Brief paper

Event-based model predictive control for nonlinear systems with dynamic disturbance[☆]

Pengfei Li [a,b], Tao Wang [a], Yu Kang [a,b,c,*], Kun Li [c], Yun-Bo Zhao [a,b]

[a] *Department of Automation, University of Science and Technology of China, Hefei, 230027, China*
[b] *Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei, 230088, China*
[c] *Institute of Advanced Technology, University of Science and Technology of China, Hefei, 230088, China*

ARTICLE INFO

ABSTRACT

In this paper, we investigate the event-based model predictive control (MPC) for constrained nonlinear systems with dynamic disturbance. An event-triggered disturbance prediction MPC (DPMPC) scheme and a self-triggered counterpart, which explicitly consider the disturbance dynamics, are proposed. For the event-triggered DPMPC scheme, the triggering condition relying on the state prediction error and the predicted disturbance sequence, updates at each time step based on the system states. For the self-triggered DPMPC scheme, the next triggering instant is determined by using the optimal state sequence and predicted disturbance sequence. In both event-based schemes, the optimal control problems are solved only at triggering instants, thus reducing the consumption of computational resource. The effectiveness of the two schemes is demonstrated by a simulation example.

1. Introduction

Model predictive control (MPC) is an advanced technique in achieving high control performance while explicitly considering the system constraints (Rawlings, Mayne, & Diehl, 2017). Therefore, it attracts much attention in recent years and has found application in diverse fields such as process control (Griffith, Biegler, & Patwardhan, 2018), transportation systems (Ye et al., 2019), and automotive systems (Cheng, Li, Guo, Chen, & Song, 2019). However, the major drawback of the MPC approaches is the heavy computational burden induced by the optimal control problem (OCP), hindering their usage in practical systems. An effective remedy is integrating with event-based control techniques, e.g., event-triggered control and self-triggered control, in order to save the computational resource.

By now there exist a great amount of works investigating the event-triggered and self-triggered MPC aiming at enlarging the triggering interval (Hashimoto, Adachi, & Dimarogonas, 2017; Liu, Gao, Li, & Xu, 2018; Sun, Xia, Dai, & Campoy, 2020; Wang, Sun, & Chen, 2019). For the event-triggered MPC, a state-related triggering condition is implemented at the sensor side and is checked at each time step. Only when the condition is contravened, the controller updates the state information and solves the OCP to yield the control sequence. Therefore, the triggering condition is vital to saving computational resource. One type of typical triggering conditions is derived from the deviation between the actual state and the predicted one in order to ensure the recursive feasibility, see, e.g., Hashimoto et al. (2017), Liu et al. (2018), Sun et al. (2020), Wang et al. (2019). Another type is derived from perspective of stability, through guaranteeing the decrement of Lyapunov function (Hashimoto, Adachi, & Dimarogonas, 2015; He & Shi, 2015; Zou, Su, Li, Niu, & Li, 2019). It is noted that the implementation of an event-triggered mechanism requires periodic or continuous monitoring system states, aggravating the sensing cost. To address this issue, self-triggered MPC is proposed via simultaneously solving the OCP and determining the next triggering instant. Similar to the event-triggered ones, the triggering instants are determined from the view of feasibility (Cui & Li, 2022), stability (Eqtami, Heshmati-Alamdari, Dimarogonas, & Kyriakopoulos, 2013) or both (Li, Kang, Zhao, & Wang, 2021; Sun, Dai, Liu, Dimarogonas, & Xia, 2019). However, the triggering instants of the event are predicted by using the worst case of the disturbance, resulting in a more conservative result than the event-triggered one.

* Corresponding author at: Department of Automation, University of Science and Technology of China, Hefei, 230027, China.
E-mail addresses: puffylee@ustc.edu.cn (P. Li), wanghtao@mail.ustc.edu.cn (T. Wang), kangduyu@ustc.edu.cn (Y. Kang), zkdlk@ustc.edu.cn (K. Li), ybzhao@ustc.edu.cn (Y.-B. Zhao).

Observed from the triggering conditions in Cui and Li (2022), Hashimoto et al. (2017), Liu et al. (2018), Sun et al. (2020), Wang et al. (2019), the triggering frequency is susceptible to the state prediction precision. In fact, the state prediction error between the actual state and the predicted one is resulted from the disturbance. Hence, existing works try to make use of extra disturbance information to improve the prediction precision. One widely used methodology is to make use of the disturbance rejection technique to suppress disturbance. Specifically, Incremona, Ferrara, and Magni (2017) and Rubagotti, Raimondo, Ferrara, and Magni (2011) adopted the integral sliding-mode control to compensate matched disturbance, reducing the uncertainty of the prediction model. Sun, Xia, Dai, Liu, and Ma (2017) employed the disturbance observer and a feed-forward control law to estimate and compensate the uncertainties. Another effective routine, known as disturbance prediction technique, involves integrating the predicted disturbance into the prediction model to reduce the state error. For example, Li et al. (2021) and Lin and Görges (2020) considered slowly varying disturbance and took advantage of the disturbance change rate to generate the predicted disturbance sequence. Yoo and Johansson (2021) established the relationship between disturbance, control inputs and system states by statistical learning method, and based on which generated the predicted disturbance sequence.

In this paper, we focus on both the event-triggered and self-triggered MPC for discrete-time nonlinear systems subject to additive disturbance. Different from the conventional bounded disturbance, the one considered here is described by a dynamical model that relies on the system states and disturbance. This modeling provides a more accurate description of system uncertainties, and thus has potential benefits in achieving the aim of improving prediction precision. However, such problem settings pose unique challenges to designing the event-based MPC schemes. First, how to formulate the state-dependent dynamic disturbance in the OCP. Second, how to explicitly analyze the state prediction error under the novel designed OCP. Third, how to design the triggering condition to lower the triggering frequency while ensuring recursive feasibility and stability.

To solve the above challenges, we propose event-triggered and self-triggered disturbance prediction MPC (DPMPC). The predicted disturbance sequence is generated by employing the latest obtained predicted state sequence and actual disturbance. The state and disturbance prediction errors are analyzed simultaneously via the state extension method. Both the event-triggering and the self-triggering conditions are designed by taking advantage of the predicted state with high precision. As a result, the triggering frequency is significantly reduced. Furthermore, the recursive feasibility of both DPMPC schemes and the input-to-state stability (ISS) of the closed-loop systems are guaranteed.

The main contributions are summarized as follows:

1. A recursive disturbance prediction approach is proposed and the corresponding prediction error is explicitly analyzed.
2. A novel constraint tightening method that takes the state prediction error and the disturbance prediction error into consideration is developed to ensure state constraint satisfaction.
3. The event-triggering condition relying on the predicted disturbance sequence and the self-triggering condition relying on the disturbance learning error are designed, and the recursive feasibility and stability are ensured.

The structure of this paper is as follows. Section 2 describes the plant model, disturbance model, and related assumptions. Sections 3 and 4 propose and analyze the event-triggered and

self-triggered DPMPC schemes, respectively. Section 5 shows the simulation results. Section 6 concludes this paper.

Notations. The symbols $\mathbb{R}$ and $\mathbb{R}^n$ refer to the real numbers set and n-dimensional real space. For a vector $x \in \mathbb{R}^n$, x^T, $\|x\|_2$ and $\|x\|_P$ denote its transpose, its Euclidean norm and P-weighted norm, respectively. Given a square matrix $Q \in \mathbb{R}^{n \times n}$, $\lambda_{\max}(Q)$ and $\lambda_{\min}(Q)$ are its maximum and minimum eigenvalue, respectively. Given two nonempty sets $\mathbb{X}$ and $\mathbb{Y}$, the Minkowski addition set is defined by $\mathbb{X} \oplus \mathbb{Y} \triangleq \{x + y | x \in \mathbb{X}, y \in \mathbb{Y}\}$, and the Pontryagin difference set is by $\mathbb{X} \ominus \mathbb{Y} \triangleq \{x : x + y \in \mathbb{X}, \forall y \in \mathbb{Y}\}$. Given a compact set $\mathbb{W}$, the projection operator is defined by $\text{Proj}_{\mathbb{W}}(x) = \arg\min_{z \in \mathbb{W}} \|x - z\|_P^2$. Note that $x = \text{Proj}_{\mathbb{W}}(x), \forall x \in \mathbb{W}$.

2. System descriptions

In this paper, the considered discrete-time nonlinear systems have the following form:

$$x(k + 1) = f(x(k), u(k)) + w(k), \tag{1}$$

where $x \in \mathbb{X} \subseteq \mathbb{R}^n$, $u \in \mathbb{U} \subseteq \mathbb{R}^m$ and $w \in \mathbb{W} \subseteq \mathbb{R}^n$ represent the system state, control input and disturbance, respectively, sets $\mathbb{X}$ and $\mathbb{U}$ are compact sets and satisfy $0 \in \mathbb{X}$ and $0 \in \mathbb{U}$. $\mathbb{W}$ is also a compact set and satisfies $\|w\|_P \leq \bar{w}, \forall w \in \mathbb{W}$.

In particular, suppose that the state-dependent disturbance w evolves according to the following dynamics

$$w(k + 1) = g(w(k), x(k)) + v(k), \tag{2}$$

where g is a continuous function that can be determined by learning techniques, v can be regarded as the learning error and satisfies $\|v\|_P \leq \bar{v}$. In practice, it is reasonable to assume that $\bar{v} \leq \bar{w}$. Such representation of disturbance is quite general, and can be used to characterize various modeling uncertainties and exterior perturbations.

Before proceeding, we make the following assumption.

Assumption 1. The plant model (1) and the disturbance model (2) are Lipschitz continuous in their arguments, i.e., there exist constants L_f, L_w and L_x such that

$$\|f(x, u) - f(y, u)\|_P \leq L_f \|x - y\|_P \tag{3}$$

$$\|g(w, x) - g(\omega, y)\|_P \leq L_w \|w - \omega\|_P + L_x \|x - y\|_P \tag{4}$$

for all $(x, y, w, \omega, u) \in \mathbb{X} \times \mathbb{X} \times \mathbb{W} \times \mathbb{W} \times \mathbb{U}$.

Remark 1. The Lipschitz assumption in (4) can be satisfied for many disturbance dynamics. For example, Yi, Zheng, and Liu (2022) provide three typical types of disturbance dynamics, i.e., attenuated harmonic disturbance, sawtooth wave disturbance, and white noise disturbance. Since these disturbance dynamics are represented by Lipschitz continuous exogenous dynamic neural network models in that work, the Lipschitz constants can then be identified.

Remark 2. Various approaches, e.g., statistical learning (Yoo & Johansson, 2021), deep learning (Chen, Cao, Kang, Sun, & Wang, 2020), system identification (Kaheman, Brunton, & Kutz, 2020), can be adopted to learn the function g. To meet the Lipschitz continuous assumption (4), additional modifications should be made. For example, a deep neural network is used to learn g with Lipschitz constants in Shi et al. (2019) by adding an extra Lipschitz constraint into the learning error minimization problem.

The aim of this paper is to design event-based MPC schemes to reduce the consumption of computational resources as much as possible by making use of the disturbance prediction on the basis of dynamics (2).

Remark 3. State extension is a common used method to exploit the disturbance dynamics (2) in designing the prediction model, i.e., let $\zeta(k) = [x^T(k), w^T(k)]^T$ and then use both the plant model (1) and the nominal disturbance dynamics (2) to serve as the prediction models (El-Ferik, 2020). However, such a method may face challenges as the expression of g, which depends on the adopted learning method, can be extremely complex and may not be smooth, dramatically increasing the difficulty in solving the optimization problem. Moreover, the extended state $\zeta(k)$ doubles the dimension of decision variables, leading to high computational complexity. Therefore, a new method that explicitly exploits the disturbance dynamics (2) in formulating the OCP while possessing low computational complexity should be proposed.

3. Event-triggered DPMPC

In this section, an event-triggered DPMPC scheme is proposed, the recursive feasibility and stability are discussed.

3.1. Optimal control problem

For an event-based MPC scheme, the OCP is solved only at each triggering instant k_j (the jth triggering instant) to generate the optimal control and state sequence. For ease of representation, we directly give the form of the OCP in what follows. Note that $\hat{x}_i(k_j)$ and $u_i(k_j)$ represent the state and input prediction i steps ahead from time k_j, respectively.

$$
\begin{aligned}
\min_{\mathbf{u}} \; & V_N(x(k_j), \mathbf{u}(k_j)) \\
\text{s.t. } & \hat{x}_{i+1}(k_j) = f(\hat{x}_i(k_j), u_i(k_j)) + \bar{w}_i(k_j) \\
& \hat{x}_0(k_j) = x(k_j) \\
& \hat{x}_i(k_j) \in \mathbb{X}(i) \\
& \|\hat{x}_i(k_j) - \bar{x}_i(k_j)\|_P \le \alpha \\
& u_i(k_j) \in \mathbb{U}, i = 1, \ldots, N-1 \\
& \hat{x}_N(k_j) \in \mathbb{X}_f
\end{aligned}
\tag{5}
$$

where $V_N(x(k_j), \mathbf{u}(k_j)) = \sum_{i=0}^{N-1} \|\hat{x}_i(k_j)\|_Q^2 + \|u_i(k_j)\|_R^2 + \|\hat{x}_N(k_j)\|_P^2$ is the MPC value function, N is the prediction horizon, P, Q, R are all positive definite matrices, $\mathbb{X}_f$ is the terminal state constraint set, $\bar{w}_i(k_j)$ is the $(i+1)$th element of the predicted disturbance sequence $\bar{\mathbf{w}}(k_j)$ generated at time k_j, $\mathbb{X}(i)$ is a tightened state constraint set, $\bar{x}_i(k_j)$ is defined in (6), and $\alpha > 0$ is a design parameter related to the feasibility.

Once the OCP (5) is solved at k_j, the optimal state and control sequences are denoted by $\hat{\mathbf{x}}^*(k_j) = \{\hat{x}_0^*(k_j), \ldots, \hat{x}_N^*(k_j)\}$ and $\mathbf{u}^*(k_j) = \{u_0^*(k_j), \ldots, u_{N-1}^*(k_j)\}$, respectively. Moreover, the optimal MPC value function is then denoted by

$$
V_N^*(x(k_j)) = \sum_{i=0}^{N-1} \|\hat{x}_i^*(k_j)\|_Q^2 + \|u_i^*(k_j)\|_R^2 + \|\hat{x}_N^*(k_j)\|_P^2
$$

In the following parts, we will describe the design approach of the predicted disturbance sequence $\bar{\mathbf{w}}(k)$ and determine the related sets as well as the parameters of the OCP (5).

3.1.1. Determination of $\bar{\mathbf{w}}(k)$

It is known that a more accurate model representation of the system dynamics in MPC often implies high prediction precision. Due to this fact, we added a predicted disturbance generated by the disturbance dynamics (2) in the prediction model. Such a model, compared with the nominal counterpart, has a greater potential to reduce the state prediction error, leading to a lower triggering frequency.

The construction procedure of sequence $\bar{\mathbf{w}}(k)$, $k > k_j$ is designed recursively as follows:

$$
\begin{cases}
\bar{x}_i(k) = f(\bar{x}_{i-1}(k), \bar{u}_{i-1}(k)) + \bar{w}_{i-1}(k) \\
\bar{u}_i(k) = \begin{cases} \tilde{u}_{i+1}(k-1), & i = 1, \ldots, N-2 \\ \kappa(\bar{x}_{N-1}(k)), & i = N-1 \end{cases} \\
\bar{w}_i(k) = \text{Proj}_{\mathbb{W}}\big(g(\bar{w}_{i-1}(k), \bar{x}_{i-1}(k))\big)
\end{cases}
\tag{6}
$$

$$
\bar{x}_N(k) = f(\bar{x}_{N-1}(k), \kappa(\bar{x}_{N-1}(k))) + \bar{w}_{N-1}(k)
$$

where $\tilde{u}_{i+1}(k-1) = \begin{cases} u_{i+1}^*(k_j), & \text{if } k = k_j + 1 \\ \bar{u}_{i+1}(k-1), & \text{if } k > k_j + 1 \end{cases}$, $\bar{x}_0(k) = x(k)$,

$\bar{u}_0(k) = \begin{cases} u_1^*(k_j), & \text{if } k = k_j + 1 \\ \bar{u}_1(k-1), & \text{if } k > k_j + 1 \end{cases}$, and $\bar{w}_0(k) = \text{Proj}_{\mathbb{W}}(g(w(k-1), x(k-1)))$. $\kappa(x)$ is an auxiliary control law designed below. It is noteworthy that $w(k-1)$ is available at time k because $w(k-1) = x(k) - f(x(k-1), u(k-1))$. Since $g(w, x)$ may not be in set $\mathbb{W}$ although $w \in \mathbb{W}$ and $x \in \mathbb{X}$, we use the projection operator $\text{Proj}_{\mathbb{W}}(.)$ here to guarantee $\bar{w}_i(k) \in \mathbb{W}$, $i = 0, \ldots, N-1$.

Observe that from the construction procedure (6), we can generate the sequences $\bar{\mathbf{w}}(k) = \{\bar{w}_0(k), \ldots, \bar{w}_{N-1}(k)\}$, $\bar{\mathbf{x}}(k) = \{\bar{x}_0(k), \ldots, \bar{x}_N(k)\}$, $\bar{\mathbf{u}}(k) = \{\bar{u}_0(k), \ldots, \bar{u}_{N-1}(k)\}$, $\forall k > k_j$, simultaneously.

The following lemma indicates the error of the predicted disturbance and the states between two consecutive time steps.

Lemma 1. Denote $\bar{e}_w(i) = \|\bar{w}_i(k) - \bar{w}_{i+1}(k-1)\|_P$ and $\bar{e}_x(i) = \begin{cases} \|\bar{x}_i(k) - \hat{x}_{i+1}^*(k_j)\|_P, & \text{if } k = k_j + 1 \\ \|\bar{x}_i(k) - \bar{x}_{i+1}(k-1)\|_P, & \text{if } k > k_j + 1 \end{cases}$, where $i = 0, \ldots, N-1$, then the following inequality holds

$$
\begin{bmatrix} \bar{e}_x(i) \\ \bar{e}_w(i) \end{bmatrix} \le \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^i \begin{bmatrix} \bar{v} \\ L_w \bar{v} \end{bmatrix} + \Gamma(\alpha, i)
\tag{7}
$$

where $\Gamma(\alpha, i) = \sum_{s=0}^{i-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} 0 \\ L_x \alpha \end{bmatrix}$.

Proof. See Appendix A.

Remark 4. In min–max MPC scheme, all possible disturbance sequences are considered to ensure the control constraint and the control actions are obtained by minimizing the worst-case performance cost (Raimondo, Limon, Lazar, Magni, & ndez Camacho, 2009). In contrast, the DPMPC in (5) only considers a likely disturbance sequence to minimize the performance cost. Therefore, the proposed DPMPC has the advantage of less conservativeness (i.e., DPMPC has a larger feasible solution space) and lower computational overhead.

3.1.2. Design of $\mathbb{X}(i)$ and $\mathbb{X}_f$

The tightened constraint set $\mathbb{X}(i)$ is designed for state constraint satisfaction and terminal constraint set $\mathbb{X}_f$ is designed for feasibility and stability.

Based on Lemma 1, $\mathbb{X}(i)$ can be designed as

$$
\mathbb{X}(i) = \mathbb{X} \ominus \mathbb{B}(i), i = 1, \ldots, N-1
$$
$$
\mathbb{B}(i) \triangleq \{e : \|e\|_P \le \Xi(i)\}
\tag{8}
$$

where $\Xi(i) = \begin{bmatrix} 1 & 0 \end{bmatrix} \sum_{s=0}^{i-1} \left\{ \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} \bar{v} \\ L_w \bar{v} \end{bmatrix} + \Gamma(\alpha, s) \right\}$.

The following lemma shows that the true state constraint satisfaction can be guaranteed under the designed $\mathbb{X}(i)$.

Lemma 2. Suppose that the elements in $\mathbf{u}^*(k_j)$ are applied to the plant in turn during k_j and k_{j+1}, then the constraint $\hat{x}_i(k_j) \in \mathbb{X}(i)$ guarantees $x(k) \in \mathbb{X}$, $\forall k \in (k_j, k_{j+1}]$.

Proof. See Appendix B.

Next, we design the terminal state constraint set $\mathbb{X}_f$. Similar to the one in Sun et al. (2019), $\mathbb{X}_f$ is defined as $\mathbb{X}_f \triangleq \{x : \|x\|_P \leq \epsilon_f\}$. Besides, the parameters P, Q, R, ϵ_f should meet the following standard conditions.

Assumption 2. There exists an auxiliary constraint set $\mathbb{X}_a$ with the form of $\mathbb{X}_a = \{x : \|x\|_P \leq \epsilon_a\}$ and an auxiliary control law $\kappa(x) : \mathbb{X}_a \to \mathbb{U}$, such that

1. $\mathbb{X}_f \subseteq \mathbb{X}_a \subseteq \mathbb{X}(N-1)$;
2. $f(x, \kappa(x)) + w \in \mathbb{X}_f$, $\forall x \in \mathbb{X}_a$, $w \in \mathbb{W}$;
3. $\|f(x, \kappa(x))\|_P^2 - \|x\|_P^2 \leq -\|x\|_Q^2 - \|\kappa(x)\|_R^2$, $\forall x \in \mathbb{X}_a$.

Remark 5. The above assumption is rather standard. Condition (1) and (3), which are useful in ensuring recursive feasibility and stability, respectively, can be found in Rawlings et al. (2017), Sun et al. (2019). Note that from condition (3), one further obtains

$$\|f(x, \kappa(x)) + w\|_P^2 - \|x\|_P^2$$
$$\leq \|f(x, \kappa(x))\|_P^2 - \|x\|_P^2 + \bar{w}^2 + 2f(x, \kappa(x))^T P w$$
$$\leq -\|x\|_Q^2 - \|\kappa(x)\|_R^2 + \bar{w}^2 + 2\epsilon_f \bar{w}. \tag{9}$$

Let $\rho(\bar{w}) = \bar{w}^2 + 2\epsilon_f \bar{w}$, which is a class $\mathcal{K}$ function. This equation is useful in proving the ISS property. Condition (2) gives a restriction on disturbance, that is,

$$\bar{w} < \epsilon_a - \max_{x \in \mathbb{X}_a} \|f(x, \kappa(x))\|_P \tag{10}$$

which guarantees the existence of the terminal set $\mathbb{X}_f$.

3.2. Event-triggering condition and recursive feasibility

In this part, an event-triggering condition for guaranteeing the recursive feasibility is proposed. The result is demonstrated in the following theorem.

Theorem 1. *Suppose that the system state and disturbance evolve according to* (1) *and* (2), *respectively. Then OCP* (5) *is recursively feasible if* (10) *as well as the following inequality holds*

$$\begin{bmatrix} 1 & 0 \end{bmatrix} \left(\begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^N \begin{bmatrix} 0 \\ \bar{v} \end{bmatrix} + \Gamma(\alpha, N-1) \right) \leq \epsilon_a - \epsilon_f, \tag{11}$$

and the event-triggering condition is design as

$$k_{j+1} = \inf_k \left\{ k : \sum_{s=0}^{N-2} L_f^s \|\bar{w}_{N-2-s}(k) - \bar{w}_{N-1-s}(k-1)\|_P \right.$$
$$\left. + L_f^{N-1} \|x(k) - \tilde{x}(k)\|_P \geq \epsilon_a - \epsilon_f \right\} \tag{12}$$

where $\tilde{x}(k) = \begin{cases} \hat{x}_1^*(k_j), & \text{if } k = k_j + 1 \\ \bar{x}_1(k-1), & \text{if } k > k_j + 1 \end{cases}.$

Proof. To prove this theorem, we show that the generated sequences $\bar{\mathbf{u}}(k)$ and $\bar{\mathbf{x}}(k)$ in (6) meet all constraints in OCP (5) for $k_j < k \leq k_{j+1}$. To be specific, we assume that $\bar{\mathbf{u}}(k-1)$ and $\bar{\mathbf{x}}(k-1)$ are feasible solutions (This is true for $k = k_j + 1$ since $\bar{\mathbf{u}}(k-1)$ and $\bar{\mathbf{x}}(k-1)$ become $\mathbf{u}^*(k_j)$ and $\hat{\mathbf{x}}^*(k_j)$, respectively), and then prove $\bar{\mathbf{u}}(k)$ and $\bar{\mathbf{x}}(k)$ are also feasible. It should be indicated that the constraint $\|\hat{x}_i(k_j) - \bar{x}_i(k_j)\|_P \leq \alpha$ in (5) can be met for all $\alpha > 0$. The proof can be completed from four aspects.

1. $\bar{x}_i(k) \in \mathbb{X}(i)$, $i = 1, \ldots, N-2$. To verify this result, we first note that $\|\bar{x}_i(k)\|_P \leq \|\bar{x}_{i+1}(k-1)\|_P + \bar{e}_x(i)$. Then, based on (7) and (8), it holds that

$$\bar{x}_i(k) \in \mathbb{X} \ominus \mathbb{B}(i+1)$$

$$\oplus \begin{bmatrix} 1 & 0 \end{bmatrix} \left(\begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^i \begin{bmatrix} \bar{v} \\ L_w \bar{v} \end{bmatrix} + \Gamma(\alpha, i) \right)$$
$$\in \mathbb{X} \ominus \mathbb{B}(i) = \mathbb{X}(i)$$

2. $\bar{x}_{N-1}(k) \in \mathbb{X}(N-1)$. First, one notes that $\|\bar{x}_{N-1}(k) - \bar{x}_N(k-1)\|_P \leq \sum_{s=0}^{N-2} L_f^s \|\bar{w}_{N-2-s}(k) - \bar{w}_{N-1-s}(k-1)\|_P + L_f^{N-1} \|x(k) - \bar{x}_1(k-1)\|_P$ holds. Recalling that $\bar{x}_N(k-1) \in \mathbb{X}_f$ and the event-triggering condition (12), we have $\|\bar{x}_{N-1}(k)\|_P \leq \|\bar{x}_N(k-1)\|_P + \|\bar{x}_{N-1}(k) - \bar{x}_N(k-1)\|_P \leq \epsilon_f + \epsilon_a - \epsilon_f = \epsilon_a$. That is $\bar{x}_{N-1}(k) \in \mathbb{X}_a \subseteq \mathbb{X}(N-1)$.

3. $\bar{x}_N(k) \in \mathbb{X}_f$. Notice that (10) implies that condition (2) in Assumption 2 holds. Then, according to this condition, we directly have $\bar{x}_N(k) \in \mathbb{X}_f$.

4. $\bar{u}_i(k) \in \mathbb{U}$, $i = 0, \ldots, N-1$. Firstly, according to the definition of $\bar{u}_i(k)$ in (6), we directly have $\bar{u}_i(k) \in \mathbb{U}$, $i = 0, \ldots, N-2$. In addition, since $\bar{x}_{N-1}(k) \in \mathbb{X}_a$, the control constraint satisfaction can be easily guaranteed due to the definition of control law κ in Assumption 2.

For the sake of brevity, the above proof only considers the case when $k > k_j + 1$. Actually, when $k = k_j + 1$, the same result can be obtained by a similar procedure where $\hat{x}_i^*(k_j)$ and $u_i^*(k_j)$ are used instead of $\bar{x}_i(k-1)$ and $\bar{u}_i(k-1)$.

Summarizing the above statements, we show that $\bar{\mathbf{u}}(k)$ is a feasible solution to OCP (5) with $k = k_{j+1}$ ($k_j + 1 \leq k_{j+1} \leq k_j + N$). This completes the proof.

In triggering condition (12), the predicted disturbance sequences at two consecutive time steps are utilized, which seems to bring conservativeness for the event-triggered mechanism. But it is not the case in practice as the state prediction error in (7) has been significantly reduced. To be specific, one has $\|\bar{x}_{N-1}(k) - \bar{x}_N(k-1)\|_P \leq L_f^i \bar{w}$ under the conventional MPC scheme. Therefore, if $\bar{w}$ (the upper bound of disturbance) and $\bar{v}$ (the upper bound of learning error) satisfy

$$\begin{bmatrix} 1 & 0 \end{bmatrix} \left(\begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^{N-1} \begin{bmatrix} \bar{v} \\ L_w \bar{v} \end{bmatrix} + \Gamma(\alpha, N-1) \right) \leq L_w^{N-1} \bar{w},$$

then the event-triggering condition in (12) has less conservativeness. This inequality can be easily satisfied as the inequality $\bar{v} \ll \bar{w}$ often holds by using the existing learning methods. Note that $\Gamma(\alpha, N-1)$ in the above inequality is neglected as α can be selected arbitrarily small. Therefore, we claim that the proposed triggering condition is particularly suitable for the case when the learning error $\bar{v}$ is far less than $\bar{w}$.

Remark 6. The proposed event-triggering condition in this section is distinct from the conventional conditions (Hashimoto et al., 2017; Wang et al., 2019) in three aspects. Firstly, the predicted disturbance sequences are utilized in the event-triggering condition (12) to guarantee the recursive feasibility of the DPMPC scheme. Secondly, the event-triggering condition (12) is updated at each time step by generating the feasible predicted state sequence $\bar{\mathbf{x}}(k)$, which enables to enlarge the triggering interval. Thirdly, the constraint that the interval between two consecutive triggering instants should be no larger than the prediction horizon has been removed.

The event-triggered DPMPC scheme is summarized in Algorithm 1.

3.3. Stability analysis

In this part, we discuss the stability of the system under the event-triggered DPMPC scheme.

Algorithm 1 Event-Triggered DPMPC

1: **while** (1) **do**
2:    Measure the current system state $x(k)$;
3:    Generate $\bar{\mathbf{w}}(k)$, $\bar{\mathbf{x}}(k)$ and $\bar{\mathbf{u}}(k)$ based on (6);
4:    **if** condition (12) is triggered **then**
5:       Update the triggering instant $j \leftarrow j + 1$; $k_j \leftarrow k$;
6:       Solve OCP (5) to obtain $\mathbf{u}^*(k_j)$ and $\hat{\mathbf{x}}^*(k_j)$;
7:       Apply the control action $u_0^*(k_j)$ to the plant;
8:    **else**
9:       Apply the control action $\bar{u}_0(k)$ to the plant;
10:   **end if**
11:   Update the time instant $k \leftarrow k + 1$;
12: **end while**

Theorem 2. *For system* (1) *under Algorithm* 1, *suppose that Assumptions* 1 *and* 2 *hold, then the overall system is ISS, i.e., there exist* $\beta \in \mathcal{KL}$ *and* $\sigma \in \mathcal{K}$ *such that*

$$\|x(k)\| \leq \beta(\|x(k_0)\|, k - k_0) + \sigma(\bar{w}) \tag{13}$$

Proof. As indicated in Pin and Parisini (2011), we need to find an ISS-type Lyapunov function to verify the input-to-state stability, i.e., there exists a positive definite function $J(\cdot) : \mathbb{X} \to \mathbb{R}$ such that the following inequalities hold

$$\alpha_1(x) \leq J(x) \leq \alpha_2(x) + \rho(\bar{w}), \tag{14}$$

$$J(f(x, u) + w) - J(x) \leq -\alpha_3(x) + \gamma(\bar{w}), \forall x \in \mathbb{X} \tag{15}$$

where α_1, α_2, α_3 are class $\mathcal{K}_\infty$ functions and ρ, γ are class $\mathcal{K}$ functions.

In what follows, we need to show the following function is an ISS-type Lyapunov function.

$$J_N(x(k)) = \begin{cases} V_N^*(x(k_j)), & \text{if } k = k_j \\ V_N(x(k)), & \text{if } k_j < k < k_{j+1} \end{cases} \tag{16}$$

where $V_N(x(k)) = \sum_{i=0}^{N-1} \|\bar{x}_i(k)\|_Q^2 + \|\bar{u}_i(k)\|_R^2 + \|\bar{x}_N(k)\|_P^2$.

Firstly, note that

$$J_N(x(k)) \geq \alpha_1(x(k)) \triangleq \|x(k)\|_Q^2, \forall x \in \mathbb{X} \tag{17}$$

Then, following Theorem 3 in Magni, Raimondo, and Scattolini (2006), we can obtain the following inequality, with $\rho(\bar{w})$ being defined in Remark 5,

$$J_N(x(k)) \leq \|x(k)\|_P^2 + N\rho(\bar{w}), \forall x \in \mathbb{X}_a \tag{18}$$

by repeatedly using Condition 3 in Assumption 2. By using the technique reported in Rubagotti et al. (2011, Lemma 4), we can extend the upper bound (18) to $\mathbb{X}$, i.e., there exist two functions $\alpha_2 \in \mathcal{K}_\infty$ and $\bar{\rho} \in \mathcal{K}$ such that $J_N(x(k)) \leq \alpha_2(x(k)) + \bar{\rho}(\bar{w})$ for $x \in \mathbb{X}$.

Next, we need to show (15) is satisfied for $J_N(x(k))$.

If $k = k_{j+1} - 1$, we have $V_N^*(x(k_{j+1})) \leq V_N(x(k_{j+1}))$ because $\bar{\mathbf{u}}(k_{j+1})$ is a feasible solution. Therefore, for any $k \in [k_j, k_{j+1})$, one has

$$J_N(x(k+1)) - J_N(x(k))$$
$$\leq V_N(x(k+1)) - V_N(x(k))$$
$$\leq -\|x(k)\|_Q^2 + \sum_{i=0}^{N-2}(\|\bar{x}_i(k+1)\|_Q^2 - \|\bar{x}_{i+1}(k)\|_Q^2)$$
$$\quad + \|\bar{x}_{N-1}(k+1)\|_Q^2 + \|\bar{u}_{N-1}(k+1)\|_R^2$$
$$\quad + \|\bar{x}_N(k+1)\|_P^2 - \|\bar{x}_N(k)\|_P^2$$
$$\leq -\|x(k)\|_Q^2 + \sum_{i=0}^{N-2} L_Q \|\bar{x}_i(k+1) - \bar{x}_{i+1}(k)\|_P$$

$$\quad + L_P \|\bar{x}_{N-1}(k+1) - \bar{x}_N(k)\|_P + \rho(\bar{w})$$
$$\leq -\alpha_3(x(k)) + \gamma(\bar{w}) \tag{19}$$

where L_Q and L_P are constants that satisfy $\|x\|_Q^2 - \|y\|_Q^2 \leq L_Q \|x - y\|_Q$, $\forall x, y \in \mathbb{X}$ and $\|x\|_P^2 - \|y\|_P^2 \leq L_P \|x - y\|_P$, $\forall x, y \in \mathbb{X}_a$, $\gamma(\bar{w}) = L_P \begin{bmatrix} 1 & 0 \end{bmatrix} \left(\begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^{N-1} \begin{bmatrix} \bar{w} \\ L_w \bar{w} \end{bmatrix} + \Gamma(\alpha, N-1) \right) + L_Q \sqrt{\frac{\lambda_{\max}(Q)}{\lambda_{\min}(P)}} \begin{bmatrix} 1 & 0 \end{bmatrix} \sum_{i=0}^{N-2} \left(\begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^i \begin{bmatrix} \bar{w} \\ L_w \bar{w} \end{bmatrix} + \Gamma(\alpha, i) \right) + \rho(\bar{w})$.

Notice that the above inequality still holds for the case $k = k_j$ by using $\hat{x}_i^*(k_j)$ to replace $\bar{x}_i(k)$.

Incorporating the above statements, we show that $J_N(x)$ is an ISS-type Lyapunov function, which implies the ISS of the overall system. $\blacksquare$

4. Self-triggered DPMPC

In this section, we develop a self-triggered DPMPC scheme to compute the optimal predictive control and state sequences and determine the next triggering instant simultaneously. Compared with the event-triggered counterpart, the actual states during two consecutive triggering instants are not available in the self-triggered DPMPC scheme. Therefore, the disturbance prediction technique in (6) should be modified and the self-triggering condition cannot be directly derived from the event-triggering condition (12).

In what follows, we first modify the disturbance prediction technique and then derive the self-triggering condition to formulate the self-triggered DPMPC scheme.

The predicted disturbance sequence at k_{j+1} is constructed based on the optimal control sequence $\mathbf{u}^*(k_j)$. The construction procedure is shown as follows.

$$\begin{cases} \bar{x}_i(k_{j+1}) = f(\bar{x}_{i-1}(k_{j+1}), \bar{u}_{i-1}(k_{j+1})) + \bar{w}_{i-1}(k_{j+1}) \\ \bar{u}_i(k_{j+1}) = \begin{cases} \bar{u}_{i+\Delta_j}(k_j), & i = 1, \ldots, N - \Delta_j - 1 \\ \kappa(\bar{x}_i(k_{j+1})), & i = N - \Delta_j, \ldots, N - 1 \end{cases} \\ \bar{w}_i(k_{j+1}) = \text{Proj}_{\mathbb{W}}\left(g(\bar{w}_{i-1}(k_{j+1}), \bar{x}_{i-1}(k_{j+1}))\right) \end{cases} \tag{20}$$
$$\bar{x}_N(k_{j+1}) = f(\bar{x}_{N-1}(k_{j+1}), \kappa(\bar{x}_{N-1}(k_{j+1}))) + \bar{w}_{N-1}(k_{j+1})$$

where $\Delta_j = k_{j+1} - k_j$, $\bar{x}_0(k_j) = x(k_j)$, $\bar{u}_0(k_{j+1}) = u_{\Delta_j}^*(k_j)$, and $\bar{w}_0(k_{j+1}) = \text{Proj}_{\mathbb{W}}\left(g(w(k_{j+1} - 1), x(k_{j+1} - 1))\right)$.

Note that this procedure, in contrast to the one (6) in event-triggered DPMPC scheme, is performed only at each triggering instant. To obtain $\bar{w}_0(k_{j+1})$, extra sampling of the system state before the triggering instant (i.e., $x(k_{j+1} - 1)$) is needed.

Next, we follow the idea of Sun et al. (2019) to design the self-triggered DPMPC scheme from recursive feasibility and control performance. The sub-optimal performance, defined in Sun et al. (2019), is

$$\sum_{k=0}^{\Delta_j} \|\hat{x}_i^*(k_j)\|_Q^2 + \|u_i^*(k_j)\|_R^2 \leq \frac{V_N^*(x(k_{j+1})) - V_N^*(x(k_j))}{\beta}$$

with $\beta < 1$ will be satisfied.

It is noted that in this self-triggered DPMPC scheme, the formulation of the OCP and its relevant parameters are the same as the ones designed in (5).

The following theorem validates the recursive feasibility of the DPMPC with the designed self-triggering condition.

Theorem 3. *Suppose that the system state and disturbance evolve according to* (1) *and* (2), *respectively. Then OCP* (5) *is recursively feasible if* (10) *and* (11) *hold, and the self-triggering condition is design as*

$$k_{j+1} = \min\{k_j + N, \bar{r}_{j+1}, \underline{r}_{j+1}\} \tag{21}$$

$$\bar{r}_{j+1} = \sup_{k}\left\{k : \sum_{s=N-(k-k_j)}^{N-1} \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} \bar{v} \\ L_w\bar{v} \end{bmatrix}\right.$$

$$\left. + \Gamma(\alpha, N-1) \le \epsilon_a - \epsilon_f \right\} \tag{22}$$

$$\underline{r}_{j+1} = \sup_{k}\left\{k : \sum_{i=k-k_j}^{N-1} \left[2\sqrt{\frac{\lambda_{\max}(Q)}{\lambda_{\min}(P)}} \|\hat{x}_i^*(k_j)\|_Q h(i) \right.\right.$$

$$\left. + \frac{\lambda_{\max}(Q)}{\lambda_{\min}(P)} h^2(i) \right] + (\epsilon_a + \epsilon_f)h(N)$$

$$\left. \le (1-\beta) \sum_{i=0}^{k-k_j-1} \left(\|\hat{x}_i^*(k_j)\|_Q^2 + \|u_i^*(k_j)\|_R^2 \right) \right\} \tag{23}$$

where $h(l) = \begin{bmatrix} 1 & 0 \end{bmatrix} \left(\sum_{s=0}^{k-k_j-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^{s+l-k+k_j} \begin{bmatrix} \bar{v} \\ L_w\bar{v} \end{bmatrix} \right) + \Gamma(\alpha, l-1)$.

To prove this theorem, we need the following lemma.

Lemma 3. *Considering the sequences $\bar{x}(k_{j+1})$, $\bar{u}(k_{j+1})$ and $\bar{w}(k_{j+1})$ constructed according to (20) and denoting $\bar{\xi}_x(i) \triangleq \|\bar{x}_i(k_{j+1}) - \hat{x}^*_{i+\Delta_j}(k_j)\|_P$, $\bar{\xi}_w(i) \triangleq \|\bar{w}_i(k_{j+1}) - \bar{w}_{i+\Delta_j}(k_j)\|_P$, $\xi_x(i) \triangleq \|x(k_j + i) - \hat{x}_i^*(k_j)\|_P$ and $\xi_w(i) \triangleq \|w(k_j+i) - \bar{w}_i(k_j)\|_P$, the following inequalities hold for $0 \le i \le N - \Delta_j$.*

$$\begin{bmatrix} \bar{\xi}_x(i) \\ \bar{\xi}_w(i) \end{bmatrix} \le \sum_{s=0}^{\Delta_j-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^{s+i} \begin{bmatrix} \bar{v} \\ L_w\bar{v} \end{bmatrix} + \Gamma(\alpha, i + \Delta_j - 1) \tag{24}$$

Proof. See Appendix C

The proof of Theorem 3: Recalling the sequences $\bar{x}(k_{j+1})$ and $\bar{u}(k_{j+1})$ constructed by (20), this proof can be completed by following the same lines of the logic given in Theorem 1 within the following four steps. In particular,

1. $\bar{x}_i(k_{j+1}) \in \mathbb{X}(i)$, $i = 1, \ldots, N - \Delta_j$. To verify this result, we observe that $\|\bar{x}_i(k_{j+1})\|_P \le \|\hat{x}^*_{i+\Delta_j}(k_j)\|_P + \|\bar{x}_i(k_{j+1}) - \hat{x}^*_{i+\Delta_j}(k_j)\|_P$. Note that

$$\sum_{s=0}^{i+\Delta_j-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} \bar{v} \\ L_w\bar{v} \end{bmatrix} - \sum_{s=0}^{i-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} \bar{v} \\ L_w\bar{v} \end{bmatrix}$$

$$= \sum_{s=0}^{\Delta_j-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^{s+i} \begin{bmatrix} \bar{v} \\ L_w\bar{v} \end{bmatrix} \tag{25}$$

and

$$\sum_{s=0}^{i+\Delta_j-1} \Gamma(\alpha, s) - \Gamma(\alpha, i + \Delta_j - 1) \ge \sum_{s=0}^{i-1} \Gamma(\alpha, s) \tag{26}$$

Then, based on (25) and (26), we can verify that

$$\bar{x}_i(k_{j+1}) \in \mathbb{X} \ominus \mathbb{B}(i + \Delta_j) \oplus \begin{bmatrix} 1 & 0 \end{bmatrix} \left(\Gamma(\alpha, i + \Delta_j - 1) \right.$$

$$\left. + \sum_{s=0}^{\Delta_j-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^{s+i} \begin{bmatrix} \bar{v} \\ L_w\bar{v} \end{bmatrix} \right)$$

$$\in \mathbb{X} \ominus \mathbb{B}(i) = \mathbb{X}(i)$$

2. $\bar{x}_i(k_{j+1}) \in \mathbb{X}(i)$, $i = N - \Delta_j + 1, \ldots, N - 1$. To validate this claim, we need to show from the triggering condition (21) that $\bar{x}_{N-\Delta_j+1}(k_{j+1}) \in \mathbb{X}_a$. In fact, $\|\bar{x}_{N-\Delta_j+1}(k_{j+1})\|_P \le \|\hat{x}_N^*(k_j)\|_P + \bar{\xi}_x(N - \Delta_j + 1) \le \epsilon_f + \sum_{s=0}^{\Delta_j-1} \begin{bmatrix} 1 & 0 \end{bmatrix}$

$$\begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^{N-\Delta_j+s+1} \begin{bmatrix} \bar{v} \\ L_w\bar{v} \end{bmatrix} \le \epsilon_f + \epsilon_a - \epsilon_f = \epsilon_a, \text{ which}$$

implies $\bar{x}_{N-\Delta_j+1}(k_{j+1}) \in \mathbb{X}_a$. Incorporating the construction procedure (20) and Assumption 2, one has $\bar{x}_i(k_{j+1}) \in \mathbb{X}_a \subseteq \mathbb{X}(i)$, $\forall i \ge N - \Delta_j + 1$.

3. $\bar{x}_N(k_{j+1}) \in \mathbb{X}_f$. According to condition (2) in Assumption 2, we directly verify this claim.

4. $\bar{u}_i(k_{j+1}) \in \mathbb{U}$, $i = 0, \ldots, N-1$. This claim can be verified by the same argument of the control constraint satisfaction in Theorem 1.

This completes the proof. ∎

The self-triggered DPMPC scheme is summarized in Algorithm 2.

Algorithm 2 Self-Triggered DPMPC

1: **while** (1) **do**
2:     **if** $k = k_j$ **then**
3:         Measure $x(k_j)$ and generate $\bar{\mathbf{w}}(k_j)$ based on (20);
4:         Solve OCP (5) to obtain $\mathbf{u}^*(k_j)$ and $\hat{\mathbf{x}}^*(k_j)$;
5:         Set $j \leftarrow j+1$ and determine the next triggering instant k_j according to (21);
6:         Apply the control action $u_0^*(k_j)$ to the plant;
7:     **else if** $k = k_{j+1} - 1$ **then**
8:         Measure the system state $x(k)$;
9:         Apply the control action $u_{k-k_j}^*(k_j)$ to the plant;
10:     **else**
11:         Apply the control action $u_{k-k_j}^*(k_j)$ to the plant;
12:     **end if**
13:     Update the time instant $k \leftarrow k + 1$;
14: **end while**

Finally, we present the stability result for the system under the self-triggered DPMPC algorithm. As the proof is similar to Theorem 2 in Sun et al. (2019), it is omitted for simplicity.

Theorem 4. *For system (1) under Algorithm 2, suppose that Assumptions 1 and 2 hold, then the overall system is ISS and the suboptimal performance is guaranteed.*

Remark 7. Note that if the dual-mode strategy is adopted as in Sun et al. (2019), i.e., the self-triggered DPMPC is adopted only when $x \in \mathbb{X} \setminus \mathbb{X}_f$ and the auxiliary controller law κ is adopted when $x \in \mathbb{X}_f$, then it can be verified that the state starts from set $\mathbb{X} \setminus \mathbb{X}_f$ will enter $\mathbb{X}_f$ in finite time and never leave $\mathbb{X}_f$.

5. Simulation example

The attitude regulation problem of a three-DOF helicopter model is discussed in this section to exhibit the effectiveness of the event-triggered and self-triggered DPMPC schemes. The continuous-time system dynamic is given by Yan, Le, and Wang (2016)

$$\dot{x}_1 = x_2 + w_1$$
$$\dot{x}_2 = p_1 \cos x_1 + p_2 \sin x_1 + p_3 x_2 + p_4(u_1 + u_2)\cos x_3 + w_2$$
$$\dot{x}_3 = x_4 + w_3$$
$$\dot{x}_4 = p_5 \cos x_3 + p_6 \sin x_3 + p_7 x_4 + p_8(u_1 - u_2) + w_4$$

where the states x_1, x_2, x_3, x_4 represent the elevation angle, the elevation rate, the pitch angle and the pitch angle rate, respectively, the inputs u_1 and u_2 represent the voltages applied to the front and back motor, respectively. $w = [w_1, w_2, w_3, w_4]^T$ represents the external disturbance. The system constraints are given by $\mathbb{X} = \{x : -6 \le x_1 \le 6, -3 \le x_2 \le 3, -4 \le x_3 \le$

Table 1
Parameters description.

Symbols	Description	Value
M_f	Mass of the front section	0.95
M_b	Mass of the back section	0.95
M_c	Mass of the count-weight	2.21
L_d	Length of pendulum for the elevation axis	0.08
L_c	Distance from the pivot point to count-weight	0.52
L_a	Distance from the pivot point to helicopter body	0.65
L_ε	Length of pendulum for pitch axis	0.2
L_h	Distance from pitch axis to either motor	0.28
g	Gravitational acceleration	9.81
J_ε	Moment of inertia about elevation axis	1.2
J_θ	Moment of inertia about pitch axis	0.08
η_e	Coefficient of viscous friction about elevation axis	0.001
η_θ	Coefficient of viscous friction about pitch axis	0.001

Fig. 1. Disturbance evolution under Algorithm 1.

$4, -3 \le x_4 \le 3\}$ and $\mathbb{U} = \{u : -3 \le u_1 \le 7, -3 \le u_2 \le 7\}$. The coefficients $p_i, i = 1, \ldots, p_8$ and their specific physical meanings are shown below

$$p_1 = [-(M_f + M_b)gL_a + M_cgL_c]/J_\varepsilon$$
$$p_2 = [-(M_f + M_b)gL_a \tan \delta_a + M_cgL_c \tan \delta_c]/J_\varepsilon$$
$$p_3 = -\eta_e/J_\varepsilon, \quad p_4 = K_mL_a/J_\varepsilon, \quad p_5 = (-M_f + M_b)gL_h/J_\theta$$
$$p_6 = -(M_f + M_b)gL_h \tan \delta_h/J_\theta, \quad p_7 = -\eta_\theta/J_\theta,$$
$$p_8 = K_mL_h/J_\theta, \quad \delta_a = \tan^{-1}((L_d + L_\varepsilon)/L_a)$$
$$\delta_c = \tan^{-1}(L_d/L_c), \delta_h = \tan^{-1}(L_\varepsilon/L_h)$$

with the relevant parameters given in Table 1.

The equilibrium of the system is $x^e = [-0.62, 0, 0, 0]^T$, $u^e = [0.89, 0.89]^T$. Then, the discrete-time system can be obtained by defining $\tilde{x} = [x_1, x_2, x_3, x_4]^T - x^e$, $u = [u_1, u_2]^T - u^e$ and by adopting the forward-Euler discretized method with sampling period $T_s = 0.2s$. Assume that w is the attenuated harmonic disturbance, a most common type of disturbance in aerospace systems (Yi et al., 2022), and has the following simple form,

$$w(k + 1) = Aw(k) + g(x(k))w(k) + v(k)$$

where

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0.85 & 0 & -0.52 \\ 0 & 0 & 0 & 0 \\ 0 & 0.38 & 0 & 0.76 \end{bmatrix}, \quad g(x(k)) = \begin{bmatrix} 0 \\ 0.2x_2(k) \\ 0 \\ 0.2x_4(k) \end{bmatrix},$$

$v = [0, v_2, 0, v_4]^T$ is the learning error with $\bar{v} = 0.0007$, and the initial disturbance is set as $w(0) = [0, 0.01, 0, -0.01]^T$. It can be verified by simulation that the disturbance constraint set satisfies $\mathbb{W} = \{w : \|w\|_p \le 0.0312\}$.

Firstly, we set the related parameters of the proposed event-triggered DPMPC (Algorithm 1) as follows. The prediction horizon is $N = 9$. The weighted matrices are designed as $Q = \text{diag}(1, 1, 1, 1)$, $R = 0.05\text{diag}(1, 1)$,

$$P = \begin{bmatrix} 11.2168 & 3.8259 & 0 & 0 \\ 3.8259 & 6.2817 & 0 & 0 \\ 0 & 0 & 9.9572 & 1.6724 \\ 0 & 0 & 1.6724 & 1.9888 \end{bmatrix}$$

The Lipschitz constants are $L_f = 1.4502$, $L_x = 0.0024$ and $L_w = 1.1246$. The parameters of the terminal set $\mathbb{X}_f$ and the auxiliary set $\mathbb{X}_a$ are $\epsilon_f = 1.4627$ and $\epsilon_a = 1.5395$, respectively. The auxiliary controller is designed as

$$\kappa(\tilde{x}) = \begin{bmatrix} -1.6360 & -3.9255 & 0.2514 & -1.4419 \\ -1.6360 & -3.9255 & -0.2514 & 1.4419 \end{bmatrix} \tilde{x}$$

which meets Assumption 2. We also set $\alpha = 0.2$ in OCP (5). To illustrate the effectiveness of Algorithm 1, we compare our

results with the conventional event-triggered MPC in Hashimoto et al. (2017) where the nominal system serves as the prediction model, $\epsilon'_f = 1.4315$, and time-varying horizon $N(k) \equiv 11$. The simulation results are shown in Figs. 1–4, respectively. The disturbance w_2 and w_4 under Algorithm 1 is shown in Fig. 1. One can also observe from Figs. 2 and 3 that the system constraints are satisfied under both MPC schemes. Notice that the abrupt change of the control signal obtained by Algorithm 1 is caused by the switching from the MPC-based controller to the auxiliary controller (The triggering interval is larger than N). It can be seen from Fig. 4 that Algorithm 1 and ETMPC in Hashimoto et al. (2017) are triggered 11 and 25 times, respectively, which presents a significant reduction of the triggering frequency. Such profit mainly comes from the disturbance prediction technique, the update of the triggering condition at each time step and the cancel of the constraint that the triggering interval should be no greater than prediction horizon N.

Secondly, we present the effectiveness of the self-triggered DPMPC scheme (Algorithm 2). Note that the self-triggered scheme is much more conservative than the event-triggered one. In this case, we set the bound of the learning error as $\bar{v} = 0.0002$. The simulation results are also shown in Figs. 2–4. It can be seen that periodic triggering occurs after 4s because condition (23) is always transgressed. For this case, we can simply use the auxiliary controller $\kappa(x)$ instead of the MPC scheme to reduce the consumption of the computational resource.

6. Conclusion

Two event-based DPMPC schemes have been proposed for constrained nonlinear systems with additive dynamic disturbance. It has been demonstrated that the predicted disturbance sequence generated by utilizing the disturbance dynamics enables to improve the state prediction precision, hence enlarging the triggering interval and lowering the consumption of computational resource. A numerical example shows that the triggering frequency has been significantly lowered by the proposed event-triggered and self-triggered DPMPC schemes. Further investigations will focus on its implementation and application to practical systems.
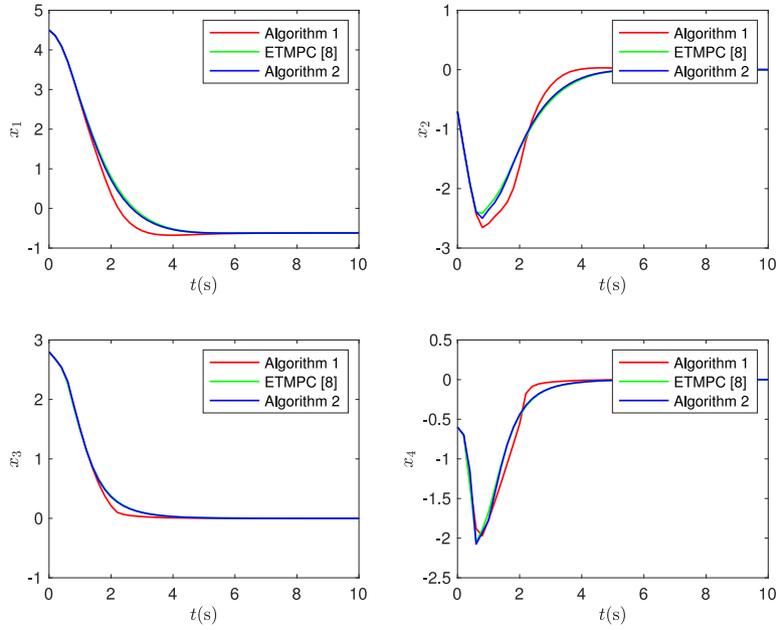
Fig. 2. State evolution under three different event-based MPC algorithms.
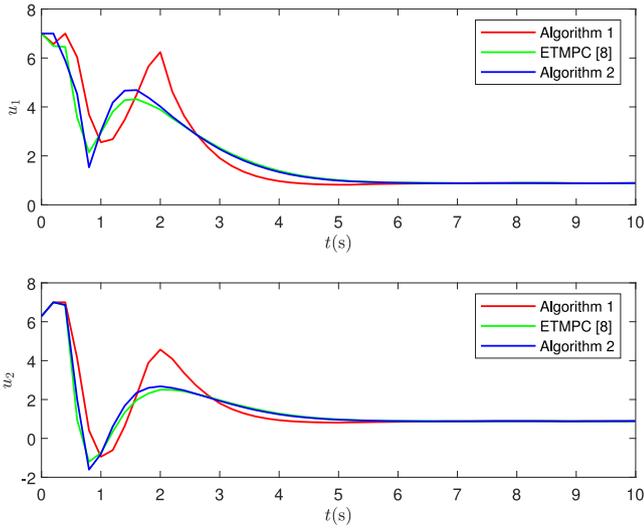
Fig. 3. Control signal obtained by three different event-based MPC algorithms.
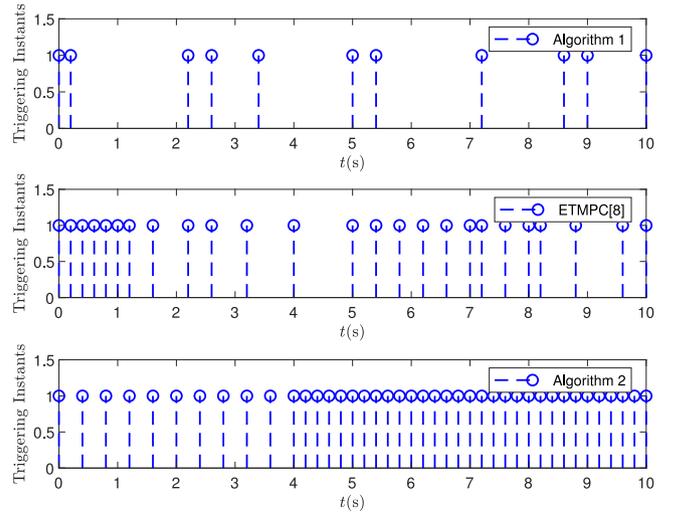
Fig. 4. Triggering instants under three different event-based MPC algorithms.

Appendix A

For a start, we analyze $\bar{e}_x(0)$ and $\bar{e}_w(0)$. According to their definitions, disturbance dynamics (2) and the fact $\|\text{Proj}_{\mathbb{W}}(x) - \text{Proj}_{\mathbb{W}}(y)\| \leq \|x - y\|$ (Bertsekas, 2003, Proposition 2.2.1), we have

$$
\begin{aligned}
\bar{e}_x(0) = & \ \|x(k) - \bar{x}_1(k-1)\|_P \\
\leq & \ \| f(x(k-1), \bar{u}_0(k-1)) + w(k-1) \\
& \ - f(x(k-1), \bar{u}_0(k-1)) + \bar{w}_0(k-1) \|_P \leq \bar{v}
\end{aligned}
\tag{27}
$$

and

$$
\begin{aligned}
\bar{e}_w(0) = & \ \|\bar{w}_0(k) - \bar{w}_1(k-1)\|_P \\
\leq & \ \|g(w(k-1), x(k-1)) - g(\bar{w}_0(k-1), x(k-1))\|_P \\
\leq & \ L_w \|w(k-1) - \bar{w}_0(k-1)\| \leq L_w \bar{v}
\end{aligned}
\tag{28}
$$

In what follows, we derive $\bar{e}_x(i)$ and $\bar{e}_w(i)$ with $i \geq 1$. When $k = k_j + 1$, recalling the recursive process in (6) obtains

$$
\begin{aligned}
\bar{e}_x(i) = & \ \|\bar{x}_i(k) - \hat{x}_{i+1}^*(k_j)\|_P \\
= & \ \| f(\bar{x}_{i-1}(k), \bar{u}_{i-1}(k)) + \bar{w}_{i-1}(k) - f(\hat{x}_i^*(k_j), u_i^*(k_j)) \\
& \ - \bar{w}_i(k_j) \|_P \\
\leq & \ L_f \bar{e}_x(i-1) + \bar{e}_w(i-1)
\end{aligned}
\tag{29}
$$

and

$$
\begin{aligned}
\bar{e}_w(i) = & \ \|\bar{w}_i(k) - \bar{w}_{i+1}(k_j)\|_P \\
\leq & \ \| g(\bar{w}_{i-1}(k), \bar{x}_{i-1}(k)) - g(\bar{w}_i(k-1), \hat{x}_i^*(k_j)) \\
& \ + g(\bar{w}_i(k-1), \hat{x}_i^*(k_j)) - g(\bar{w}_i(k-1), \bar{x}_i(k_j)) \|_P \\
\leq & \ L_x \bar{e}_x(i-1) + L_w \bar{e}_w(i-1) + L_x \alpha
\end{aligned}
\tag{30}
$$

When $k > k_j + 1$, the above two inequalities can be derived by replacing $\hat{x}_i^*(k_j)$ and $u_i^*(k_j)$ with $\bar{x}_i(k-1)$ and $\bar{u}_i(k-1)$, respectively.

Note that for this case, (29) still holds while (30) becomes

$$\bar{e}_w(i) \le L_x \bar{e}_x(i-1) + L_w \bar{e}_w(i-1)$$

Combining the above two cases yields

$$\begin{bmatrix} \bar{e}_x(i) \\ \bar{e}_w(i) \end{bmatrix} \le \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix} \begin{bmatrix} \bar{e}_x(i-1) \\ \bar{e}_w(i-1) \end{bmatrix} + \begin{bmatrix} 0 \\ L_x \alpha \end{bmatrix}. \tag{31}$$

Then, (7) can be easily verified.

Appendix B

To complete the proof, we first need to discuss the prediction error of the state and the disturbance, i.e., $e_x(i) \triangleq \|x(k+i) - \bar{x}_i(k)\|_P$ and $e_w(i) \triangleq \|w(k+i) - \bar{w}_i(k)\|$, $i = 0, \ldots, N-1$. The analysis method follows closely along the lines of Lemma 1. To be specific, we have

$$\begin{aligned} e_x(i) =& \|x(k+i) - \bar{x}_i(k)\|_P \\ =& \| f(x(k+i-1), \bar{u}_{i-1}(k)) + w(k+i-1) \\ & - f(\bar{x}_{i-1}(k), \bar{u}_{i-1}(k)) - \bar{w}_{i-1}(k) \|_P \\ \le& L_f e_x(i-1) + e_w(i-1) \end{aligned} \tag{32}$$

and

$$\begin{aligned} e_w(i) =& \|w(k+i) - \bar{w}_i(k)\|_P \\ \le& \| g(w(k+i-1), x(k+i-1)) + v(k+i-1) \\ & - g(\bar{w}_{i-1}(k), \bar{x}_{i-1}(k)) \|_P \\ \le& L_x e_x(i-1) + L_w e_w(i-1) + \bar{v} \end{aligned} \tag{33}$$

Note that $e_x(0) = 0$ and $e_w(0) = \bar{v}$. Then the above two inequalities can be written in the following vector form

$$\begin{aligned} \begin{bmatrix} e_x(i) \\ e_w(i) \end{bmatrix} \le& \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix} \begin{bmatrix} e_x(i-1) \\ e_w(i-1) \end{bmatrix} + \begin{bmatrix} 0 \\ \bar{v} \end{bmatrix} \\ \le& \sum_{s=0}^{i} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} 0 \\ \bar{v} \end{bmatrix}. \end{aligned} \tag{34}$$

In the remaining part of the proof, we show that $x(k) \in \mathbb{X}, \forall k > k_j$ as long as $\bar{x}_{k-k_j}(k_j) \in \mathbb{X}(k - k_j)$. Let $i = k - k_j$, we have

$$\begin{aligned} \|x(k_j + i)\|_P \le& \|\bar{x}_i(k_j)\|_P + \begin{bmatrix} 1 & 0 \end{bmatrix} \sum_{s=0}^{i} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} 0 \\ \bar{v} \end{bmatrix} \\ =& \|\bar{x}_i(k_j)\|_P + \begin{bmatrix} 1 & 0 \end{bmatrix} \sum_{s=0}^{i-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} \bar{v} \\ L_w \bar{v} \end{bmatrix} \end{aligned}$$

Since $\bar{x}_i(k_j) \in \mathbb{X}(i)$, thus

$$x(k) \in \mathbb{X} \ominus \mathbb{B}(i) \oplus \begin{bmatrix} 1 & 0 \end{bmatrix} \sum_{s=0}^{i-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} \bar{v} \\ L_w \bar{v} \end{bmatrix}$$

$$\in \mathbb{X}.$$

This completes the proof.

Appendix C

According to the definition, one has

$$\begin{aligned} \bar{\xi}_x(i) =& \| f(\bar{x}_{i-1}(k_{j+1}), u^*_{i+\Delta_j-1}(k_j)) + \bar{w}_{i-1}(k_{j+1}) \\ & - f(\hat{x}^*_{i+\Delta_j-1}(k_j), u^*_{i+\Delta_j-1}(k_j)) - \bar{w}_{i+\Delta_j-1}(k_j) \|_P \\ \le& L_f \bar{\xi}_x(i-1) + \bar{\xi}_w(i-1) \end{aligned} \tag{35}$$

and

$$\bar{\xi}_w(i) \le \| g(\bar{w}_{i-1}(k_{j+1}), \bar{x}_{i-1}(k_{j+1}))$$

$$\begin{aligned} & - g(\bar{w}_{i+\Delta_j-1}(k_j), \bar{x}_{i+\Delta_j-1}(k_j)) \|_P \\ \le& L_w \bar{\xi}_w(i-1) + L_x \|\bar{x}_{i-1}(k_{j+1}) - \bar{x}_{i+\Delta_j-1}(k_j)\|_P \\ \le& L_w \bar{\xi}_w(i-1) + L_x \bar{\xi}_x(i-1) + L_x \alpha \end{aligned} \tag{36}$$

Note that

$$\begin{aligned} \bar{\bar{\xi}}_x(0) =& \|x(k_{j+1}) - \hat{x}^*_{\Delta_j}(k_j)\|_P \\ =& \| f(x(k_{j+1} - 1), u^*_{\Delta_j-1}(k_j)) + w(k_{j+1} - 1) \\ & - f(\hat{x}^*_{\Delta_j-1}(k_j), u^*_{\Delta_j-1}(k_j)) + \bar{w}_{\Delta_j-1}(k_j) \|_P \\ \le& L_f \xi_x(\Delta_j - 1) + \xi_w(\Delta_j - 1) \end{aligned} \tag{37}$$

and

$$\begin{aligned} \bar{\bar{\xi}}_w(0) \le& \| g(w(k_{j+1} - 1), x(k_{j+1} - 1)) \\ & - g(\bar{w}_{\Delta_j-1}(k_j), \bar{x}_{\Delta_j-1}(k_j)) \|_P \\ \le& L_w \xi_w(\Delta_j - 1) + L_x \xi_x(\Delta_j - 1) + L_x \alpha \end{aligned} \tag{38}$$

Incorporating the above equations, one can iteratively obtain

$$\begin{aligned} \begin{bmatrix} \bar{\bar{\xi}}_x(i) \\ \bar{\bar{\xi}}_w(i) \end{bmatrix} \le& \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^i \begin{bmatrix} \bar{\bar{\xi}}_x(0) \\ \bar{\bar{\xi}}_w(0) \end{bmatrix} + \sum_{s=0}^{i-1} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} 0 \\ L_x \alpha \end{bmatrix} \\ \le& \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^{i+1} \begin{bmatrix} \xi_x(\Delta_j - 1) \\ \xi_w(\Delta_j - 1) \end{bmatrix} + \sum_{s=0}^{i} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} 0 \\ L_x \alpha \end{bmatrix} \end{aligned} \tag{39}$$

Similar to (35) and (36), we have

$$\begin{aligned} \xi_x(i) =& \| f(x(k_j + i - 1), u^*_{i-1}(k_j)) + w(k_j + i - 1) \\ & - f(\hat{x}^*_{i-1}(k_j), u^*_{i-1}(k_j)) + \bar{w}_{i-1}(k_j) \|_P \\ \le& L_f \xi_x(i-1) + \xi_w(i-1) \end{aligned} \tag{40}$$

and

$$\begin{aligned} \xi_w(i) \le& \| g(w(k_j + i - 1), x(k_j + i - 1)) + v(k_j + i - 1) \\ & - g(\bar{w}_{i-1}(k_j), \bar{x}_{i-1}(k_j)) \|_P \\ \le& L_w \xi_w(i-1) + L_x \xi_x(i-1) + L_x \alpha + \bar{v} \end{aligned} \tag{41}$$

It should be note that

$$\begin{aligned} \xi_w(1) \le& \|g(w(k_j), x(k_j)) + v(k_j) - g(\bar{w}_0(k_j), \bar{x}_0(k_j))\|_P \\ \le& L_w \xi_w(0) + L_x \xi_x(0) + \bar{v} \end{aligned} \tag{42}$$

and $\xi_w(0) = \bar{v}$ and $\xi_x(0) = 0$; Similarly, from (40) to (42), we have

$$\begin{bmatrix} \xi_x(i) \\ \xi_w(i) \end{bmatrix} \le \sum_{s=0}^{i} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} 0 \\ \bar{v} \end{bmatrix} + \sum_{s=0}^{i-2} \begin{bmatrix} L_f & 1 \\ L_x & L_w \end{bmatrix}^s \begin{bmatrix} 0 \\ L_x \alpha \end{bmatrix} \tag{43}$$

Substituting (43) into (39) easily obtains the results of Lemma 3. This completes the proof.

References

Bertsekas, D. P. (2003). *Convex analysis and optimization*. Belmont, MA: Athena Scientific.

Chen, S., Cao, Y., Kang, Y., Sun, B., & Wang, X. (2020). Deep representation-based packetized predictive compensation for networked nonlinear systems. *Neural Computing and Applications*, 1–13.

Cheng, S., Li, L., Guo, H.-Q., Chen, Z.-G., & Song, P. (2019). Longitudinal collision avoidance and lateral stability adaptive control system based on MPC of autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 21(6), 2376–2385.

Cui, D., & Li, H. (2022). Dual self-triggered model-predictive control for nonlinear cyber-physical systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(6), 3442–3452.

El-Ferik, S. (2020). RMPC for uncertain nonlinear systems with non-additive dynamic disturbances and noisy measurements. *IEEE Access*, 8, 44846–44857.

Eqtami, A., Heshmati-Alamdari, S., Dimarogonas, D. V., & Kyriakopoulos, K. J. (2013). Self-triggered model predictive control for nonholonomic systems. In *Proc. Eur. control conf.* (pp. 638–643). Zurich, Switzerland.

Griffith, D. W., Biegler, L. T., & Patwardhan, S. C. (2018). Robustly stable adaptive horizon nonlinear model predictive control. *Journal of Process Control*, 70, 109–122.

Hashimoto, K., Adachi, S., & Dimarogonas, D. V. (2015). Distributed aperiodic model predictive control for multi-agent systems. *IET Control Theory & Applications*, 9(1), 10–20.

Hashimoto, K., Adachi, S., & Dimarogonas, D. V. (2017). Event-triggered inter-mittent sampling for nonlinear model predictive control. *Automatica*, 81, 148–155.

He, N., & Shi, D. (2015). Event-based robust sampled-data model predictive control: A non-monotonic Lyapunov function approach. *IEEE Transactions on Circuits and Systems I*, 62(10), 2555–2564.

Incremona, G. P., Ferrara, A., & Magni, L. (2017). Asynchronous networked MPC with ISM for uncertain nonlinear systems. *IEEE Transactions on Automatic Control*, 62(9), 4305–4317.

Kaheman, K., Brunton, S. L., & Kutz, J. N. (2020). Automatic differentiation to simultaneously identify nonlinear dynamics and extract noise probability distributions from data. arXiv preprint arXiv:2009.08810.

Li, P., Kang, Y., Zhao, Y.-B., & Wang, T. (2021). A novel self-triggered MPC scheme for constrained input-affine nonlinear systems. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 68(1), 306–310.

Lin, X., & Görges, D. (2020). Robust model predictive control of linear systems with predictable disturbance with application to multiobjective adaptive cruise control. *IEEE Transactions on Control Systems Technology*, 28(4), 1460–1475.

Liu, C., Gao, J., Li, H., & Xu, D. (2018). Aperiodic robust model predictive control for constrained continuous-time nonlinear systems: An event-triggered approach. *IEEE Transactions on Cybernetics*, 48(5), 1397–1405.

Magni, L., Raimondo, D. M., & Scattolini, R. (2006). Regional input-to-state stability for nonlinear model predictive control. *IEEE Transactions on Automatic Control*, 51(9), 1548–1553.

Pin, G., & Parisini, T. (2011). Networked predictive control of uncertain constrained nonlinear systems: Recursive feasibility and input-to-state stability analysis. *IEEE Transactions on Automatic Control*, 56(1), 72–87.

Raimondo, D. M., Limon, D., Lazar, M., Magni, L., & ndez Camacho, E. F. (2009). Min-max model predictive control of nonlinear systems: A unifying overview on stability. *European Journal of Control*, 15(1), 5–21.

Rawlings, J. B., Mayne, D. Q., & Diehl, M. (2017). *Model predictive control: Theory, computation, and design*. USA: Nob Hill Publishing.

Rubagotti, M., Raimondo, D. M., Ferrara, A., & Magni, L. (2011). Robust model predictive control with integral sliding mode in continuous-time sampled-data nonlinear systems. *IEEE Transactions on Automatic Control*, 56(3), 556–570.

Shi, G., Shi, X., O'Connell, M., Yu, R., Azizzadenesheli, K., Anandkumar, A., et al. (2019). Neural lander: Stable drone landing control using learned dynamics. In *2019 International conference on robotics and automation* (pp. 9784–9790). Montreal, Canada.

Sun, Z., Dai, L., Liu, K., Dimarogonas, D. V., & Xia, Y. (2019). Robust self-triggered MPC with adaptive prediction horizon for perturbed nonlinear systems. *IEEE Transactions on Automatic Control*, 64(11), 4780–4787.

Sun, Z., Xia, Y., Dai, L., & Campoy, P. (2020). Tracking of unicycle robots using event-based MPC with adaptive prediction horizon. *IEEE/ASME Transactions on Mechatronics*, 25(2), 739–749.

Sun, Z., Xia, Y., Dai, L., Liu, K., & Ma, D. (2017). Disturbance rejection MPC for tracking of wheeled mobile robot. *IEEE/ASME Transactions on Mechatronics*, 22(6), 2576–2587.

Wang, M., Sun, J., & Chen, J. (2019). Input-to-state stability of perturbed nonlinear systems with event-triggered receding horizon control scheme. *IEEE Transactions on Industrial Electronics*, 66(8), 6393–6403.

Yan, Z., Le, X., & Wang, J. (2016). Tube-based robust model predictive control of nonlinear systems via collective neurodynamic optimization. *IEEE Transactions on Industrial Electronics*, 63(7), 4377–4386.

Ye, B.-L., Wu, W., Ruan, K., Li, L., Chen, T., Gao, H., et al. (2019). A survey of model predictive control methods for traffic signal control. *IEEE/CAA Journal of Automatica Sinica*, 6(3), 623–640.

Yi, Y., Zheng, W. X., & Liu, B. (2022). Adaptive anti-disturbance control for systems with saturating input via dynamic neural network disturbance modeling. *IEEE Transactions on Cybernetics*, 52(6), 5290–5300.

Yoo, J., & Johansson, K. H. (2021). Event-triggered model predictive control with a statistical learning. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(4), 2571–2581.

Zou, Y., Su, X., Li, S., Niu, Y., & Li, D. (2019). Event-triggered distributed predictive control for asynchronous coordination of multi-agent systems. *Automatica*, 99, 92–98.

Pengfei Li received the Ph.D. degree in control science and engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2020.

He is an Associate Research Fellow with the School of Information Science and Technology, USTC. His research interests include networked control systems, model predictive control, and learning based control.
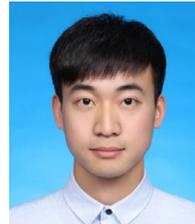
Tao Wang received the B.S. degree from Harbin Engineering University, Harbin, China, in 2018. He is currently pursuing the M.S. degree with the Department of Automation, University of Science and Technology of China, Hefei, China.

His research interests include model predictive control and networked control systems.

Yu Kang received the Ph.D. degree in control theory and control engineering from the University of Science and Technology of China, Hefei, China, in 2005.

From 2005 to 2007, he was a Postdoctoral Fellow with the Academy of Mathematics and Systems Science, Chinese Academy of Sciences. He is currently a Professor with the Department of Automation and the Institute of Advanced Technology, University of Science and Technology of China. His current research interests include monitoring of vehicle emissions, adaptive/robust control, variable structure control, mobile manipulator, and Markovian jump systems.

Kun Li received the Ph.D. degree in control science and engineering from University of Science and Technology of China (USTC), Hefei, China, in 2019.

He is currently a Postdoctoral Fellow with Institute of Advanced Technology, USTC. His research interests include nonlinear control theory and its application, robot control system.

Yun-Bo Zhao received his B.Sc. degree in mathematics from Shandong University, Jinan, China in 2003, M.Sc. degree in systems sciences from the Key Laboratory of Systems and Control, Chinese Academy of Sciences, Beijing, China in 2007, and Ph.D. degree in control engineering from the University of South Wales (formerly University of Glamorgan), Pontypridd, UK in 2008, respectively.

He is currently a Professor with University of Science and Technology of China, Hefei, China. He is mainly interested in AI-driven control and automation, specifically, AI-driven networked intelligent control, AI-driven human–machine autonomies and AI-driven machine gaming.