2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Greedy Synthesis of Event- and Self-Triggered Controls with Control Lyapunov-Barrier Function

Masako Kishida, *Senior Member*

***Abstract*— This paper addresses the co-design problem of control inputs and execution decisions for event- and self-triggered controls subject to constraints given by the control Lyapunov function and control barrier function. The proposed approach computes the control input in a way that allows for longer inter-execution intervals, which distinguishes it from many existing event- and self-triggered controllers or control Lyapunov-barrier function controllers. The proposed approach guarantees lower bounds on the minimum inter-execution times. The effectiveness of the proposed approach is demonstrated and compared with existing approaches using a numerical example.**

I. INTRODUCTION

Control Barrier Function (CBF) [1] is a function used to design control inputs to satisfy safety requirements. As the use of automatic control increases in safety-critical systems, CBF is attracting much attention recently. Applications of CBF can be found such as in robotics [2] and adaptive cruise control [3]. Furthermore, CBF has been combined with signal temporal logic [4], model predictive control [5] and extended to assuring risk-sensitive safety [6], making it more versatile for a wider range of applications. On the other hand, Control Lyapunov Function (CLF) [7], [8], an extension of the Lyapunov function, has been widely used to design stabilizing controllers for many different problems (see e.g., [9]–[12]). After an integration of CLF and CBF was proposed in [13], [14], it has been shown that CLF and CBF can be combined to form a quadratic program (QP) for computing control inputs that ensure safety while aiming at stability in [15], [16]. Since then, the CLF-CBF QP approach has been used to solve various problems, such as autonomous surface vehicles [17] and safe stabilization [18].

With the increasing prevalence of networked control systems, where the communication bandwidth is shared with other tasks or batteries are used in the system elements, it has become crucial to design systems that use communication bandwidth and energy efficiently. Event- and self-triggered control approaches are effective solutions to minimize unnecessary communication and energy consumption for such systems [19]–[21].

Several approaches have been proposed for safety-critical systems using event- or self-triggered strategies. An event-triggered control based on input-to-state safe barrier functions by using a state feedback control law $u = k(x)$ was proposed in [22]. The approach is to bound the difference between the current state and the state used to compute the control input, i.e., the state at the previous execution time instance, to guarantee that the value of the barrier function monotonically decreases. More recently, [23] proposed an approach of event-triggered control for multi-agent systems with unknown dynamics. It synthesizes an event-triggered control with an adaptive affine dynamics that are updated based on the error states to estimate the real system state. A combination of self-triggered control and CLF-CBF QP was proposed in [24]; however, it does not seem to guarantee the existence of the control input. Moreover, it possibly results in continuous control updates if the optimal control input is achieved on the boundary of the constraint of the QP.

The main contribution of this paper is to introduce approaches to co-designing control inputs and execution time instances for event- and self- triggered controls, with the aim of meeting the constraints given by the CLF and CBF while reducing the number of executions. This is achieved by computing the control input so as to obtain long inter-execution intervals in a greedy manner. The approach is different from many existing event- and self-triggered controllers that rely on constant feedback laws [20] or control Lyapunov-barrier function controllers that account for only the cost of control inputs. It is also shown that the proposed approach does not exhibit Zeno behavior and that the optimization parameters appear in a lower bound on the minimum inter-execution time.

The rest of the paper is organized as follows. After introducing the notation, system model, and basics of control Lyapunov-barrier function in Section II, Section III presents the proposed event-triggered control approach. Section IV discusses the extension to the self-triggered control approach. The performances of those proposed controllers are illustrated and compared with existing controllers in Section V, which is followed by conclusions in Section VI.

II. PRELIMINARIES

A. Notation

The sets of real numbers, real vectors of length n, and real matrices of size $n \times m$ are denoted by $\mathbb{R}$, $\mathbb{R}^n$, and $\mathbb{R}^{n\times m}$, respectively. The sets of nonnegative numbers and nonnegative integers are denoted by $\mathbb{R}_{\geq 0}$ and $\mathbb{N}$, respectively.

$L_f V(x)$ denotes the Lie derivative of $V(x)$ along the vector field $f(x)$, i.e. $L_f V(x) = \frac{\partial V(x)}{\partial x} f(x)$.

A continuous function $\gamma : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\gamma(0) = 0$. A continuous function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to belong to class $\mathcal{K}_\infty$ if it belongs to class $\mathcal{K}$ and $\alpha(r) \to \infty$ as $r \to \infty$.

*This work was supported by JST, PRESTO Grant Number JP-MJPR22C3, Japan.

Masako Kishida is with the National Institute of Informatics, Tokyo 101-8430, Japan kishida@nii.ac.jp

B. System model

This paper deals with a nonlinear control affine system in the form of

$$\dot{x} = f(x) + g(x)u, \; x(0) = x_0 \in \mathcal{C}, \tag{1}$$

where $x \in \mathcal{D} \subset \mathbb{R}^n$ and $u \in \mathbb{R}^m$ denote the state and the control input of the system, respectively, and $\mathcal{C} \subseteq \mathcal{D}$ is a safe set that is defined later. It is assumed that $\mathcal{D}$ is bounded, and the functions $f(x)$ and $g(x)$ are Lipschitz.

C. Control Lyapunov function

Definition 2.1 (Control Lyapunov Function): A positive definite function $V : \mathcal{D} \to \mathbb{R}_{\geq 0}$ is called a Control Lyapunov Function (CLF) if it satisfies

$$\inf_{u \in \mathcal{U}} L_f V(x) + L_g V(x) u \leq -\gamma(V(x)), \tag{2}$$

where γ is a class $\mathcal{K}$ function.
The existence of a CLF guarantees asymptotic stabilization of the nonlinear control system (1) with any Lipschitz continuous feedback controller $u(x)$ that satisfies (2) for all $x \in \mathcal{D}$ [16], [25].

D. Control barrier function

Let define the safe set $\mathcal{C}$ as the superlevel set of a continuously differentiable function $h : \mathcal{D} \subset \mathbb{R}^n \to \mathbb{R}$

$$\mathcal{C} = \{x \in \mathcal{D} : h(x) \geq 0\}. \tag{3}$$

Definition 2.2 (Control Barrier Function): A function $h : \mathcal{D} \to \mathbb{R}$ is called a Control Barrier Function (CBF) if it satisfies

$$\sup_{u \in \mathcal{U}} L_f h(x) + L_g h(x) u \geq -\alpha(h(x)) \tag{4}$$

for all $x \in \mathcal{D}$, where α is a class $\mathcal{K}_\infty$ function.
The existence of a CBF guarantees that the control system is safe [15].

E. CLF-CBF-based QP

Motivated by the results on CLF and CBF, it is of interest to obtain a controller that satisfies

$$L_f V(x) + L_g V(x) u \leq -\gamma(V(x)), \tag{5}$$
$$L_f h(x) + L_g h(x) u \geq -\alpha(h(x)) \tag{6}$$

for all $x \in \mathcal{D}$ so that it is a safe stabilizing controller.

To design such a controller, a CLF-CBF QP was constructed in [15], [26]:

$$\begin{aligned} \mathbf{v}^*(x) = & \underset{\mathbf{v}=[u,\delta]^\top \in \mathbb{R}^{m+1}}{\arg\min} \frac{1}{2}\mathbf{v}^\top Q \mathbf{v} + c^\top \mathbf{v} \\ \text{s.t.} \quad & \begin{bmatrix} L_g V(x) & -1 \\ -L_g h(x) & 0 \end{bmatrix} \mathbf{v} \leq \begin{bmatrix} b_{\text{clf}}(x) \\ b_{\text{cbf}}(x) \end{bmatrix}, \end{aligned} \tag{7}$$

where

$$\begin{aligned} b_{\text{clf}}(x) &= -L_f V(x) - \gamma(V(x)), \\ b_{\text{cbf}}(x) &= L_f h(x) + \alpha(h(x)), \end{aligned} \tag{8}$$

a positive definite matrix $Q \in \mathbb{R}^{(m+1)\times(m+1)}$ and $c \in \mathbb{R}^{m+1}$ are weights and δ is a relaxation variable that ensures the solvability of the QP. If δ is forced to be nonpositive, then the existence of a feasible controller will guarantee the monotonic decrease of the CLF.

III. EVENT-TRIGGERED CONTROL

This section proposes a greedy event-triggered control with the control Lyapunov-barrier function for the system (1).

A. Event-triggered controller structure

The primary idea behind event-triggered control is to update the control input only when necessary to achieve a specified performance condition, thereby reducing the frequency of updates. In line with the standard structure of an event-triggered controller, we consider the control inputs that are maintained constant between successive event times, i.e.,

$$u(t) = u_k, \; t \in [t_k, t_{k+1}), \tag{9}$$

where u_k is the control input computed at time t_k, which is the time instance when the control input is re-computed and the actuator signals are updated. The time instance t_k is determined by

$$\begin{aligned} &t_0 = 0, \\ &t_{k+1} = \inf\{t \in \mathbb{R} : t > t_k \text{ and trigger condition is met}\}. \end{aligned} \tag{10}$$

B. Greedy control update

Here, we introduce a greedy approach for computing the control input u_k at trigger time t_k. To implement it into an event-triggered control, we are interested in a control law that maximizes the inter-execution time. For this purpose, we seek the control input that brings the state away from the boundaries of the constraints (5), (6). This is achieved by maximizing the slack variables ρ_1 and ρ_2 in the new constraints:

$$\begin{aligned} L_g V(x) u + \rho_1 &\leq b_{\text{clf}}(x), \\ -L_g h(x) u + \rho_2 &\leq b_{\text{cbf}}(x), \end{aligned} \tag{11}$$

where $b_{\text{clf}}(x)$ and $b_{\text{cbf}}(x)$ are defined in (8).

Based on the constraints (11), we propose to modify the CLF-CBF QP in (7) as follows:

$$\begin{aligned} \mathbf{v}^*(x) = & \underset{\mathbf{v}=[u,\rho_1,\rho_2]^\top \in \mathbb{R}^{m+2}}{\arg\min} \frac{1}{2}\mathbf{v}^\top Q \mathbf{v} + c^\top \mathbf{v} \\ \text{s.t.} \quad & \begin{bmatrix} L_g V(x) & 1 & 0 \\ -L_g h(x) & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \mathbf{v} \leq \begin{bmatrix} b_{\text{clf}}(x) \\ b_{\text{cbf}}(x) \\ -\varepsilon_{\text{cbf}} \end{bmatrix}, \end{aligned} \tag{12}$$

where a positive semidefinite matrix $Q = Q^\top \in \mathbb{R}^{(m+2)\times(m+2)}$ and $c \in \mathbb{R}^{m+2}$ are weights, ρ_1 and ρ_2 are slack variables, and $\varepsilon_{\text{cbf}} > 0$ is a constant (design parameter) that forces ρ_2 be sufficiently large in (11), thus guarantees that the states to be away from the safety boundary $-L_g h(x) u = b_{\text{cbf}}(x)$. This is still QP and the solvability of the QP is still ensured as long as ρ_1 is not constrained. Let $[u^*(x), \rho_1^*(x), \rho_2^*(x)]^\top = \mathbf{v}^*(x)$.

Typically, a small norm control input u is desired to minimize the control effort while large ρ_1 and ρ_2 are desired to have a longer the inter-execution time. Hence, a possible choice for Q and c is

$$Q = \begin{bmatrix} w_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ c = \begin{bmatrix} 0 & -w_2 & -w_3 \end{bmatrix}^\top, \qquad (13)$$

where $w_1 \in \mathbb{R}^{m\times m}$ is positive definite, and $w_2, w_3 \geq 0$.

With (12), the proposed controller implements the control input u_k defined by

$$u_k = u^*(x_k) = \begin{bmatrix} \mathbf{1}^\top & 0 & 0 \end{bmatrix} \mathbf{v}^*(x_k), \qquad (14)$$

where $x_k = x(t_k)$.

C. Trigger conditions

For the states to remain in the safe region, the safety constraint (6) should be always satisfied. However, the satisfaction of the stability constraint (5) cannot be guaranteed together with the satisfaction of (6) in general. This is the same for the proposed controller. Yet, we do our best to minimize the time in which the stability constraint (5) is violated.

Define

$$p(x) = p_k(x),\ t \in [t_k, t_{k+1}), \qquad (15)$$
$$q(x) = q_k(x),\ t \in [t_k, t_{k+1}), \qquad (16)$$

where

$$p_k(x) = -L_f V(x) - L_g V(x) u_k, \qquad (17)$$
$$q_k(x) = L_g h(x) u_k + b_{\text{cbf}}(x). \qquad (18)$$

Note that $\dot{V}(x) = -p_k(x)$, thus $p_k(x) \geq 0$ is desired for the stability, while $q_k(x) \geq 0$ is desired for the safety.

We set the trigger condition in (10) as

1) if $p_k(x_k) \geq \varepsilon_{\text{clf}}$:

$$p_k(x) = 0 \text{ or } q_k(x) = 0 \qquad (19)$$

2) else:

$$q_k(x) = 0 \text{ or } t = t_k + \tau_{bd} \qquad (20)$$

where $\varepsilon_{\text{clf}} > 0$ and $\tau_{bd} > 0$ are design parameters.

The first case is that, if $\dot{V}$ is sufficiently negative at the time of update t_k, then the next update time is when either the time-derivative of the CLF becomes zero or the safety constraint (6) is satisfied with equality. This guarantees the satisfaction of both stability and safety between t_k and t_{k+1}.

The second case is that, if $\dot{V}$ is close to zero or positive at the time of update t_k, then we compromise the controller design only focusing on the safety constraint. The next update time is when the safety constraint (6) is satisfied with equality or small time τ_{bd} passes, whichever occurs first. This limits the duration of time during which the stability constraint is violated to τ_{bd} with the same control input.

D. Lower bound on minimum inter-execution time

This subsection shows that the events cannot be triggered an infinite number of times in any finite time period with the proposed controller, i.e., the proposed controller is Zeno-free under mild conditions.

Define the inter-execution time

$$\tau_k = t_{k+1} - t_k,\ k \in \mathbb{N}. \qquad (21)$$

It is of interest to show the existence of a lower bound τ^* such that $\tau_k \geq \tau^*$ for all $k \in \mathbb{N}$.

Assumption 3.1: We assume the followings hold on $\mathcal{D}$:

- $\|f(x) + g(x)u\|$ is bounded above, and
- There exists a Lipschitz constant $L_{\text{clf}} > 0$ for $p(x)$
- There exists a Lipschitz constant $L_{\text{cbf}} > 0$ for $q(x)$

If the problem is considered in a finite time horizon, the first assumption is sufficient. There are reasonable assumptions; the first assumption only requires that the time derivative of the state, $\dot{x}$, is bounded, the second and third assumptions asks for bounded Lie derivatives, bounded $h(x)$ and bounded control inputs.

Theorem 3.2: Under Assumption 3.1, for the time sequence (10) for the system (1) with the event-triggered controller (9), (14) with the trigger condition (19)-(20), there exists $\tau^* > 0$ such that $\tau_k \geq \tau^*$ for all $k \in \mathbb{N}$.

Proof: First, we consider if $p_k(x_k) \geq \varepsilon_{\text{clf}}$, then how long it takes to achieve

$$p_k(x) = 0 \qquad (22)$$

for the first time after t_k. Let this time instance be $\bar{t}_{\text{clf}}$ and the corresponding state be $\bar{x}_{\text{clf}}$.

For this, we first show that

$$\|\bar{x}_{\text{clf}} - x_k\| \geq \frac{\varepsilon_{\text{clf}}}{L_{\text{clf}}}. \qquad (23)$$

Clearly,

$$p_k(x_k) - p_k(\bar{x}_{\text{clf}}) \geq \varepsilon_{\text{clf}}, \qquad (24)$$

and

$$L_{\text{clf}}\|x_k - \bar{x}_{\text{clf}}\| \geq p_k(x_k) - p_k(\bar{x}_{\text{clf}}). \qquad (25)$$

Together, it follows that

$$\|\bar{x}_{\text{clf}} - x_k\| \geq \frac{\varepsilon_{\text{clf}}}{L_{\text{clf}}}. \qquad (26)$$

Next, we show that the time difference $\bar{t}_{\text{clf}} - t_k$ is lower bound away from zero. By the fundamental theorem of calculus, we have

$$\begin{aligned} \|\bar{x}_{\text{clf}} - x_k\| &= \left\| \int_{t_k}^{\bar{t}_{\text{clf}}} \dot{x}(\tau) d\tau \right\| \\ &\leq \sup \|f(x) + g(x)u\| (\bar{t}_{\text{clf}} - t_k). \end{aligned} \qquad (27)$$

Because $\|f(x)+g(x)u\|$ is bounded above, there exists $M > 0$ such that $M = \sup\|f(x) + g(x)u\|$, then it follows that

$$\bar{t}_{\text{clf}} - t_k \geq \frac{1}{M}\|\bar{x}_{\text{clf}} - x_k\| \geq \frac{\varepsilon_{\text{clf}}}{M L_{\text{clf}}}. \qquad (28)$$

Thus, at least the time interval of $\frac{\varepsilon_{\text{clf}}}{M L_{\text{clf}}}$ takes to achieve (22).

Similarly, we consider how long it takes to achieve

$$q_k(x) = 0 \tag{29}$$

for the first time after t_k. Let this time instance $\bar{t}_{\text{cbf}}$ and the corresponding state $\bar{x}_{\text{cbf}}$.

Using the fact that $q_k(x_k) \geq \rho_2^*(x_k) \geq \varepsilon_{\text{cbf}}$, we can show that

$$\|\bar{x}_{\text{cbf}} - x_k\| \geq \frac{\varepsilon_{\text{cbf}}}{L_{\text{cbf}}} \tag{30}$$

and then

$$\bar{t}_{\text{cbf}} - t_k \geq \frac{1}{M}\|\bar{x}_{\text{cbf}} - x_k\| \geq \frac{\varepsilon_{\text{cbf}}}{ML_{\text{cbf}}}. \tag{31}$$

In summary, the inter-execution time is lower bounded by $\tau^* = \min(\frac{\varepsilon_{\text{clf}}}{ML_{\text{clf}}}, \frac{\varepsilon_{\text{cbf}}}{ML_{\text{cbf}}}, \tau_{bd})$, which is strictly positive. This completes the proof. ■

Here, we see the parameter $\varepsilon_{\text{cbf}} > 0$ in (12) can be used as a parameter to tune the inter-execution time together with other design parameters ε_{clf} and τ_{bd}.

E. Special cases

1) Feasibility is guaranteed: Suppose that the feasibility of the constraints (5) and (6) is guaranteed for all $x \in \mathcal{D}$ with some margins, i.e., there exist $s_1, s_2 > 0$ such that for all $x \in \mathcal{D}$, there exists an control input u that depends on x that satisfies

$$\begin{aligned} L_gV(x)u + s_1 &\leq b_{\text{clf}}(x), \\ -L_gh(x)u + s_2 &\leq b_{\text{cbf}}(x). \end{aligned} \tag{32}$$

Then, we may consider not only forcing the relaxation variable δ to be 0 in (7), but securing certain distances from the boundaries of the constraints. This allows us to add a constraint on ρ_1 in (12) and the resulting QP is:

$$\begin{aligned} \mathbf{v}^*(x) = &\underset{\mathbf{v}=[u,\rho_1,\rho_2]^\top \in \mathbb{R}^{m+2}}{\arg\min} \frac{1}{2}\mathbf{v}^\top Q\mathbf{v} + c^\top \mathbf{v} \\ &\text{s.t.} \begin{bmatrix} L_gV(x) & 1 & 0 \\ -L_gh(x) & 0 & 1 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \mathbf{v} \leq \begin{bmatrix} b_{\text{clf}}(x) \\ b_{\text{cbf}}(x) \\ -\varepsilon_1 \\ -\varepsilon_2 \end{bmatrix}, \end{aligned} \tag{33}$$

where $\varepsilon_1 \in (0, s_1), \varepsilon_2 \in (0, s_2)$ are constants, a positive semidefinite matrix $Q \in \mathbb{R}^{(m+2)\times(m+2)}$, and $c \in \mathbb{R}^{m+2}$ are the weights. In this case, the trigger condition (19)-(20) can be combined and modified as:

$$(L_gV(x)u - b_{\text{clf}}(x))(L_gh(x)u + b_{\text{cbf}}(x)) = 0 \tag{34}$$

to guarantee both stability and safety.

2) Control input is not penalized: In addition that the feasibility is guaranteed, if the control input u is not penalized as in other event-triggered controls, then, the problem simplifies to solving a linear program:

$$\begin{aligned} u^*(x) = &\underset{u \in \mathbb{R}^m}{\arg\min} \left(-w_2L_gV(x) - w_3L_gh(x)\right)u \\ &\text{s.t.} \begin{bmatrix} L_gV(x) \\ -L_gh(x) \end{bmatrix} u \leq \begin{bmatrix} b_{\text{clf}}(x) - \varepsilon_1 \\ b_{\text{cbf}}(x) - \varepsilon_2 \end{bmatrix}, \end{aligned} \tag{35}$$

where, again, $\varepsilon_1 \in (0, s_1), \varepsilon_2 \in (0, s_2)$ are constants, using some weights $w_2, w_3 \geq 0$.

IV. Self-triggered control

This section develops a greedy self-triggered control based on the results in Section III.

A. Self-triggered controller structure

As in Section III, let t_k be the triggering time instance. Self-triggered control computes the control input u_k as well as the next time instance t_{k+1} at execution time t_k. Similar to the event-triggered control (9), the control input u_k remains constant in between t_k and t_{k+1}. Unlike the event-triggered condition, however, no sampling or computation is required between t_k and t_{k+1}.

As for the event-triggered control, the controller implements the control input u_k in (14) by solving (12).

The next execution time t_{k+1} is computed based on the measured state $x(t_k) = x_k$ at t_k:

$$\begin{aligned} t_0 &= 0, \\ t_{k+1} &= t_k + \Gamma(x_k), \end{aligned} \tag{36}$$

where the map $\Gamma : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ determines the triggering time t_{k+1} as a function of the state x_k at the time t_k. Thus, the inter-execution time is given by $\Gamma(x)$, i.e., $\tau_k = \Gamma(x_k)$.

In the following, we present approaches to how to design the map $\Gamma(x)$.

B. Computing the next execution time instance

In Section III, the trigger condition (19)-(20) is designed to satisfy the safety constraint (6) and minimize the violation of the stability constraint (5). Here, again, we design the map Γ that satisfies the safety constraint (6) all the time, while allowing the violation of the stability constraint:

$$\begin{aligned} \Gamma(x_k) \leq \sup\{&t > t_k : \\ &\text{if } p_k(x_k) \geq \varepsilon_{\text{clf}} : \\ &\quad p_k(x(\tau)) > 0 \text{ and } q_k(x(\tau)) > 0 \text{ for all } \tau < t \\ &\text{else:} \\ &\quad q_k(x(\tau)) > 0 \text{ for all } \tau < t \text{ and } t \leq t_k + \tau_{bd}\} \\ &- t_k. \end{aligned} \tag{37}$$

This basically requires the same conditions as for (19)-(20).

Ideally, we would like to find t that achieves an equality in (37). However, it is difficult in general for nonlinear systems, thus, we aim at finding a lower bound by revising the approach in Section III-D.

Again, we assume that Assumption 3.1 is satisfied. Then, we may use the following map:

Theorem 4.1: Under Assumption 3.1,

$$\Gamma(x_k) = \begin{cases} \min\left(\frac{\varepsilon_{\text{clf}}}{L_{\text{clf}}M_k}, \frac{\varepsilon_{\text{cbf}}}{L_{\text{cbf}}M_k}\right), & \text{if } p_k(x_k) \geq \varepsilon_{\text{clf}} \\ \min\left(\frac{\varepsilon_{\text{cbf}}}{L_{\text{cbf}}M_k}, \tau_{bd}\right), & \text{otherwise,} \end{cases} \tag{38}$$

where

$$M_k = \sup_{t \in [t_k, t_k + \Gamma(x_k))} \|f(x) + g(x)u_k\| \tag{39}$$

satisfies (37).

Proof: Here, again note that after solving (12), it is guaranteed that

$$q_k(x_k) \geq \rho_2^*(x_k) \geq \varepsilon_{\text{cbf}}. \tag{40}$$

For the first case of $p_k(x_k) \geq \varepsilon_{\text{clf}}$, we show

$$\Gamma(x_k) = \min\left(\frac{\varepsilon_{\text{clf}}}{L_{\text{clf}} M_k}, \frac{\varepsilon_{\text{cbf}}}{L_{\text{cbf}} M_k}\right), \tag{41}$$

satisfies (37). By the fundamental theorem of calculus, the equation (41) implies that for $t \in [t_k, t_k + \Gamma(x_k))$,

$$\begin{aligned} \|x - x_k\| &\leq \int_{t_k}^{t} \|f(x) + g(x)u_k\| dt \\ &\leq (t - t_k) M_k \\ &\leq (t - t_k) \frac{\min\left(\frac{\varepsilon_{\text{clf}}}{L_{\text{clf}}}, \frac{\varepsilon_{\text{cbf}}}{L_{\text{cbf}}}\right)}{\Gamma(x_k)} \\ &< \min\left(\frac{\varepsilon_{\text{clf}}}{L_{\text{clf}}}, \frac{\varepsilon_{\text{cbf}}}{L_{\text{cbf}}}\right) \\ \Leftrightarrow &\begin{cases} \varepsilon_{\text{clf}} - L_{\text{clf}} \|x - x_k\| > 0, \\ \varepsilon_{\text{cbf}} - L_{\text{cbf}} \|x - x_k\| > 0. \end{cases} \end{aligned} \tag{42}$$

On the other hand,

$$\begin{aligned} \varepsilon_{\text{clf}} - p_k(x) &\leq p_k(x_k) - p_k(x) \leq L_{\text{clf}} \|x - x_k\|, \\ \varepsilon_{\text{cbf}} - q_k(x) &\leq q_k(x_k) - q_k(x) \leq L_{\text{cbf}} \|x - x_k\|. \end{aligned} \tag{43}$$

Hence, it follows that

$$p_k(x) > 0, \; q_k(x) > 0. \tag{44}$$

The second case is clear from the above discussions. Together, it completes the proof. ■

The difference between this map Γ and the lower bound τ^* in the event-triggered control is only the upper bound on the norm of $f(x) + g(x)u$. Although a uniform M may be used to design a constant $\Gamma = \tau^*$, such a law will shorten the inter-execution time because M is likely to be much larger than M_k. However, one difficulty of implementing this approach is actually in the computation of M_k in (47) where M_k appears in both sides of the equation. To compute exact M_k, it is required to compute the evolution of (1) starting at t_k by gradually increasing the time duration. In the actual implementation, this might not be a big problem because implementation is done digitally, which we discuss next.

C. Digital implementation

Similarly to [20], we now consider the following discrete-time versions of $p_k(x)$ and $q_k(x)$ based on a sampling time $\Delta > 0$, which are defined by

$$\begin{aligned} \bar{p}_k(n) &= p_k(x(t_k + n\Delta)), \\ \bar{q}_k(n) &= q_k(x(t_k + n\Delta)). \end{aligned} \tag{45}$$

Let $\tau_{\min}$ and $\tau_{\max}$ be design parameters and let $N_{\min} = \lfloor \tau_{\min}/\Delta \rfloor$, $N_{\max} = \lfloor \tau_{\max}/\Delta \rfloor$.

Then, by choosing $\tau_{bd} = \Delta$, the map can be simplified as

$$\Gamma(x_k) = \max\{\tau_{\min}, n(x_k)\Delta\}, \tag{46}$$

where

$$\begin{aligned} n(x) = \max_{n \in \mathbb{N}} \{ n \leq N_{\max} : \bar{p}_k(m) \geq 0 \text{ and } \bar{q}_k(m) \geq 0, \\ m = 1, \cdots, n \}. \end{aligned} \tag{47}$$

This satisfies

$$\bar{p}_k(n) \geq 0 \text{ and } \bar{q}_k(n) \geq 0, \; \forall n \in \left[0, \left\lceil \frac{t_{k+1} - t_k}{\Delta} \right\rceil \right) \tag{48}$$

and $n(x) \in [N_{\min}, N_{\max}]$.

With a smaller choice of Δ, the stability constraint is more strictly forced, i.e., while the stability constraint is violated, the control input is updated every sampling time. Such a choice of Δ may result in an increase in the number of execution. Of course, it is possible to use different values for τ_{bd} and Δ with a slight modification.

We may simply choose $\tau_{\min} = 0$ and a sufficiently large $\tau_{\max}$. However, the value of $\tau_{\max}$ enforces the robustness of the implementation and limits the computational complexity [20].

V. NUMERICAL EXAMPLE

This section demonstrates the effectiveness of the proposed approach using an example of a double integrator.

Let $x = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^\top$, where x_1 is the position and x_2 is the velocity. The dynamics of a double integrator is given as follows:

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u. \tag{49}$$

The CLF and CBF are selected as

$$\begin{aligned} V(x) &= x_1^2 + x_1 x_2 + x_2^2, \\ h(x) &= (x_1 - 0.5)^2 + (x_2 + 0.5)^2 - 0.3^2. \end{aligned} \tag{50}$$

Moreover, the α and γ functions are chosen to be identity maps and the parameters $\varepsilon_{\text{clf}} = \varepsilon_{\text{cbf}} = 0.1$, $\tau_{bd} = 0.5$, $\Delta = 0.2$, $\tau_{\min} = 0$ and $\tau_{\max} = 4$ are selected. The weights for QP in (12) are selected as

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; c = \begin{bmatrix} 0 & -1 & 0 \end{bmatrix}^\top. \tag{51}$$

Here, the performances of the following five controllers are compared for the duration of time 15, starting at $x_0 = [1, 1]$.

- Greedy ET: the controller that solves (12) and implements (14) when the trigger condition (19) or (20) is met
- Greedy ST: the controller that solves (12) and implements (14) at time instances determined by (36) and (46)
- Greedy: the controller that solves (12) and implements (14) continuously
- CLF-CBF QP: CLF-CBF QP controller in [16] with $H(x) = 2$, $p = 1$
- SF: a standard state-feedback controller with the controller gain $K = [-0.5 \; -1]$, which corresponds to

the case of weight for the states is $\begin{bmatrix} 0.25 & -0.5 \\ -0.5 & 0 \end{bmatrix}$ and weight for the control input is 1, thus $P = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$ in the algebraic Riccati equation. No constraints are considered.

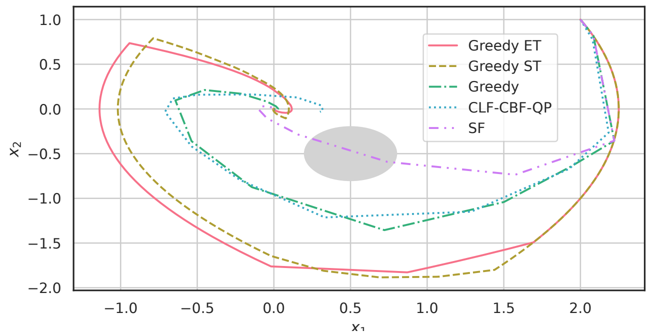


Fig. 1: Phase portrait: the grey region indicates the unsafe region, $h(x) < 0$

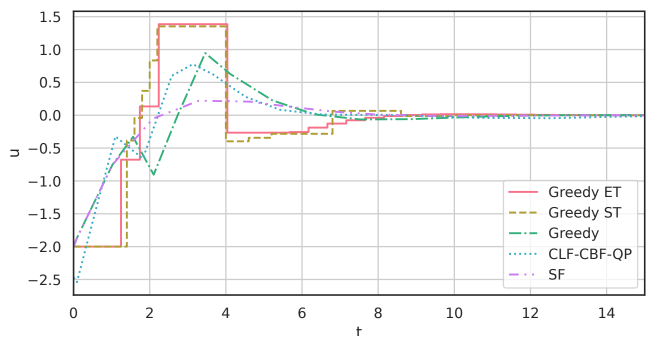


Fig. 2: Control inputs

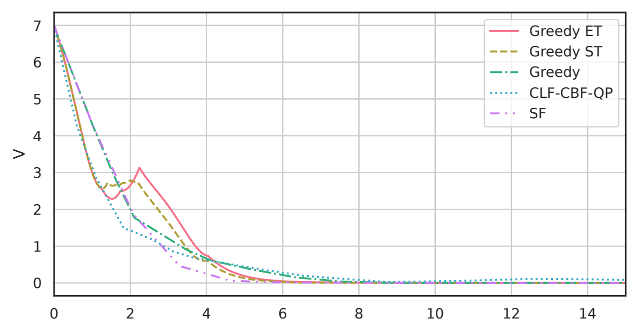
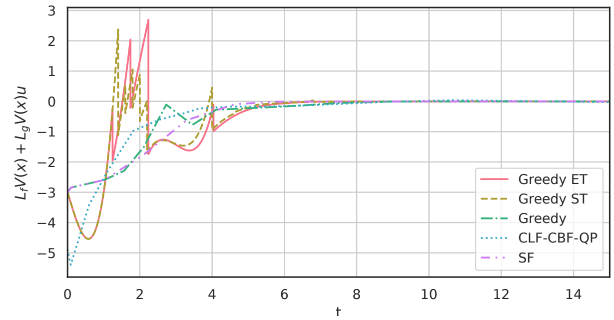


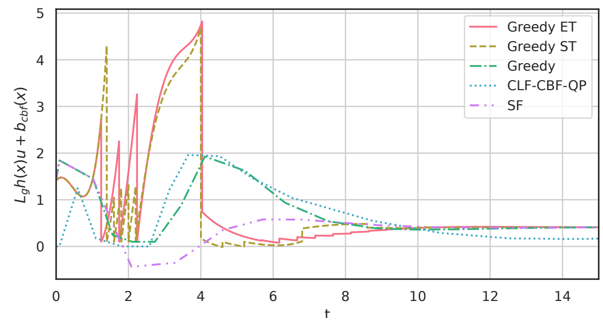
Fig. 3: CLF values, $V(x) = x_1^2 + x_1x_2 + x_2^2$, versus time

Figure 1 shows the phase portraits for the five controllers. It is observed that the trajectory with SF goes into the unsafe region, which motivates us to use the barrier function to remain in the safe region. Also, the trajectories of Greedy and CLF-CBF-QP are close to each other, but the state of CLF-CBF-QP is further from the origin at the end of the simulation compared with the other four methods. Moreover, both trajectories of Greedy ET and Greedy SF take longer paths compared with other methods.

Figure 2 shows the control input trajectories. It can be seen that Greedy ET and Greedy ST do not require frequent



(a) Stability constraint used for trigger condition, $L_fV(x) + L_gV(x)u$. The values are desired to be negative.



(b) Safety constraint used for trigger condition, $L_gh(x)u + b_{cbf}(x)$. The values are desired to be nonnegative.

Fig. 4: Trigger conditions

control updates. In fact, the numbers of control updates for Greedy ET and Greedy ST were 24 and 26, respectively. With Greedy ST, a smaller sampling time Δ forces the derivative of the CLF to be negative more strictly thus tends to increase the update frequency.

Figure 3 shows how the CLF values varies as functions of time. Both Greedy ET and Greedy ST admit increases of the CLF values for certain periods. However, those trajectories approach zero quickly, while CLF-CBF-QP is still away from zero at the end of the simulation. Note that the original CLF-CBF-QP controller also allows increases of the CLF values to guarantee the feasibility of the optimization problem [16].

Figure 4 shows the trajectories of values used for triggers. Because Greedy ET and Greedy ST compromised the stability constraints for the sake of reducing the update frequencies, Figure 4a indicates the values of $L_fV(x) + L_gV(x)u$ go above zero sometimes. On the other hand, the safety constraints are always satisfied by all the controllers except for the SF that ignored the existence of unsafe region in Figure 4b. In Figure 4b, we also observe that the trajectory $L_gh(x)u + b_{cbf}(x)$ of CLF-CBF-QP stays near zero around time 2. Thus, we cannot implement an event-triggered strategy with CLF-CBF-QP because the trigger condition is violated or close to violate already at the time of the update.

VI. CONCLUSIONS

In this paper, we have presented a set of greedy approaches for synthesizing event-triggered and self-triggered controls with the control Lyapunov-Barrier function. Our proposed approach computes each control input to maximize the distance from the safety boundary, which is a departure from existing approaches to the control Lyapunov-Barrier function. By doing so, our approach ensures a positive lower bound on the minimum inter-execution time, while also maintaining the safety of the control system and reducing the frequency of control input updates and/or samplings. This is particularly beneficial in the context of networked control systems, where safety is of paramount concern. The effectiveness of our proposed approach has been illustrated through a numerical example, which highlights its potential for real-world implementation.

References

[1] P. Wieland and F. Allgöwer, "Constructive safety using control barrier functions," *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 462–467, 2007.

[2] A. Agrawal and K. Sreenath, "Discrete control barrier functions for safety-critical control of discrete systems with application to bipedal robot navigation," *Robotics: Science and Systems*, 2017.

[3] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *Proc. of IEEE Conference on Decision and Control*, 2014, pp. 6271–6278.

[4] L. Lindemann and D. V. Dimarogonas, "Control barrier functions for signal temporal logic tasks," *IEEE Control Systems Letters*, vol. 3, no. 1, pp. 96–101, 2019.

[5] J. Zeng, B. Zhang, and K. Sreenath, "Safety-critical model predictive control with discrete-time control barrier function," in *Proc. of American Control Conference*, 2021, pp. 3882–3889.

[6] A. Singletary, M. Ahmadi, and A. D. Ames, "Safe control for nonlinear systems with stochastic uncertainty via risk control barrier functions," *IEEE Control Systems Letters*, vol. 7, pp. 349–354, 2023.

[7] P. Khargonekar and M. Rotea, "Stabilization of uncertain systems with norm bounded uncertainty using control lyapunov functions," in *Proc. of IEEE Conference on Decision and Control*, 1988, pp. 503–507.

[8] H. K. Khalil, *Nonlinear systems*. Upper Saddle River, NJ": Prentice-Hall, 2002.

[9] M. Krstić and P. V. Kokotović, "Control lyapunov functions for adaptive nonlinear stabilization," *Systems & Control Letters*, vol. 26, no. 1, pp. 17–23, 1995.

[10] R. Freeman and J. Primbs, "Control lyapunov functions: new ideas from an old source," in *Proc. of IEEE Conference on Decision and Control*, vol. 4, 1996, pp. 3926–3931.

[11] J. A. Primbs, V. Nevistić, and J. C. Doyle, "Nonlinear optimal control: A control lyapunov function and receding horizon perspective," *Asian Journal of Control*, vol. 1, no. 1, pp. 14–24, 1999.

[12] P. Ogren, M. Egerstedt, and X. Hu, "A control lyapunov function approach to multi-agent coordination," in *Proc. of IEEE Conference on Decision and Control*, vol. 2, 2001, pp. 1150–1155 vol.2.

[13] M. Z. Romdlony and B. Jayawardhana, "Uniting control Lyapunov and control barrier functions," in *Proc. of IEEE Conference on Decision and Control*, 2014, pp. 2293–2298.

[14] ——, "Stabilization with guaranteed safety using control lyapunov–barrier function," *Automatica*, vol. 66, pp. 39–47, 2016.

[15] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.

[16] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *Proc. of European Control Conference*, 2019, pp. 3420–3431.

[17] E. A. Basso, E. H. Thyri, K. Y. Pettersen, M. Breivik, and R. Skjetne, "Safety-critical control of autonomous surface vehicles in the presence of ocean currents," in *Proc. of IEEE Conference on Control Technology and Applications*, 2020, pp. 396–403.

[18] P. Mestres and J. Cortés, "Optimization-based safe stabilizing feedback with guaranteed region of attraction," *IEEE Control Systems Letters*, vol. 7, pp. 367–372, 2023.

[19] R. Postoyan, P. Tabuada, D. Nešić, and A. Anta, "Event-triggered and self-triggered stabilization of distributed networked control systems," in *Proc. of IEEE Conference on Decision and Control and European Control Conference*, 2011, pp. 2565–2570.

[20] W. Heemels, K. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *Proc. of IEEE Conference on Decision and Control*, 2012, pp. 3270–3285.

[21] M. Mazo and P. Tabuada, "On event-triggered and self-triggered control over sensor/actuator networks," in *Proc. of IEEE Conference on Decision and Control*, 2008, pp. 435–440.

[22] 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.

[23] W. Xiao, C. Belta, and C. G. Cassandras, "Event-triggered control for safety-critical systems with unknown dynamics," *IEEE Transactions on Automatic Control*, pp. 1–16, 2022.

[24] G. Yang, C. Belta, and R. Tron, "Self-triggered control for safety critical systems using control barrier functions," in *Proc. of American Control Conference*, 2019, pp. 4454–4459.

[25] A. D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle, "Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics," *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 876–891, 2014.

[26] A. D. Ames and M. Powell, *Towards the Unification of Locomotion and Manipulation through Control Lyapunov Functions and Quadratic Programs*. Heidelberg: Springer International Publishing, 2013, pp. 219–240.