

Lebesgue-Approximation Model Predictive Control of Nonlinear Sampled-Data Systems

Jie Tao , Lixing Yang , Zheng-Guang Wu , Xiaofeng Wang , Member, IEEE,
and Hongye Su , Senior Member, IEEE

Abstract—This article studies discrete-time implementation of model predictive control (MPC) algorithms in continuous-time nonlinear sampled-data systems. We present a discrete-time and aperiodic nonlinear MPC algorithm to stabilize continuous-time nonlinear dynamics, based on the Lebesgue approximation model (LAM). In this LAM-based MPC (LAMPC), the sampling instants are triggered by a self-triggered scheme, and the predicted states and transition time instants in the optimal control problem are calculated in an aperiodic manner subject to the LAM. Sufficient conditions are derived on feasibility and stability of the resulting closed-loop systems. According to these conditions, the parameters in LAMPC are designed with the guarantee of exclusion of Zeno behavior. Meanwhile, it is shown that the periodic task model is a special case in our framework with appropriate choice of the parameters in the LAM. Simulation results indicate that LAMPC can dynamically adjust the computation periods and has the potential to reduce the computational costs compared with periodic approaches.

Index Terms—Lebesgue approximation model (LAM), nonlinear model predictive control (NMPC), sampled-data systems.

I. INTRODUCTION

IN PAST decades, model predictive control (MPC) has established itself as an efficient tool to control constrained systems and has been used in a wide range of applications, such as process control, power grids, transportation systems, and manufacturing, to name a few [1]–[3]. It guarantees certain levels of optimality

Manuscript received September 17, 2018; revised April 25, 2019; accepted November 6, 2019. Date of publication November 13, 2019; date of current version September 25, 2020. This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 2018FZA5009, in part by the Science Fund for Creative Research Groups of the National Natural Science Foundation of China under Grant 61621002, in part by the National Natural Science Foundation of China under Grant 61903093, in part by the Open Research Project of the State Key Laboratory of Industrial Control, Technology, Zhejiang University, China, under Grant ICT1900311, and in part by the National Science Foundation under Grant IIS-1525900 and Grant ECCS-1739886. Recommended by Associate Editor M. Alamir. (Corresponding author: Zheng-Guang Wu.)

J. Tao, Z.-G. Wu, and H. Su are with the State Key Laboratory of Industrial Control Technology, Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou 310027, China (e-mail: jtiao@ipc.zju.edu.cn; nashwzhg@126.com; hysu@ipc.zju.edu.cn).

L. Yang and X. Wang are with the Department of Electrical Engineering, University of South Carolina, Columbia, SC 29208 USA (e-mail: lixing@email.sc.edu; wangxi@cec.sc.edu).

Color versions of one or more of the figures in this article are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TAC.2019.2953147

in the behavior of controlled systems subject to state and/or input constraints. Standard implementation of MPC predicts the (near) optimal control inputs based on a mathematical model that attempts to approximate the actual dynamical system of interest (though most of the time such a model cannot completely represent the actual dynamics).

In computer-controlled systems, even when the physical process is continuous-time, MPC algorithms have to be discrete-time, given the digital environment for implementation. Such discretization is twofold.

- 1) The time instants to sample the states and compute the optimal solution to a finite-horizon optimal control problem (FHOC) should be triggered in a discrete-time manner.
- 2) The computation of the FHOC itself, including the predictive model and the cost function, should be discretized.

Most of the existing approaches focus only on 1). Traditional methods often consider periodic sampling, in which the sampling period is fixed [4]–[8]. This approach could be conservative in applications with limited computation resources because it may trigger the computation of the solution to the FHOC more frequent than necessary and, therefore, lead to significant overprovisioning to the processor. Aperiodic sampled-data MPC, therefore, has received a lot of attentions recently, which can reduce the frequency of solving the FHOC. The work in [9] and [10] considers uncertain continuous-time linear systems. The maximum intersampling time interval must satisfy some linear matrix inequalities to ensure stability. A different approach is event-triggered MPC. In this case, the sampling instants are identified by occurrence of some predefined events [11]–[18]. Different from event-triggered methods, self-triggered MPC has the next sampling time instant expressed explicitly as a function of the past information [19]–[23]. In all this work, however, either discrete-time plant is considered, which automatically leads to a discrete-time model in the FHOC, or the FHOC directly takes continuous-time models. (In this case, the controller still needs to solve a continuous-time FHOC at each sampling instant, which, by itself, is computationally expensive).

The work considering both 1) and 2) focuses on linear time-invariant (LTI) systems with fixed sampling period [24], [25]. In [24], the FHOC was reformulated to guarantee constraint satisfaction using polytopic inclusions. Tubed-based approaches were studied in [25] for sampled-data implementation of MPC. Since both of them consider LTI systems, model discretization becomes simple, given the fact that a continuous-time LTI

system can be perfectly discretized without approximation errors at the sampling time instants. Using this discretized system model, one can directly develop discrete-time MPC algorithms, with some treatments on the errors during the intersampling period to avoid violations of the constraints. For nonlinear systems, however, this approach does not work because there is no perfect discretization for those systems in general. Even if using a discrete-time approximation model instead, the computational cost raises another concern, because a relatively accurate approximation model requires a small sampling period, which implies more frequent execution of the FHOCP.

This article focuses on nonlinear MPC (NMPC), investigating two questions: 1) how to schedule sampling and computation tasks in MPC for continuous-time nonlinear systems? 2) how to take advantage of aperiodic task models in prediction for possible computation reduction in solving the FHOCP? We address these questions through the introduction of the Lebesgue approximation model (LAM) [26] into NMPC, which presents aperiodic discretization of the continuous-time nonlinear dynamics. The basic idea of LAM is to approximate the system dynamics by updating the state and the related transition time instants together only when the predicted state meets some thresholds.¹ With the LAM, the prediction horizon, in the continuous-time framework, may potentially be longer than that in periodic task models, given the same number of steps in the FHOCP. In other words, the controller can optimize the system performance over a longer prediction horizon without increasing the number of steps in the FHOCP. To interpret it in a different way, given the same length of the prediction horizon, the LAM will take fewer state transitions than periodic task models, which can dramatically save computational costs when solving the discretized FHOCP.

The contributions of this article are listed as follows.

- 1) We present a completely discrete-time, aperiodic LAM-based MPC (LAMPC) algorithm to stabilize continuous-time nonlinear systems in the presence of state and input constraints. In this algorithm, sampling is triggered by a self-triggering scheme and the FHOCP is aperiodically discretized based on the LAM with modified constraint sets. To the best of our knowledge, this is the first work examining “completely” discrete-time and aperiodic NMPC for continuous-time nonlinear systems. “Completely” means that both the sampling instants and the FHOCP are discrete-time so that continuous-time behaviors can be eliminated in the controller. The aperiodic feature has the potential to generate longer intersampling time intervals and prediction horizons and, therefore, reduce computational costs. Our preliminary results on LAMPC were presented in [27].
- 2) Sufficient conditions are derived on feasibility and stability of the resulting closed-loop systems. We also show that under LAMPC, the state and input constraints will not be violated. Based on these conditions, threshold functions in the self-triggered sampling scheme and the LAM are developed without exhibitions of Zeno behavior.

¹ A fundamental difference between the LAM and event/self-triggering is that the latter only determines the time instants and the states are directly sampled from the plant, while the former updates both the state and the time instants, which is necessary for prediction.

- 3) As a special case of LAMPC, we study periodic task models in NMPC and derive the bound on the maximum allowable sampling period that meets the feasibility and stability conditions.
- 4) A benchmark example is studied to evaluate performance of the LAMPC algorithm. It is shown that LAMPC can generate longer intersampling time intervals and prediction horizons during the transience, compared with the periodic model. Meanwhile, it is robust to actuation delays.

The remainder of this article is organized as follows. Section II formulates the problem. The LAMPC algorithm is introduced in Section III. Feasibility and stability analysis can be found in Section IV. Section V derives the thresholds that meet the stability conditions developed in Section IV. Section VI shows a special case where the LAM becomes periodic. Simulation results are presented in Section VII. Section VIII summarizes the results. All proofs are given in the Appendix.

II. PROBLEM FORMULATION

Notations: We denote by $\mathbb{R}^n$ the n -dimensional real vector space, by $\mathbb{R}^+$ the set of the real positive numbers, and by $\mathbb{R}_0^+$ the set of the real non-negative numbers. We use $\|\cdot\|$ to denote the Euclidean norm of a vector and the induced 2-norm of a matrix. The symbol “e” denotes the exponential function. $\hat{\cdot}$ denotes the predicted variable. $\cdot^*$ denotes the optimal variable. For all $A, B \in \mathbb{R}^n$, $A \ominus B = \{z \in A : z + b \in A, \text{ for all } b \in B\}$, $\mathcal{B}_\epsilon = \{x \in \mathbb{R}^n : \|x\|_2 \leq \epsilon\}$ is the closed ball of $\|\cdot\|_2$ with radius $\epsilon > 0$. To simplify the notations in subsequent sections, for given scalar ϵ , we denote the set $\mathcal{X} \ominus \mathcal{B}_\epsilon$ as $\mathcal{X} - \epsilon$.

Definition 1: A continuous function $\alpha : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$ belongs to class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$. A function $\alpha : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$ belongs to class $\mathcal{K}_\infty$ if it belongs to class $\mathcal{K}$ and $\lim_{r \rightarrow \infty} \alpha(r) = \infty$.

Definition 2: The state $x(t)$ of a system $\dot{x} = f(x)$ is called *uniformly ultimately bounded (UUB)* with ultimate bound b if there exist positive constants b and c , independent of $t_0 \geq 0$, and for every $a \in (0, c)$, there is $T = T(a, b) \geq 0$, independent of t_0 , such that $\|x(t_0)\| \leq a$ implies $\|x(t)\| \leq b$ for any $t \geq t_0 + T$.

Consider a nonlinear control system

$$\dot{x}(t) = f(x(t), u(t)) \quad (1a)$$

$$x(t_0) = x_0 \quad (1b)$$

where $x : \mathbb{R}_0^+ \rightarrow \mathcal{X}$ is the system state, $u : \mathbb{R}_0^+ \rightarrow \mathcal{U}$ is the control input, and $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is a known continuous function satisfying $f(0, 0) = 0$. The compact sets $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{U} \subseteq \mathbb{R}^m$ describe the state and input constraints, respectively. In other words

$$x(t) \in \mathcal{X}, \mathcal{X} \text{ contains the origin} \quad (2)$$

$$u(t) \in \mathcal{U} \quad (3)$$

must hold for any $t \geq 0$. We assume that $f(x, u)$ is Lipschitz in x over $\mathcal{X}$ uniformly in u , i.e., there exists a positive constant L_f such that for any $x, y \in \mathcal{X}$ and any $u \in \mathcal{U}$, the following inequality holds:

$$\|f(x, u) - f(y, u)\| \leq L_f \|x - y\|. \quad (4)$$

To stabilize the system subject to the state and input constraints, we use MPC. Let t_k denote the time instant that triggers the k th sampling of the state. It can be mathematically defined by the following equation:

$$t_{k+1} = t_k + \phi(x(t_k), u(t_k)) \quad (5)$$

where $\phi : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^+$ is the function to predict the next sampling time instant.

The main idea of MPC is described as follows: At time t_k , the sensor samples the state $x(t_k)$. With the sampled state as the initial condition, the controller solves an FHOCPC over the time interval $[t_k, t_k + T_k]$, where T_k is the horizon length of the FHOCPC at the k th computation. The solution to the FHOCPC, i.e., the optimal control input, will be sent for actuation over the time interval $[t_k, t_{k+1})$, where $t_{k+1} \leq t_k + T_k$. Then, the horizon window will move to the next computation cycle, starting at t_{k+1} .

When implementing MPC in digital environments, the algorithm has to be discrete-time, which can be interpreted from two aspects: 1) the sampling time instants that trigger the FHOCPC must be discrete; and 2) the calculation of the optimal solution to the FHOCPC must be completely discrete and based on a discrete-time model.

Assume that the original continuous-time cost function of the FHOCPC is

$$J[\hat{u}_k | x(t_k)] = \int_{t_k}^{t_k + T_k} \ell(\hat{x}_k(\tau), \hat{u}_k(\tau)) d\tau + V_f(\hat{x}_k(t_k + T_k)) \quad (6)$$

where $\ell : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^+$ is the running cost function that is continuous, positive definite, and locally Lipschitz, and $V_f : \mathbb{R}^n \rightarrow \mathbb{R}^+$ is the terminal cost function. The predicted state and input, $\hat{x}_k(t) : \mathbb{R}^+ \rightarrow \mathbb{R}^n$ and $\hat{u}_k(t) : \mathbb{R}^+ \rightarrow \mathbb{R}^m$, are subject to the discrete-time model

$$\hat{x}_k(t_k^{i+1}) = \hat{f}(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i)) \quad (7a)$$

$$\hat{x}_k(t_k^0) = x(t_k) \quad (7b)$$

$$t_k^{i+1} = t_k^i + \hat{g}(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i)) \quad (7c)$$

$$t_k^0 = t_k \quad (7d)$$

where the to-be-determined functions $\hat{f}$ and $\hat{g}$ describe the transitions in state and time, respectively, and t_k^i represents the i th transition time instant at the k th computation of the FHOCPC. Notice that the predicted state $\hat{x}_k(t_k^i)$ is the prediction of $x(t_k^i)$, calculated at t_k . $t_k^{i+1} - t_k^i$ is called the i th “intertransition time interval” when computing the FHOCPC for the k th time, and $t_{k+1} - t_k$ is called the k th “intersampling time interval” (or “sampling period” if sampling is periodic).

Since the cost function in (6) needs continuous-time $\hat{x}_k(\tau)$ and $\hat{u}_k(\tau)$, the state and the input over $[t_k, t_{k+1})$ can be approximated using interpolation methods. In this article, we simply consider zero-order-hold (ZOH) as follows:

$$\hat{x}_k(t) = \hat{x}_k(t_k^i) \quad (8)$$

$$\hat{u}_k(t) = \hat{u}_k(t_k^i) \quad (9)$$

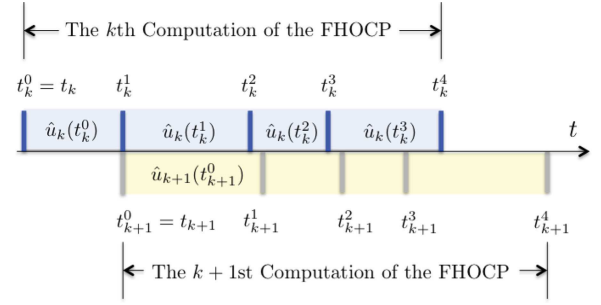


Fig. 1. Relation between t_k and t_k^i .

for any $t \in [t_k^i, t_k^{i+1})$. The discrete-time model in (7), together with the interpolation in (8), is expected to approximate the continuous-time plant in (1). The discretization of the cost function in (6) will be based on this discrete-time model with ZOH.

Remark 1: Once the function ϕ in (5) is defined, the time instant t_{k+1} will, then, be triggered by a self-triggering scheme [28], [29]. A special case is to set $\phi \equiv T_s$ where T_s is a positive constant. Then, calculation of the FHOCPC becomes periodically triggered. Also, the discrete-time model in (7) is general enough to include both aperiodic and periodic task models. If $\hat{g}(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i)) \equiv T_s$, the model becomes periodic. Notice that the model in (7) is different from self-triggering. In self-triggered control, only the time instants are iteratively calculated and the states are directly sampled from the plant, while in (7), both the state and the time instants are updated, which is necessary in prediction.

Remark 2: The relation between t_k and t_k^i is described in Fig. 1. Let $N \in \mathbb{N}$ be the prediction horizon of the discrete-time FHOCPC. Then, at t_k , the model in (7a) will be iterated for N steps to obtain the optimal solution. Accordingly, the optimal solution will generate a sequence of future time instants $t_k^0, t_k^1, \dots, t_k^N$, based on (7c) and (7d).

Remark 3: It is assumed in this article that there is no delay in sampling, solving the FHOCPC, and actuation. In this case, we can focus more on the LAMPC algorithm itself. A more practical assumption is to consider delays in the closed loop and investigate their impacts on system performance. In Section VII, we examine actuation delays through simulations. A more rigorous analysis on LAMPC of delayed systems will be provided in the future.

The objective of this article is to design the self-triggered scheduling scheme that triggers the FHOCPC and develop the discrete-time sporadic model used in the FHOCPC, such that the overall continuous-time system can be stabilized 1) without violating the state and input constraints and 2) in a computationally cost-efficient way.

III. LAMPC ALGORITHM

This section presents the LAMPC algorithm. The key is to identify the functions ϕ in (5) and $\hat{f}, \hat{g}$ in (7). Meanwhile, the parameters and the functions in the FHOCPC, such as T_k, V_f , and ℓ , also need to be appropriately chosen to ensure stability. First

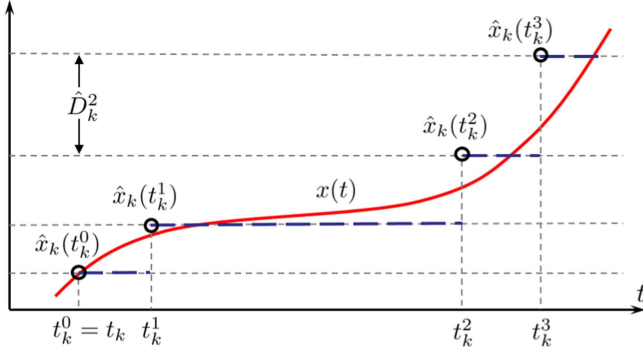


Fig. 2. State trajectory generated by LAM.

of all, let us introduce the LAM, which will be used in solving the FHOCP. It is aperiodic discretization of the continuous-time system in (1) in the case when $\|d\hat{x}_k^i\| > 0$, where $d\hat{x}_k^i \in \mathbb{R}^n$ is a simplified notation to denote $f(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i))$, i.e., $d\hat{x}_k^i = f(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i))$

$$\hat{x}_k(t_k^{i+1}) = \hat{x}_k(t_k^i) + \hat{D}_k^i \frac{d\hat{x}_k^i}{\|d\hat{x}_k^i\|} \quad (10a)$$

$$\hat{x}_k(t_k^0) = x(t_k) \quad (10b)$$

$$t_k^{i+1} = t_k^i + \frac{\hat{D}_k^i}{\|d\hat{x}_k^i\|} \quad (10c)$$

$$t_k^0 = t_k \quad (10d)$$

where $\hat{D}_k^i$ is the discretization (or quantization) level of the LAM in the i th transition at time t_k . For the case $\|d\hat{x}_k^i\| = 0$, it means $f(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i)) = 0$ and the predicted state reaches the equilibrium of the actual plant. In this case, we set $\hat{x}_k(t_k^{i+1}) = \hat{x}_k(t_k^i)$ and $t_k^{i+1} = t_k^i + \varpi_{\max}$, where $\varpi_{\max}$ is a positive constant that defines the maximum intertransition time interval for safety reasons. In the following discussion, we focus on the case when $\|d\hat{x}_k^i\| > 0$.

We approximate the states between $\hat{x}_k(t_k^i)$ and $\hat{x}_k(t_k^{i+1})$ by $\hat{x}_k(t_k^i)$ such that $\hat{x}_k(t)$ is continuous-time. It means $\hat{x}_k(t) = \hat{x}_k(t_k^i)$ and $\hat{u}_k(t) = \hat{u}_k(t_k^i)$ for any $t \in [t_k^i, t_k^{i+1})$.

Remark 4: The trajectory of the states generated by the LAM is plotted in Fig. 2. At time t_k , the model starts from the accurate measurement $\hat{x}_k(t_k^0) = x(t_k)$, which serves as the base to generate the approximated states. Once we have $\hat{x}_k(t_k^0)$, ZOH approximation is used to approximate $\hat{x}_k(t)$ until the next triggering time. Notice that the states generated by the LAM will not be exactly the same as the actual states but some predicted values. For an aperiodic model, a critical issue is related to the Zeno behavior [30], [31]. Since the time instants are generated aperiodically according to (10c), the value $\frac{\hat{D}_k^i}{\|d\hat{x}_k^i\|}$ has to be strictly greater than zero; otherwise, transitions may take place infinite times over a finite time horizon (Zeno). Thus, the quantization level $\hat{D}_k^i$ should be selected in a way to avoid Zeno behavior.

Since the parameter $\hat{D}_k^i$ may depend on the predicted state and input, we define the positive function $D : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^+$

to describe such dependence

$$\hat{D}_k^i = D(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i)) \quad (11)$$

where the function D is to be determined.

Assumption 1: For any $x \in \mathcal{X}$ and any $u \in \mathcal{U}$, $\frac{D(x,u)}{\|f(x,u)\|} > 0$ if $x \neq 0$ and $u \neq 0$.

This assumption implies $\frac{\hat{D}_k^i}{\|d\hat{x}_k^i\|}$ is strictly positive when $\hat{x}_k(t_k^i)$ and $\hat{u}_k(t_k^i)$ are nonzero. It is important for the discussions in this section. We will discuss the selection of $D(\cdot, \cdot)$ in Section V to relax this assumption.

Let $N \in \mathbb{N}$ be the prediction horizon of the LAM, i.e., the number of steps in the FHOCP. Therefore, the horizon length in the cost function (6) is $T_k = t_k^N - t_k^0$ and $J[\hat{u}_k|x(t_k)]$ can be rewritten as

$$\begin{aligned} J[\hat{u}_k|x(t_k)] &= \int_{t_k^0}^{t_k^N} \ell(\hat{x}_k(\tau), \hat{u}_k(\tau)) d\tau + V_f(\hat{x}_k(t_k + T_k)) \\ &= \sum_{i=0}^{N-1} \int_{t_k^i}^{t_k^{i+1}} \ell(\hat{x}_k(\tau), \hat{u}_k(\tau)) d\tau + V_f(\hat{x}_k(t_k^N)). \end{aligned}$$

Since $\hat{x}_k(\tau) = \hat{x}_k(t_k^i)$ and $\hat{u}_k(\tau) = \hat{u}_k(t_k^i)$ are constant over $[t_k^i, t_k^{i+1})$ based on the LAM, so is $\ell(\hat{x}_k(\tau), \hat{u}_k(\tau))$. Then, the cost function can be further simplified as

$$\begin{aligned} J[\hat{u}_k|x(t_k)] &= \sum_{i=0}^{N-1} \ell(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i)) (t_k^{i+1} - t_k^i) + V_f(\hat{x}_k(t_k^N)) \\ &= \sum_{i=0}^{N-1} \ell(\hat{x}_k(t_k^i), \hat{u}_k(t_k^i)) \frac{\hat{D}_k^i}{\|d\hat{x}_k^i\|} + V_f(\hat{x}_k(t_k^N)) \end{aligned} \quad (12)$$

where the last equivalence comes from (10c). For notational simplicity, let $\hat{x}_k^i = \hat{x}_k(t_k^i)$ and $\hat{u}_k^i = \hat{u}_k(t_k^i)$.

Notice that the cost function in (12) only depends on the predicted states and inputs of the LAM, but does not explicitly depend on the time index t . With this observation, the FHOCP at t_k can be formally stated as a discrete-time optimal control problem

$$V(x(t_k)) = \min_{\hat{u}_k^i \in \mathcal{U}, i=0, \dots, N-1} J[\hat{u}_k(t)|x(t_k)]$$

$$\text{subject to } \hat{x}_k^{i+1} = \hat{x}_k^i + \hat{D}_k^i \frac{d\hat{x}_k^i}{\|d\hat{x}_k^i\|} \quad (13a)$$

$$\hat{x}_k^0 = x(t_k) \quad (13b)$$

$$\hat{x}_k^i \in \mathcal{X}_i, i = 1, \dots, N-1 \quad (13c)$$

$$\hat{x}_k^N \in \mathcal{X}_N = \mathcal{X}_f \quad (13d)$$

where $\mathcal{X}_i$ are the compact sets to be determined. The state $\hat{x}_k^1$ must stay inside a reduced set from $\mathcal{X}$ due to the model difference between the continuous-time system and the LAM. This point will be further discussed in the later sections. Let $\hat{u}_k^{i,*}$ for $i = 0, \dots, N-1$ be the optimal solution to the FHOCP and $\hat{x}_k^{i,*}$ be the corresponding optimal state. Then, to map the predicted discrete-time states back to the continuous-time horizon, we

use (10c) to identify the related time instants

$$t_k^{i+1,*} = t_k^{i,*} + \frac{\hat{D}_k^{i,*}}{\|d\hat{x}_k^{i,*}\|}$$

$$t_k^{0,*} = t_k$$

where $\hat{D}_k^{i,*} = D(\hat{x}_k^{i,*}, \hat{u}_k^{i,*})$ and $d\hat{x}_k^{i,*} = f(\hat{x}_k^{i,*}, \hat{u}_k^{i,*})$. Therefore, the prediction horizon is $T_k = t_k^{N,*} - t_k^{0,*}$ and the actual input $u(t)$ will be

$$u(t) = \hat{u}_k^{0,*} \quad \forall t \in [t_k^{0,*}, t_k^{1,*}). \quad (14)$$

With this equation, it is natural to set the next computation time as

$$t_{k+1} = t_k^{1,*} = t_k^{0,*} + \frac{\hat{D}_k^{0,*}}{\|d\hat{x}_k^{0,*}\|}$$

$$= t_k + \frac{D(x(t_k), u(t_k))}{\|f(x(t_k), u(t_k))\|} \quad (15)$$

which indicates a self-triggered feedback scheme.

IV. FEASIBILITY AND STABILITY ANALYSIS

A. State Constraints

This section discusses how the state constraints can be ensured. Notice that at the k th computation, $\hat{x}_k^1 = \hat{x}_k(t_{k+1}) \in \mathcal{X}$ does not guarantee $x(t_{k+1}) \in \mathcal{X}$ due to the modeling error between the LAM in (10) and the actual continuous-time plant (1). Therefore, to ensure $x(t) \in \mathcal{X}$, we make $\mathcal{X}_1$ a reduced set from $\mathcal{X}$. The basic idea is to quantify the difference between $x(t_{k+1})$ and $\hat{x}_k^1$ and, then, reduce the constraint sets in Problem (13) such that $\hat{x}_k^1 \in \mathcal{X}_1$ implies $x(t) \in \mathcal{X}$ for any $t \in [t_k, t_{k+1}]$.

In order to derive the upper bound on the state error, we construct a continuous-time system over $[t_k, t_{k+1}]$

$$\dot{z}_k(t) = d\hat{x}_k^{0,*} = f(\hat{x}_k^{0,*}, \hat{u}_k^{0,*}) = f(x(t_k), u(t_k))$$

$$z_k(t_k) = x(t_k). \quad (16)$$

Notice that

$$z_k(t) = z_k(t_k) + d\hat{x}_k^{0,*}(t - t_k) \quad \forall t \in [t_k, t_{k+1}] \quad (17)$$

$$z_k(t_{k+1}) = z_k(t_k) + d\hat{x}_k^{0,*}(t_{k+1} - t_k)$$

$$= x(t_k) + d\hat{x}_k^{0,*} \frac{\hat{D}_k^{0,*}}{\|d\hat{x}_k^{0,*}\|} = \hat{x}_k^{1,*} \quad (18)$$

where the last two equivalence come from (15) and (13a), respectively.

We consider the error between $z_k(t)$ and $x(t)$ over the time interval $[t_k, t_{k+1}]$. Notice that both $x(t)$ and $z_k(t)$ are generated by the same control input $\hat{u}_k^{0,*}$, starting at the same state $x(t_k)$. The state error appears only because of their difference in models.

Lemma 1: Consider system (1) and the state generated by (16). For any $t \in [t_k, t_{k+1}]$, the following inequality holds:

$$\|x(t) - z_k(t)\| \leq \epsilon_k \triangleq D(x(t_k), u(t_k))$$

$$\times \left(e^{L_f \frac{D(x(t_k), u(t_k))}{\|f(x(t_k), u(t_k))\|}} - 1 \right) \quad (19)$$

for any $t \in [t_k, t_{k+1}]$.

Proof: The proof is similar to the proof of [26, Prop. 5.3] and, therefore, omitted. ■

With the upper bound on the state error derived in 1, we can define the constraint set $\mathcal{X}_1$ in (13c) and ensure $x(t) \in \mathcal{X}$ over the entire time horizon. This is formally stated as follows.

Theorem 1: Suppose that there exists a positive constant ϵ such that

$$D(x, u) \left(e^{L_f \frac{D(x, u)}{\|f(x, u)\|}} - 1 \right) \leq \epsilon \quad (20)$$

for any $x \in \mathcal{X}$ and any $u \in \mathcal{U}$. If $x(t_0) \in \mathcal{X} - \epsilon$ and

$$\mathcal{X}_1 = \mathcal{X} - 2\epsilon \quad (21)$$

then, $x(t) \in \mathcal{X}$ for any $t \geq t_0$ under the LAMPC in (13).

Remark 5: Inequality (20) places the uniform bound ϵ on $\|x(t) - z_k(t)\|$. It is used to guarantee $x(t) \in \mathcal{X}$ for any $t \in [t_k, t_{k+1}]$. This cannot be achieved by using time-varying bounds on $\|x(t) - z_k(t)\|$, which may only ensure $x(t_k) \in \mathcal{X}$ but not those intermediate states, $x(t)$, between $x(t_k)$ and $x(t_{k+1})$. Notice that if $x(t_k) \in \mathcal{X}$ and $u(t_k) \in \mathcal{U}$, then $\epsilon_k \leq \epsilon$.

B. Feasibility

This section discusses feasibility of the LAMPC. We first introduce some assumptions on the constraint sets and the terminal function $V_f(x)$, which are similar to those in the standard MPC approaches:

Assumption 2: There exist a class $\mathcal{K}_\infty$ function $\alpha : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$, and a function $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ with $h(0) = 0$ such that:

- i) $\mathcal{X}_f + \epsilon(L_s + 1)^{N-1} \subseteq \mathcal{X}_U \triangleq \{x \in \mathcal{X}_{N-1} | h(x) \in \mathcal{U}\}$, and $0 \in \text{int}(\mathcal{X}_f)$;
- ii) If $x \in \mathcal{X}_f + \epsilon(L_s + 1)^{N-1}$, then $x + D(x, h(x)) \frac{f(x, h(x))}{\|f(x, h(x))\|} \in \mathcal{X}_f$;
- iii) The following inequality holds:

$$V_f \left(x + D(x, h(x)) \frac{f(x, h(x))}{\|f(x, h(x))\|} \right) - V_f(x)$$

$$\leq -\ell(x, h(x)) \frac{D(x, h(x))}{\|f(x, h(x))\|} \quad (22)$$

$$V_f(x) \leq \alpha(\|x\|). \quad (23)$$

With Assumption 2, we can construct control inputs for the discrete-time model in (13a) with the initial condition $\hat{x}_{k+1}^0 = x(t_{k+1})$, which will be admissible to the FHOCPC at t_{k+1}

$$\hat{u}_{k+1}^i = \begin{cases} \hat{u}_k^{i+1,*}, & i = 0, 1, \dots, N-2 \\ h(\hat{x}_{k+1}^{N-1}), & i = N-1. \end{cases} \quad (24)$$

Let the resulting state be denoted by $\hat{x}_{k+1}^0, \hat{x}_{k+1}^1, \dots, \hat{x}_{k+1}^N$. Before presenting the main result on feasibility, we introduce

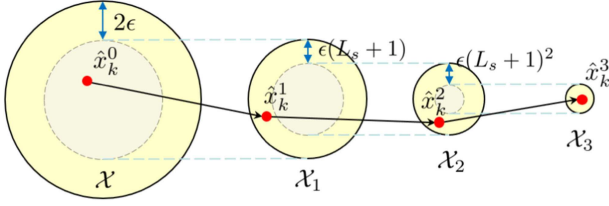


Fig. 3. Reduced sets in the FHOCP.

the following lemma to quantify the difference between $\hat{x}_{k+1}^i$ and $\hat{x}_k^{i+1,*}$.

Lemma 2: If $\frac{D(x,u)f(x,u)}{\|f(x,u)\|}$ is Lipschitz in x over $\mathcal{X}$ uniformly in u , i.e., for any $x \in \mathcal{X}$ and $u \in \mathcal{U}$, there exists a positive constant L_s such that

$$\left\| \frac{D(x,u)f(x,u)}{\|f(x,u)\|} - \frac{D(y,u)f(y,u)}{\|f(y,u)\|} \right\| \leq L_s \|x - y\| \quad (25)$$

holds, then

$$\|\hat{x}_{k+1}^i - \hat{x}_k^{i+1,*}\| \leq \epsilon_k (L_s + 1)^i \quad (26)$$

for $i = 0, 1, \dots, N-1$, where ϵ_k is defined in (19).

To guarantee feasibility, we still need to define $\mathcal{X}_i$ for $i = 1, 2, 3, \dots, N-1$ given the bounds derived in Theorem 1 and Lemma 2. They are basically the reduced sets from $\mathcal{X}$ due to state error between $x(t)$ and $\hat{x}_k^i$

$$\mathcal{X}_i \triangleq \begin{cases} \mathcal{X} - 2\epsilon & i = 1 \\ \mathcal{X} - \epsilon \left(\sum_{p=2}^i (L_s + 1)^{p-1} + 2 \right) & i = 2, \dots, N-1. \end{cases} \quad (27)$$

The basic idea is demonstrated in Fig. 3.

Theorem 2 (Feasibility): Assume that the hypotheses in Theorem 1 and Lemma 2 hold. Also assume that Assumption 2 holds. Then, the LAMPC problem in (13) is feasible for $k = 0, 1, 2, \dots$

With the definition of $\mathcal{X}_i$, the LAMPC algorithm can be summarized as follows.

Algorithm 1: LAMPC at the k th Computation of the FHOCP.

Input:

- The initial time instant of the k th FHOCP, $t_k^0 = t_k$;
- The initial state of the k th FHOCP, $\hat{x}_k^0 = x(t_k)$;
- The constraint sets, $\mathcal{X}_i$ defined in (27), $\mathcal{X}_f$ defined in Assumption 2, and $\mathcal{U}$;

Output:

- The $k+1$ st sampling instant, t_{k+1} (which is also the initial time instant of the $k+1$ st FHOCP);
- The control input over $[t_k, t_{k+1})$, $u(t)$;
- 1: Solve FHOCP in (13) for the optimal time and control sequences, $\{t_k^{i,*}\}_{i=0}^N$ and $\{\hat{u}_k^{i,*}\}_{i=0}^{N-1}$;
- 2: Set $t_{k+1} = t_k^{1,*}$;
- 3: Set $u(k) = \hat{u}_k^{0,*}$ over the time interval $[t_k, t_{k+1})$;
- 4: At time t_{k+1} , sample the state $x(t_{k+1})$;
- 5: The horizon window will move to the next computation cycle, starting at t_{k+1} and $x(t_{k+1})$.

C. Stability

The following theorem presents the stability result of the closed-loop system under LAMPC.

Theorem 3 (Stability): Suppose that the hypotheses in Theorem 2 hold. If for any $x, y \in \mathcal{X} \setminus \{0\}$ and $u \in \mathcal{U} \setminus \{0\}$, there exist positive constants $L_c, L_{V_f}, r \in \mathbb{R}^+$, $\rho \in (0, 1)$ and a class $\mathcal{K}_\infty$ function $\beta : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$ such that

$$|V_f(x) - V_f(y)| \leq L_{V_f} \|x - y\| \quad (28)$$

$$\left| \frac{\ell(x,u)D(x,u)}{\|f(x,u)\|} - \frac{\ell(y,u)D(y,u)}{\|f(y,u)\|} \right| \leq L_c \|x - y\| \quad (29)$$

$$\frac{\ell(x,u)D(x,u)}{\|f(x,u)\|} \geq \beta(\|x\|) \quad (30)$$

$$\forall \|x\| > r, \frac{\rho \ell(x,u)}{\|f(x,u)\|} \geq \theta \left(e^{L_f \frac{D(x,u)}{\|f(x,u)\|}} - 1 \right) \quad (31)$$

hold where $\theta = L_c \frac{(L_s+1)^{N-1}-1}{L_s} + L_{V_f} (L_s+1)^{N-1}$, the system in (1) under the MPC algorithm (13) is UUB with the ultimate bound $\beta^{-1}(\alpha(r))$.

Remark 6: Notice that if inequality (31) holds for $r = 0$, then the closed-loop system is asymptotically stable. Of course, we can intentionally pick $D(x,u)$ satisfying inequality (31) with $r = 0$. Then, we must examine if the selected $D(x,u)$ will lead to Zeno behavior. If so, the threshold $D(x,u)$ must be modified.

D. Example

In this example, we will show how the results can be applied to a simple system

$$\dot{x}_1(t) = f_1(x(t), u(t)) = x_2(t)$$

$$\dot{x}_2(t) = f_2(x(t), u(t)) = -\sin(x_1(t)) - x_2^2(t) + u(t)$$

where $f(x,u) = [f_1(x,u) \ f_2(x,u)]^\top$, $x = [x_1 \ x_2]^\top$, which must satisfy $|x_i(t)| \leq 3$ for any $i = 1, 2$. The input constraint is $-3 \leq |u(t)| \leq 3$. The running cost function is $\ell(x,u) = x^\top x + 0.01 u^\top u$ and the terminal cost function is $V_f(x) = x^\top x$.

In the following, we will calculate the relevant Lipschitz constants and verify the necessary assumptions:

$$\begin{aligned} & \|f(x,u) - f(y,u)\| \\ & \leq \max_{x \in \mathcal{X}} \lambda \left(\frac{\partial f(x,u)}{\partial x} \right) \cdot \|(x-y)\| \\ & = \max_{x \in \mathcal{X}} \lambda \left(\begin{bmatrix} 0 & 1 \\ -\cos(x_1) & -2x_2 \end{bmatrix} \right) \cdot \|(x-y)\| \\ & = \max_{x \in \mathcal{X}} \left\{ -x_2 - \sqrt{x_2^2 - \cos(x_1)}, \right. \\ & \quad \left. -x_2 + \sqrt{x_2^2 - \cos(x_1)} \right\} \cdot \|(x-y)\| \\ & \leq (3 + \sqrt{3^2 + 1}) \cdot \|(x-y)\|. \end{aligned}$$

Then, we have $L_f = 3 + \sqrt{10}$

$$\begin{aligned}\|V_f(x) - V_f(y)\| &= \|x^\top x - y^\top y\| \\ &= \|(x + y)^\top (x - y)\| \\ &\leq (\|x\| + \|y\|)\|x - y\|.\end{aligned}$$

Then, we have $L_{V_f} = 6\sqrt{2}$.

According to (31), the quantization size $D(x, u)$ can be selected as

$$D(x, u) = \frac{\|f(x, u)\|}{L_f}.$$

Then, inequality (31) will be satisfied

$$\begin{aligned}&\left\| \frac{\ell(x, u)D(x, u)}{\|f(x, u)\|} - \frac{\ell(y, u)D(y, u)}{\|f(y, u)\|} \right\| \\ &= \left\| \frac{\ell(x, u)}{L_f} - \frac{\ell(y, u)}{L_f} \right\| \\ &\leq \frac{L_{V_f}}{L_f} \|x - y\|.\end{aligned}$$

Then, we have $L_c = \frac{6\sqrt{2}}{3+\sqrt{10}}$. Furthermore, $\frac{\ell(x, u)D(x, u)}{\|f(x, u)\|} = \frac{\ell(x, u)}{L_f} \geq \frac{x^\top x}{3+\sqrt{10}}$, which means the satisfaction of inequality (30)

$$\begin{aligned}&\left\| \frac{D(x, u)f(x, u)}{\|f(x, u)\|} - \frac{D(y, u)f(y, u)}{\|f(y, u)\|} \right\| \\ &= \left\| \frac{f(x, u)}{L_f} - \frac{f(y, u)}{L_f} \right\| \\ &\leq \|x - y\|.\end{aligned}$$

Therefore, $L_s = 1$.

V. THRESHOLD DESIGN

This section studies the design of the threshold $D(x, u)$. By the results discussed in the previous sections, we know that, in order to ensure feasibility and stability, $D(x, u)$ must satisfy inequalities (20), (25), and (29)–(31). Besides these inequalities, $D(x, u)$ should also be selected in a way to avoid Zeno behavior. To do so, we can choose $D(x, u)$ as the function such that $D(x, u) = 0$ only when $x = 0$ and $u = 0$. Meanwhile

$$\lim_{x, u \rightarrow 0} \frac{D(x, u)}{\|f(x, u)\|} \neq 0. \quad (32)$$

In this case, since $x(t), u(t)$ are always inside compact sets, the intertransition time interval will be always greater than a positive constant, given (10c).

Let us first consider the satisfaction of inequality (31). Obviously, if we let

$$D(x, u) = \frac{\|f(x, u)\|}{L_f} \log \left(\frac{\rho \ell(x, u)}{\theta \|f(x, u)\|} + 1 \right) \quad (33)$$

inequality (31) will be trivially satisfied with $r = 0$, which implies asymptotic stability.

Remark 7: In fact, it is enough to choose the threshold as

$$D(x, u) \leq \frac{\|f(x, u)\|}{L_f} \log \left(\frac{\rho \ell(x, u)}{\theta \|f(x, u)\|} + 1 \right) \quad (34)$$

as long as Lipschitz conditions and continuity of $D(x, u)$ can be guaranteed. Meanwhile, Zeno behavior must be avoided.

With this threshold, as long as

$$\liminf_{x, u \rightarrow 0} \frac{\ell(x, u)}{\|f(x, u)\|} \neq 0 \quad (35)$$

then (32) holds and Zeno behavior can be avoided. So we can present the following result.

Theorem 4: Given the threshold in (33), if (35) holds and there exists a class $\mathcal{K}$ function $\eta(\|x\|)$ such that $\ell(x, u) \geq \eta(\|x\|)$ for any $x \in \mathcal{X}$ and any $u \in \mathcal{U}$, then inequalities (25) and (29)–(31) hold. Also, the intertransition time intervals are greater than a positive constant.

To ensure inequality (20), we can simply pick ρ such that

$$\max_{x \in \mathcal{X}, u \in \mathcal{U}} D(x, u) \left(e^{L_f \frac{D(x, u)}{\|f(x, u)\|}} - 1 \right) = \epsilon$$

with $D(x, u)$ defined in (33). Notice that such a threshold will guarantee asymptotic stability since $r = 0$ in this case.

If $\lim_{x, u \rightarrow 0} \frac{\ell(x, u)}{\|f(x, u)\|} = 0$, we may need to slightly modify the threshold in order to avoid Zeno behavior. For example, we can pick a small enough positive constant δ and define the threshold as

$$D(x, u) = \frac{\|f(x, u)\|}{L_f} \log \left(\max \left\{ \frac{\rho \ell(x, u)}{\theta \|f(x, u)\|}, \delta \right\} + 1 \right). \quad (36)$$

In this case, inequality (31) will not hold for $r = 0$, but a positive constant r . As a result, the system will become UUB instead of asymptotically stable.

Remark 8: In the presence of noise, $\frac{D(x, u)}{\|f(x, u)\|}$ may approach to the infinity, which will lead to infinite intersampling time intervals and prediction horizons. To avoid this case, we can introduce a proper positive constant $\delta_{\max}$ and define the threshold as

$$\begin{aligned}D(x, u) &= \frac{\|f(x, u)\|}{L_f} \log \\ &\times \left(\min \left\{ \max \left\{ \frac{\rho \ell(x, u)}{\theta \|f(x, u)\|}, \delta \right\}, \delta_{\max} \right\} + 1 \right).\end{aligned} \quad (37)$$

Recalling the definition $\varpi_{\max}$ below (10), if the predicted state reaches an equilibrium of the actual plant (i.e., $f(x, u) = 0$), then

$$t_k^{i+1} = t_k^i + \varpi_{\max} = t_k^i + \frac{1}{L_f} \log(\delta_{\max} + 1).$$

VI. PERIODIC CASE

The previous sections present a general framework for LAMPC where the LAM is aperiodic discretization of the continuous-time system in (1). In this section, we study a special case where the LAM takes the periodic model, i.e., $\frac{\hat{D}_k^i}{\|d\hat{x}_k^i\|} = T$

where T is a positive constant as the period. Therefore, the LAM in (10) can be simplified as follows, with which the sufficient conditions for feasibility and stability can be further simplified. To distinguish it from the general LAM, we use $\bar{x}$ instead of $\hat{x}$ in the following discussion for this special case:

$$\bar{x}_k(t_k^{i+1}) = \bar{x}_k(t_k^i) + d\bar{x}_k^i T \quad (38a)$$

$$\bar{x}_k(t_k^0) = x(t_k) \quad (38b)$$

$$t_k^{i+1} = t_k^i + T \quad (38c)$$

$$t_k^0 = t_k \quad (38d)$$

where $d\bar{x}_k^i = f(\bar{x}_k(t_k^i), \bar{u}_k(t_k^i))$. For convenience, in the following, let $\bar{x}_k^i = \bar{x}_k(t_k^i)$ and $\bar{u}_k^i = \bar{u}_k(t_k^i)$. With preceding model, the FHOCP in (13) reduces to

$$\begin{aligned} V(x(t_k)) &= \min_{\bar{u}_k^i \in \mathcal{U}, i=0, \dots, N-1} J[\bar{u}_k | x(t_k)] \\ &= \min_{\bar{u}_k^i \in \mathcal{U}, i=0, \dots, N-1} \sum_{i=0}^{N-1} \ell(\bar{x}_k^i, \bar{u}_k^i) T \\ &\quad + V_f(\bar{x}_k^N) \end{aligned}$$

subject to

$$(38a) \text{ and } (38b) \quad (39a)$$

$$\bar{x}_k^i = \bar{x}_k(t_k + iT) \in \bar{\mathcal{X}}_i, \quad i = 1, \dots, N-1 \quad (39b)$$

$$\bar{x}_k^N = \bar{x}_k(t_k + NT) \in \bar{\mathcal{X}}_N = \bar{\mathcal{X}}_f. \quad (39c)$$

The results are presented in the following corollaries. The proofs of these corollaries are similar to those for the general LAMPC and, therefore, omitted due to space limit.

To guarantee the satisfaction of the state constraints, i.e., $x(t) \in \mathcal{X}$, we have the following corollary that is directly from Theorem 1.

Corollary 1 (State Constraints): Suppose that there exists a positive constant $\bar{\epsilon}$ such that

$$\|f(x, u)\|T (e^{L_f T} - 1) \leq \bar{\epsilon} \quad (40)$$

for any $x \in \mathcal{X}$ and any $u \in \mathcal{U}$. Let

$$\bar{\mathcal{X}}_i \triangleq \mathcal{X} - \bar{\epsilon} \left(\sum_{p=2}^i (L_f T + 1)^{p-1} + 2 \right). \quad (41)$$

If $x(t_0) \in \mathcal{X} - \bar{\epsilon}$, then $x(t) \in \mathcal{X}$ for any $t \geq t_0$ under the LAMPC in (39).

Remark 9: Notice that $f(x, u)$ is bounded because f is locally Lipschitz and $\mathcal{X}, \mathcal{U}$ are compact sets. So inequality (40) can always be satisfied. Meanwhile, $\bar{\epsilon}$ can be arbitrarily small as long as T is small enough.

To guarantee feasibility, we make the following assumption.

Assumption 3: There exist a class $\mathcal{K}_\infty$ function $\bar{\alpha} : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$, and a function $\bar{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ with $\bar{h}(0) = 0$ such that

- 1) $\bar{\mathcal{X}}_f + \bar{\epsilon}(L_f T + 1)^{N-1} \subseteq \bar{\mathcal{X}}_U \triangleq \{x \in \bar{\mathcal{X}}_{N-1} | \bar{h}(x) \in \mathcal{U}\}$, and $0 \in \text{int}(\bar{\mathcal{X}}_f)$;
- 2) If $x \in \bar{\mathcal{X}}_f + \bar{\epsilon}(L_f T + 1)^{N-1}$, then $x + f(x, \bar{h}(x))T \in \bar{\mathcal{X}}_f$;

3) The following inequality holds:

$$V_f(x + f(x, \bar{h}(x))T) - V_f(x) \leq -\ell(x, \bar{h}(x))T \quad (42)$$

$$V_f(x) \leq \bar{\alpha}(\|x\|). \quad (43)$$

Corollary 2 (Feasibility): Assume that the hypotheses in Corollary 1 and Assumption 3 hold. Then, the LAMPC problem in (39) is feasible for $k = 0, 1, 2, \dots$

The following corollary presents the result on stability of the closed-loop system under the LAMPC in (39), which directly follows Theorem 3.

Corollary 3 (Stability): Suppose that the hypotheses in Corollary 2 hold. If for any $x, y \in \mathcal{X}$ and $u \in \mathcal{U}$, there exist positive constants $L_c, L_{V_f}, r \in \mathbb{R}^+, \rho \in (0, 1)$ and a class $\mathcal{K}_\infty$ function $\bar{\beta} : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$ such that

$$|V_f(x) - V_f(y)| \leq L_{V_f} \|x - y\| \quad (44)$$

$$|\ell(x, u) - \ell(y, u)|T \leq L_c \|x - y\| \quad (45)$$

$$\ell(x, u)T \geq \bar{\beta}(\|x\|) \quad (46)$$

$$\forall \|x\| > r, \frac{\rho \ell(x, u)}{\|f(x, u)\|} \geq \bar{\theta} (e^{L_f T} - 1) \quad (47)$$

hold where $\bar{\theta} = L_c \frac{(L_f T + 1)^{N-1} - 1}{L_f T} + L_{V_f} (L_f T + 1)^{N-1}$, the system in (1) under the LAMPC algorithm (39) is UUB with the ultimate bound $\bar{\beta}^{-1}(\bar{\alpha}(r))$.

Remark 10: Notice that in the periodic LAM, inequality (25) is automatically satisfied because $f(x, u)$ is locally Lipschitz. Inequality (29) is replaced by a weaker assumption that $\ell(x, u)$ is locally Lipschitz. Meanwhile, inequality (30) is simplified by the positive definiteness of $\ell(x, u)$. Finally, inequality (47) provides a rule to choose the period where T must satisfy $T \leq \min_{x \in \mathcal{X}/B(r), u \in \mathcal{U}} \frac{1}{L_f} \log(\frac{\rho \ell(x, u)}{\bar{\theta} \|f(x, u)\|} + 1)$.

VII. SIMULATIONS

This section presents the simulation results to demonstrate the performance of the LAMPC. We adopt the example in [32] and consider a crane with the horizontal trolley position $x_1(t)$, the trolley velocity $x_2(t)$, the excitation angle $x_3(t)$, and the angular velocity of the mass point $x_4(t)$. Let $x(t) = [x_1(t) \ x_2(t) \ x_3(t) \ x_4(t)]^\top \in \mathbb{R}^4$, where $x_i(t)$ is the i th coordinate of $x(t)$. Then, the crane model is governed by the following differential equations:

$$\dot{x}(t) = f(x, u) = \begin{bmatrix} x_2(t) \\ u(t) \\ x_4(t) \\ -g \sin(x_3(t)) - u(t) \cos(x_3(t)) - b x_4(t) \end{bmatrix}$$

$$x(0) = [1 \quad -1 \quad 0 \quad 0.1]^\top$$

where $g = 9.81 \text{ m/s}^2$ and $b = 0.2 \text{ Js}$. The state and input constraints are assumed to be

$$-3 \leq x_i(t) \leq 3$$

$$-0.5 \leq u(t) \leq 0.5.$$

Notice that the open-loop system is unstable.

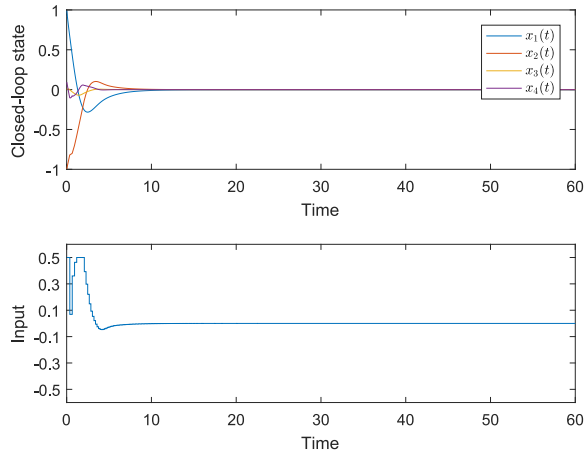


Fig. 4. State and input trajectories of the closed-loop system under the LAMPC algorithm.

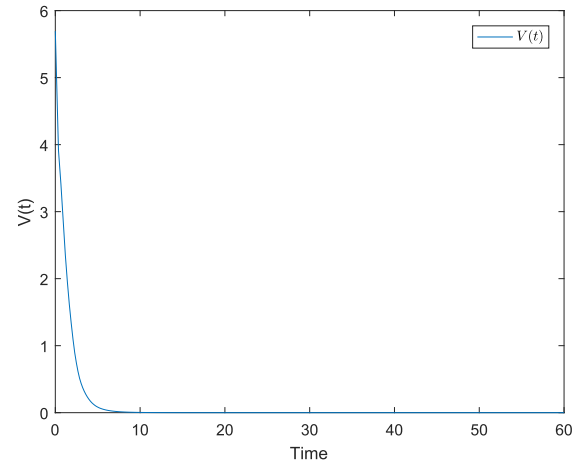


Fig. 6. Time history of the Lyapunov function V .

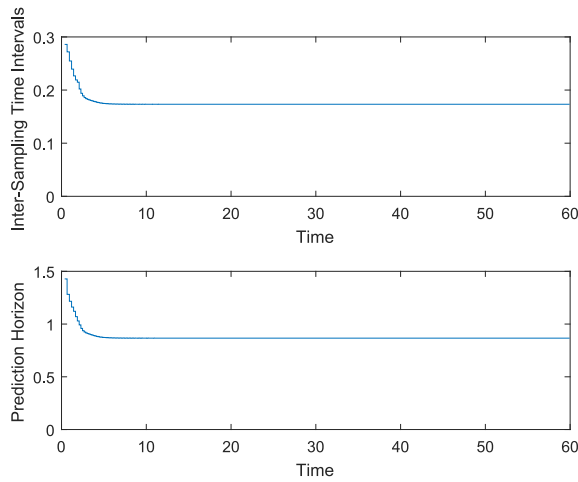


Fig. 5. History of the intersampling time intervals and the prediction horizons generated by the LAMPC algorithm.

The prediction horizon in (12) is chosen as $N = 5$. The running cost function and the terminal cost function are defined as follows:

$$\ell(x(t), u(t)) = \|f(x(t), u(t))\| (\|x(t)\|^2 + \|u(t)\|^2 + 1) \quad (48)$$

$$V_f(x) = 10\|x(t)\|^2. \quad (49)$$

With this choice, (35) is satisfied, which avoids Zeno behavior.

To examine the LAMPC algorithm, the threshold function $D(x, u)$ takes the form in (33). We use the `fmincon` solver in MATLAB optimization toolbox to solve the nonlinear FHOC. Fig. 4 plots the state and input trajectories of the closed-loop system. Both of them approach zero, which indicates the system is stabilized. Also, notice that both state and input constraints are met. Meanwhile, the Lyapunov function is strictly decreasing as shown in Fig. 6. Fig. 5 plots the history of the intersampling time intervals generated by the self-triggered scheme (top) and the prediction horizons at each sampling instants (bottom). It is clear that those intervals are time-varying and strictly greater than zero. One thing worth mentioning is that both of them seem to converge to positive constants as the state and the input approach

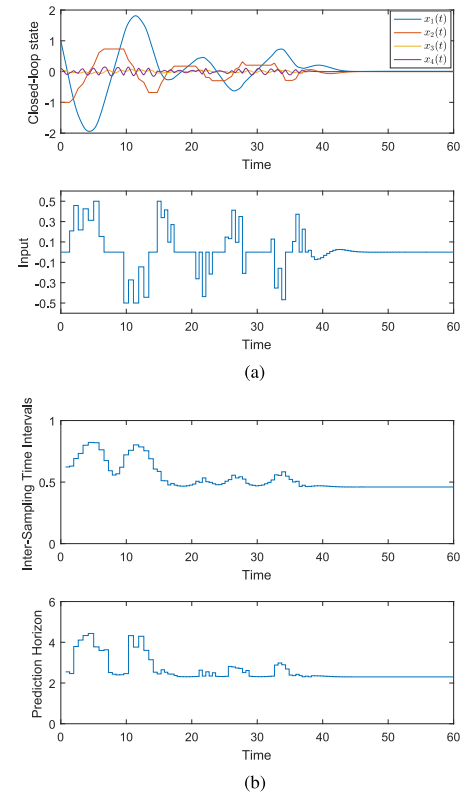


Fig. 7. System under the LAMPC that generates the maximum average inter-sampling time interval. (a) State and input trajectories of the closed-loop system. (b) History of the inter-sampling time intervals and the prediction horizons.

zero. It is consistent with the theoretical findings because based on (10c) and the definitions of $\ell(x, u)$ in (48) and $D(x, u)$ in (33), we have $\lim_{x, u \rightarrow 0} \frac{D(x, u)}{\|f(x, u)\|} = \frac{1}{L_f} \log\left(\frac{\rho}{\theta} + 1\right)$.

By varying the parameter ρ in (33) with the guarantee of the feasibility and stability conditions, we simulate the system and generate different sets of the intersampling time intervals. We calculate the average intersampling time interval based on the data in each set. The maximum of these average intersampling time intervals is 0.4602 s (see Fig. 7). Meanwhile, the maximum

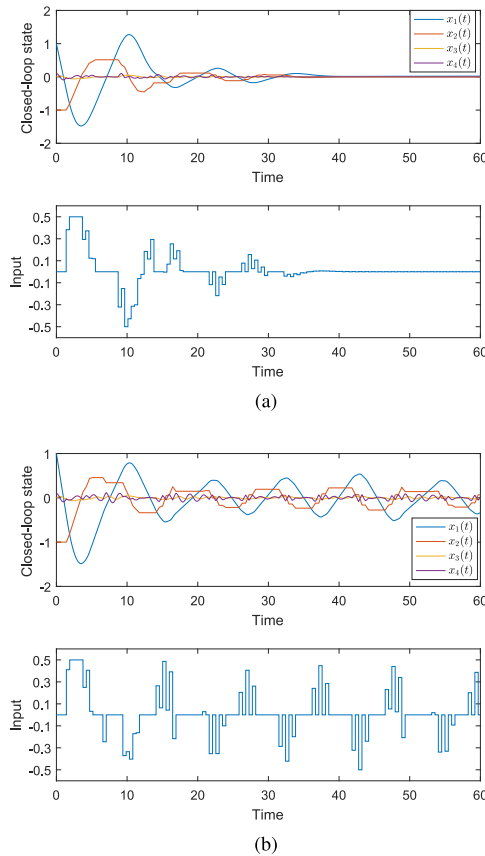


Fig. 8. State and input trajectories of the closed-loop system with different sampling periods. (a) $T = 0.46$. (b) $T = 0.47$.

average intersampling time interval during the transience (over the time interval $[0, 45]$) is 0.5528 s.

We then study the special case of the periodic LAM and compare the performance between this case and the general LAMPC. By simulations, we find that the maximum allowable period, for which the system is stable, is a value between 0.46 and 0.47 [see Fig. 8(a) and (b)], which is very close to the maximum average intersampling time intervals obtained in the previous section.

By comparing Figs. 7(a) and 8(a), it can be observed that the aperiodic model has a little larger overshoot. Also, we find that the aperiodic model triggers fewer times solving the FHOCP (81 times) during the transience, compared with the periodic model (98 times). Based on this simulation, we see that the aperiodic LAMPC algorithm may generate longer intersampling time intervals than the periodic case during the transience at the expense of incremental overshoot.

Finally, we investigate the impact of actuation delay on the control performance, where the state equations can be rewritten as

$$\dot{x}(t) = \begin{bmatrix} x_2(t) \\ u(t) \\ x_4(t) \\ -g \sin(x_3(t)) - u(t - t_d) \cos(x_3(t)) - b x_4(t) \end{bmatrix}.$$

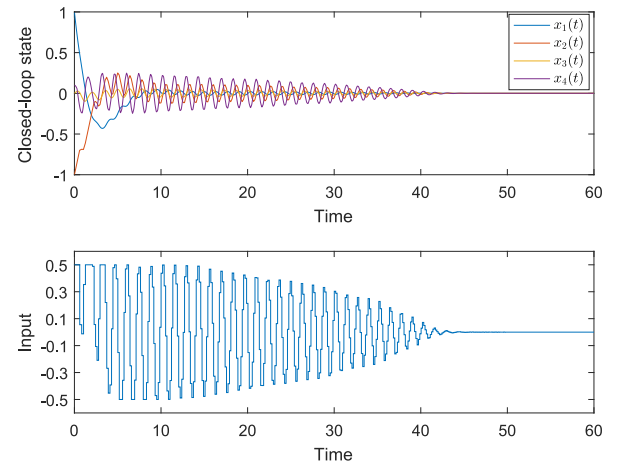


Fig. 9. State and input trajectories of the closed-loop system with actuation delays.

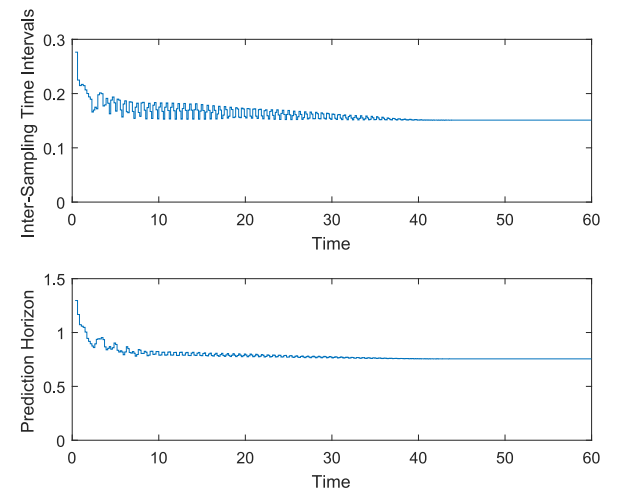


Fig. 10. Intersampling time intervals and the prediction horizons in the presence of actuation delays.

Fig. 9 plots the state and input trajectories of the system when the delay is set to be $t_d = 0.3$. It is shown that although system stability is still preserved, the state and input signals oscillate a lot during the transience. Fig. 10 plots the history of the intersampling time intervals and the prediction horizons. A significant decrease is found in the length of these intervals, compared with the delay-free case in the first simulation. It means that the FHOCP is triggered more frequently, which requires more computational resources.

Fig. 11 plots the relation between the actuation delays and the average intersampling time intervals generated by the LAMPC. The average intersampling time interval is taken to be the maximum allowable average interval for stability and feasibility under a specific value of t_d . It shows that as the actuation delay t_d increases, the allowable average intersampling time interval decreases. There is a dramatic drop in the average intersampling time interval when $t_d > 0.2$. From these simulation results, we can see that the actuation delay has negative impacts on both control performance and computational efficiency.

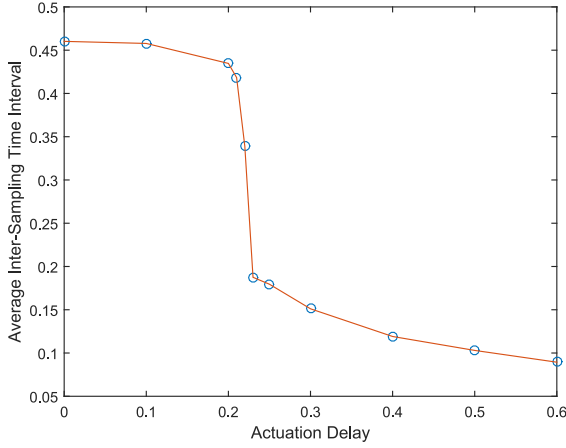


Fig. 11. Relation between actuation delays and average intersampling time intervals.

VIII. CONCLUSION

This article presents the LAMPC algorithm for nonlinear sampled-data systems. The LAMPC samples the state following a self-triggering mechanism in order to reduce the frequency of solving the FHOCP. More importantly, the FHOCP itself is also discretized in an aperiodic manner based on the LAM, which makes the algorithm completely discrete-time and computer-friendly. The aperiodic nature in the FHOCP enables the controller to optimize the system performance over a longer prediction horizon without increasing the number of transitions. We show sufficient conditions for feasibility and stability of the resulting closed-loop systems.

There are still many open problems to be addressed in the future. Notice that the parameters in the LAM may be over-conservative due to the global nature of the Lipschitz constants (We compute the Lipschitz constants over the entire state and input constraint sets). An alternative way is to refine Lipschitz constants at each computation of the FHOCP. As the state approaches the origin, we can use the Lipschitz constants calculated based on a smaller neighborhood of the sampled state. By this way, the Lipschitz constants will become smaller and smaller, which makes the approach less conservative. The work in [33] also provides three possible methods to reduce conservativeness of Lipschitz constants, which certainly deserve further investigation. One important problem is to further reduce the complexity in solving the FHOCP. It is shown that the choice of the threshold function and the cost function may affect the complexity. Appropriate cost function and threshold can significantly simplify the computation and, therefore, reduce the complexity. Another interesting problem is to quantify the impact of disturbances and delays, such as measurement noises, actuation errors, and sensing/actuation delays, in LAMPC. Following the spirit of the work in [33], a possible solution is to further reduce the constraint sets, and redefine appropriate terminal cost functions and terminal sets, so that the system can preserve enough feasibility and stability margins to tolerate the negative impacts raised by disturbances and delays. Besides deterministic formulations, a stochastic framework for robust LAMPC over

the probability space is also worth investigation [34], [35]. We will study both cases in the future work.

APPENDIX PROOFS

A. Proof of Theorem 1

Proof: First of all, notice that given (17), for any $t \in [t_k, t_{k+1}]$ and any $k \in \mathbb{Z}_0^+$, $z_k(t)$ can be written as

$$z_k(t) = (1 - \theta)z_k(t_k) + \theta z_k(t_{k+1}) \quad (50)$$

where $\theta = \frac{t-t_k}{t_{k+1}-t_k} \in [0, 1]$.

Then, we prove the statement in a recursive way. For $k = 0$, we have $x(t_0) = z_0(t_0) = \hat{x}_0^{0,*} \in \mathcal{X} - \epsilon$ by the assumption. Also notice that $z_0(t_1) = \hat{x}_0^{1,*} \in \mathcal{X}_1 - 2\epsilon$ by (18). Because $\mathcal{X} - \epsilon$ is still compact and $z_0(t_0), z_0(t_1) \in \mathcal{X} - \epsilon$, (50) implies $z_0(t) \in \mathcal{X} - \epsilon$ for any $t \in [t_0, t_1]$. By Lemma 1

$$\|x(t) - z_0(t)\| \leq D(x(t_0), u(t_0)) \left(e^{L_f \frac{D(x(t_0), u(t_0))}{\|f(x(t_0), u(t_0))\|}} - 1 \right) \leq \epsilon$$

over $[t_0, t_1]$, where the second inequality comes from (20) since $x(t_0) \in \mathcal{X} - \epsilon$ and $u(t_0) \in \mathcal{U}$ by (13). Thus, $z_0(t) \in \mathcal{X} - \epsilon$ implies $x(t) \in \mathcal{X}$ for any $t \in [t_0, t_1]$.

For $k = 1$, by (21), $\hat{x}_0^{1,*} = z_0(t_1) \in \mathcal{X} - 2\epsilon$. By Lemma 1, we know $\|x(t_1) - z_0(t_1)\| \leq \epsilon$. So $x(t_1) = \hat{x}_1^{0,*} = z_1(t_1) \in \mathcal{X} - \epsilon$. Also $z_1(t_2) = \hat{x}_1^{1,*} \in \mathcal{X} - 2\epsilon$ holds by (13c). Therefore, (50) implies that $z_1(t)$ stays in $\mathcal{X} - \epsilon$ for any $t \in [t_1, t_2]$. Again, by Lemma 1 and inequality (20), $\|x(t) - z_1(t)\| \leq \epsilon$ holds and, therefore, $x(t) \in \mathcal{X}$ for any $t \in [t_1, t_2]$. Keeping this process, we show that $x(t) \in \mathcal{X}$ for any $t \in [t_k, t_{k+1}]$ and any $k \in \mathbb{Z}_0^+$. ■

B. Proof of Lemma 2

Proof: We prove the statement by mathematical induction. By Lemma 1, we know $\|x(t_{k+1}) - z_k(t_{k+1})\| \leq \epsilon_k$. Since $x(t_{k+1}) = \hat{x}_{k+1}^0$ and $z_k(t_{k+1}) = \hat{x}_k^{1,*}$, we know inequality (26) holds for $i = 0$.

Assume that inequality (26) holds for $i = p - 1$, i.e.,

$$\|\hat{x}_{k+1}^{p-1} - \hat{x}_k^{p,*}\| \leq \epsilon_k(L_s + 1)^{p-1}. \quad (51)$$

We now show that this inequality also holds for $i = p$. Notice that by (13a)

$$\begin{aligned} \hat{x}_k^{p+1,*} &= \hat{x}_k^{p,*} + D(\hat{x}_k^{p,*}, \hat{u}_k^{p,*}) \frac{f(\hat{x}_k^{p,*}, \hat{u}_k^{p,*})}{\|f(\hat{x}_k^{p,*}, \hat{u}_k^{p,*})\|} \\ \hat{x}_{k+1}^p &= \hat{x}_{k+1}^{p-1} + D(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*}) \frac{f(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*})}{\|f(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*})\|} \end{aligned}$$

hold. Therefore

$$\begin{aligned} &\|\hat{x}_{k+1}^p - \hat{x}_k^{p+1,*}\| \\ &\leq \left\| \hat{x}_{k+1}^{p-1} + D(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*}) \frac{f(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*})}{\|f(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*})\|} - \hat{x}_k^{p,*} \right. \\ &\quad \left. - D(\hat{x}_k^{p,*}, \hat{u}_k^{p,*}) \frac{f(\hat{x}_k^{p,*}, \hat{u}_k^{p,*})}{\|f(\hat{x}_k^{p,*}, \hat{u}_k^{p,*})\|} \right\| \end{aligned}$$

$$\begin{aligned}
&\leq \|\hat{x}_{k+1}^{p-1} - \hat{x}_k^{p,*}\| + \left\| D(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*}) \frac{f(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*})}{\|f(\hat{x}_{k+1}^{p-1}, \hat{u}_k^{p,*})\|} \right. \\
&\quad \left. - D(\hat{x}_k^{p,*}, \hat{u}_k^{p,*}) \frac{f(\hat{x}_k^{p,*}, \hat{u}_k^{p,*})}{\|f(\hat{x}_k^{p,*}, \hat{u}_k^{p,*})\|} \right\| \\
&\leq (1 + L_s) \|\hat{x}_{k+1}^{p-1} - \hat{x}_k^{p,*}\| \leq \epsilon_k (L_s + 1)^p
\end{aligned}$$

where the third inequality is obtained because of inequality (25) and the last inequality comes from inequality (51). By mathematical induction, inequality (26) holds and the proof is completed. ■

C. Proof of Theorem 2

Proof: We will prove that if $\hat{x}_k^{i,*} \in \mathcal{X}_i$ for $i = 1, 2, \dots, N$, there is a feasible solution of the optimization problem in $k+1$, which is $\hat{u}_{k+1}^i$ defined in (24), based on the optimal solution in k , $\hat{u}_k^{i,*}$.

First, we show that $\hat{u}_{k+1}^i \in \mathcal{U}$ for $i = 0, 1, \dots, N-1$. Based on (24), $\hat{u}_{k+1}^i = \hat{u}_k^{i+1,*} \in \mathcal{U}$ for $i = 0, 1, \dots, N-2$ because of the feasibility of $\hat{u}_k^{i+1,*}$. By Lemma 2 and inequality (20), we know

$$\|\hat{x}_{k+1}^{N-1} - \hat{x}_k^{N,*}\| \leq \epsilon_k (L_s + 1)^{N-1} \leq \epsilon (L_s + 1)^{N-1}.$$

Since $\hat{x}_k^{N,*} \in \mathcal{X}_f$, $\hat{x}_{k+1}^{N-1} \in \mathcal{X}_f + \epsilon (L_s + 1)^{N-1}$ holds. By condition (i) in Assumption 2

$$\hat{x}_{k+1}^{N-1} \in \mathcal{X}_f + \epsilon (L_s + 1)^{N-1} \subseteq X_U$$

and, therefore, $\hat{u}_{k+1}^{N-1} \in \mathcal{U}$.

Next, we show that $\hat{x}_{k+1}^i \in \mathcal{X}_i$ for $i = 1, \dots, N-1$ and $\hat{x}_{k+1}^N \in \mathcal{X}_f$. Because $\hat{x}_{k+1}^{N-1} \in \mathcal{X}_f + \epsilon (L_s + 1)^{N-1}$, we have $\hat{x}_{k+1}^N \in \mathcal{X}_f$ by condition (ii) in Assumption 2. Also, by Lemma 2 and inequality (20), we know $\|\hat{x}_{k+1}^i - \hat{x}_k^{i+1,*}\| \leq \epsilon (L_s + 1)^i$ for $i = 1, \dots, N-1$. Also notice that $\hat{x}_k^{i+1,*} \in \mathcal{X}_{i+1} = \mathcal{X} - \epsilon (\sum_{p=2}^{i+1} (L_s + 1)^{p-1} + 2)$ for $i = 1, \dots, N-2$ and $\hat{x}_k^{N,*} \in \mathcal{X}_f \subseteq \mathcal{X}_{N-1}$. So

$$\begin{aligned}
\hat{x}_{k+1}^i &\in \mathcal{X}_{i+1} + \epsilon (L_s + 1)^i \\
&= \mathcal{X} - \epsilon \left(\sum_{p=2}^i (L_s + 1)^{p-1} + 2 \right) = \mathcal{X}_i
\end{aligned}$$

for $i = 1, \dots, N-1$. ■

D. Proof of Theorem 3

Proof: Consider

$$\begin{aligned}
&J[\hat{u}_{k+1}(t)|x(t_{k+1})] - V(x(t_k)) \\
&= \sum_{i=0}^{N-1} \ell(\hat{x}_{k+1}^i, \hat{u}_{k+1}^i) \frac{\hat{D}_{k+1}^i}{\|d\hat{x}_{k+1}^i\|} + V_f(\hat{x}_{k+1}^N) - V(x(t_k)) \\
&= \sum_{i=0}^{N-2} \ell(\hat{x}_{k+1}^i, \hat{u}_{k+1}^i) \frac{\hat{D}_{k+1}^i}{\|d\hat{x}_{k+1}^i\|} + V_f(\hat{x}_{k+1}^{N-1}) \\
&\quad + \ell(\hat{x}_k^{0,*}, \hat{u}_k^{0,*}) \frac{\hat{D}_k^{0,*}}{\|d\hat{x}_k^{0,*}\|} - V(x(t_k)) \Big\} \Phi
\end{aligned}$$

$$\begin{aligned}
&+ \ell(\hat{x}_{k+1}^{N-1}, \hat{u}_{k+1}^{N-1}) \frac{\hat{D}_{k+1}^{N-1}}{\|d\hat{x}_{k+1}^{N-1}\|} + V_f(\hat{x}_{k+1}^N) - V_f(\hat{x}_{k+1}^{N-1}) \\
&- \ell(\hat{x}_k^{0,*}, \hat{u}_k^{0,*}) \frac{\hat{D}_k^{0,*}}{\|d\hat{x}_k^{0,*}\|}.
\end{aligned} \tag{52}$$

By inequality (22) in Assumption 2

$$\ell(\hat{x}_{k+1}^{N-1}, \hat{u}_{k+1}^{N-1}) \frac{\hat{D}_{k+1}^{N-1}}{\|d\hat{x}_{k+1}^{N-1}\|} + V_f(\hat{x}_{k+1}^N) - V_f(\hat{x}_{k+1}^{N-1}) \leq 0 \tag{53}$$

because $\hat{u}_{k+1}^{N-1} = h(\hat{x}_{k+1}^{N-1})$ and

$$\begin{aligned}
\hat{x}_{k+1}^N &= \hat{x}_{k+1}^{N-1} + \hat{D}_{k+1}^{N-1} \frac{d\hat{x}_{k+1}^{N-1}}{\|d\hat{x}_{k+1}^{N-1}\|} \\
&= \hat{x}_{k+1}^{N-1} + D(\hat{x}_{k+1}^{N-1}, \hat{u}_{k+1}^{N-1}) \frac{f(\hat{x}_{k+1}^{N-1}, \hat{u}_{k+1}^{N-1})}{\|f(\hat{x}_{k+1}^{N-1}, \hat{u}_{k+1}^{N-1})\|}.
\end{aligned}$$

Consider Φ . Notice that $V(x(t_k))$ can be rewritten as

$$V(x(t_k)) = \sum_{i=-1}^{N-2} \ell(\hat{x}_k^{i+1,*}, \hat{u}_k^{i+1,*}) \frac{\hat{D}_k^{i+1,*}}{\|d\hat{x}_k^{i+1,*}\|} + V_f(\hat{x}_k^{N,*}).$$

Therefore

$$\begin{aligned}
\Phi &= \sum_{i=0}^{N-2} \ell(\hat{x}_{k+1}^i, \hat{u}_{k+1}^i) \frac{\hat{D}_{k+1}^i}{\|d\hat{x}_{k+1}^i\|} + V_f(\hat{x}_{k+1}^{N-1}) \\
&\quad - \sum_{i=0}^{N-2} \ell(\hat{x}_k^{i+1,*}, \hat{u}_k^{i+1,*}) \frac{\hat{D}_k^{i+1,*}}{\|d\hat{x}_k^{i+1,*}\|} - V_f(\hat{x}_k^{N,*}) \\
&\leq \sum_{i=0}^{N-2} \left| \frac{\ell(\hat{x}_{k+1}^i, \hat{u}_{k+1}^i) \hat{D}_{k+1}^i}{\|d\hat{x}_{k+1}^i\|} - \frac{\ell(\hat{x}_k^{i+1,*}, \hat{u}_k^{i+1,*}) \hat{D}_k^{i+1,*}}{\|d\hat{x}_k^{i+1,*}\|} \right| \\
&\quad + |V_f(\hat{x}_{k+1}^{N-1}) - V_f(\hat{x}_k^{N,*})| \\
&\leq \sum_{i=0}^{N-2} L_c \|\hat{x}_{k+1}^i - \hat{x}_k^{i+1,*}\| + L_{V_f} \|\hat{x}_{k+1}^{N-1} - \hat{x}_k^{N,*}\| \\
&\leq \sum_{i=0}^{N-2} L_c \epsilon_k (L_s + 1)^i + L_{V_f} \epsilon_k (L_s + 1)^{N-1} \\
&= \epsilon_k \left(L_c \frac{(L_s + 1)^{N-1} - 1}{L_s} + L_{V_f} (L_s + 1)^{N-1} \right) = \epsilon_k \theta
\end{aligned}$$

where the second inequality comes from inequality (29) and the last inequality is obtained using Lemma 2. Applying the inequality above and inequality (53) into (52) yields

$$\begin{aligned}
&J[\hat{u}_{k+1}(t)|x(t_{k+1})] - V(x(t_k)) \\
&\leq -\ell(\hat{x}_k^{0,*}, \hat{u}_k^{0,*}) \frac{\hat{D}_k^{0,*}}{\|d\hat{x}_k^{0,*}\|} + \epsilon_k \theta \\
&= -\ell(x(t_k), u(t_k)) \frac{\hat{D}(x(t_k), u(t_k))}{\|f(x(t_k), u(t_k))\|} + \epsilon_k \theta.
\end{aligned}$$

Therefore

$$\begin{aligned}
V(x(t_{k+1})) - V(x(t_k)) &= \min_{\hat{u}_{k+1}(t)} J[\hat{u}_{k+1}(t)|x(t_{k+1})] \\
&\quad - V(x(t_k)) \\
&\leq -\ell(\hat{x}_k^{0,*}, \hat{u}_k^{0,*}) \frac{\hat{D}_k^{0,*}}{\|d\hat{x}_k^{0,*}\|} + \epsilon_k \theta \\
&= -\ell(x(t_k), u(t_k)) \frac{D(x(t_k), u(t_k))}{\|f(x(t_k), u(t_k))\|} + \epsilon_k \theta \\
&= -D(x(t_k), u(t_k)) \\
&\quad \cdot \left(\frac{\ell(x(t_k), u(t_k))}{\|f(x(t_k), u(t_k))\|} - \theta \left(e^{L_f \frac{D(x(t_k), u(t_k))}{\|f(x(t_k), u(t_k))\|}} - 1 \right) \right)
\end{aligned}$$

where the last equivalence is obtained by the definition of ϵ_k in (19). By (31), we know that for any $\|x(t_k)\| \geq r$

$$\begin{aligned}
V(x(t_{k+1})) - V(x(t_k)) &\leq -\frac{D(x(t_k), u(t_k)) (1 - \rho) \ell(x(t_k), u(t_k))}{\|f(x(t_k), u(t_k))\|} \\
&\leq -(1 - \rho) \beta (\|x(t_k)\|)
\end{aligned} \tag{54}$$

holds. By Assumption 2, we know

$$V_f(\hat{x}_k^{i+1}) - V_f(\hat{x}_k^i) \leq -\ell(\hat{x}_k^i, h(\hat{x}_k^i)) \frac{D(\hat{x}_k^i, h(\hat{x}_k^i))}{\|f(\hat{x}_k^i, h(\hat{x}_k^i))\|}.$$

Summing up the inequality above for $i = 0, 1, \dots, N-1$

$$V_f(\hat{x}_k^N) - V_f(\hat{x}_k^0) \leq -\sum_{i=0}^{N-1} \ell(\hat{x}_k^i, h(\hat{x}_k^i)) \frac{D(\hat{x}_k^i, h(\hat{x}_k^i))}{\|f(\hat{x}_k^i, h(\hat{x}_k^i))\|}.$$

Thus, by the definition of $V(x(t_k))$

$$\begin{aligned}
V(x(t_k)) &\leq \sum_{i=0}^{N-1} \ell(\hat{x}_k^i, h(\hat{x}_k^i)) \frac{D(\hat{x}_k^i, h(\hat{x}_k^i))}{\|f(\hat{x}_k^i, h(\hat{x}_k^i))\|} + V_f(\hat{x}_k^N) \\
&\leq V_f(\hat{x}_k^0) = V_f(x(t_k)) \leq \alpha(\|x(t_k)\|)
\end{aligned} \tag{55}$$

holds, which, together with inequality (54), implies that $x(t_k)$ will ultimately stay in the set $\Omega \triangleq \{x | V(x) \leq \alpha(r)\}$. Since $V(x(t)) \leq V(x(t_k))$ for any $t \in (t_k, t_{k+1})$, we can conclude that $x(t)$ will eventually stay in Ω , which means that the resulting closed-loop system is UUB. Also notice that $V(x) \geq \ell(x, u) \frac{D(x, u)}{\|f(x, u)\|} \geq \beta(\|x\|)$. Then, we can derive the ultimate bound as $\beta^{-1}(\alpha(r))$. ■

E. Proof of Theorem 4

Proof: First of all, by the definition of $D(x, u)$ in (33), inequality (31) will be trivially satisfied with $r = 0$.

Since $\ell(x, u)$ and $f(x, u)$ are continuous, $\frac{\ell(x, u)}{\|f(x, u)\|}$ is continuous over $\mathcal{X} \times \mathcal{U} \setminus \{(0, 0)\}$. This, together with (35), implies that there exists a positive constant ξ such that $\frac{\ell(x, u)}{\|f(x, u)\|} \geq \xi$ for any $x \in \mathcal{X}$ and any $u \in \mathcal{U}$. Applying this inequality into (33) means that the intertransition time intervals are bounded from below by some positive constant. Therefore, Zeno behavior can be avoided.

To show the satisfaction of inequality (25), we can simply apply (33) into $\frac{D(x, u)f(x, u)}{\|f(x, u)\|}$ to replace $D(x, u)$. Because $\ell(x, u)$ and $f(x, u)$ are locally Lipschitz, $\frac{D(x, u)f(x, u)}{\|f(x, u)\|}$ is also locally Lipschitz. Therefore, inequality (25) holds. By a similar analysis, we can show that $\frac{D(x, u)\ell(x, u)}{\|f(x, u)\|}$ is locally Lipschitz and inequality (29) holds.

Finally, by the assumption that $\ell(x, u) \geq \eta(\|x\|)$, we have

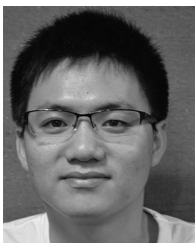
$$\begin{aligned}
\frac{\ell(x, u)D(x, u)}{\|f(x, u)\|} &= \frac{\ell(x, u)}{L_f} \log \left(\frac{\rho \ell(x, u)}{\theta \|f(x, u)\|} + 1 \right) \\
&\geq \frac{\eta(\|x\|)}{L_f} \log \left(\frac{\rho \xi}{\theta} + 1 \right).
\end{aligned}$$

Thus, inequality (30) is satisfied. ■

REFERENCES

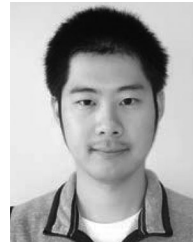
- [1] E. F. Camacho and C. B. Alba, *Model Predictive Control*. Berlin, Germany: Springer, 2013.
- [2] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.
- [3] A. Bemporad and M. Morari, "Robust model predictive control: A survey," in *Robustness in Identification and Control*. Berlin, Germany: Springer, 1999, pp. 207–226.
- [4] F. A. Fontes, "A general framework to design stabilizing nonlinear model predictive controllers," *Syst. Control Lett.*, vol. 42, no. 2, pp. 127–143, 2001.
- [5] L.-S. Hu, B. Huang, and Y.-Y. Cao, "Robust digital model predictive control for linear uncertain systems with saturations," *IEEE Trans. Autom. Control*, vol. 49, no. 5, pp. 792–796, May 2004.
- [6] L. Magni and R. Scattolini, "Model predictive control of continuous-time nonlinear systems with piecewise constant control," *IEEE Trans. Autom. Control*, vol. 49, no. 6, pp. 900–906, Jun. 2004.
- [7] M. Rubagotti, D. M. Raimondo, A. Ferrara, and L. Magni, "Robust model predictive control with integral sliding mode in continuous-time sampled-data nonlinear systems," *IEEE Trans. Autom. Control*, vol. 56, no. 3, pp. 556–570, Mar. 2011.
- [8] J. Sjöberg, R. Findeisen, and F. Allgöwer, "Model predictive control of continuous time nonlinear differential algebraic systems," *IFAC Proc. Vol.*, vol. 40, no. 12, pp. 48–53, 2007.
- [9] T. Raff, D. Sinz, and F. Allgöwer, "Model predictive control of uncertain continuous-time systems with piecewise constant control input: a convex approach," in *Proc. Am. Control Conf.*, 2008, pp. 1109–1114.
- [10] T. Shi and H. Su, "Sampled-data MPC for LPV systems with input saturation," *IET Control Theory Appl.*, vol. 8, no. 17, pp. 1781–1788, 2014.
- [11] D. Lehmann, E. Henriksson, and K. H. Johansson, "Event-triggered model predictive control of discrete-time linear systems subject to disturbances," in *Proc. IEEE Eur. Control Conf.*, 2013, pp. 1156–1161.
- [12] A. Eqtami, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Event-triggered control for discrete-time systems," in *Proc. Am. Control Conf.*, 2010, pp. 4719–4724.
- [13] A. Eqtami, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Event-triggered strategies for decentralized model predictive controllers," *Proc. 18th IFAC World Congr.*, vol. 44, no. 1, pp. 10068–10073, 2011.
- [14] H. Li and Y. Shi, "Event-triggered robust model predictive control of continuous-time nonlinear systems," *Automatica*, vol. 50, no. 5, pp. 1507–1513, 2014.
- [15] A. Eqtami, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Novel event-triggered strategies for model predictive controllers," in *Proc. 50th IEEE Conf. Decis. Control Eur. Control Conf.*, 2011, pp. 3392–3397.
- [16] A. Ferrara, G. P. Incremona, and L. Magni, "Model-based event-triggered robust MPC/ISM," in *Proc. Eur. Control Conf.*, 2014, pp. 2931–2936.
- [17] J. Sijts, M. Lazar, and W. Heemels, "On integration of event-based estimation and robust MPC in a feedback loop," in *Proc. 13th ACM Int. Conf. Hybrid Syst.: Comput. Control*, 2010, pp. 31–40.
- [18] F. D. Brunner, W. Heemels, and F. Allgöwer, "Robust event-triggered MPC for constrained linear discrete-time systems with guaranteed average sampling rate," *IFAC-PapersOnLine*, vol. 48, no. 23, pp. 117–122, 2015.

- [19] J. B. Berglind, T. Gommans, and W. Heemels, "Self-triggered MPC for constrained linear systems and quadratic costs," in *Proc. 4th IFAC Nonlinear Model Predictive Control Conf.*, vol. 45, no. 17, pp. 342–348, 2012.
- [20] F. Brunner, W. Heemels, and F. Allgöwer, "Robust self-triggered MPC for constrained linear systems," in *Proc. Eur. Control Conf.*, 2014, pp. 472–477.
- [21] A. Eqtami, S. Heshmati-alamdari, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Self-triggered model predictive control for nonholonomic systems," in *Proc. IEEE Eur. Control Con.*, 2013, pp. 638–643.
- [22] E. Henriksson, D. E. Quevedo, H. Sandberg, and K. H. Johansson, "Self-triggered model predictive control for network scheduling and control," *IFAC Proc. Vol.*, vol. 45, no. 15, pp. 432–438, 2012.
- [23] T. Gommans and W. Heemels, "Resource-aware MPC for constrained nonlinear systems: A self-triggered control approach," *Syst. Control Lett.*, vol. 79, pp. 59–67, 2015.
- [24] P. Sopasakis, P. Patrinos, and H. Sarimveis, "MPC for sampled-data linear systems: Guaranteeing constraint satisfaction in continuous-time," *IEEE Trans. Autom. Control*, vol. 59, no. 4, pp. 1088–1093, Apr. 2014.
- [25] M. Farina and R. Scattolini, "Tube-based robust sampled-data MPC for linear continuous-time systems," *Automatica*, vol. 48, no. 7, pp. 1473–1476, 2012.
- [26] X. Wang and B. Zhang, "Lebesgue approximation model of continuous-time nonlinear dynamic systems," *Automatica*, vol. 64, pp. 234–239, 2016.
- [27] X. Wang and L. Yang, "Sporadic model predictive control using Lebesgue approximation," in *Proc. Am. Control Conf.*, 2017, pp. 5768–5773.
- [28] X. Wang and M. Lemmon, "Self-triggered feedback control systems with finite-gain L_2 stability," *IEEE Trans. Autom. Control*, vol. 54, no. 3, pp. 452–467, Mar. 2009.
- [29] A. Anta and P. Tabuada, "To sample or not to sample: Self-triggered control for nonlinear systems," *IEEE Trans. Autom. Control*, vol. 55, no. 9, pp. 2030–2042, Sep. 2010.
- [30] D. Liberzon, *Switching in Systems and Control Systems and Control: Foundations and Applications*. Boston, MA, USA: Birkhauser, 2003.
- [31] X. Wang, "Event-triggering in cyber-physical systems," Ph.D. dissertation, Univ. Notre Dame, Notre Dame, IN, USA, 2009.
- [32] B. Houska, H. J. Ferreau, and M. Diehl, "An auto-generated real-time iteration algorithm for nonlinear MPC in the microsecond range," *Automatica*, vol. 47, no. 10, pp. 2279–2285, 2011.
- [33] D. L. Marruedo, T. Alamo, and E. Camacho, "Input-to-state stable MPC for constrained discrete-time nonlinear systems with bounded additive uncertainties," in *Proc. 41th IEEE Conf. Decis. Control*, 2002, vol. 4, pp. 4619–4624.
- [34] H.-H. Kuo, "Stochastic differential equations," in *Introduction to Stochastic Integration*. Berlin, Germany: Springer, 2006, pp. 185–230.
- [35] B. Øksendal, "Stochastic differential equations," in *Stochastic Differential Equations*. Berlin, Germany: Springer, 2003, pp. 65–84.



Jie Tao received the B.S. degree from the Harbin Institute of Technology, Harbin, China, in 2013 and the Ph.D. degree from the Department of Control Science and Engineering, Zhejiang University, Hangzhou, China, in 2018.

He is currently with the Guangdong Provincial Key Laboratory of Intelligent Decision and Cooperative Control, Guangdong University of Technology, Guangzhou, China. His current research interests include dissipative control and filtering, Markov jump systems, and event-based control.



Lixing Yang received the double B.S. degrees from Florida International University, Miami, FL, USA, and the Hebei University of Technology, Hongqiao, China, in 2013. He received the M.S. degree from the New Jersey Institute of Technology, Newark, NJ, USA, in 2015. He has been working toward the Ph.D. degree with the research group of Xiaofeng Wang, University of South Carolina, Columbia, SC, USA, since 2015.

His research interests include model predictive control, robust adaptive control, and robotics.



Zheng-Guang Wu was born in 1982. He received the B.S. and M.S. degrees from Zhejiang Normal University, Jinhua, China, in 2004 and 2007, respectively, and the Ph.D. degree from Zhejiang University, Hangzhou, China, in 2011.

He is currently with the Institute of Cyber-Systems and Control, Zhejiang University. His current research interests include hybrid systems, networked systems and computational intelligence.



Xiaofeng Wang (M'09) received the B.S. and M.S. degrees in mathematics from East China Normal University, Shanghai, China, in 2000 and 2003, respectively, and the Ph.D. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, USA, in 2009.

After working as a Postdoctoral Research Associate with the Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, he joined the University of South Carolina at Columbia, Columbia, SC, USA, in 2012, and is an Associate Professor with the Department of Electrical Engineering. His research interests include networked control systems, event-based control, robust adaptive control, cyber-physical systems, and robotics.



Hongye Su (SM'14) was born in 1969. He received the B.S. degree in industrial automation from the Nanjing University of Chemical Technology, Jiangsu, China, in 1990, and the M.S. and Ph.D. degrees from Zhejiang University, Hangzhou, China, in 1993 and 1995, respectively.

He was a Lecturer with the Department of Chemical Engineering, Zhejiang University, from 1995 to 1997. From 1998 to 2000, he was an Associate Professor with the Institute of Advanced Process Control, Zhejiang University. He is currently a Professor with the Institute of Cyber-Systems and Control, Zhejiang University. His current research interests include the robust control, time-delay systems, and advanced process control theory and applications.