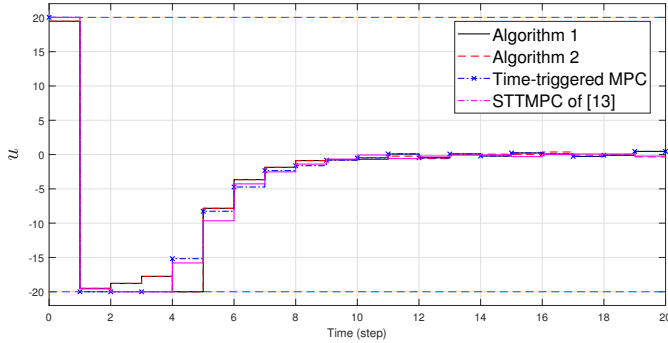


Fig. 9. Input trajectories under Algorithm 1, Algorithm 2, the time-triggered RMPC algorithm and STTMPC algorithm of [13].

condition $x(0) = [0 \ 0]^T$, $T_{\text{run}} = 100$, and different values of c_1 , and evaluate the average inter-sampling time and the asymptotic bound. Table II shows that the average inter-sampling time increases with c_1 . This is due to the fact from (40), that a larger predicted optimal cost at next sampling instant is allowed when a larger c_1 is used, thereby leading to a larger sampling interval. On the other hand, increasing ε

increases both the asymptotic bound on the MPC cost and the asymptotic bound, $\mathcal{X} \cap \mathcal{X}(\varepsilon)$, on the system state. Therefore, the parameter ε can be used to make a trade-off between the control performance and the usage of computational resources, and can be tuned to fit practical setups.

IX. CONCLUSION AND FUTURE WORK

This paper proposes a self-triggered formulation of MPC for linear systems with additive disturbances subject to hard constraints. The inter-sampling time is determined through a fast convergence self-triggered mechanism to maximize the difference between the optimal predicted costs at consecutive sampling instants. Constraint satisfaction is ensured by using worst-case predictions of the future states and control inputs to construct suitable tightened constraints, which are applied to the nominal predicted trajectories. The resulting MPC optimization problem guarantees constraint satisfaction and recursive feasibility, despite the presence of additive disturbances. The closed-loop system is stable in a mean-square sense. To further reduce the amount of computation, a local self-triggered control law is implemented in a terminal set, which ensures the convergence of states to an RPI set. Relative to periodically-triggered MPC schemes, the proposed algorithms realize a significant reduction in the number of control updates without appreciably affecting performance. Simultaneously they avoid bursts in computation at sampling instants, thus providing a significant improvement over existing RMPC-based self-triggered control algorithms. Future work will consider how the fast self-triggered framework can be reformulated for the case of continuous-time systems and nonlinear systems.

REFERENCES

- [1] Qin, S. J., & Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control engineering practice*, 11(7), 733–764.
- [2] Herzog, F., Keel, S., Dondi, G., Schumann, L. M., & Geering, H. P. (2006, June). Model predictive control for portfolio selection. In *Proceedings of the American control conference* (pp. 1252–1259).
- [3] Wang, W., Rivera, D. E., & Kempf, K. G. (2007). Model predictive control strategies for supply chain management in semiconductor manufacturing. *International Journal of Production Economics*, 107(1), 56–77.
- [4] Heemels, W. P. M. H., Johansson, K. H., & Tabuada, P. (2012, December). An introduction to event-triggered and self-triggered control. In *Proceedings of the IEEE Conference on Decision and Control* (pp. 3270–3285).

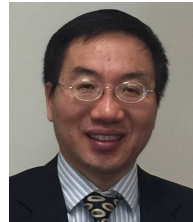
- [5] Hetel, L., Fiter, C., Omran, H., Seuret, A., Fridman, E., Richard, J. P., & Niculescu, S. I. (2017). Recent developments on the stability of systems with aperiodic sampling: An overview. *Automatica*, 76, 309–335.
- [6] Henriksson, E., Quevedo, D. E., Sandberg, H., & Johansson, K. H. (2012). Self-triggered model predictive control for network scheduling and control. *IFAC Proceedings Volumes*, 45(15), 432–438.
- [7] Henriksson, E., Quevedo, D. E., Peters, E. G., Sandberg, H., & Johansson, K. H. (2015). Multiple-loop self-triggered model predictive control for network scheduling and control. *IEEE Transactions on Control Systems Technology*, 23(6), 2167–2181.
- [8] Berglind, J. B., Gommans, T. M. P., & Heemels, W. P. M. H. (2012). Self-triggered MPC for constrained linear systems and quadratic costs. *IFAC Proceedings Volumes*, 45(17), 342–348.
- [9] Antunes, D., & Heemels, W. P. M. H. (2014). Rollout event-triggered control: Beyond periodic control performance. *IEEE Transactions on Automatic Control*, 59(12), 3296–3311.
- [10] Gommans, T., Antunes, D., Donkers, T., Tabuada, P., & Heemels, M. (2014). Self-triggered linear quadratic control. *Automatica*, 50(4), 1279–1287.
- [11] Brunner, F. D., Heemels, W. P. M. H., & Allgöwer, F. (2014, June). Robust self-triggered MPC for constrained linear systems. In *Proceedings of the European control conference* (pp. 472–477).
- [12] Aydin, E., Brunner, F. D., Heemels, W. P. M. H., & Allgöwer, F. (2015). Robust self-triggered model predictive control for constrained discrete-time LTI systems based on homothetic tubes. In *Proceedings of the European control conference* (pp. 1587–1593).
- [13] Brunner, F. D., Heemels, M., & Allgöwer, F. (2016). Robust self-triggered MPC for constrained linear systems: A tube-based approach. *Automatica*, 72, 73–83.
- [14] Hashimoto, K., Adachi, S., & Dimarogonas, D. V. (2017). Self-triggered model predictive control for nonlinear input-affine dynamical systems via adaptive control samples selection. *IEEE Transactions on Automatic Control*, 62(1), 177–189.
- [15] Eqtami, A., Heshmati-alamdari, S., Dimarogonas, D. V., & Kyriakopoulos, K. J. (2013). Self-triggered model predictive control for nonholonomic systems. In *Proceedings of the European control conference* (pp. 638–643).
- [16] 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.
- [17] Raković, S. V., Kouvaritakis, B., Findeisen, R., & Cannon, M. (2012). Homothetic tube model predictive control. *Automatica*, 48(8), 1631–1638.
- [18] Chisci, L., Rossiter, J. A., & Zappa, G. (2001). Systems with persistent disturbances: predictive control with restricted constraints. *Automatica*, 37(7), 1019–1028.
- [19] Kouvaritakis, B., & Cannon, M. (2016). *Model Predictive Control: Classical, Robust, and Stochastic*. Springer International.
- [20] Gilbert, E. G., & Tan, K. T. (1991). Linear systems with state and control constraints: The theory and application of maximal output admissible sets. *IEEE Transactions on Automatic control*, 36(9), 1008–1020.
- [21] QuadProgBB Solver. <https://github.com/sburer/QuadProgBB>.
- [22] CPLEX Optimizer. <https://www.ibm.com/analytics/cplex-optimizer>.
- [23] Gurobi Solver. <http://www.gurobi.com>.
- [24] Brunner, F. D., Heemels, W. P. M. H., & Allgöwer, F. (2016). Numerical evaluation of a robust self-triggered MPC algorithm. *IFAC-PapersOnLine*, 49(22), 151–156.
- [25] Grant, M., & Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 1.21. <http://cvxr.com/cvx>, Feb 2011.



Li Dai received the B.S. degree in Information and Computing Science in 2010 and the Ph.D. degree in Control Science and Engineering in 2016, both from Beijing Institute of Technology, Beijing, China. Now she is an assistant professor in the School of Automation, Beijing Institute of Technology. Her research interests include model predictive control, distributed control, data-driven control, stochastic systems, and networked control systems.



Mark Cannon received an M.Eng. in Engineering Science in 1993, a D.Phil. in Control Engineering in 1998, both from the University of Oxford, and received the M.S. in Mechanical Engineering in 1995 from Massachusetts Institute of Technology. He is currently an associate professor of Engineering Science, Oxford University and a fellow of St. John's College, Oxford.



Fuwen Yang (M'03-SM'04) received the Ph.D degree in control engineering from Huazhong University of Science and Technology, China, in 1990.

He is currently an Associate Professor with Griffith School of Engineering at Griffith University, Australia. He was a Research Fellow with Brunel University and King's College London, U.K., a Professor with Fuzhou University and East China University of Science and Technology, China, and an Associate Professor with Central Queensland University, Australia. He had also held Visiting Professor appointments with the University of Manchester, U.K., and the University of Hong Kong, China. He is an Associate Editor on the IEEE CSS Conference Editorial Board and a senior member of IEEE. He has published more than 200 journal and conference papers.

His main research interests include networked control systems, distributed filtering and sensing, reliable fault detection and diagnosis, distributed control and filtering for smart grids, and solar PV power generation systems.



Shuhao Yan received the B.Eng degree in Electrical and Electronic Engineering from City, University of London, UK, in 2015, the B.Eng degree in Automation from Nanjing University of Aeronautics and Astronautics, China, in 2015 and the M.Sc degree in Control Systems from Imperial College London, UK, in 2016. He is currently a DPhil student at the Department of Engineering Science, University of Oxford, UK.