2024 American Control Conference (ACC)
July 8-12, 2024. Toronto, Canada

Robust Model Predictive Control for Networked Control Systems with Timing Perturbations

Renke Wang Ningshi Yao

Abstract— **Our earlier work established a contention-resolving model predictive control (or MPC) framework to co-design priorities and control for networked control systems (or NCSs), assuming the system models are perfect. However, due to the differences between a model and the real world, there are inevitable perturbations. In this paper, we focus on perturbations of the predicted timings, which are rarely addressed in the literature, and present a robust MPC design to achieve guaranteed performance under such timing perturbations. We propose a state-feedback correction term with dynamic gain, added to the nominal contention-resolving MPC policy, which can eliminate the state deviation between perturbed and nominal systems at the predicted task completion time. Additionally, we identified the largest tolerable timing perturbation for such a robust MPC design. Under the tolerable timing perturbations, we analytically proved that the state deviation can be bounded by a forward invariant set (or FIS) for all time. The robust MPC policy can be then designed based on the FIS, such that perturbed system trajectories are guaranteed to satisfy all the original state and control constraints. The effectiveness of our proposed method is verified through simulation.**

I. INTRODUCTION

Differences between system models and real system performances are unavoidable, which can be viewed as perturbations. Perturbations in NCSs, especailly for timing, can be caused by unexpected network congestion, sampling delays, packet dropouts, etc. The control performance can be dramatically degraded by perturbations. More seriously, the system states may violate its safety constraints. Traditionally, the effect of perturbations is often analyzed using input-to-state stability techniques [1] with strict Lyapunov functions [2]. When the goal is to show invariance for a specific set, barrier Lyapunov functions [3], control barrier functions [4] and other popular Lyapunov-like approaches can be used. However, these traditional analytical techniques can be conservative in the sense that they find bounds on tolerable disturbances that are much smaller than the largest tolerable disturbance set. If the bounds are too conservative, then they may sacrifice optimality or even lose feasibility.

In recent years, the tube-based MPC approach [5]–[7] was developed to compensate for perturbations in system models, and has been applied to linear time-invariant (or LTI) systems [8], [9], linear time-varying (or LTV) systems [10], [11], and nonlinear systems [12], [13]. However, the studies proposed so far merely focus on the bounded additive perturbation in system dynamics. In NCSs, another type of perturbation can affect the system behavior as well but was less explored, which is perturbations on the time when a control law is updated or applied to a control system. Incorrect timing has been shown to even sabotage the stability of a control system [14]–[18] in extreme cases. The current tube-based MPC designs for additive system disturbances have difficulty incorporating disturbances of the timing because the asynchronization between the nominal and perturbed systems makes it difficult to analyze the error dynamics.

A few works have considered the problem with inaccurate timing for control systems, assuming that timing perturbation is either a constant or obeys a certain distribution. The authors in [19]–[23] discussed the cases of the delayed control signal, and [24]–[26] studied the delays in state measurement, where the timing perturbation is treated as a fixed constant value. However, those methods are not applicable for timing perturbations in NCSs because the perturbations may be introduced by random instances such as unpredictable system failures. In [27], the random time delays from the sensor to the controller and from the controller to the actuator were modeled as homogeneous Markov Chain, and a method for stabilization of networked control systems with random delays was presented. The work in [28] also assumes that the sampling periods can be modeled by a Markov Chain. In reality, a timing perturbation can be purely random without knowing its distribution. How to design controllers to compensate for general random timing perturbations is still an open question.

In this paper, we propose a new tube-based MPC with dynamic gain for coupled control systems to compensate for the random timing perturbations existing in an NCS. The main contributions of this paper are summarized as follows:

1. A feedback controller is designed based on the error between nominal system and perturbed system states at the perturbed control updating times. Distinguishing from the traditional tube-based MPC with the constant feedback gain, we design the feedback gain as a function of timing perturbations, so that the error is guaranteed to be zero at each predicted control updating time. When the timing perturbations no longer exist, real system trajectories converge to nominal system trajectories immediately under the proposed controller design, which cannot be achieved by the traditional tube-based MPC.

2. We identify the tolerable range of the timing perturbation to guarantee the stability of the perturbed system. Within such a range, the close-loop error dynamics can be proven to be asymptotically stable, which guarantees that a forward invariant set (or FIS) for the error dynamics exists,

The research work is supported by NSF grant ECCS-2218517.

Renke Wang and Ningshi Yao are with the Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA, 22030, USA. Email: rwang26@gmu.edu, nyao4@gmu.edu

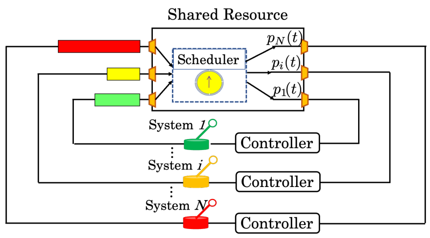

Fig. 1. Networked control system with a shared resource.

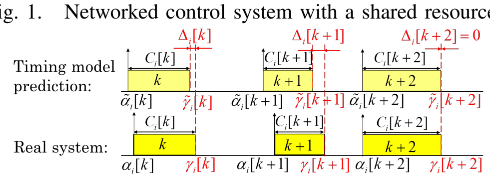

Fig. 2. The timing of the real-time tasks with time perturbations.

i.e., perturbed system trajectories are bounded by a tube around nominal trajectories. After computing the FIS, we can leverage a standard MPC to design the nominal control law under the shrunk state and control constraints (original constraints subtract the FIS). The final controller then is the nominal controller plus the feedback controller for the error dynamics. The perturbed solution is then guaranteed to stay within the tube, satisfying all the original constraints.

II. Problem Formulation

Consider N control systems with recurring requests use a shared resource to complete control tasks, as shown in Fig. 1. Such a shared resource is widely adapted in NCSs to better utilize limited bandwidth, and we assume that only one system can occupy the shared resource at the same time instant. Such an NCS setup can be seen in a shared processor, or communication media like the control area network (or CAN) [29]. The i-th system dynamics is

$$\dot{x}_i(t) = A_i x_i(t) + B_i u_i(t), i=1,\dots,N \tag{1}$$

where the matrix $A_i \in \mathbb{R}^{n\times n}$ is nonsingular and has exclusively real eigenvalues, $B_i \in \mathbb{R}^{n\times m}$ is nonsingular and has linearly independent rows, $x_i(t) \in \mathbb{R}^n$ is the state vector and $u_i(t) \in \mathbb{R}^m$ is the control command. They are subject to the state and control constraints, respectively

$$x_i(t) \in \mathbb{X}_i, \text{ and } u_i(t) \in \mathbb{U}_i, \text{for all } t. \tag{2}$$

A. Priority-based Scheduling for Shared Resource

The i-th system has a sequence of tasks to operate, as shown in Fig. 2, where each task execution time may vary from each other, which is denoted by $C_i[k]$, where $k \in \mathbb{N}$ is the index of the k-th task. The task generation time is denoted as $\alpha_i[k]$. When a task finishes the occupation of the shared resource, the completion time is denoted as $\gamma_i[k]$. Since the measurement of the system state $x_i(t)$ is only accessible to the controller at the task completion time $\gamma_i[k]$, the control signal of the i-th system can only be updated at $\gamma_i[k]$, i.e.,

$$u_i(t) = u_i[k], \text{ for } t \in [\gamma_i[k], \gamma_i[k+1]), \tag{3}$$

meaning that the control u_i is a piece-wise constant. This design follows the idea of the zeroth-order-hold (or ZOH) mechanism that is frequently used in sampling-based control.

Because the shared resource only allows one system to occupy it at any time, when multiple systems request to use the shared resource simultaneously, a contention occurs. In this case, priorities are assigned to the contended systems to determine which system can access the shared resource first, where the priority order can be predetermined or obtained by solving an optimization problem [17]. Only the system with the highest priority can occupy the shared resource immediately at $\alpha_i[k]$. All the other systems will have time delays. Denote $\delta_i[k]$ as the time delay from the task generation time $\alpha_i[k]$ to the actual time when the task begins to occupy the resource, namely, $\gamma_i[k]$ satisfies $\gamma_i[k] = \alpha_i[k] + \delta_i[k] + C_i[k]$. Both $\delta_i[k]$ and $\gamma_i[k]$ are actually implicit functions of specific priority assignments, timing parameters $\alpha_i[k]$ and $C_i[k]$. Our previous work [17] presented an *analytical timing model*, which can predict $\delta_i[k]$ and $\gamma_i[k]$.

Remark 1: Note that for timing model to accurately compute $\delta_i[k]$ and $\gamma_i[k]$, the parameters $\alpha_i[k]$ and $C_i[k]$ need to be known for all k.

B. Perturbed Timing

In real applications, the execution time $C_i[k]$ can be obtained easily because they are normally pre-defined by the design of tasks. The timing parameter $\alpha_i[k]$ is generated at run-time and may not be known precisely, due to reasons such as the clocks of each system in the NCS are not synchronized. Hence, we assume that $C_i[k]$ are known for all k, while $\alpha_i[k]$ is unknown. The best we can do is to estimate $\alpha_i[k]$. The work in [30] proposed a method to estimate $\alpha_i[k]$, which was denoted as $\tilde{\alpha}_i[k]$ on CAN bus, and proved that the difference between $\alpha_i[k] - \tilde{\alpha}_i[k]$ is non-negative and is non-increasing as k increases. Therefore, to predict any future time delay, we can only use $\tilde{\alpha}_i[k]$ in the timing model, which can obtain the estimated $\tilde{\gamma}_i[k]$, as shown in Fig. 2. In this paper, we specifically look into the difference between the actual completion time and the estimated completion time, denoted as $\Delta_i[k] = \gamma_i[k] - \tilde{\gamma}_i[k]$, because this is the direct perturbation affecting the control applied to each system.

Assumption 1: $0 \le \Delta_i[k] \le \overline{\Delta}_i$ is a random variable with a finite constant upper bound $\overline{\Delta}_i$ for all k.

The timing perturbations would cause the system states to deviate from the predicted system behavior. To distinguish the difference, we define the predicted system behavior as the nominal system under the estimated completion time instants, which follows the dynamic equations

$$\begin{aligned} \dot{x}_i^n(t) &= A_i x_i^n(t) + B_i u_i^n(t), \\ u_i^n(t) &= u_i^n[k], \text{ for } t \in [\tilde{\gamma}_i[k], \tilde{\gamma}_i[k+1]), \end{aligned} \tag{4}$$

where $x_i^n(t) \in \mathbb{R}^n$ is the nominal system state, and $u_i^n(t) \in \mathbb{R}^m$ is the nominal system control signal.

C. Formulation of Model Predictive Control

The MPC is triggered whenever a task is completed and the controller obtains a new state measurement $x_i(t_0)$ at time t_0. We establish MPC problems during the time interval $[t_0, t_0 + T_p]$, where the current time t_0 is the initial time of the prediction horizon of MPC optimization and T_p is the

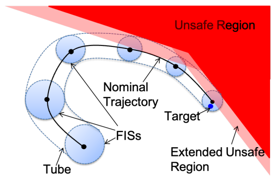

Fig. 3. Tube-based MPC illustration. MPC steers tube towards targeted equilibrium point, while satisfying state constraints.

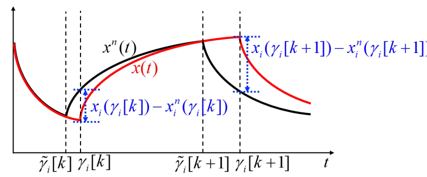

Fig. 4. Illustration of state-feedback controller for the perturbed system.

prediction length. We set the initial run-time timing parameters for this prediction horizon as $\alpha_i[0]=\gamma_i[0]=t_0$. Since the system state measurement is only accessible to the controller at the future task completion time $\gamma_i[k] > t_0$, a control objective function can only be evaluated at $\gamma_i[k]$. Define $x_i[k]=x_i(\gamma_i[k])$, $u_i[k]=u_i(\gamma_i[k])$ for $k=0,1,\ldots,K$, where K is the largest integer such that $\gamma_i[K]\le t_0+T_p$, and $\mathbf{u}[k] = [u_1[k],\ldots,u_N[k]]$. Given the initial state value $x_i(t_0)$, the initial control law $u_i[0] = u_i(t_0)$, and a specific priority order, the optimization problem for MPC is represented as

$$\min_{\mathbf{u}[k]} \sum_{i=1}^{N} \left[\sum_{k=0}^{K-1} \left(\|x_i[k]\|^2_{Q_i} + \|u_i[k]\|^2_{R_i} \right) + \|x_i[K]\|^2_{P_i} \right], \quad \text{s.t. } (1),\ (2),\ (3), \tag{5}$$

where Q_i, R_i, P_i are positive definite matrices, and $\|x_i[k]\|^2_{Q_i}$ denotes the quadratic form of vector $x_i[k]$, i.e., $x_i^T[k]Q_i x_i[k]$. The proper $\mathbf{u}[k]$ needs to be found as the decision variable to minimize the cost function.

However, since the actual task completion time $\gamma_i[k]$ in (3) entails the random variable and is not deterministic in the future prediction horizon, the system state $x_i[k]$ in the future is also undetermined. Thus, directly solving this optimization problem in (5) of the perturbed system is impossible.

In this case, a tube-based MPC approach is leveraged to compensate for perturbations in system timing, as illustrated by Fig. 3. The tube-based robust MPC design is decomposed into two stages. First, the system dynamics is decomposed into nominal system dynamics and the error dynamics at $\gamma_i[k]$. A feedback controller is designed for the error dynamics only and an FIS (or its upper bound), denoted as $\mathbb{S}_i$, for the error dynamics under the designed feedback controller can be computed. The blue balls in Fig. 3 represent the FIS for the error dynamics. Second, the nominal dynamics plus the FIS are considered as a tube. The state and input constraint sets for the nominal MPC design are extended to accommodate the tube. Define $x_i^n[k] = x_i^n(\tilde{\gamma}_i[k])$ and $u_i^n[k] = u_i^n(\tilde{\gamma}_i[k])$ for $k = 0,1,2,...,\overline{K}$, where $\overline{K}$ is the largest integer such that $\tilde{\gamma}_i[\overline{K}] \le t_0+T_p$. Given the initial nominal state value $x_i^n(t_0) = x_i(t_0)$, the initial nominal control law $u_i^n[0]=u_i[0]=u_i(t_0)$, the initial estimated timing parameters $\tilde{\gamma}_i[0]= \gamma_i[0]=t_0$ and the same priority order, the optimal nominal control $\mathbf{u}^{n*}[k]$ can be obtained by solving the following optimization problem

$$\min_{\mathbf{u}^n[k]} \sum_{i=1}^{N} \left[\sum_{k=0}^{\overline{K}-1} \left(\|x_i^n[k]\|^2_{Q_i} + \|u_i^n[k]\|^2_{R_i} \right) + \|x_i^n[\overline{K}]\|^2_{P_i} \right], \quad \text{s.t. } (4),\ x_i^n(t) \in \mathbb{X}_i \ominus \mathbb{S}_i, \tag{6}$$

where the set subtraction is defined as $A \ominus B \coloneqq \{x\in\mathbb{R}^n | \{x\}\oplus B \subseteq A\}$, in which the set addition is $A\oplus B \coloneqq \{a+b | a\in A, b\in B\}$ for $A,B\in\mathbb{R}^n$, and $\mathbf{u}^n[k] = [u_1^n[k],\ldots,u_N^n[k]]$. The standard MPC solver can compute the nominal control once $\mathbb{S}_i$ is given. The final controller for the perturbed system is the nominal controller plus the feedback controller for the error dynamics.

III. Tube-Based MPC with Dynamic Gain

We first explain why a traditional tube-based MPC discussed in work such as [6], [31] cannot work for random timing perturbations and then present our robust MPC design.

Since the control command is a piece-wise constant function , we can discretize the continuous-time system into a discrete-time system $x_i[k+1] = A_i[k]x_i[k] + B_i[k]u_i[k]$, where $x_i[k+1] = x_i(\gamma_i[k+1])$, $x_i[k] = x_i(\gamma_i[k])$, $A_i[k] = e^{A_i(\gamma_i[k+1]-\gamma_i[k])} = e^{A_i(\tilde{\gamma}_i[k+1]+\Delta_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} = e^{A_i(\Delta_i[k+1]-\Delta_i[k])}e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}$ (depending on $\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]$), and $B_i[k] = (A_i[k]-I)A_i^{-1}B_i$ with I as an $n\times n$ identity matrix. The traditional tube-based MPC within time interval $t\in[\gamma_i[k],\gamma_i[k+1])$ is

$$u_i[k] = u_i^n[k] - K_c\left[x_i(\gamma_i[k]) - x_i^n(\gamma_i[k])\right]. \tag{7}$$

Fig. 4 presents an illustration of the tube-based MPC design with the feedback correction term utilizing the error between the actual states and nominal states at the actual task completion time. Once a task is completed, we can directly measure the current system state value $x_i(\gamma_i[k])$. The nominal system state $x_i^n(\gamma_i[k])$ can be easily obtained through simulations.

Here the constant feedback gain K_c should be designed such that $A_i[k]-B_i[k]K_c$ is Hurwitz for all $0\le k\le K$. However, since both $A_i[k]$ and $B_i[k]$ depend on random timing perturbations, $A_i[k]$ and $B_i[k]$ are time-varying and undetermined. In this case, it is very hard to find a K_c to guarantee $A_i[k]-B_i[k]K_c$ stable for all k. In [11], a tube-based MPC has been presented for LTV systems. But the method in [11] requires all $(A_i[k],B_i[k]) \in Conv\{(A_j,B_j), \forall j = 1,2,\ldots,L\}$ where (A_j,B_j) are vertices of a convex hull and L is the number of vertices of the convex hull, i.e., any $(A_i[k],B_i[k])$ is the convex combination of (A_j,B_j), which does not apply here. The best we can do to design a constant K_c is to leverage the estimated task completion time $\tilde{\gamma}_i[k]$ to approximate $A_i[k]$ and $B_i[k]$ as $\tilde{A}_i[k]=e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}$ and $\tilde{B}_i[k] = (\tilde{A}_i[k]-I)A_i^{-1}B_i$, and design K_c such that $\tilde{A}_i[k]-\tilde{B}_i[k]K_c$ is Hurwitz. But the actual $A_i[k]-B_i[k]K_c$ can be unstable for some k due to the timing perturbations. To resolve this, we present a dynamic feedback gain $K(\Delta_i[k])$ design that can ensure $A_i[k]-B_i[k]K(\Delta_i[k])$ Hurwitz for all k, which will be introduced in the next subsection.

A. Dynamic Feedback Gain Design

Our dynamic state-feedback controller design for the perturbed system is

$$u_i[k] = u_i^n[k] - K(\Delta_i[k])\left[x_i(\gamma_i[k]) - x_i^n(\gamma_i[k])\right], \quad (8)$$

where the feedback controller gain $K(\Delta_i[k])$ is equal to

$$K(\Delta_i[k]) = B_i^\dagger A_i\left[e^{A_i(\tilde{\gamma}_i[k+1]-\gamma_i[k])} - I\right]^{-1} e^{A_i(\tilde{\gamma}_i[k+1]-\gamma_i[k])}$$
$$= B_i^\dagger A_i\left[e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} - I\right]^{-1} e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} \quad (9)$$

where $B_i^\dagger \in \mathbb{R}^{m\times n}$ is pseudo-inverse of B_i with $B_i B_i^\dagger = I$.

Remark 2: Notice that the proposed dynamic gain $K(\Delta_i[k])$ is computable at run-time. Once a task is completed, we can directly measure the current time, i.e. $\gamma_i[k]$. The estimated task completion time instants $\tilde{\gamma}_i[k]$ and $\tilde{\gamma}_i[k+1]$ can be predicted by the timing model presented in [17].

Here, we present the following *Lemma* which shows that under the proposed feedback control law, the perturbed system trajectory will oscillate around the nominal system trajectory, and the perturbed system state equals the nominal system state at each estimated task completion time.

Lemma 1: With the proposed state-feedback controller design in (8) and (9), the perturbed and nominal system states are identical to each other at each estimated task completion time, i.e., $x_i(\tilde{\gamma}_i[k]) = x_i^n(\tilde{\gamma}_i[k])$, for all $k = 0, 1, \ldots, \overline{K}$.

Proof. We will prove this *Lemma* by mathematical induction. Initially, since $x_i(t_0) = x_i^n(t_0) = x_{i,0}$ and $\tilde{\gamma}_i[0] = t_0$, we have $x_i(\tilde{\gamma}_i[0]) = x_i^n(\tilde{\gamma}_i[0])$.

Assume $x_i(\tilde{\gamma}_i[k]) = x_i^n(\tilde{\gamma}_i[k])$ for an arbitrary k. We can prove $x_i(\tilde{\gamma}_i[k+1]) = x_i^n(\tilde{\gamma}_i[k+1])$. Due to the nominal system dynamics in (4), we can get the analytical form of $x_i^n(\tilde{\gamma}_i[k+1])$, which is $x_i^n(\tilde{\gamma}_i[k+1]) = e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}x_i^n(\tilde{\gamma}_i[k]) + \int_{\tilde{\gamma}_i[k]}^{\tilde{\gamma}_i[k+1]} e^{A_i(\tilde{\gamma}_i[k+1]-\tau)}B_i u_i^n(\tau)d\tau$. Since matrix B_i is constant, and $u_i^n(\tau) = u_i^n[k]$ for any $\tau \in [\tilde{\gamma}_i[k], \tilde{\gamma}_i[k+1])$ in the nominal system, the integral item in the above equation can be simplified as $e^{A_i\tilde{\gamma}_i[k+1]}\int_{\tilde{\gamma}_i[k]}^{\tilde{\gamma}_i[k+1]} e^{-A_i\tau}d\tau B_i u_i^n[k]$. Take the integration of the exponential function, we have $\int_{\tilde{\gamma}_i[k]}^{\tilde{\gamma}_i[k+1]} e^{-A_i\tau}d\tau = -\left(e^{-A_i\tilde{\gamma}_i[k+1]} - e^{-A_i\tilde{\gamma}_i[k]}\right)A_i^{-1}$. Thus,

$$x_i^n(\tilde{\gamma}_i[k+1]) = e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}x_i^n(\tilde{\gamma}_i[k]) + [e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])} - I]A_i^{-1}B_i u_i^n[k]. \quad (10)$$

Similarly, the perturbed system state $x_i(\tilde{\gamma}_i[k+1])$ at the estimated completion time $\tilde{\gamma}_i[k+1]$ can be represented as $x_i(\tilde{\gamma}_i[k+1]) = e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}x_i(\tilde{\gamma}_i[k]) + \int_{\tilde{\gamma}_i[k]}^{\tilde{\gamma}_i[k+1]} e^{A_i(\tilde{\gamma}_i[k+1]-\tau)}B_i u_i(\tau)d\tau$. Since the perturbed system's control commands are $u_i(\tau) = u_i[k-1]$ for $\tau \in [\tilde{\gamma}_i[k], \gamma_i[k])$ and $u_i(\tau) = u_i[k]$ for $\tau \in [\gamma_i[k], \tilde{\gamma}_i[k+1])$, the above equation can be rewritten as $x_i(\tilde{\gamma}_i[k+1]) = e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}x_i(\tilde{\gamma}_i[k]) + \int_{\tilde{\gamma}_i[k]}^{\gamma_i[k]} e^{A_i(\gamma_i[k]-\tau)}B_i u_i[k-1]d\tau + \int_{\gamma_i[k]}^{\tilde{\gamma}_i[k+1]} e^{A_i(\tilde{\gamma}_i[k+1]-\tau)}B_i u_i[k]d\tau$, which leads to

$$x_i(\tilde{\gamma}_i[k+1]) = e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}x_i(\tilde{\gamma}_i[k])$$
$$+ \left[e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])} - e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}\right]A_i^{-1}B_i u_i[k-1]$$
$$+ \left[e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} - I\right]A_i^{-1}B_i u_i[k]. \quad (11)$$

Subtracting (11) by (10), the difference between the perturbed and the nominal systems can be computed

$$x_i(\tilde{\gamma}_i[k+1]) - x_i^n(\tilde{\gamma}_i[k+1])$$
$$= e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}\left[x_i(\tilde{\gamma}_i[k]) - x_i^n(\tilde{\gamma}_i[k])\right]$$
$$+ \left[e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])} - e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}\right]A_i^{-1}B_i u_i[k-1]$$
$$+ \left[e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} - I\right]A_i^{-1}B_i u_i[k]$$
$$- \left[e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])} - I\right]A_i^{-1}B_i u_i^n[k]. \quad (12)$$

Since we assume $x_i^n(\tilde{\gamma}_i[k]) = x_i(\tilde{\gamma}_i[k])$, then first term $e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}\left[x_i(\tilde{\gamma}_i[k]) - x_i^n(\tilde{\gamma}_i[k])\right]$ in (12) equals 0. To further simplify (12), we will represent $u_i[k]$ as a function of $u_i[k-1]$ and $u_i^n[k]$, based on (8) and (9). The feedback term $x_i(\gamma_i[k]) - x_i^n(\gamma_i[k])$, based on the analytical form for LTI, can be derived as $x_i(\gamma_i[k]) = e^{A_i(\gamma_i[k]-\tilde{\gamma}_i[k])}x_i(\tilde{\gamma}_i[k]) + \left[e^{A_i(\gamma_i[k]-\tilde{\gamma}_i[k])} - I\right]A_i^{-1}B_i u_i[k-1]$ because the perturbed system control for $t \in [\tilde{\gamma}_i[k], \gamma_i[k])$ is $u_i[k-1]$, and $x_i^n(\gamma_i[k]) = e^{A_i(\gamma_i[k]-\tilde{\gamma}_i[k])}x_i^n(\tilde{\gamma}_i[k]) + \left[e^{A_i(\gamma_i[k]-\tilde{\gamma}_i[k])} - I\right]A_i^{-1}B_i u_i^n[k]$ because the nominal system control for $t \in [\tilde{\gamma}_i[k], \gamma_i[k])$ is $u_i^n[k]$. We can derive

$$x_i(\gamma_i[k]) - x_i^n(\gamma_i[k]) = e^{A_i\Delta_i[k]}x_i(\tilde{\gamma}_i[k]) - e^{A_i\Delta_i[k]}x_i^n(\tilde{\gamma}_i[k])$$
$$+ (e^{A_i\Delta_i[k]} - I)A_i^{-1}B_i u_i[k-1] - (e^{A_i\Delta_i[k]} - I)A_i^{-1}B_i u_i^n[k]$$
$$= (e^{A_i\Delta_i[k]} - I)A_i^{-1}B_i(u_i[k-1] - u_i^n[k]), \quad (13)$$

because $x_i(\tilde{\gamma}_i[k]) = x_i^n(\tilde{\gamma}_i[k])$ as assumed.

For simplicity of the rest of this proof, we define $M_1 = e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])}$, and $M_2 = e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}$. The term $M_2 = e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}$ can be rewritten as $e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}\left(e^{A\Delta_i[k]} - I\right)\left(e^{A_i\Delta_i[k]} - I\right)^{-1} = \left[e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k])} - e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}\right]\left(e^{A\Delta_i[k]} - I\right)^{-1} = (M_1 - M_2)\left(e^{A\Delta_i[k]} - I\right)^{-1}$. Bringing M_1, M_2 and $x_i(\gamma_i[k]) - x_i^n(\gamma_i[k])$ in (8) leads to control law for perturbed system

$$u_i[k] = u_i^n[k] - B_i^\dagger A_i(M_2 - I)^{-1}M_2\left[x_i(\gamma_i[k]) - x_i^n(\gamma_i[k])\right]$$
$$= u_i^n[k] - B_i^\dagger A_i(M_2 - I)^{-1}(M_1 - M_2)\left(e^{A_i\Delta_i[k]} - I\right)^{-1}$$
$$\cdot\left(e^{A_i\Delta_i[k]} - I\right)A_i^{-1}B_i\left(u_i[k-1] - u_i^n[k]\right) \quad (14)$$
$$= u_i^n[k] + B_i^\dagger A_i(M_2 - I)^{-1}(M_2 - M_1)A_i^{-1}B_i(u_i[k-1] - u_i^n[k]).$$

Using M_1 and M_2 in (12), we have

$$x_i(\tilde{\gamma}_i[k+1]) - x_i^n(\tilde{\gamma}_i[k+1]) = (M_1 - M_2)A_i^{-1}B_i u_i[k-1]$$
$$-(M_1 - I)A_i^{-1}B_i u_i^n[k] + (M_2 - I)A_i^{-1}B_i u_i[k], \quad (15)$$

where the last term $(M_2 - I)A_i^{-1}B_i u_i[k] = (M_2 - I)A_i^{-1}B_i$ $[u_i^n[k] + B_i^\dagger A_i(M_2 - I)^{-1}(M_2 - M_1)A_i^{-1}B_i(u_i[k-1] - u_i^n[k])]$ with (15). Multiplying $(M_2 - I)A_i^{-1}B_i$ inside the bracket, it equals $(M_2 - I)A_i^{-1}B_i u_i^n[k] + (M_2 - I)A_i^{-1}B_i B_i^\dagger A_i(M_2 - I)^{-1}(M_2 - M_1)A_i^{-1}B_i(u_i[k-1] - u_i^n[k])$. Since $B_i B_i^\dagger = I$, then $(M_2 - I)\cdot A_i^{-1}\cdot B_i B_i^\dagger A_i(M_2 - I)^{-1} = I$. Then we can derive $(M_2 - I)A_i^{-1}B_i u_i[k] = (M_2 - I)A_i^{-1}B_i u_i^n[k] + (M_2 - M_1)A_i^{-1}B_i(u_i[k-1] - u_i^n[k])$. Thus, the equation (12) is equivalent to $x_i(\tilde{\gamma}_i[k+1]) - x_i^n(\tilde{\gamma}_i[k+1]) = (M_1 - M_2)A_i^{-1}B_i u_i[k-1] - (M_1 - I)A_i^{-1}B_i u_i^n[k] + (M_2 - I)A_i^{-1}B_i u_i^n[k] + (M_2 - M_1)A_i^{-1}B_i(u_i[k-1] - u_i^n[k])$. After grouping all the terms corresponding to $u_i[k-1]$ and $u_i^n[k]$ together, respectively, we obtain $x_i(\tilde{\gamma}_i[k+1]) - x_i^n(\tilde{\gamma}_i[k+1]) = [(M_1 - M_2) + (M_2 - M_1)]u_i[k-1] + [-(M_1 - I) + (M_2 - I) - (M_2 - M_1)]u_i^n[k] = 0$, which completes the proof. □

B. Error Dynamics

In the proof of *Lemma* 1, we can see that $x_i(\gamma_i[k]) \neq x_i^n(\gamma_i[k])$ if $u_i[k-1] \neq u_i^n[k]$, according to (13). The perturbations on timing cause the perturbed solution to deviate from the nominal solution. It does not exist a steady state for the deviation. What can hope for is to impose tight bounds on the perturbed solution, so that it can stay in a tube with changing volume over time. Hence, we analyze the error dynamics between the perturbed and nominal systems at the actual task completion time $\gamma_i[k]$. Define the error $e_i[k] = x_i[k] - x_i^n[k] = x_i(\gamma_i[k]) - x_i^n(\gamma_i[k])$.

Lemma 2: The error dynamics can be represented as

$$e_i[k+1] = A_i^{cl}[k]e_i[k] + w_i[k], \tag{16}$$

where $A_i^{cl}[k] = -[e^{A_i\Delta_i[k+1]} - I][e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} - I]^{-1}e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}$ and $w_i[k] = (e^{A_i\Delta_i[k+1]} - I)A_i^{-1}B_i(u_i^n[k] - u_i^n[k+1])$.

Proof. To compute $e_i[k+1] = x_i[k+1] - x_i^n[k+1]$, we first need to obtain $x_i[k+1]$ and $x_i^n[k+1]$. Since $u_i(t)$ is a constant $u_i[k]$ within $[\gamma_i[k], \gamma_i[k+1])$, it is easy to compute $x_i[k+1] = e^{A_i(\gamma_i[k+1]-\gamma_i[k])}x_i[k] + [e^{A_i(\gamma_i[k+1]-\gamma_i[k])} - I]A_i^{-1}B_iu_i[k]$. For the nominal system, since $u_i^n(t) = u_i^n[k]$ for $t \in [\gamma_i[k], \tilde{\gamma}_i[k+1])$ and $u_i^n(t) = u_i^n[k+1]$ for $t \in [\tilde{\gamma}_i[k+1], \gamma_i[k+1])$, we can compute that $x_i^n[k+1] = e^{A_i(\gamma_i[k+1]-\gamma_i[k])}x_i^n[k] + [e^{A_i(\gamma_i[k+1]-\gamma_i[k])} - e^{A_i\Delta_i[k+1]}]A_i^{-1}B_iu_i^n[k] + (e^{A_i\Delta_i[k+1]} - I)A_i^{-1}B_iu_i^n[k+1]$. Hence, the error

$$\begin{aligned} e_i[k+1] &= e^{A_i(\gamma_i[k+1]-\gamma_i[k])}(x_i[k] - x_i^n[k]) \\ &+ [e^{A_i(\gamma_i[k+1]-\gamma_i[k])} - I]A_i^{-1}B_iu_i[k] - [e^{A_i(\gamma_i[k+1]-\gamma_i[k])} \\ &- e^{A_i\Delta_i[k+1]}]A_i^{-1}B_iu_i^n[k] - [e^{A_i\Delta_i[k+1]} - I]A_i^{-1}B_iu_i^n[k+1]. \end{aligned} \tag{17}$$

For simplicity, let $M_3 = e^{A_i(\gamma_i[k+1]-\gamma_i[k])}$. The control law (8) is $u_i[k] = u_i^n[k] - K(\Delta_i[k])(x_i[k] - x_i^n[k]) = u_i^n[k] - K(\Delta_i[k])e_i[k]$, bring it into (17) to replace $u_i[k]$, we get $e_i[k+1] = M_3e_i[k] + (M_3 - I)A_i^{-1}B_i\,[u_i^n[k] - K(\Delta_i[k])e_i[k]] - (M_3 - e^{A_i\Delta_i[k+1]})A_i^{-1}B_iu_i^n[k] - \left(e^{A_i\Delta_i[k+1]} - I\right)A_i^{-1}B_iu_i^n[k+1]$. Group all terms corresponding to $e_i[k]$, $u_i^n[k]$ and $u_i^n[k+1]$, it has $e_i[k+1] = \left[M_3 - (M_3 - I)A_i^{-1}B_iK(\Delta_i[k])\right]e_i[k] + \left[(M_3 - I) - (M_3 - e^{A_i\Delta_i[k+1]})\right]A_i^{-1}B_iu_i^n[k] - (e^{A_i\Delta_i[k+1]} - I)A_i^{-1}B_iu_i^n[k+1]$, where the system matrix $A_i^{cl}[k] = M_3 - (M_3 - I)A_i^{-1}B_iK(\Delta_i[k])$, and the additive disturbance is

$$w_i[k] = [e^{A_i\Delta_i[k+1]} - I]A_i^{-1}B_i\left(u_i^n[k] - u_i^n[k+1]\right). \tag{18}$$

Plug controller gain $K(\Delta_i[k])$ from (9) into $A_i^{cl}[k]$, it has

$$\begin{aligned} A_i^{cl}[k] &= M_3 - (M_3 - I)A_i^{-1}B_iB_i^{\dagger}A_i(M_2 - I)^{-1}M_2 \\ &= M_3 - (M_3 - I)(M_2 - I)^{-1}M_2. \end{aligned} \tag{19}$$

Since $M_3 = e^{A_i(\tilde{\gamma}_i[k+1]+\Delta_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} = e^{A_i\Delta_i[k+1]}e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} = e^{A_i\Delta_i[k+1]}M_2$, we can rewrite the term $M_3 - I$ as $M_3 - e^{A_i\Delta_i[k+1]} + e^{A_i\Delta_i[k+1]} - I = \left(e^{A_i\Delta_i[k+1]}M_2 - e^{A_i\Delta_i[k+1]}\right) + e^{A_i\Delta_i[k+1]} - I = e^{A_i\Delta_i[k+1]}(M_2 - I) + (e^{A_i\Delta_i[k+1]} - I)$, and take it back into (19), we obtain $A_i^{cl}[k] = M_3 - [e^{A_i\Delta_i[k+1]}(M_2 - I) + (e^{A_i\Delta_i[k+1]} - I)](M_2 - I)^{-1}M_2$. Multiply $(M_2 - I)^{-1}M_2$ into the bracket, then we have $A_i^{cl}[k] = M_3 - e^{A_i\Delta_i[k+1]}(M_2 - I)(M_2 - I)^{-1}M_2 - (e^{A_i\Delta_i[k+1]} - I)(M_2 - I)^{-1}M_2 = M_3 - e^{A_i\Delta_i[k+1]}M_2 - (e^{A_i\Delta_i[k+1]} - I)(M_2 - I)^{-1}M_2$, which equals $-(e^{A_i\Delta_i[k+1]} - I)(M_2 - I)^{-1}M_2$ since $M_3 = e^{A_i\Delta_i[k+1]}M_2$. Thus, the closed-loop system matrix for error dynamics is

$$\begin{aligned} A_i^{cl}[k] = -[e^{A_i\Delta_i[k+1]} - I][e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} - I]^{-1} \\ \cdot e^{A_i(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}, \end{aligned} \tag{20}$$

which completes this proof. □

Since the state deviation satisfies (16), it means $e_i[k] = (\prod_{j=0}^{k-1} A_i^{cl}[j])e_i[0] + \sum_{j=0}^{k-1}\left[(\prod_{q=j+1}^{k-1} A_i^{cl}[q])w_i[j]\right]$, where $e_i[0] = x_i[0] - x_i^n[0] = 0$. Then set $\mathbb{S}_i[k]$ combined by $e_i[k]$ can be computed by $\mathbb{S}_i[k] = \sum_{j=0}^{k-1}\left[\left(\prod_{q=j+1}^{k-1} A_i^{cl}[q]\right)\mathbb{W}_i[j]\right] = \mathbb{W}_i[k-1] \oplus A_i^{cl}[k-1]\mathbb{W}_i[k-2] \oplus \ldots \oplus \left(\prod_{q=1}^{k-1} A_i^{cl}[q]\right)\mathbb{W}_i[0]$. When k goes to infinity, we get $\mathbb{S}_{i,\infty} = \lim_{k\to\infty}\mathbb{S}_i[k]$, which is the outer bound of $\mathbb{S}_i[k]$.

C. Forward Invariant Set for the Error Dynamics

This sub-section shows that there exists an FIS for the error dynamics (16) if the upper bound of the timing perturbation satisfies the following condition.

Define Δt_m as the minimal time interval of two consecutive estimated task completion time, i.e., $\Delta t_m = \min_k\{\tilde{\gamma}_i[k+1] - \tilde{\gamma}_i[k]\}$ for $k = 0, 1, \ldots, \overline{K}$, and λ_j to be an eigenvalue of the system matrix A_i, with $j = 1, \ldots, n$. We have a further assumption on the bound of the timing perturbation.

Condition 1: If matrix A_i only has negative eigenvalues, then the timing perturbation $\Delta_i[k]$ satisfies $\Delta_i[k] < \frac{1}{2}\Delta t_m, \forall k$. If matrix A_i has any positive eigenvalues, then the timing perturbation $\Delta_i[k]$ satisfies $\Delta_i[k] < \Delta t_m - \frac{1}{\lambda_M}\ln\left\{(e^{\lambda_M\Delta t_m} + 1)/2\right\}, \forall k$, where λ_M is the largest eigenvalue of matrix A_i.

Proposition 1: The matrix $A_i^{cl}[k]$ is stable if the timing perturbation $\Delta_i[k]$ satisfies *Assumption* 1 and *Condition* 1 for all $k = 1, 2, \ldots, \overline{K}$.

Proof. Define $A_i^{cl}[k] = f(A_i)$ as a function of A_i. According to the property of eigenvalues in [32], an eigenvalue of $f(A_i)$ equals $f(\lambda_j) = -[e^{\lambda_j\Delta_i[k+1]} - 1][e^{\lambda_j(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} - 1]^{-1}e^{\lambda_j(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}$, where λ_j is an eigenvalue of A_i. To show that $A_i^{cl}[k]$ is stable, we need to show eigenvalues magnitudes of $A_i^{cl}[k]$ are less than 1, i.e.,

$$|f(\lambda_j)| = \frac{\left|e^{\lambda_j\Delta_i[k+1]} - 1\right|\left|e^{\lambda_j(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])}\right|}{\left|e^{\lambda_j(\tilde{\gamma}_i[k+1]-\tilde{\gamma}_i[k]-\Delta_i[k])} - 1\right|} < 1. \tag{21}$$

Denote $\Delta t = \tilde{\gamma}_i[k+1] - \tilde{\gamma}_i[k]$, then the magnitude of $f(\lambda_j)$ can be demonstrated to be smaller than 1 in two cases.

Case 1: For $\lambda_j < 0$, since $\Delta_i[k] < \frac{1}{2}\Delta t$ for all k from *Condition* 1, we have $\Delta_i[k] + \Delta_i[k+1] < \Delta t$, which can be rewritten as $\Delta_i[k+1] < \Delta t - \Delta_i[k]$. Since λ_j is negative, we have $\lambda_j\Delta_i[k+1] > \lambda_j(\Delta t - \Delta_i[k])$. Because the exponential function is monotonically increasing, it is easy to obtain $e^{\lambda_j\Delta_i[k+1]} > e^{\lambda_j(\Delta t - \Delta_i[k])}$. Since both $\lambda_j\Delta_i[k+1]$ and

$\lambda_j(\Delta t - \Delta_i[k])$ are non-positive, we have $0 \geq e^{\lambda_j \Delta_i[k+1]} - 1 > e^{\lambda_j(\Delta t - \Delta_i[k])} - 1$. Note that both sides of the inequality are smaller than 0, thus $\left|e^{\lambda_j \Delta_i[k+1]} - 1\right| < \left|e^{\lambda_j(\Delta t - \Delta_i[k])} - 1\right|$, i.e., $\left|e^{\lambda_j \Delta_i[k+1]} - 1\right| / \left|e^{\lambda_j(\Delta t - \Delta_i[k])} - 1\right| < 1$. Also because $e^{\lambda_j(\Delta t - \Delta_i[k])} \in (0,1)$, it has $|e^{\lambda_j \Delta_i[k+1]} - 1| \cdot |e^{\lambda_j(\Delta t - \Delta_i[k])}| / |e^{\lambda_j(\Delta t - \Delta_i[k])} - 1| < 1$, i.e., $|f(\lambda_j)| < 1$.

Case 2: For $\lambda_j > 0$, we define the function $g(\Delta t, \lambda_j) = \Delta t - \frac{1}{\lambda_j} \ln\left(\frac{e^{\lambda_j \Delta t}+1}{2}\right)$. Take the partial derivative of $g(\Delta t, \lambda_j)$ with respect to λ_j, we have

$$\frac{\partial g(\Delta t, \lambda_j)}{\partial \lambda_j} = \frac{\ln\left(\frac{e^{\lambda_j \Delta t}+1}{2}\right) - \lambda_j \Delta t \frac{e^{\lambda_j \Delta t}}{e^{\lambda_j \Delta t}+1}}{\lambda_j^2}, \tag{22}$$

and we can show $\frac{\partial g(\Delta t, \lambda_j)}{\partial \lambda_j} \leq 0$ (see detailed proof in the Appendix), meaning that $g(\Delta t, \lambda_j)$ is non-increasing with respect to λ_j. Therefore, $g(\Delta t, \lambda_M) \leq g(\Delta t, \lambda_j)$. Besides, the partial derivative of $g(\Delta t, \lambda_M)$ to Δt is $\frac{\partial g(\Delta t, \lambda_M)}{\partial \Delta t} = 1/(e^{\lambda_M \Delta t} + 1) > 0$, which means $g(\Delta t, \lambda_M)$ is monotonically increasing with respect to Δt. So, it has $g(\Delta t_m, \lambda_M) \leq g(\Delta t, \lambda_M)$. Therefore, the function $g(\Delta t, \lambda_j)$ obtains minimum at $(\Delta t_m, \lambda_M)$. Also because the *Condition* 1 must be satisfied by $\overline{\Delta}_i$, we have that $\overline{\Delta}_i < g(\Delta t_m, \lambda_M) \leq g(\Delta t, \lambda_j) = \Delta t - \frac{1}{\lambda_j} \ln(\frac{e^{\lambda_j \Delta t}+1}{2})$. Rearrange this inequality, we have $\ln\left[(e^{\lambda_j \Delta t} + 1)/2\right] < \lambda_j(\Delta t - \overline{\Delta}_i)$ Since the exponential function is monotonically increasing, the inequation sign would not change to get $(e^{\lambda_j \Delta t} + 1)/2 < e^{\lambda_j(\Delta t - \overline{\Delta}_i)}$, then $e^{\lambda_j \Delta t} + 1 < 2e^{\lambda_j(\Delta t - \overline{\Delta}_i)}$. Rearrange the inequation again, it has $e^{\lambda_j \Delta t} - e^{\lambda_j(\Delta t - \overline{\Delta}_i)} < e^{\lambda_j(\Delta t - \overline{\Delta}_i)} - 1$. Since $e^{\lambda_j \Delta t} - e^{\lambda_j(\Delta t - \overline{\Delta}_i)} = e^{\lambda_j(\Delta t - \overline{\Delta}_i)}(e^{\lambda_j \overline{\Delta}_i} - 1)$, we can rewrite this inequality as $e^{\lambda_j(\Delta t - \overline{\Delta}_i)}(e^{\lambda_j \overline{\Delta}_i} - 1) < e^{\lambda_j(\Delta t - \overline{\Delta}_i)} - 1$. Note that both sides of the inequality are positive, let them be divided by the right side item, we have $\frac{e^{\lambda_j(\Delta t - \overline{\Delta}_i)}(e^{\lambda_j \overline{\Delta}_i} - 1)}{e^{\lambda_j(\Delta t - \overline{\Delta}_i)} - 1} = \frac{e^{\lambda_j(\Delta t - \overline{\Delta}_i)}}{e^{\lambda_j(\Delta t - \overline{\Delta}_i)} - 1} \cdot \left(e^{\lambda_j \overline{\Delta}_i} - 1\right) < 1$. Because $\frac{e^{\lambda_j(\Delta t - x)}}{e^{\lambda_j(\Delta t - x)} - 1} = 1 + \frac{1}{e^{\lambda_j(\Delta t - x)} - 1}$ is increasing as x increases, and $\Delta_i[k] \leq \overline{\Delta}_i$, then $0 < \frac{e^{\lambda_j(\Delta t - \Delta_i[k])}}{e^{\lambda_j(\Delta t - \Delta_i[k])} - 1} \leq \frac{e^{\lambda_j(\Delta t - \overline{\Delta}_i)}}{e^{\lambda_j(\Delta t - \overline{\Delta}_i)} - 1}$. Also because the function $e^{\lambda_j x} - 1$ is monotonically increasing, and $\Delta_i[k+1] \leq \overline{\Delta}_i$, it is easy to acquire $0 < e^{\lambda_j \Delta_i[k+1]} - 1 \leq e^{\lambda_j \overline{\Delta}_i} - 1$. Combining these two inequalities, we have $\frac{e^{\lambda_j(\Delta t - \Delta_i[k])}}{e^{\lambda_j(\Delta t - \Delta_i[k])} - 1}\left(e^{\lambda_j \Delta_i[k+1]} - 1\right) < 1$, which means $|f(\lambda_j)| = \left|(e^{\lambda_j \Delta_i[k+1]} - 1)\right| \left|e^{\lambda_j(\Delta t - \Delta_i[k])}\right| / \left|e^{\lambda_j(\Delta t - \Delta_i[k])} - 1\right| < 1$.

Combining these two cases, all eigenvalues magnitudes of $A_i^{cl}[k]$ are smaller than 1, $\forall k$, which completes the proof. □

Thus, $e_i[k+1] = A_i^{cl}[k] e_i[k]$ is asymptotic stable.

Lemma 3: The set $\mathbb{W}_i = \{w_i[k]\}$, comprising all additive disturbance $w_i[k]$, is a compact set that contains the origin.

Proof. Since $\mathbb{U}_i$ is compact, it is bounded and closed due to Heine Borel's Theorem [33]. Then the set $\{u_i^n[k] - u_i^n[k+1]\}$ made of two elements subtraction in $\mathbb{U}_i$ is also bounded and closed. Since $\Delta_i[k] \in [0, \overline{\Delta}_i]$ is bounded and closed due to *Assumption* 1, with constant matrices A_i and B_i, the term $\left[e^{A_i \Delta_i[k+1]} - I\right] A_i^{-1} B_i(u_i^n[k] - u_i^n[k+1])$ as a function of $\Delta_i[k]$ is bounded and closed. So, the set $w_i[k] = \left[e^{A_i \Delta_i[k+1]} - I\right] A_i^{-1} B_i(u_i^n[k] - u_i^n[k+1])$ is bounded and closed, i.e., compact. Also, $w_i[k] = 0$ if $\Delta_i[k+1] = 0$ or $u_i^n[k] = u_i^n[k+1]$. So, $\mathbb{W}_i$ contains the origin. □

Theorem 1: The set $\mathbb{S}_{i,\infty}$ exists and is a positive forward invariant set (or FIS).

Proof. Since the closed-loop matrix $A_i^{cl}[k]$ is proven to be asymptotic stable for all k in *Proposition* 1, and the set $Y = \{w_i[k]\}$ is proven to be a compact set in *Lemma* 3, then the set $\mathbb{S}_{i,\infty}$ must exist, referring to [34].

The set $\mathbb{S}_{i,\infty}$ is normally hard or computationally expensive to compute. An outer bound of $\mathbb{S}_{i,\infty}$, denoted as $\tilde{\mathbb{S}}_i$ can be leveraged to approximate the FIS of the tube, using either analytical methods [34] or numerical methods [35].

IV. Simulation

This section presents four scalar systems using one shared resource with timing perturbations. The nominal system models are $\dot{x}_i^n(t) = a_i x_i^n(t) + u_i^n(t)$, where $a_i = [1, \frac{6}{5}, \frac{4}{3}, \frac{3}{2}]$ for $i = 1,2,3,4$. Their initial conditions are $x_i(0) = x_i^n(0) = 1$. The time horizon is from 0 to 6 seconds. Given nominal system control constraints $u_i^n(t) \in [-3, 3]$, we use the same schedule as in [17], where the task completion time $\tilde{\gamma}_1[k] = [0, 1, 1.3, 2.5, 3.5, 4.35, 5.3]$, $\tilde{\gamma}_2[k] = [0, 0.7, 1.8, 2.8, 4.05, 5.6]$, $\tilde{\gamma}_3[k] = [0, 0.4, 1.7, 3.3, 4.7]$, and $\tilde{\gamma}_4[k] = [0, 0.2, 2.2, 4.75]$. The cost function is $\frac{1}{2}\sum_{i=1}^{4}\{\sum_{k=0}^{\overline{K}_i - 1}(x_i^n[k]^2 + 0.0001 u_i^n[k]^2) + x_i^n[\overline{K}_i]^2\}$, for $\overline{K}_i = [6, 5, 4, 3]$. Note that all nominal systems are stabilized.

For perturbed systems, *Condition* 1 should be satisfied. Since all the open-loop systems are unstable, timing perturbations of system 1 satisfy $\Delta_1[k] < 0.3 - \frac{1}{1} ln[(e^{0.3 \times 1}) + 1)/2] \approx 0.1387$. Similarly, the other satisfies $\Delta_2[k] < 0.3582, \Delta_3[k] < 0.3978$, and $\Delta_4[k] < 0.4297$. With these conditions, we set timing perturbations to be $\Delta_1[k] = [0, 0.138, 0, 0.138, 0.13, 0.135, 0.09]$, $\Delta_2[k] = [0, 0.35, 0.08, 0.3, 0, 0.35]$, $\Delta_3[k] = [0, 0.2, 0.1, 0.39, 0]$, and $\Delta_4[k] = [0, 0.42, 0.05, 0.15]$ in simulation. For traditional tube-based MPC, we design constant gains $\mathbf{K}_c = [1.83, 1.64, 1.68, 1.565]$ for all i.

The simulation results are shown in Fig. 5. First, all four systems become unstable if there is no compensation. Thus, we do need the robust MPC design for timing perturbations. With the traditional tube-based MPC (shown by green lines), the perturbed system trajectories can be stabilized but the performance is worse than our method (larger deviations from nominal trajectories). This is because $A_i[k] - B_i[k] K_c$ is unstable in some time intervals. For example, system 2 is unstable in $[4.05, 5.95]$s, and system 3 is unstable in $[2.09, 3.3]$s. For our control law (8) with dynamic gain $K(\Delta_i[k])$ (showing by red lines), the perturbed and nominal system states are identical at $\tilde{\gamma}_i[k]$, and perturbed systems are bounded at $\gamma_i[k]$. Also, all perturbed systems can be stabilized, verifying that our control policy is more effective.

V. Conclusions and Future Works

In this paper, we provided a tube-based MPC policy for the LTI system to compensate for timing perturbations. We also identified the upper bound of the allowed timing perturbations for system stability. Additionally, we have proven

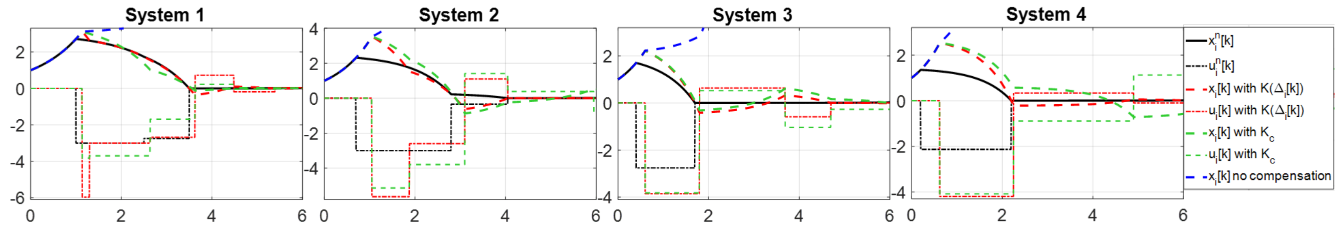

Fig. 5. States of four systems. The x axes represent time and the y axes represent $x_i(t)$ in each sub-figure, respectively. The black solid lines show the nominal system. The blue dashed lines show the perturbed system without compensation. The green dashed lines show the perturbed system with tube-based MPC with constant K_c. The red dashed lines show the perturbed system with our robust control policy (8).

that the state deviation is bounded by an FIS, which can help find a feasible solution to MPC optimization problems. For future work, we will generalize the proposed design to time-varying and nonlinear systems.

VI. Appendix

Denote $\lambda_j \Delta t$ as x, we can define the numerator of (22) as $L(x)=\ln\left(\frac{e^x+1}{2}\right)-\frac{xe^x}{e^x+1}$. When $x=0$, it is trivial that $L(0)=0$. Taking derivative of $L(x)$ with respect to x, we have $\frac{dL(x)}{dx}=\frac{e^x}{e^x+1}-\frac{(xe^x)'(e^x+1)-xe^x\cdot e^x}{(e^x+1)^2}=\frac{e^x}{e^x+1}-\frac{e^{2x}+xe^x+e^x}{(e^x+1)^2}=\frac{e^x(e^x+1)-(e^{2x}+xe^x+e^x)}{(e^x+1)^2}=-\frac{xe^x}{(e^x+1)^2}$. Since $\lambda_j>0$ in **Case 2**, we have $x=\lambda_j\Delta t\geq 0$ and $\frac{dL(x)}{dx}=-\frac{xe^x}{(e^x+1)^2}\leq 0$, meaning that $L(x)$ is non-increasing for $x\geq 0$, which leads to $L(x)\leq 0$ when $x\geq 0$. Therefore, $\frac{\partial g(\Delta t,\lambda_j)}{\partial \lambda_j}=\frac{L(\lambda_j\Delta t)}{\lambda_j^2}\leq 0$.

References

[1] E. Sontag, "Input to state stability: Basic concepts and results," in *Nonlinear and optimal control theory*. Springer, 2008, pp. 163–220.

[2] M. Malisoff and F. Mazenc, *Constructions of strict Lyapunov functions*. Springer Science & Business Media, 2009.

[3] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier Lyapunov functions for the control of output-constrained nonlinear systems," *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.

[4] Ames *et al.*, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[5] 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.

[6] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design*. Nob Hill Pub., 2009.

[7] D. Q. Mayne, E. C. Kerrigan, E. Van Wyk, and P. Falugi, "Tube-based robust nonlinear model predictive control," *International Journal of Robust and Nonlinear Control*, vol. 21, no. 11, pp. 1341–1353, 2011.

[8] J. Sieber, A. Zanelli, S. Bennani, and M. N. Zeilinger, "System level disturbance reachable sets and their application to tube-based MPC," *European Journal of Control*, vol. 68, p. 100680, 2022.

[9] J. Sieber, S. Bennani, and M. N. Zeilinger, "A system level approach to tube-based model predictive control," *IEEE Control Systems Letters*, vol. 6, pp. 776–781, 2021.

[10] M. M. Morato, J. E. Normey-Rico, and O. Sename, "Model predictive control design for linear parameter varying systems: A survey," *Annual Reviews in Control*, vol. 49, pp. 64–80, 2020.

[11] P. Bumroongsri, "Tube-based robust MPC for linear time-varying systems with bounded disturbances," *International Journal of Control, Automation and Systems*, vol. 13, pp. 620–625, 2015.

[12] P. Falugi and D. Q. Mayne, "Getting robustness against unstructured uncertainty: A tube-based MPC approach," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1290–1295, 2013.

[13] R. Wang and J. Bao, "A differential Lyapunov-based tube MPC approach for continuous-time nonlinear processes," *Journal of Process Control*, vol. 83, pp. 155–163, 2019.

[14] N. Yao *et al.*, "Contention resolving optimal priority assignment for event-triggered model predictive controllers," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 2357–2362.

[15] N. Yao, M. Malisoff, and F. Zhang, "Contention-resolving model predictive control for coordinating automated vehicles at a traffic intersection," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 2233–2238.

[16] N. Yao and F. Zhang, "Contention-resolving model predictive control for an intelligent intersection traffic model," *Discrete Event Dynamic Systems*, pp. 1–31, 2021.

[17] N. Yao, M. Malisoff, and F. Zhang, "Contention-resolving model predictive control for coupled control systems with a shared resource," *Automatica*, vol. 122, p. 109219, 2020.

[18] N. Yao and F. Zhang, "Event-triggered scheduling and control co-design for networked control systems with sub-schedulability," in *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 1733–1738.

[19] Z. Sun, X. Bao, and N. Sharma, "Tube-based model predictive control of an input delayed functional electrical stimulation," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 5420–5425.

[20] T. L. Santos, J. E. Normey-Rico, and D. Limón, "Explicit input-delay compensation for robustness improvement in MPC," *IFAC Proceedings Volumes*, vol. 43, no. 2, pp. 384–389, 2010.

[21] H. Yu, Z. Wang, and Y. Zheng, "On the model-based networked control for singularly perturbed systems," *Journal of Control Theory and Applications*, vol. 6, no. 2, pp. 153–162, 2008.

[22] V. Léchappé, E. Moulay, F. Plestan, A. Glumineau, and A. Chriette, "New predictive scheme for the control of lti systems with input delay and unknown disturbances," *Automatica*, vol. 52, pp. 179–184, 2015.

[23] A. A. Pyrkin and A. A. Bobtsov, "Adaptive controller for linear system with input delay and output disturbance," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 4229–4234, 2015.

[24] B. Zhou, "Input delay compensation of linear systems with both state and input delays by nested prediction," *Automatica*, vol. 50, no. 5, pp. 1434–1443, 2014.

[25] Q. Weiwei, L. Jieyu, H. Xiaoxiang, H. Bing, and L. Gang, "Tube-based active robust mpc for uncertain constrained linear systems with time delays," *IEEE Access*, vol. 7, pp. 125 552–125 561, 2019.

[26] H.-S. Yan and Z.-Y. Duan, "Tube-based model predictive control using multidimensional taylor network for nonlinear time-delay systems," *IEEE Transactions on Automatic Control*, vol. 66, no. 5, pp. 2099–2114, 2020.

[27] Zhang *et al.*, "A new method for stabilization of networked control systems with random delays," *IEEE Transactions on automatic control*, vol. 50, no. 8, pp. 1177–1181, 2005.

[28] Z. Cao, Z. Wang, Y. Niu, J. Song, and H. Liu, "Sliding mode control for sampled-data systems subject to deception attacks: Handling randomly perturbed sampling periods," *IEEE Transactions on Cybernetics*, 2022.

[29] R. B. GmbH, "CAN specification (version 2.0)," 1991.

[30] Z. Shi and F. Zhang, "Model predictive control under timing constraints induced by controller area networks," *Real-Time Systems*, vol. 53, pp. 196–227, 2017.

[31] Limón *et al.*, "Robust tube-based MPC for tracking of constrained linear systems with additive disturbances," *Journal of Process Control*, vol. 20, no. 3, pp. 248–260, 2010.

[32] C. D. Meyer, *Matrix analysis and applied linear algebra*. Siam, 2000, vol. 71.

[33] H. K. Khalil, *Nonlinear control*. Pearson New York, 2015, vol. 406.

[34] I. Kolmanovsky, E. G. Gilbert *et al.*, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Mathematical problems in engineering*, vol. 4, pp. 317–367, 1998.

[35] P. Varnell, S. Mukhopadhyay, and F. Zhang, "Discretized boundary methods for computing smallest forward invariant sets," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 6518–6524.