Self-triggered MPC with Performance Guarantee for Tracking Piecewise Constant Reference Signals

Liang Lu, Daniel Limon and Ilya Kolmanovsky

Abstract— **This paper considers a self-triggered MPC controller design strategy for tracking piecewise constant reference signals. The proposed triggering scheme is based on the relaxed dynamic programming inequality and the idea of reference governor; such a scheme computes both the updated control action and the next triggering time. The resulting self-triggered tracking MPC control law preserves stability and constraint satisfaction and also satisfies certain a priori chosen performance requirements without the need to impose stabilizing terminal conditions. An illustrative example shows the effectiveness of this self-triggered tracking MPC implementation.**

I. INTRODUCTION

In conventional Model Predictive Control (MPC) [18], [19] an open-loop optimal control problem is solved at each sampling time resulting in a sequence of control inputs of which only the first one is implemented. This leads to an algorithmically defined feedback control law.

The classical MPC can be viewed as "time-driven" because the control input profile is updated repeatedly after each fixed time interval. Specifically, at each sampling time a sequence of control values is computed, but only the first element is applied to the system while the rest is discarded. One may ask whether we can continue using the sequence we have computed for as long as possible while still guaranteeing stability and required performance. The answer leads to the so-called "event-driven" MPC in which the control computations only happen when some prescribed "event" occurs. This reduces the frequency of MPC updates and average computing power required. For instance, in [2], [3], the MPC computation is activated by comparing the measured state and its past prediction. However, this requires continuously taking measurements and monitoring the system.

In self-triggered MPC, the necessary measurement and computation only take place at a triggered time at which both the updated MPC control actions and the next triggering time are determined. In particular, in [7] a self-triggered linear quadratic control (LQR) strategy is developed for linear systems without constraints. The paper [11] considers a self-triggered receding horizon controller for multiple-loop unconstrained linear time-invariant (LTI) systems and proposes a co-design of control and sensor sampling strategy. For self-triggered MPC of constrained systems, in [8], [12], [16], [23], the control law and triggering conditions are co-designed to satisfy a specified closed loop performance requirement. In particular, in [8], a self-triggered scheme is derived to get "group sparse" control signals by holding the control value at the triggering time to be constant for as long as possible. A similar idea is also pursued in [12], where control signals are kept the same between the triggering time instants. However, keeping the control signal constant may cause the control and state mismatch as the state is evolving with time. Thus, consecutive updates may result which is undesirable. This kind of consecutive triggerings also happen when uncertainty is present [23] and it is necessary to guarantee a specified performance.

In order to guarantee recursive feasibility in MPC setting most of the existing results including [2], [3], [8], [12], [23] exploit terminal cost and terminal set constraints. However, making the MPC controller satisfy terminal constraints can degrade performance, and even cause infeasibility in engineering practice.

In comparison, the recent paper by Lu and Maciejowski [16] provides an alternative approach for self-triggered MPC, which makes full use of non-constant control sequences MPC computes at triggering times and can maintain stability and satisfy certain performance requirements without terminal constraints and penalties. Unlike [2], [3], [8], [12], [23], where terminal cost and terminal constraints are used, the asymptotic stability is ensured in [16] by exploiting the relaxed dynamic programming (RDP) inequality without terminal constraints [9], [10], [15]. Furthermore, the occurrence of consecutive updates is significantly reduced by introducing an extra slack variable in the RDP condition.

While all the exisiting self-triggered MPC schemes have been extensively studied in the regulation case, i.e. when the goal is to control the state of the system to the origin (a fixed setpoint), a control law and triggering condition co-design for reference tracking has not been addressed when setpoint is changing with time, e.g., is a piecewise constant function in time. The synthesis approach of stabilizing MPC may not be viable at the new setpoint and the constraints could be violated [4], [13], [14]. For these reasons, under piecewise constant reference signals, a regulation self-triggered MPC, even with the extra slack variable technique introduced [16], may exhibit consecutive updates (cf. Fig. 3), which should be avoided to guarantee certain inter-event time that is required

Liang Lu is with Ningbo Research Institute, Zhejiang University, Ningbo, P. R. China liangup@gmail.com

Daniel Limon is with the Departamento de Ingeniería de Sistemas y Automática, Escuela Técnica Superior de Ingeniería, Universidad de Sevilla, Sevilla, Spain dlm@us.es

Ilya Kolmanovsky is with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, United States ilya@umich.edu

This work was supported by the Agencia Estatal de Investigacin (AEI)-Spain under Grant PID2019-106212RB-C41/AEI/10.13039/501100011033. The third author acknowledges support by the U.S. National Science Foundation Award CMMI 1904394.

by wireless networked control systems (WNCS) [17], [21]. Otherwise, if consecutive updates exist, the MPC updates must be computed intensively during some periods. The merits of event-driven MPC are lost.

In this paper, a reference governor [5], [6] is integrated into a self-triggering MPC scheme to govern the tail of shifted MPC sequences and guarantee constraint satisfaction, stability at the equilibrium and performance in a reference command tracking setting.

The organization of the paper is as follows. In Section II, we give definitions and preliminary results that will be used in the rest of the paper to formulate the self-triggered tracking MPC problem. In Section III, the relaxed dynamic programming approach is proposed with an integration of reference governor design which for piecewise constant in time reference commands is able to reduce the number consecutive triggering times. An illustrative example is presented in Section IV, and we conclude the paper in Section V.

Notation: Let $\mathbb{R}$, $\mathbb{R}_+$, $\mathbb{Z}$ and $\mathbb{Z}_+$ denote the set of real numbers, non-negative real numbers, integers, and non-negative integers, respectively, and let $\mathbb{Z}_{[a,b)}$ denote the set $\{\phi \in \mathbb{Z} \mid a \leq \phi < b\}$. Throughout this paper, t denotes sampling time, and k denotes the count of time-steps within the prediction horizon. Given two sets $\mathscr{X}$, $\mathscr{Y} \subseteq \mathbb{R}^n$, the Minkowski set addition is defined by $\mathscr{X} \oplus \mathscr{Y} := \{x+y|x \in \mathscr{X}, y \in \mathscr{Y}\}$. The Pontryagin set difference is defined by $\mathscr{X} \ominus \mathscr{Y} := \{z|z \oplus \mathscr{Y} \subseteq \mathscr{X}\}$. The ball of radius ϵ is denoted by $\mathscr{B}(\epsilon) = \{x \in \mathbb{R}^n : \|x\| \leq \epsilon\}$. For a given set $\mathscr{P}$ containing the origin, we let $\text{int}_\epsilon(\mathscr{P})$ denote the ϵ-interior of $\mathscr{P}$, i.e., $\text{int}_\epsilon(\mathscr{P}) \triangleq \mathscr{P} \ominus \mathscr{B}(\epsilon)$. Finally $\|x\|_Q^2 := \frac{1}{2}x^\text{T}Qx$.

II. PROBLEM SETUP

Consider a linear system,

$$x(t+1) = Ax(t) + Bu(t), \quad x(0) = \bar{x}, \tag{1}$$

$$y(t) = Cx(t) + Du(t) - g(t), \tag{2}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^p$ and $g(t) \in \mathbb{R}^p$ are the state, input, (generalized) output and external signal at time instant t, respectively. The convex sets $\mathbb{X} \subseteq \mathbb{R}^n$ and $\mathbb{U} \subseteq \mathbb{R}^m$ are closed sets which represent state and input constraints and contain the origin in their interiors. We assume that:

(i) (A, B) is stabilizable.
(ii) Matrix D is of full column rank.

For the tracking problem, we can view $g(t)$ as an *artificial setpoint*, g_sp, or an *artificial reference function*, for instance, $g_\text{sp} = Cx_\text{s} + Du_\text{s}$ where (x_s, u_s) is a steady state and input pair which guarantees the artificial setpoint g_sp is equal (or closest) to the desired setpoint r_sp.

For unconstrained systems, g_sp would essentially coincide with and be set to the desired reference setpoint r_sp. Then, from (1)-(2), the steady state satisfies

$$\begin{bmatrix} I-A & -B \\ C & D \end{bmatrix} \begin{bmatrix} x_\text{s} \\ u_\text{s} \end{bmatrix} = \begin{bmatrix} 0 \\ g_\text{sp} \end{bmatrix}. \tag{3}$$

The matrix in the left-hand side of equation (3) is an $(n+p)\times(n+m)$ matrix. From linear algebra, for the linear equation (3) to have a solution for all g_sp, it is sufficient that the rows of the matrix on the left-hand side are linearly independent, which requires $p \leq m$. However, normally we have more (generalized) outputs than manipulated references. So we choose a matrix H and set $g_c = Hg$ to select particular linear combinations of the generalized outputs. The variable $g_c \in \mathbb{R}^{n_c}$ is referred to as the *controlled variable*. In particular, if the columns of the matrix on the left-hand side are linearly independent, the linear equation (3) has a unique solution. If the solution is non-unique, the steady state pair (x_s, u_s) can be determined by solving an optimization problem

$$\begin{aligned} \min_{x_\text{s},u_\text{s},g_\text{sp}} \quad & (u_\text{s} - u_\text{sp})^\text{T} R_\text{s} (u_\text{s} - u_\text{sp}) + \phi(Hg_\text{sp} - r_\text{sp}), \\ \text{s.t.} \quad & \begin{bmatrix} I-A & -B \\ HC & HD \end{bmatrix} \begin{bmatrix} x_\text{s} \\ u_\text{s} \end{bmatrix} = \begin{bmatrix} 0 \\ Hg_\text{sp} \end{bmatrix}, \\ & x_\text{s} \in \text{int}_\epsilon(\mathbb{X}), \\ & u_\text{s} \in \text{int}_\epsilon(\mathbb{U}), \end{aligned} \tag{4}$$

where the first term penalizes the control effort w.r.t the desired steady input u_sp and $\phi(Hg_\text{sp} - r_\text{sp}) = \|Hg_\text{sp} - r_\text{sp}\|_\infty$ penalizes the deviation between the desired setpoint r_sp and the artificial setpoint g_sp.

The set of admissible setpoints such that the constraints are not active is defined as follows:

$$\mathscr{G}_\text{sp} = \{g_\text{sp} = Cx_\text{s} + Du_\text{s} : x_\text{s} \in \text{int}_\epsilon(\mathbb{X}),\ u_\text{s} \in \text{int}_\epsilon(\mathbb{U})\}.$$

A. Finite-horizon Tracking MPC

The objective of the reference tracking problem is to steer the output $y(t)$ to zero, while keeping Hg_sp as close as possible to r_sp. Here g_sp can be computed based on (4).

In tracking MPC, we take a finite horizon $N \in \mathbb{Z}_+$ and solve the following optimization problem at each sampling time, t:

$$\begin{aligned} \min_{\mathbf{u} \triangleq [u_0^\text{T}, \cdots, u_{N-1}^\text{T}]^\text{T}} \quad & J^{(N)}(\bar{x}, \bar{g}, \mathbf{u}) \triangleq \sum_{k=0}^{N-1} \|y_k\|_2^2, \\ \text{s.t.} \quad & x_k \in \mathbb{X}, \quad k = 1, \ldots, N, \\ & u_k \in \mathbb{U}, \quad k = 0, 1, \ldots, N-1, \\ & x_0 = \bar{x}, \\ & g_t = \bar{g}, \\ & x_{k+1} = Ax_k + Bu_k, \quad k = 0, 1, \ldots, N-1, \\ & y_k = Cx_k + Du_k - g_t, \quad k = 0, 1, \ldots, N-1. \end{aligned} \tag{5}$$

Solving the above optimization problem at each sampling time for a particular $\bar{x}$ and $\bar{g}$ leads to a unique sequence of optimal control laws from time t to time $t+N-1$, given by $\mathbf{u}^*(\bar{x}, \bar{g}) = [u_0^{*\text{T}}(\bar{x}, \bar{g}), u_1^{*\text{T}}(\bar{x}, \bar{g}), \ldots, u_{N-1}^{*\text{T}}(\bar{x}, \bar{g})]^\text{T}$.

The (finite-horizon) value function is defined as

$$V^{(N)}(x(t), g(t)) \triangleq J^{(N)}(x(t), g(t), \mathbf{u}^*). \tag{6}$$

The tracking MPC control law is given by applying the first control move of the open-loop optimal control sequence

$\mathbf{u}^*(x(t), g(t))$ to the system, i.e.

$$u(t) = \mu(x(t), g(t)) := u_0^*(x(t), g(t)). \quad (7)$$

Then the closed-loop system is given by

$$x(t+1) = Ax(t) + B\mu(x(t), g(t)), \quad (8)$$

$$y_\mu(t) = Cx(t) + D\mu(x(t), g(t)) - g(t). \quad (9)$$

B. Recursive Feasibility

In this subsection we review the background definitions and results from [16] needed for the subsequent developments.

Definition 1: A control sequence $\mathbf{u} = \{u(0), u(1), \ldots, u(N-1)\}$ is said to be *admissible* for $x(0) \in \mathbb{X}$, if $(x(t), u(t)) \in \mathbb{X} \times \mathbb{U}$ for all $t \in \{0, 1, \ldots, N-1\}$. The set of all admissible control sequences of length N is denoted by $\mathscr{U}^N(x(0))$.

The N step feasible region is defined as

$$\mathbb{I}_N := \{x \in \mathbb{X} : \mathscr{U}^N(x) \neq \emptyset\}.$$

The region $\mathbb{I}_\infty$ is called *viability kernel* [1], which characterizes the set of the infinite horizon feasible initial conditions of system (1) subject to input and state constraints.

Remark 1: Any admissible equilibrium point x_s is in the viability kernel. If the initial state is in an equilibrium point, then the proposed tracking MPC will be feasible.

The sequence of feasible sets $\mathbb{I}_N$'s becomes *stationary*, if there exists $N_0 \in \mathbb{Z}_+$, such that $\mathbb{I}_N = \mathbb{I}_{N_0}$ holds for all $N \geq N_0$.

Definition 2: A set $\mathscr{P} \subseteq \mathbb{R}^n$ is called a *(controlled) positively invariant (PI) set* or a *viable set* for the closed-loop system (1), if $\mathscr{P} \subseteq \mathbb{X}$ and for all $x \in \mathscr{P}$, there is a $u \in \mathbb{U}$, such that $Ax + Bu \in \mathscr{P}$.

$\mathbb{I}_\infty$ is also called the *maximal positively invariant (MPI) set*, which includes all the possible PI set $\mathscr{P}$, i.e. $\mathscr{P} \subseteq \mathbb{I}_\infty$.

Definition 3: A set $\mathscr{P}$ is called *RH N-invariant* or *recursively feasible with respect to a horizon* $N \in \mathbb{Z}_+$ if $\mathscr{P} \subseteq \mathbb{I}_N$ is a PI set for the closed-loop system (8) under the MPC controller (7) with a receding horizon (RH) N, i.e.

$$x(0) \in \mathscr{P} \Rightarrow x(t) \in \mathscr{P}, \quad \forall t \in \mathbb{Z}_+.$$

The following proposition shows that for a sufficiently large horizon N, MPC controller will generate recursive feasibility on the whole feasibility kernel $\mathbb{I}_\infty$. This property is inferred from *stationarity* of the feasible sets $\mathbb{I}_N$'s [1].

Proposition 1: Suppose that $g_{\text{sp}} \in \mathscr{G}_{\text{sp}}$ in (5). If $V^{(\infty)}(x(t), g_{\text{sp}}) < c$ holds for some $c \in \mathbb{R}_+$ and all $x(t) \in \mathbb{I}_\infty$ and for all $g_{\text{sp}} \in \mathscr{G}_{\text{sp}}$, the feasible sets $\mathbb{I}_N$'s become stationary for some $N_0 \in \mathbb{Z}_+$, i.e., $\mathbb{I}_{N_0} = \mathbb{I}_{N_0+1} = \mathbb{I}_{N_0+2} = \cdots = \mathbb{I}_\infty$.

Assumption 1: In this paper it is assumed that

$$\sup_{x \in \mathbb{I}_\infty, g \in \mathscr{G}_{\text{sp}}} V^{(\infty)}(x, g) = c < \infty.$$

For a given horizon N and a positive scalar ν, in order to determine a RH N-invariant set, we define the sub-level $S_\nu^N \subset \mathbb{R}^n \times \mathbb{R}^{n_c}$ of finite horizon value function $V^N(x, g)$

$$S_\nu^N = \{(x, g) \in \mathbb{I}_N \times \mathbb{R}^{n_c} : V^{(N)}(x, g) \leq \nu\}.$$

C. Relaxed Dynamic Programming

We use the relaxed dynamic programming result in [15] to develop a triggering condition for self-triggered MPC.

The next proposition is a variant of the main proposition stated in [10], [15] for approximating the Bellman's equation based on the finite-horizon value function $V^{(N)}(x(t), g(t))$ defined in Section II-A and its corresponding optimal control policy $\mu(x(t), g(t))$.

Proposition 2: Consider the system (1)-(2) with the feedback control law $\mu(x, g)$, and suppose that the following inequality is satisfied:

$$V^{(N)}(x(t), g(t)) \geq V^{(N)}(x(t+1), g(t+1)) + \alpha \|y_\mu(t)\|_2^2, \quad (10)$$

for a given scalar $\alpha \in (0, 1]$ and all $(x(t), g(t)) \in S_\nu^N$. Then,

$$\alpha \sum_{t=0}^{\infty} \|y_\mu(t)\|_2^2 \leq V^{(\infty)}(x(0)), \quad (11)$$

where $x(t+1)$ and $y_\mu(t)$ are obtained by applying $\mu(x(t), g(t))$ to the closed-loop system, i.e., $x(t+1) = Ax(t) + B\mu(x(t), g(t))$ and $y_\mu(t) = Cx(t) + D\mu(x(t), g(t)) - g(t)$.

Throughout this paper, we make the following assumption.

Assumption 2: The control horizon $N \geq N^*$ is known for which the RDP inequality (10) is satisfied for a specified $\alpha \in (0, 1]$ and all $(x(t), g(t)) \in S_\nu^N$.

While under reasonable assumptions on the system (1), we can ensure N^* exists, it may be difficult to compute it a priori. Our triggering mechanism will detect if N was chosen too small, such that it can be adapted [16].

D. Reference Governor

Consider a control law for tracking a constant reference $r(t) = r_{\text{sp}}$, of the form,

$$u(t) = Kx + \Gamma r, \quad (12)$$

where K is a feedback gain matrix such that $A + BK$ is Schur and Γ is a feedforward gain.

The constraints $(x(t), u(t)) \in \mathbb{X} \times \mathbb{U}$ can be expressed as inequality constraints

$$Ex(t) + Fu(t) \leq h, \quad t \geq 0 \quad (13)$$

Applying the control law (12) and replacing the desired reference $r(t)$ with $v(t)$ to the system (1), the closed loop system has the form,

$$x(t+1) = \Phi x(t) + Gv(t), \quad x(0) = \bar{x}, \quad (14)$$

where $\Phi = A + BK$ is Schur and $G = B\Gamma$. The inequality constraints (13) can be restated as

$$\begin{bmatrix} \Psi & \Theta \end{bmatrix} \begin{bmatrix} v(t) \\ x(t) \end{bmatrix} \leq h, \quad t \geq 0 \quad (15)$$

where $\Theta = E + FK$ and $\Psi = F\Gamma$.

The reference governor [5] behaves as a pre-filter which, based on the current state $x(t)$ and the desired reference $r(t)$, generates a modified reference $v(t)$ which fulfills the constraints $(x(t), u(t)) \in \mathbb{X} \times \mathbb{U}$. The updates for $v(t)$ take the form

$$v(t) = v(t-1) + \kappa(t)(r(t) - v(t-1)), \tag{16}$$

where the scalar $\kappa(t) \in [0,1]$ is chosen by solving the optimization problem,

$$\begin{aligned} \kappa(t) \triangleq \max_{\kappa \in [0,1]} \ & \kappa \\ \text{s.t.} \ & v = v(t-1) + \kappa(r(t) - v(t-1)), \\ & (v, x(t)) \in \tilde{O}_\infty, \end{aligned} \tag{17}$$

where $\tilde{O}_\infty$ is a finitely determined approximation to maximum output admissible set [6] for system (14)-(15).

III. RDP-Based Approach for Tracking

In this section, we will adapt the relaxed dynamic programming inequality in Proposition 2 to the self-triggered tracking MPC setting.

Define the triggering times $\{t_l \mid l \in \mathbb{Z}_+\}$, which satisfy $t_{l+1} > t_l$ for all $l \in \mathbb{Z}_+$ and $t_{l+1} - t_l < N$. Within the time interval $[t_l, t_{l+1})$, we set

$$u(t) = \tilde{\mu}(t, x(t_l), g(t_l)) := u^*_{(t-t_l)}(x(t_l), g(t_l)), t \in \mathbb{Z}_{[t_l, t_{l+1})}. \tag{18}$$

When MPC update (5) is triggered at time t_l, we have to decide on both the control and the next triggering time t_{l+1} which should be as large as possible while reference tracking is achieved and a certain required performance is guaranteed. The computation of t_{l+1} will be based on checking of the RDP inequality and for the setpoint changes.

In the self-triggered tracking MPC setting, multiple open-loop control moves of MPC sequence at time t_l may be applied before the next MPC update at time t_{l+1} is executed. We keep $g(t_l)$ constant in-between the triggering times t_l and t_{l+1}, and we amend the RDP condition as follows:

$$\begin{aligned} V^{(N)}(x(t_l), g(t_l)) \geq & V^{(N)}(x(t_{l+1}), g(t_{l+1})) \\ & + \alpha \sum_{t=t_l}^{t_{l+1}-1} \|y_{\tilde{\mu}}(t)\|_2^2, \end{aligned} \tag{19}$$

where $\sum_{t=t_l}^{t_{l+1}-1} \|y_{\tilde{\mu}}(t)\|_2^2$ denotes the sum of the running costs at the triggering times $t_l, t_l + 1, \ldots, t_{l+1} - 1$ with the control policy $u(t)$ defined as in (18). As the optimal value $V^{(N)}(x(t_{l+1}), g(t_{l+1}))$ at the next triggering time is not available at t_l, we will construct an upper bound for it. Besides, we will also exploit an extra slack variable which reflects the decay of the Lyapunov function $V^{(N)}(x(t), g(t))$ at the previous triggering times. The main theorem of this paper is presented below. It demonstrates that after all of the above mentioned modifications to the RDP inequality, a certain bound on performance and reference tracking are still guaranteed.

Theorem 1: Suppose $(x(0), g(0)) \in S_\nu^N$ and an upper bound $\bar{V}^{(N)}(x(t), g(t))$ can be found for $t \in \{t_l \mid l \in \mathbb{Z}_+\}$ such that

$$\bar{V}^{(N)}(x(t), g(t)) \geq V^{(N)}(x(t), g(t)). \tag{20}$$

Suppose, furthermore, the inequality

$$\begin{aligned} V^{(N)}(x(t_l), g(t_l)) - \bar{V}^{(N)}(x(t_{l+1}), g(t_l)) \geq \\ e(t_l) + \alpha \sum_{t=t_l}^{t_{l+1}-1} \|y_{\tilde{\mu}}(t)\|_2^2, \end{aligned} \tag{21}$$

is enforced for a given scalar $\alpha \in (0,1)$, where the sequence $\{e(t_l)\}$ is defined in (22) for all $l \geq 2$ and if $g(t_l) \neq g(t_{l-1})$ or $t = t_1$, we set $\bar{V}^{(N)}(x(t_l), g(t_{l-1})) = V^{(N)}(x(t_l), g(t_l))$. Then:

$$\alpha \sum_{t=t_0}^{\infty} \|y_{\tilde{\mu}}(t)\|_2^2 \leq \sup_{g \in \mathscr{G}_{\mathrm{sp}}} V^{(\infty)}(x(t_0), g) = c, \tag{23}$$

and

$$\lim_{t \to \infty} y_{\tilde{\mu}}(t) = 0. \tag{24}$$

Proof. The proof is an extension of Theorem 1 in [16] for tracking piecewise constant signals and is omitted due to lack of space. □

At an MPC update time $t_l \in \mathbb{Z}_+$ with $l \in \mathbb{Z}_+$, we compute the MPC control update according to (5). In order to implement our RDP-based triggering scheme, after obtaining $\mathbf{u}^*(x(t_l), g(t_l)) = [u_0^{*\intercal}(x(t_l), g(t_l)), u_1^{*\intercal}(x(t_l), g(t_l)), \ldots, u_{N-1}^{*}{}^{\intercal}(x(t_l), g(t_l))]^\intercal$ at time t_l, the first step is to find the last component $u^*_{\bar{N}-1}$ in $\mathbf{u}^*$ sequence such that $x(t_l + \bar{N}) \in \mathit{Proj}_x(P)$, where $\bar{N} \in \mathbb{Z}_{[1,N-1]}$ and P is a closed set satisfying

$$P \subseteq \tilde{O}_\infty, \tag{25}$$

which can be expressed as set of linear inequalities of the form

$$P = \{(v, x) : M_x x + M_v v \leq b\}. \tag{26}$$

The next MPC update time t_{l+1} can be calculated by

$$t_{l+1} = t_l + \mathscr{N}_{t_l}(x(t_l)), \tag{27}$$

where the inter-triggering interval $\mathscr{N}_{t_l}(x(t_l))$ is given by

$$\mathscr{N}_{t_l}(x(t_l)) \triangleq \max\{N_{t_l} \in \mathbb{Z}_{[1,\bar{N}-1]}\} \tag{28}$$

$$\begin{aligned} \text{s.t.} \ (i) \ & V^{(N)}(x(t_l), g(t_l)) - \bar{V}^{(N)}(x(t_l + N_{t_l}), g(t_l)) \\ & \geq e(t_l) + \alpha \left(\sum_{t=t_l}^{t_l+N_{t_l}-1} \|y_{\tilde{\mu}}(t)\|_2^2 \right), \end{aligned} \tag{29}$$

$$(ii) \ g(t_l + N_{t_l}) = g(t_l). \tag{30}$$

In order to calculate the upper bound $\bar{V}^{(N)}(x(t_l + N_{t_l}), g(t_l))$ and the forward predicted state $\bar{x}(t_l + \bar{N} + i)$ for $i \in \mathbb{Z}_{[1,N_{t_l}]}$ at time t_l, we construct an input sequence $\bar{U}_N(x(t_l + N_{t_l})) = [u^*_{N_{t_l}}{}^{\intercal}(x(t_l), g(t_l)), \ldots, u^*_{\bar{N}-1}{}^{\intercal}(x(t_l), g(t_l)), \bar{u}^\intercal(t_l + \bar{N}), \ldots, \bar{u}^\intercal(t_l + N_{t_l} + \bar{N} - 1)]^\intercal$.

For a given $x(t) \in \mathit{Proj}_x(P)$, we can determine $v(t)$ to guarantee $(v(t), x(t)) \in P \subseteq \tilde{O}_\infty$, for instance, by solving the following QP:

$$\begin{aligned} \min_{v} \ & (v(t) - g(t))^\intercal (v(t) - g(t)), \\ \text{s.t.} \ & M_x x(t) + M_v v(t) \leq b. \end{aligned} \tag{31}$$

$$e(t_l) = \begin{cases} 0, & \text{if } g(t_l) \neq g(t_{l-1}) \text{ or } t = t_0, \\ e(t_{l-1}) + \alpha \sum_{t=t_{l-1}}^{t_l - 1} \|y_{\bar{\mu}}(t)\|_2^2 + \bar{V}^{(N)}(x(t_l), g(t_{l-1})) - \bar{V}^{(N)}(x(t_{l-1}), g(t_{l-2})), & \text{otherwise.} \end{cases} \tag{22}$$

To get $\bar{u}(t_l + \bar{N} + i - 1), \quad i \in \mathbb{Z}_{[1,N_{t_l}]}$, we set $v(t_l + \bar{N})$ from (31) and solve (17) for $v(t_l + \bar{N} + i)$ for $i \in \mathbb{Z}_{[1,N_{t_l}]}$. The sequence $\bar{x}(t_l + \bar{N} + i)$ can be obtained by forward simulation.

Hence, the upper bound can be defined as

$$\bar{V}^{(N)}(t_l + N_{t_l}, x(t_l + N_{t_l})) \triangleq J^{(N)}(t_l + N_{t_l}, x(t_l + N_{t_l}), \bar{U}_N(x(t))). \tag{32}$$

IV. Illustrative Example

We consider a helicopter flight envelope protection example studied in [20], [22]. The linearized continuous-time model for the helicopter dynamics is described by

$$\dot{x} = A_c x + B_c u,$$

where five states and one input are:

- γ: forward speed;
- q: pitch rate;
- θ: pitch angle;
- a: pitch angle of the virtual rotor disc;
- c: angle of the rotor stabilizer bar;
- δ_s: swash plate angle;

and

$$A_c = \begin{bmatrix} -0.0505 & 0 & -9.81 & -9.81 & 0 \\ -0.0561 & 0 & 0 & 82.6 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & -21.7391 & 14 \\ 0 & -1 & 0 & 0 & -0.342 \end{bmatrix},$$

$$B_c = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -2.174 \\ -0.7573 \end{bmatrix}, \quad x = \begin{bmatrix} v \\ q \\ \theta \\ a \\ c \end{bmatrix}, \quad u = \delta_s.$$

The discrete-time linear model of (1) is obtained assuming a sampling frequency of 60Hz. As the problem is a tracking problem, the output is chosen to be $y_k = Cx_k + Du_k - g_{k+t}$, where

$$C = \begin{bmatrix} 7.0711 & 0 & 0 & 0 & 0 \\ 0 & 0.3162 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 3.1623 & 0 \\ 0 & 0 & 0 & 0 & 0.3162 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$HC = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix}, HD = 0.$

The state and input constraints are enforced within the ranges

$$\begin{bmatrix} -5 \\ -5 \\ -3 \\ -1 \\ -2 \end{bmatrix} \leq x \leq \begin{bmatrix} 10 \\ 5 \\ 3 \\ 1 \\ 2 \end{bmatrix}, -5 \leq \delta_s \leq 5.$$

The control horizon is chosen as $N = 40$ and the performance degradation parameter as $\alpha = 0.7$. The simulation results are presented in Figs. 1-2.

If we do not add the self-triggering mechanism to the MPC algorithm, it takes 510 MPC updates. The results by the Lu et al. [16] are shown in the first and third rows in Fig. 1, which has 78 MPC updates. The response with the proposed triggering scheme are shown in the second and fourth rows in Fig. 1. In this case, MPC only needs 19 updates. Hence, the proposed strategy can significantly reduce the number of MPC update times while achieving reference tracking. The triggering instants are recorded in Fig. 3.

V. CONCLUSIONS

This paper proposed a self-triggered tracking MPC co-design procedure for constrained linear systems based on the relaxed dynamic programming inequality and reference governor scheme. The inter-triggering time is maximized by governing the tails of shifted control sequences for constructing triggering conditions of tracking MPC such that the overall closed-loop system can not only maintain asymptotic stability, but also achieve a certain prescribed performance level. The illustrative example showed that the number of consecutive updates in the self-triggered tracking MPC is significantly reduced compared to the existing self-triggered MPC schemes for regulation problem. An extension of the idea to robust case is being explored currently.

References

[1] A. Boccia, L. Grüne, and K. Worthmann. Stability and feasibility of state constrained MPC without stabilizing terminal constraints. *Systems & Control Letters*, 72:14–21, 2014.

[2] A. Eqtami, D. V. Dimarogonas, and K. J. Kyriakopoulos. Event-triggered strategies for decentralized model predictive controllers. *IFAC Proceedings Volumes*, 44(1):10068–10073, 2011.

[3] A. Eqtami, D. V. Dimarogonas, and K. J. Kyriakopoulos. Event-based model predictive control for the cooperation of distributed agents. In *2012 American Control Conference (ACC)*, pages 6473–6478. IEEE, 2012.

[4] A. Ferramosca. *Model Predictive Control of Systems with Changing Setpoints. Control predictivo de sistemas con punto de operación cambiante*. Ph.D. dissertation, Univ. de Sevilla, Sevilla, Spain, 2011.

[5] E. Garone, S. Di Cairano, and I. Kolmanovsky. Reference and command governors for systems with constraints: A survey on theory and applications. *Automatica*, 75:306–328, 2017.

[6] E. G. Gilbert and I. Kolmanovsky. Fast reference governors for systems with state and control constraints and disturbance inputs. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 9(15):1117–1141, 1999.

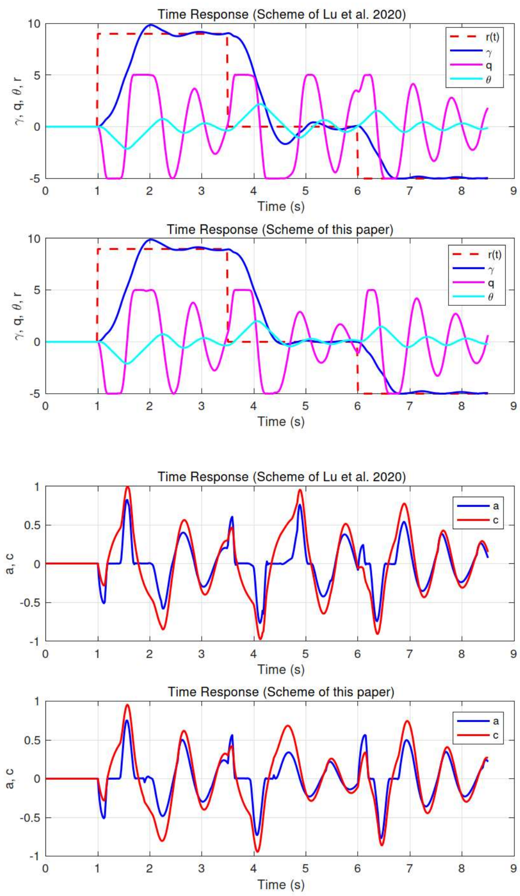

Fig. 1: State response trajectories with $\alpha = 0.7$. The upper figure shows the fuselage states and the reference (the speed reference is in red dash line, γ is in blue, q is in magenta and θ is in cyan), and the bottom figure shows the rotor states (a is in blue and c is in red).

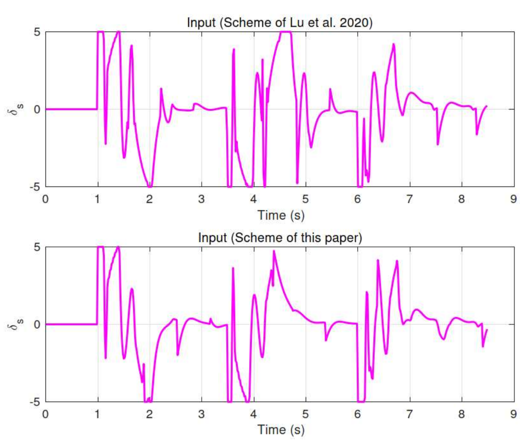

Fig. 2: Control input trajecory with $\alpha = 0.7$.

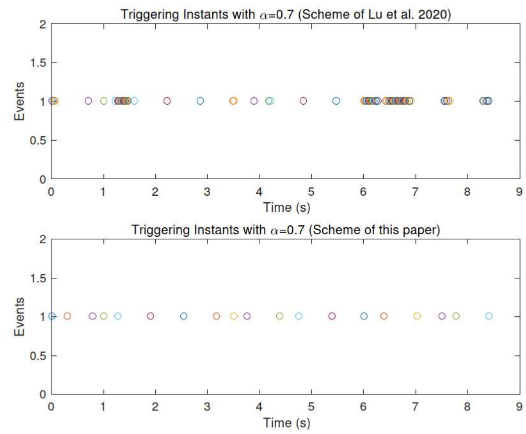

Fig. 3: Event triggering instants. The triggering instants are marked with the circles with the value 1.

[7] T. Gommans, D. Antunes, T. Donkers, P. Tabuada, and M. Heemels. Self-triggered linear quadratic control. *Automatica*, 50(4):1279–1287, 2014.

[8] T. Gommans and W. Heemels. Resource-aware mpc for constrained nonlinear systems: A self-triggered control approach. *Systems & Control Letters*, 79:59–67, 2015.

[9] L. Grüne and J. Pannek. Stability and suboptimality without stabilizing terminal conditions. In *Nonlinear Model Predictive Control*, pages 121–176. Springer, 2017.

[10] L. Grüne and A. Rantzer. On the infinite horizon performance of receding horizon controllers. *IEEE Transactions on Automatic Control*, 53(9):2100–2111, 2008.

[11] E. Henriksson, D. E. Quevedo, E. G. Peters, H. Sandberg, and K. H. Johansson. Multiple-loop self-triggered model predictive control for network scheduling and control. *IEEE Transactions on Control Systems Technology*, 23(6):2167–2181, 2015.

[12] H. Li, W. Yan, and Y. Shi. Triggering and control codesign in self-triggered model predictive control of constrained systems: With guaranteed performance. *IEEE Transactions on Automatic Control*, 63(11):4008–4015, 2018.

[13] D. Limón, T. Alamo, and E. F. Camacho. Mpc for tracking piecewise constant references for constrained linear systems. *Automatica*, 44(9):2382–2387, 2008.

[14] D. Limon, A. Ferramosca, I. Alvarado, and T. Alamo. Nonlinear mpc for tracking piece-wise constant reference signals. *IEEE Transactions on Automatic Control*, 63(11):3735–3750, 2018.

[15] B. Lincoln and A. Rantzer. Relaxing dynamic programming. *IEEE Transactions on Automatic Control*, 51(8):1249–1260, 2006.

[16] L. Lu and J. M. Maciejowski. Self-triggered MPC with performance guarantee using relaxed dynamic programming. *Automatica*, 114:108803, 2020.

[17] S. Lucia, M. Kögel, P. Zometa, D. E. Quevedo, and R. Findeisen. Predictive control, embedded cyberphysical systems and systems of systems–a perspective. *Annual Reviews in Control*, 41:193–207, 2016.

[18] J. M. Maciejowski. *Predictive Control: with Constraints*. Pearson education, 2002.

[19] D. Q. Mayne. Model predictive control: Recent developments and future promise. *Automatica*, 50(12):2967–2986, 2014.

[20] B. Mettler. *Identification modeling and characteristics of miniature rotorcraft*. Springer Science & Business Media, 2013.

[21] P. Park, S. C. Ergen, C. Fischione, C. Lu, and K. H. Johansson. Wireless network design for control systems: A survey. *IEEE Communications Surveys & Tutorials*, 20(2):978–1013, 2017.

[22] D. Simon. *Fighter Aircraft Maneuver Limiting Using MPC: Theory and Application*, volume 1881. Linköping University Electronic Press, 2017.

[23] K. Zhang, C. Liu, and Y. Shi. Self-triggered adaptive model predictive control of constrained nonlinear systems: A Min-Max approach. *arXiv preprint arXiv:1912.06978*.