

IEICE **TRANSACTIONS**

on Fundamentals of Electronics, Communications and Computer Sciences

DOI:10.1587/transfun.2024MAP0003

Publicized:2024/10/22

This advance publication article will be replaced by
the finalized version after proofreading.



A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY

The Institute of Electronics, Information and Communication Engineers

Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

Shield-Based Safe Control with Compensation of Computation Delays of Nonlinear Discrete-Time Systems

Toshimitsu USHIO^{†a)}, *Fellow*

SUMMARY We consider a nonlinear discrete-time system and a control specification including safety properties. In general, there exists a computation delay to compute a control input in a controller and the delay degrades control performances. To compensate the computation delay, we propose a safe controller using Mita's method and the shield synthesis, where the satisfaction of the safety properties is decided using a high-order control barrier function. First, we show an effect of the computation delay on the relative degree of the system. Next, we show the reduction of the number of functions that specify the high-order control barrier function. Finally, we propose a safe controller consisting of a predictor, a controller, and a shield. Then, we illustrate the proposed method on safe control of mobile robots.

key words: *Nonlinear discrete-time system, computation delay, safe control, Control barrier function.*

1. Introduction

In highly automated systems such as robots, self driving cars, and smart grid, it is an important issue to guarantee safety properties [1]. In state-based approaches, the safety properties are formulated as a set of safe states, which is called a safe set, and Nagumo shows that a necessary and sufficient condition for the satisfaction of the safety properties in continuous-time systems is positive invariance of the corresponding safe set [2]–[4].

From the practical point of view, a Lyapunov-like function based approach is useful to prove the invariance of the safe set [5], [6]. In this approach, we introduce a function called a control barrier function and the safe set is described by a set of states where the function is nonnegative. Then, the invariance of the safe set is checked by its derivative along the trajectory of the systems. The control barrier functions have been applied to many control problems such as control guaranteeing input-to-state safety against input noises [7], control under signal linear temporal logic constraints [8], model predictive control satisfying safety properties [9], self/event-triggered control for safety-critical systems [10], [11], and applications to safe reinforcement learning [12] and robotic grasping [13]. The control barrier function based approaches to discrete-time systems have been proposed [14] and applied to safety-critical model predictive control [15], formal synthesis of controllers for discrete-time stochastic systems [16], and self-driving cars [17]. Moreover, sym-

bolic control barrier functions have been proposed and applied to safety-critical embedded control [18] and opaque control [19]. For a system whose relative degree is mode than 2, a safe controller cannot be synthesized using the control barrier function since the derivative of the function along the trajectory of the system vanishes. To overcome this problem in continuous-time systems, high-order control barrier functions have been introduced [20]–[22] and have been extended to discrete-time systems [23]–[25].

In general, for a control specification including safety properties, it is sufficient to pay attention to the safety properties only when the state of the system is very close to the corresponding unsafe set. Thus, from the practical point of view, we design a controller that satisfies the control specification except the safety properties and attach a component that modifies the control input computed by the controller at runtime so as to keep the safety properties. Such a component is called a shield [26] or a safety filter [6], [27]. In other words, the shield decides if the control input keeps the safety properties and modifies it only if it violates the safety properties. The shield has been extended to an adaptive one in uncertain systems [28]. It is applied to safe reinforcement learning [29], [30], synthesis of reactive systems [31], and path planning of mobile robots [32], [33].

On the other hand, in digital control systems, the computation of the control inputs causes a delay called a computation delay, which degrades the control performances and sometimes makes the system unstable [34]. Mita proposes a method to compensate the computation delay using a predicted state [35]. Mita's method has been leveraged in real-time control systems [36], [37] and reinforcement learning of networked control systems [38]. Recently, it has been extended to continuous-time input delay systems [39].

In this paper, we propose a safe controller with compensation of computation delays of nonlinear discrete-time systems using Mita's method and the shield synthesis. The proposed safe controller consists of a predictor that predicts a future state of the systems, a controller that computes a control input candidate based on the predicted state, and a shield that modifies the control input candidate only if it violates the safety properties.

The rest of the paper is organized as follows. Section 2 reviews discrete-time control barrier functions briefly. Section 3 discusses relative degree of digital control systems with computation delays. Section 4 proposes a synthesis method of safe digital controllers with computation delays using Mita's method and the shield synthesis. Section 5

[†]The author is with the Faculty of Science and Technology, Nanzan University, 18 Yamazato, Showa, Nagoya, Aichi, 466-8673, Japan

a) E-mail: tushio@nanzan-u.ac.jp

illustrates the proposed method on safe control of mobile robots. Section 6 concludes this paper.

2. Preliminaries

We consider the following nonlinear discrete-time system.

$$x(t+1) = f(x(t), u(t)), \quad (1)$$

where $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^m$ are the state and the control input of the system at time $t \in \mathbb{N}$, respectively, and $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is a continuous function that describes the dynamics of the system.

We briefly review the high-order control barrier function. We assume that a safe set C is represented by

$$C = \{x \in \mathbb{R}^n \mid h(x) \geq 0\}, \quad (2)$$

where $h : \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuous function.

For $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$, denoted by $\bar{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is $\bar{f}(x) = f(x, \mathbf{0}_m)$, where $\mathbf{0}_m$ is the m -dimensional zero vector, and $\bar{f}^i : \mathbb{R}^n \rightarrow \mathbb{R}^n$ for each $i \in \mathbb{N}$ is defined as follows [40], [41].

$$\begin{aligned} \bar{f}^0(x) &= x, \\ \bar{f}^{i+1}(x) &= \bar{f}(\bar{f}^i(x)) \quad \forall i \in \mathbb{N}. \end{aligned}$$

We assume that the function h has relative degree r with respect to (1), that is, h satisfies the following two conditions.

1. $\frac{\partial}{\partial u} [h \circ \bar{f}^j(f(x, u))] \equiv \mathbf{0}_m^\top \forall j \in \{0, \dots, r-2\}$.
2. $\frac{\partial}{\partial u} [h \circ \bar{f}^{r-1}(f(x, u))] \neq \mathbf{0}_m^\top$.

Then, for the function h , we introduce a sequence of r functions $\psi_j : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ ($j \in \{0, 1, \dots, r\}$) as follows.

$$\begin{aligned} \psi_0(x, u) &= h(x), \\ \psi_j(x, u) &= \Delta\psi_{j-1}(x, u) + \alpha_j(\psi_{j-1}(x, \mathbf{0}_m)) \\ &\quad \forall j \in \{1, 2, \dots, r\}, \end{aligned} \quad (3)$$

where $\Delta\psi_j(x, u) = \psi_j(f(x, u), \mathbf{0}_m) - \psi_j(x, \mathbf{0}_m)$ and $\alpha_j : \mathbb{R} \rightarrow \mathbb{R}$ is an extended class \mathcal{K} function satisfying $\alpha_j(\ell) \leq \ell$ for all $\ell \in \mathbb{R}$. For each function ψ_j ($j \in \{0, 1, \dots, r-1\}$), we define the set C_j as follows.

$$C_j = \{x \in \mathbb{R}^n \mid \psi_j(x, \mathbf{0}_m) \geq 0\}. \quad (4)$$

Note that $C_0 = C$. A set $\hat{C} \subseteq C$ is defined by

$$\hat{C} = \bigcap_{j=0}^{r-1} C_j. \quad (5)$$

The function h is called a discrete-time high-order control barrier function with relative degree r for (1) if there exists a sequence of functions ψ_j ($j \in \{0, 1, \dots, r\}$) defined by (3) such that, for each $x \in \hat{C}$, there exists $u \in \mathbb{R}^m$ satisfying

$$\psi_r(x, u) \geq 0. \quad (6)$$

The control input u satisfying (6) for each $x \in \hat{C}$ is called a

permissive control input. For each state x , a set $K_{\psi_r}(x)$ of permissive control inputs is given by

$$K_{\psi_r}(x) = \{u \in \mathbb{R}^m \mid \psi_r(x, u) \geq 0\}. \quad (7)$$

Then, if a controller $K : \mathbb{R}^n \rightarrow \mathbb{R}^m$ satisfies $K(x) \in K_{\psi_r}(x)$ for each $x \in \hat{C}$ and the initial state x_0 is in the set \hat{C} , then the state $x(t)$ of (1) controlled by a controller $u(t) = K(x(t))$ is in \hat{C} for each $t \in \mathbb{N}$, that is, \hat{C} is forward invariant for (1) controlled by the controller $u(t) = K(x(t))$.

3. Control system with computation delay

We consider the case where it takes $\tau \in \mathbb{N}$ time steps for a controller to compute the control input $u(t)$ using the state $x(t)$ of nonlinear discrete-time system (1). Then, the controlled system is described by

$$x(t+1) = f(x(t), u(t-\tau)). \quad (8)$$

We introduce the following function $F_\tau : \mathbb{R}^{n+\tau m} \times \mathbb{R}^m \rightarrow \mathbb{R}^{n+\tau m}$.

$$F_\tau(x_\tau, u) = \begin{bmatrix} f(x, v_1) \\ v_2 \\ \vdots \\ v_\tau \\ u \end{bmatrix}, \quad (9)$$

where $x_\tau = [x^\top, v_1^\top, \dots, v_\tau^\top]^\top \in \mathbb{R}^{n+\tau m}$ and $v_j \in \mathbb{R}^m$ ($j \in \{1, 2, \dots, \tau\}$). Then, (8) is rewritten as the following augmented nonlinear system whose state at time t is denoted by $x_\tau(t) = [x(t)^\top, v_1(t)^\top, \dots, v_\tau(t)^\top]^\top$.

$$x_\tau(t+1) = F_\tau(x_\tau(t), u(t)). \quad (10)$$

The following proposition holds.

Proposition 1: For each $x_\tau = [x^\top, v_1^\top, \dots, v_\tau^\top]^\top \in \mathbb{R}^{n+\tau m}$, $x^j \in \mathbb{R}^n$ ($j \in \{1, 2, \dots, \tau\}$) is defined by

$$x^j = f(f(\dots(f(x, v_1), \dots), v_{j-1}), v_j). \quad (11)$$

Assume that the function $h : \mathbb{R}^n \rightarrow \mathbb{R}$ has relative degree r with respect to (1). Then, the function $h_\tau : \mathbb{R}^{n+\tau m} \rightarrow \mathbb{R}$ defined by $h_\tau(x_\tau) = h(x)$ for each $x_\tau = [x^\top, v_1^\top, \dots, v_\tau^\top]^\top \in \mathbb{R}^{n+\tau m}$ has relative degree $r + \tau$ with respect to (10).

Proof. The function $\bar{F}_\tau : \mathbb{R}^{n+\tau m} \rightarrow \mathbb{R}^{n+\tau m}$ is defined by $\bar{F}_\tau(x_\tau) = F_\tau(x, \mathbf{0}_m)$. Then, for each $j \in \{1, 2, \dots, \tau\}$,

$$\begin{aligned} \bar{F}_\tau^j(F_\tau(x_\tau, u)) &= \\ &= \begin{bmatrix} f(f(\dots(f(x, v_1), \dots, v_j), v_{j+1})) \\ v_{j+2} \\ \vdots \\ v_\tau \\ u \\ \mathbf{0}_m \\ \vdots \\ \mathbf{0}_m \end{bmatrix} = \begin{bmatrix} f(x^j, v_{j+1}) \\ v_{j+2} \\ \vdots \\ v_\tau \\ u \\ \mathbf{0}_m \\ \vdots \\ \mathbf{0}_m \end{bmatrix}, \end{aligned} \quad (12)$$

where $v_{\tau+1} = u$, and we have

$$h_{\tau}(\bar{F}_{\tau}^j(F_{\tau}(x_{\tau}, u))) = h(f(x^j, v_{j+1})). \quad (13)$$

Thus, for each $j \in \{1, 2, \dots, \tau\}$,

$$\frac{\partial}{\partial u} h_{\tau}(\bar{F}_{\tau}^j(F_{\tau}(x_{\tau}, u))) = \mathbf{0}_m,$$

and, for each $j > \tau$,

$$h_{\tau}(\bar{F}_{\tau}^j(F_{\tau}(x_{\tau}, u))) = h(\bar{f}^{j-\tau}(f(x^{\tau}, u))). \quad (14)$$

Thus, by the assumption of h , h_{τ} has relative degree $\tau + r$ with respect to (10). \square

4. Design of shield-based safe controller

By Proposition 1, when we use an existing method for the design of a safe controller with a computation delay τ for (1) using a high-order control barrier function, we check whether the function $h_{\tau} : \mathbb{R}^{n+\tau m} \rightarrow \mathbb{R}$ defined by $h_{\tau}(x_{\tau}) = h(x)$ is a high-order control barrier function with relative degree $r + \tau$ for (10), where $x_{\tau} = [x^{\top}, v_1^{\top}, \dots, v_{\tau}^{\top}]^{\top}$. Thus, we select $r + \tau$ extended class \mathcal{K} functions α_j ($j \in \{1, 2, \dots, r + \tau\}$) that is used to construct a sequence of functions given by (3). To reduce the number of the extended class \mathcal{K} functions, we propose a novel safe controller for (1) where we use a high-order control barrier function.

For the given safe set \mathcal{C} for (1) given by (2), it is assumed that h has relative degree r with respect to (1) and is a high-order control barrier function with relative degree r for (1). Then, we have a sequence of r functions ψ_j ($j \in \{1, 2, \dots, r\}$) defined by (3). For the set $\hat{\mathcal{C}} \subset \mathbb{R}^n$ given by (5), we induce a safe set $\hat{\mathcal{C}}_{\tau}$ for (10) as follows.

$$\begin{aligned} \hat{\mathcal{C}}_{\tau} = \{ & [x^{\top}, v_1^{\top}, \dots, v_{\tau}^{\top}]^{\top} \in \mathbb{R}^{n+\tau m} \mid x \in \hat{\mathcal{C}}, v_1 \in K_{\psi_r}(x), \\ & v_j \in K_{\psi_r}(f(f(\dots f(f(x, v_1), v_2), \dots, v_{j-2}), v_{j-1}))) \\ & \forall j \in \{2, 3, \dots, \tau\}\}. \end{aligned} \quad (15)$$

Then, the following proposition holds.

Proposition 2: For any $x_{\tau} = [x^{\top}, v_1^{\top}, \dots, v_{\tau}^{\top}]^{\top} \in \hat{\mathcal{C}}_{\tau}$, consider $u \in \mathbb{R}^m$ satisfying

$$u \in K_{\psi_r}(f(f(\dots f(f(x, v_1), v_2), \dots, v_{\tau-1}), v_{\tau}))). \quad (16)$$

Then,

$$F(x_{\tau}, u) \in \hat{\mathcal{C}}_{\tau}. \quad (17)$$

Proof. Let $x'_{\tau} = [x'^{\top}, v_1'^{\top}, \dots, v_{\tau}'^{\top}]^{\top} = F(x_{\tau}, u)$. Then, since $v_1 \in K_{\psi_r}(x)$, we have

$$x' = f(x, v_1) \in \hat{\mathcal{C}}.$$

For each $j \in \{1, 2, \dots, \tau - 1\}$, since $v'_j = v_{j+1}$, we have

$$\begin{aligned} v'_j & \in K_{\psi_r}(f(f(\dots f(f(x, v_1), v_2), \dots, v_{j-1}), v_j))) \\ & = K_{\psi_r}(f(f(\dots f(x', v'_1), \dots, v'_{j-2}), v'_{j-1}))). \end{aligned}$$

Since $v'_{\tau} = u$, (16) implies that

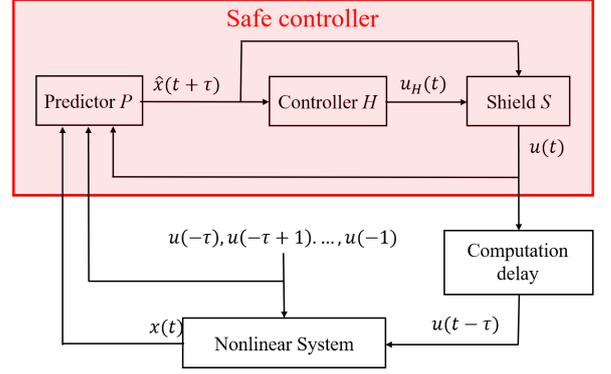


Fig. 1 Block diagram of a controlled system by a safe controller with a predictor and a shield.

$$\begin{aligned} v'_{\tau} & \in K_{\psi_r}(f(f(\dots f(f(x, v_1), v_2), \dots, v_{\tau-1}), v_{\tau}))) \\ & = K_{\psi_r}(f(f(\dots f(x', v'_1), \dots, v'_{\tau-2}), v'_{\tau-1}))). \end{aligned}$$

Thus, $x'_{\tau} \in \hat{\mathcal{C}}_{\tau}$. \square

By Proposition 2, we compute a permissive control input at state x using the set $K_{\psi_r}(x)$.

In the following, we assume that we have two types of control specifications. One is called a safety specification which is related to a safety property given by the safe set (2) and the other is called a stability specification which is related to the convergence to a target state. For the stability specification, using standard control system design approaches, we design a state feedback controller $H : \mathbb{R}^n \rightarrow \mathbb{R}^m$ beforehand for the nonlinear discrete-time system (1) without taking the computation time into consideration. We assume that the priority of the safety specification is the highest among the control specifications. Then, we propose a safe controller by which the controlled system never violates the safety specification as long as a permissive control input exists.

We consider the case where it takes τ time steps for the computation of the control input by the proposed safe controller. Then, it is an important issue to avoid the violation of the control specifications by the computation delay τ . Our proposed approach is based on Mita's method [35] and the shield synthesis [26]. The proposed safe controller is composed of a predesigned controller H satisfying the stability specification, a predictor P depending on the computation delay τ , and a shield S satisfying the safety specification. Shown in Fig. 1 is its block diagram. The predictor P predicts the state $\hat{x}(t + \tau)$ after τ time steps from the current time t . Then, the controller H computes a control input candidate $u_H(t)$. The shield S injects $u_H(t)$ to the system if $u_H(t)$ is a permissive control input, but otherwise modifies it so as to keep the controlled trajectory of the system in the safe set and injects the modified one to the system. In the following, we describe the detail of the safe controller.

Step 1. Computation of the set of permissive control inputs:

Let r be the relative degree of h with respect to (1). Then, we determine a sequence of the functions ψ_j ($j = 0, 1, \dots, r$) defined by (3) and compute the safe set $\hat{\mathcal{C}}$ given

by (5) and the set $K_{\psi_r}(x)$ of permissive control inputs given by (7).

Step 2. Computation of a sequence of initial control inputs:

Since there exists a computation delay τ , before the control action starts, we determine a sequence of control inputs $u(-\tau), u(-\tau+1), \dots, u(-1)$ satisfying the following equations.

$$\begin{aligned} u(-\tau) &\in K_{\psi_r}(x(0)), \\ u(-\tau+j) &\in K_{\psi_r}(f(f(\dots f(f(x(0), u(-\tau)), u(-\tau+1)), \\ &\dots, u(-\tau-2+j)), u(-\tau-1+j))), \\ &\forall j \in \{1, 2, \dots, \tau-1\}. \end{aligned}$$

Then, they are stored in a memory and injected to the system from $u(-\tau)$ consecutively when the control action starts.

Step 3. Computation of the predicted state:

At each time t , the predictor P computes a predicted state $\hat{x}(t+\tau)$ at time $t+\tau$ using the control inputs $u(t-\tau), u(t-\tau+1), \dots, u(t-1)$ injected to the system as follows.

$$\begin{aligned} \hat{x}(t+\tau) &= f(f(\dots f(f(x(t), u(t-\tau)), u(t-\tau+1)), \\ &\dots), u(t-1)). \end{aligned} \quad (18)$$

Step 4. Computation of the control input candidate:

At each time t , the predesigned controller H satisfying the stability specification computes a control input candidate $u_H(t)$ using the predicted state $\hat{x}(t+\tau)$ as follows.

$$u_H(t) = H(\hat{x}(t+\tau)). \quad (19)$$

Step 5. Determination of the control input:

The shield injects the control input candidate $u_H(t)$ if it is a permissive control input, but otherwise modifies it to become a permissive control input, that is,

$$u(t) = \begin{cases} u_H(t) & \text{if } u_H(t) \in K_{\psi_r}(\hat{x}(t+\tau)), \\ u^*(t) & \text{otherwise,} \end{cases} \quad (20)$$

where $u^*(t)$ is a control input such that $u^*(t) \in K_{\psi_r}(\hat{x}(t+\tau))$. There are many ways to determine $u^*(t)$. For example, when $u^*(t)$ is a minimal modification of the control input candidate $u(t)$, it is an optimal solution of the following optimization problem.

$$\min_{u(t) \in K_{\psi_r}(\hat{x}(t+\tau))} \|u(t) - u_H(t)\|. \quad (21)$$

5. Illustrative example

We consider the following discrete-time model of a unicycle mobile robot that moves on the x - y plane with a constant speed $V > 0$ [21].

$$\begin{cases} x(t+1) &= x(t) + TV \cos \theta(t), \\ y(t+1) &= y(t) + TV \sin \theta(t), \\ \theta(t+1) &= \theta(t) + Tu(t), \end{cases} \quad (22)$$

where $T > 0$ is the step width of the time discretization and

$(x(t), y(t))$, $\theta(t)$, and $u(t)$ denote the location of the robot on the x - y plane, the heading angle, and the forward acceleration at each discrete-time t , respectively. Our control objective is that, by using the forward acceleration $u(t)$ as a control input, the robot approaches to the x -axis without entering an unsafe set S_u on the x - y plane given by

$$S_u = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 < 1\}.$$

Let $h(x, y, \theta) = x^2 + y^2 - 1$. Then the safe set C is given by $C = \{(x, y, \theta) \mid h(x, y, \theta) \geq 0\}$ and h has relative degree 2 with respect to (22).

In the following, for simplicity, we set $V = 1$. Then, the functions ψ_j defined by (3) with $r = 2$ and $\alpha_j(\ell) = a_j \ell$, where $a_j \in (0, 1]$ is a constant, are as follows.

$$\psi_1(x, y, \theta, u) = a_1 w(x, y) + 2T(x \cos \theta + y \sin \theta) + T^2, \quad (23)$$

$$\begin{aligned} \psi_2(x, y, \theta, u) &= (a_1 + a_2 - 1)w'(x, y, \theta) \\ &+ (a_1 - 1)(a_2 - 1)w(x, y) + T^2 \\ &+ 2T\sqrt{w'(x, y, \theta) + 1} \sin(Tu + \theta + \phi(x, y, \theta)), \end{aligned} \quad (24)$$

where $w(x, y)$, $w'(x, y, \theta)$, and $\phi(x, y, \theta)$ are defined by

$$\begin{aligned} w(x, y) &= x^2 + y^2 - 1, \\ w'(x, y, \theta) &= w(x + T \cos \theta, y + T \sin \theta), \\ \phi(x, y, \theta) &= \sin^{-1} \left(\frac{x + T \cos \theta}{\sqrt{w'(x, y, \theta) + 1}} \right). \end{aligned}$$

Note that the set $K_{\psi_2}(x, y, \theta)$ defined by (7) is empty if and only if the following equation holds.

$$\begin{aligned} |(a_1 + a_2 - 1)w'(x, y, \theta) + (a_1 - 1)(a_2 - 1)w(x, y) + T^2| \\ > 2T\sqrt{w'(x, y, \theta) + 1}. \end{aligned}$$

We use the following state feedback controller $H(y, \theta)$.

$$H(y, \theta) = -25y - 10\theta.$$

Then, it is easily shown that, if $u(t) = H(y(t), \theta(t))$, the robot approaches to the x -axis if $(y(0), \theta(0))$ is sufficiently close to $(0, 0)$. Denoted by τ is the computation delay and $u(t)$ is given by

$$u(t) = H(y(t-\tau), \theta(t-\tau)). \quad (25)$$

Let $\tau = 1$, $(x(0), y(0), \theta(0)) = (-5.5, 0.5, 0)$, and $T = 0.05$. We set $u(-1) = H(y(0), \theta(0)) \in K_{\psi_2}(x(0), y(0), \theta(0))$. Then, shown in Fig. 2 is the controlled trajectory of the robot on the x - y plane, where the interior of the dotted red circle is the unsafe set. The robot starts moving and approaches to the x -axis. But when it is close to the unsafe set, it moves along the boundary of the unsafe set and approaches to the x -axis again after passing the set. We introduce the following variable $SH(t)$.

$$SH(t) = \begin{cases} 1 & \text{if } u_H(t) \in K_{\psi_2}(x(t), y(t), \theta(t)), \\ -1 & \text{otherwise.} \end{cases}$$

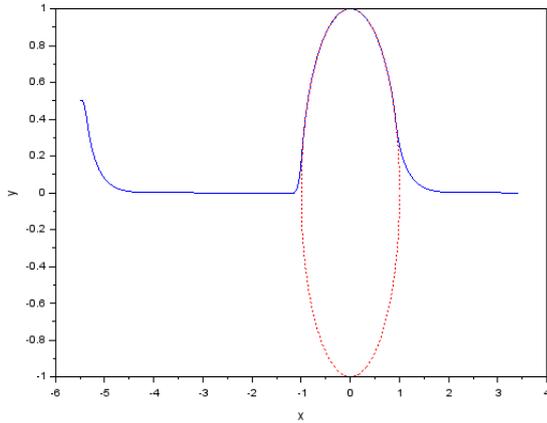


Fig. 2 Controlled trajectory on the x - y plane controlled by the proposed safe controller, where $(x(0), y(0), \theta(0)) = (-5.5, 0.5, 0)$, $T = 0.05$, $a_1 = a_2 = 0.5$.

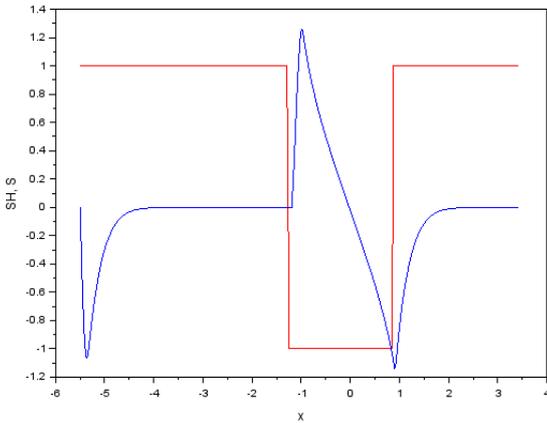


Fig. 3 Trajectories of θ and SH controlled by the proposed safe controller, where the blue and the red line represent trajectories of θ and SH , respectively, and $(x(0), y(0), \theta(0)) = (-5.5, 0.5, 0)$, $T = 0.05$, and $a_1 = a_2 = 0.5$.

Shown in Fig. 3 are the trajectories of θ and SH , respectively. When the robot starts moving, the control input candidate is injected to the robot and θ converges to 0. During the avoidance of the unsafe set, the modified control input by the shield is injected to the robot. Then, the control input candidate is injected to the robot again after passing the unsafe set. Thus, the proposed safe controller satisfies the control objective and the compensation of the computation delay is performed.

Shown in Fig. 4 is the trajectory of the robot without the compensation of the computation delay, that is, the control input candidate and the modified control input are not computed based on the predicated state but the current one. When the robot starts moving, it approaches to the x -axis, that is, the controller can stabilize the robot under the existence of the computation delay. However, the robot

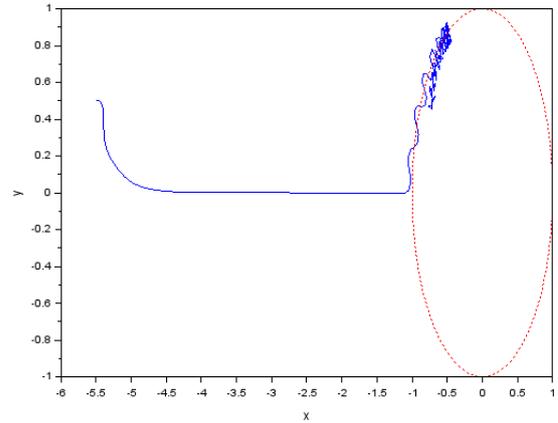


Fig. 4 Trajectory on the x - y plane without compensation of computation delay, where $(x(0), y(0), \theta(0)) = (-5.5, 0.5, 0)$, $T = 0.05$, $a_1 = a_2 = 0.5$.

cannot avoid the unsafe set and, at time $t = 210$, it is at $(x, y, \theta) = (-0.69, 0.43, 0.51)$ and there is no permissive control input so that the simulation quits. Thus, the safe control fails if we do not compensate the computation delay.

6. Conclusions

We proposed a safe controller with compensation of the computation delays of nonlinear discrete-time systems using Mita's method and the shield synthesis. The shield injects the control input candidate computed by the controller if it is a permissive control input, but otherwise replaces it by a permissive control input.

Recently, the shield is applied to safe reinforcement learning [29]. It is future work to extend the proposed method to safe reinforcement learning of nonlinear systems whose models include uncertainties.

Acknowledgements

This work was supported in part by Nanzan University Pache Research Subsidy I-A-2 for the 2024 academic year.

References

- [1] R. Alur, *Principles of Cyber-Physical Systems*, MIT Press, 2015.
- [2] M. Nagumo, "Über die Lage der Integralkurven gewöhnlicher Differentialgleichungen," *Proc. of Physico-Mathematical Society of Japan*, vol. 24, pp. 551-559, 1942.
- [3] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, pp. 1747-1767, 1999.
- [4] F. Blanchini and S. Miani, *Set-Theoretic Methods in Control*, 2nd Edition, Birkhäuser, 2015.
- [5] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Automatic Control*, vol. 62, no. 8, pp. 3861-3876, 2017.
- [6] A. D. Ames, A. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: theory and applications," in *Proc. of 2019 18th European Control Conference*, pp. 3420-3431,

- 2019.
- [7] S. Kolathaya and A. D. Ames, "Input-to-state safety with control barrier functions," *IEEE Control Systems Letters*, vol. 3, no. 1, pp. 108-113, 2019.
 - [8] L. Lindemann and D. D. Dimarogonas, "Control barrier functions for signal temporal logic tasks," *IEEE Control Systems Letters*, vol. 3, no. 1, pp. 96-101, 2019.
 - [9] Z. Wu, F. Albalawi, Z. Zhang, J. Zhang, H. Durand, and P. D. Christofides, "Control Lyapunov-barrier function-based model predictive control of nonlinear systems," *Automatica*, vol. 109, Article no. 108508, pp. 1-10, 2019.
 - [10] G. Yang, C. Belta, and R. Tron, "Self-triggered control for safety critical systems using control barrier functions," in *Proc. of 2019 American Control Conference*, pp. 4454-4459, 2019.
 - [11] A. J. Taylor, P. Ong, J. Cortés, and A. D. Ames, "Safety-critical event triggered control via input-to-state safe barrier functions," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 749-754, 2021
 - [12] J. Choi, F. Castaneda, C. J. Tomlin, and K. Sreenath, "Reinforcement learning for safety-critical control under model uncertainty, using control Lyapunov function and control barrier functions," in *Proc. of 2020 Robotics: Science and Systems*, 2020.
 - [13] W. S. Cortez, D. Oetomo, C. Manzie, and P. Choong, "Control barrier functions for mechanical systems: theory and application to robotic grasping," *IEEE Trans. Control Systems Technology*, vol. 29, no. 2, pp. 530-545, 2021.
 - [14] A. Agrawal and K. Sreenath, "Discrete control barrier functions for safety-critical control of discrete systems with application to bipedal robot navigation," in *Proc. of 2017 Robotics: Science and Systems*, 2017.
 - [15] J. Zheng, B. Zhang, and K. Sreenath, "Safety-critical model predictive control with discrete-time control barrier functions," in *Proc. of 2021 American Control Conference*, pp. 3882-3889, 2021.
 - [16] P. Jagtap, S. Soudjani, and M. Zamani, "Formal synthesis of stochastic systems via control barrier certificates," *IEEE Trans. Automatic Control*, vol. 66, no. 7, pp. 3097-3110, 2021.
 - [17] M. Khajenejad, M. Cavorsi, R. Niu, Q. S. Sze, and Z. Yong, "Tractable compositions of discrete-time control barrier functions with application to driving safety control," in *Proc. of 2021 European Control Conference*, pp. 1303-1309, 2021.
 - [18] M. Mizoguchi and T. Ushio, "Abstraction-based symbolic control barrier functions for safety-critical embedded systems," *IEEE Control Systems Letters*, vol. 6, pp. 1436-1441, 2022.
 - [19] M. Mizoguchi and T. Ushio, "Abstraction-based control under quantized observation with approximate opacity using symbolic control barrier functions," *IEEE Control Systems Letters*, vol. 6, pp. 2222-2227, 2022.
 - [20] X. Tan, W. S. Cortez, and D. V. Dimarogonas, "High-order barrier functions: robustness, safety, and performance-critical control," *IEEE Trans. Automatic Control*, vol. 67, no. 6, pp. 3021-3028, 2022.
 - [21] W. Xiao and C. Belta, "High-order control barrier functions," *IEEE Trans. Automatic Control*, vol. 67, no. 7, pp. 3655-3662, 2022.
 - [22] J. Lin, D-H. Zhai, Y. Xiong, and Y. Xia, "Safety control for UR-type robotic manipulators via high-order control barrier functions and analytical inverse kinematics," *IEEE Trans. Industrial Electronics*, vol. 71, no. 6, pp. 6150 - 6160, 2024.
 - [23] H. Ma, X. Zhang, S. E. Li, Z. Lin, Y. Lyu, and S. Zheng, "Feasibility enhancement of constrained receding horizon control using generalized control barrier function," in *Proc. of 4th IEEE International Conference on Industrial Cyber-Physical Systems*, pp. 551-557, 2021.
 - [24] Y. Xiong, D-H. Zhai, M. Tavakoli, and Y. Xia, "Discrete-time control barrier function: high-order case and adaptive case," *IEEE Trans. Cybernetics*, vol. 53, no. 5, pp. 3231-3239, 2023.
 - [25] S. Liu, J. Zweng, K. Sreenath, and C. A. Belta, "Iterative convex optimization for model predictive control with discrete-time high-order control barrier functions," in *Proc. of 2023 American Control Conference*, pp. 3368-3375, 2023.
 - [26] B. Könighofer, M. Alshiekh, R. Bloem, L. Humphrey, R. Könighofer, U. Topucu, and C. Wang, "Shield Synthesis," *Formal Methods Syst. Des.*, vol. 51, pp. 332-361, 2017.
 - [27] M. Krstic, "Inverse optimal safety filters," *IEEE Trans. Automatic Control*, vol. 69, no. 1, pp. 16-31, 2024.
 - [28] S. Pranger, B. Könighofer, M. Tappler, M. Deixelberger, N. Jansen, and R. Bloem, "Adaptive shielding under uncertainty," in *Proc. of 2021 American Control Conference*, pp. 3458-3465, 2021.
 - [29] M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, and U. Topucu, "Safe reinforcement learning via shielding," in *Proc. of 32nd AAAI Conference on Artificial Intelligence*, pp. 2669-2678, 2018.
 - [30] O. Bastani, "Safe reinforcement learning with nonlinear dynamics via model predictive shielding," in *Proc. of 2021 American Control Conference*, pp. 3479-3485, 2021.
 - [31] P. K. Pandya and A. Wakankar, "Specification and optimal reactive synthesis of run-time enforcement shields," *Information and Computation*, vol. 285, article no. 104865, pp. 1-18, 2022.
 - [32] K. Kanashima and T. Ushio, "Finite-horizon shield for path planning ensuring safety/co-safety specifications and security policies," *IEEE Access*, vol. 11, pp. 11766-11780, 2023.
 - [33] M. Mizoguchi and T. Ushio, "Abstraction-based safe control with alternating simulation-based shields and its Application to mobile robots," *IEEE Access*, vol. 12, pp. 37152-37164, 2024.
 - [34] K. G. Shin and X. Cui, "Computing time delay and its effects on real-time control systems," *IEEE Trans. Control Systems Technology*, vol. 3, no. 2, pp. 218-224, 1995.
 - [35] T. Mita, "Optimal digital feedback control systems counting computation time of control laws," *IEEE Trans. Automatic Control*, vol. AC-30, no. 6, pp. 542-548, 1985.
 - [36] P. Ramanathan, "Overload management in real-time control applications using (m, k) -firm guarantee," *IEEE Trans. on Parallel and Distributed Systems*, vol. 10, no. 6, pp. 549-559, 1999.
 - [37] T. Yoshimoto and T. Ushio, "Optimal arbitration of control tasks by job skipping in cyber-physical systems," in *Proc. of IEEE/ACM 2nd International Conference on Cyber-Physical Systems*, pp. 55-64, 2011.
 - [38] T. Fujita and T. Ushio, "Optimal digital control with uncertain network delay of linear systems using reinforcement learning," *IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E99-A, no. 2, pp. 454-461, 2016.
 - [39] Y. Masui, K. Hirata, and T. Hagiwara, "Modified state predictive control of continuous-time systems with input delay," in *Proc. of 2017 IEEE International Conference on Industrial Technology*, pp. 843-847, 2017.
 - [40] S. Monaco and D. Normand-Cyrot, "Minimum-phase nonlinear discrete-time systems and feedback stabilization," in *Proc. of 28th Conference on Decision and Control*, pp. 979-986, 1987.
 - [41] M. Sun and D. Wang, "Initial shift issues on discrete-time iterative learning control with system relative degree," *IEEE Trans. Automatic Control*, vol. 48, no. 1, pp. 144-148, 2003.



Toshimitsu Ushio received B.S., M.S. and Ph.D degrees in 1980, 1982 and 1985, from Kobe University. He was a Research Assistant at the UC Berkeley in 1985. From 1986 to 1990, he was a Research Associate in Kobe University. He joined Osaka University in 1994, and was a Professor in 1997. He joined Nanzan University as a professor in 2023. His research interests include control of discrete event and hybrid systems and analysis of nonlinear systems. He is a member of IEEE.