Safe driving with control barrier functions in mixed autonomy traffic when cut-ins occur

George Gunter, Daniel Work

Abstract— **Ensuring safety of automated vehicle (AV) control systems in multi-lane mixed-autonomy traffic is challenging. *Control barrier functions* (CBFs) represent a promising approach in which control inputs are filtered to guarantee forward-invariance of satisfaction of desired safety properties. This allows for balancing safety with performance, such as in the context of data-driven adaptive cruise control systems, which may otherwise be difficult to assure safety for. In real-world deployments to mixed-autonomy multi-lane traffic, however, external disturbances such as cut-in events can generate violations to the satisfaction of safety properties which would otherwise be met, such as maintaining a minimum time gap between vehicles. This work extends the design of CBFs for AVs by explicitly considering the effect of cut-in events. We show that a commonly proposed CBF designed to maintain time-gap cannot guarantee collision avoidance in the event of a cut-in. We show that when paired with a secondary CBF designed to maintain a positive space-gap through the use of higher-order CBFs via simple switching logic that both collisions can be avoided when cut-ins occur, and that over time the desired time-gap will be restored. Additionally, we present criteria for pole placement and string-stability of the AV when choosing CBF parameters. A series of numerical experiments are presented to demonstrate the main results.**

I. Introduction

Recent advances in *automated vehicle* (AV) technology have increased interest in how AVs will affect traffic patterns. Several works (e.g.,[1], [2]) have motivated that allowing AVs to employ control laws which manage their speed can lead to benefits for the broader traffic. Longitudinal control algorithms based on deep learning [3], [4] or other data-driven techniques [5], [6] have shown especial promise in terms of reducing energy consumption and lessening traffic waves.

While these techniques have mostly focused on the potential performance benefits, safety has received less attention. Ideally, safety would be formally verified, which is often done using reachability analysis techniques [7], [8]. Alternatively, model predictive techniques have been proposed for preventing collisions [9]. Developing formal guarantees for deep-learning or data-driven control techniques may be quite difficult however [10], [11], prompting the need for supervisory techniques which can achieve formal verification of safety properties for AVs.

Control barrier functions (CBFs) present an alternative approach for guaranteeing system safety [12], [13]. Given a safety property, CBFs can be used to derive constraints on the control inputs such that the system is *forward invariant* with respect to the desired safety property. Several works have investigated the design of CBFs for managing the control of adaptive cruise systems [12], [14], [15]. Additionally, recent work has also explored how to pair CBFs with reinforcement-learning [16], [17], including in the context of car-following.

The main contribution of this work is to consider the effect of cut-in events on AVs controlled by CBFs. A cut-in event occurs when a driver in an adjacent lane merges in front of the controlled AV. These can occur in mixed-autonomy multi-lane traffic environments. Cut-ins represent disturbances to the AV system state, which can violate a minimum time-gap safety condition, even if the control derived from the CBF would otherwise maintain the desired minimum time-gap. Using a *zeroing control barrier function* (zCBF), we consider a control law in which the control input is selected from the boundary of the permissible control inputs. This control law returns the AV to the specified minimum time-gap after a cut-in event occurs. We show how parameters in the zeroing CBF effect pole-placement and string-stability. Finally, we show that this controller may not prevent collisions in the event of a cut-in, when otherwise collisions would be avoided. This motivates the use of a second zCBF that ensures collision-avoidance. When paired with the time-gap CBF via simple switching logic, both collision-avoidance and a return to a minimum time-gap are achieved after cut-ins. All results are demonstrated with numerical experiments.

The remainder of this work is organized as follows. In Section II background related to the synthesis and design of control barrier functions is presented. Section III covers the model assumed for AV control as well as the derivation of two CBFs used for safe driving. The effect of cut-ins on the ability to guarantee system safety when using the proposed CBFs is discussed in Section IV, and subsequently new designs are proposed for how CBFs can handle cut-ins. In Section V a series of numerical experiments are shown which demonstrate the effect of cut-ins on CBF controlled AVs and demonstrate the concepts covered in III and IV. The work is concluded and future work is discussed in Section VI.

II. Preliminaries

Here we review preliminary concepts related to the design of CBFs. First, we present general background on how zeroing control barrier functions (zCBFs) are formed [15]. Next, we review how to convert zCBFs which are of lower order than the control of the system into higher-order zCBFs [18], [19].

A. Zeroing control barrier functions

Consider a *control affine* system of the following form:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t)) + g(\mathbf{x}(t))u(t), \tag{1}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the system state which evolves over time, $f(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ are the nominal dynamics, $g(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ are the controlled dynamics, and $u(t) \in \mathbb{R}^1$ is a control input.

Let the following inequality define safety at an individual state $\mathbf{x}$,

$$h(\mathbf{x}) \geq 0 \tag{2}$$

meaning a state $\mathbf{x}$ is safe if $h(\mathbf{x})$ is non-negative, where $h(\cdot) : \mathbb{R}^n \to \mathbb{R}^1$. We refer subsequently to (2) as the *safety property*, which is either satisfied or not.

Let $\mathcal{C}_h$ refer to the set of all $\mathbf{x}$ that satisfy (2) for a given choice of $h(\cdot)$. A subsequent desirable property is that (2) be *forward-invariant*, which is defined as [12]:

$$\mathbf{x}(0) \in \mathcal{C}_h \implies \mathbf{x}(t) \in \mathcal{C}_h, \forall t > 0 \tag{3}$$

meaning that if $\mathbf{x}$ begins ($t = 0$) inside $\mathcal{C}_h$ then it remains inside $\mathcal{C}_h$ for all following times t.

Via Nagumo's forward-invariance theorem [20], a given safety property as per 2 will be forward-invariant if the following inequality is also satisfied:

$$\dot{h}(\mathbf{x}) \geq -\alpha(h(\mathbf{x})), \forall \mathbf{x} \in \mathcal{C}_h \tag{4}$$

where $\dot{h}(\mathbf{x})$ is the time-derivative of $h(\mathbf{x})$, and $\alpha(\cdot)$ is a class-$\mathcal{K}$ function. Control barrier functions of this form are said to be *zeroing control barrier functions* (zCBFs) [13].

$\dot{h}(\mathbf{x})$ is calculated at a given $\mathbf{x}$ as such:

$$\begin{aligned}\dot{h}(\mathbf{x}) = L_f h(\mathbf{x}) + L_g h(\mathbf{x})u = \\ \langle \nabla h(\mathbf{x}), f(\mathbf{x})\rangle + \langle \nabla h(\mathbf{x}), g(\mathbf{x})u\rangle\end{aligned} \tag{5}$$

where $L_f h(\mathbf{x})$ and $L_g h(\mathbf{x})$ are the Lie derivatives of $h(\mathbf{x})$ with respect to $f(\mathbf{x})$ and $g(\mathbf{x})$, and u is a selected control input as previously defined. Note that choices of the control input u may effect the value of $\dot{h}(\mathbf{x})$ (so long as $\langle \nabla h(\mathbf{x}), g(\mathbf{x})\rangle \neq 0$).

Whether or not a given ($\mathbf{x}$,u) pair satisfies (4) can be stated as follows:

$$\langle \nabla h(\mathbf{x}), f(\mathbf{x})\rangle + \langle \nabla h(\mathbf{x}), g(\mathbf{x})u\rangle \geq -\alpha(h(\mathbf{x})) \tag{6}$$

Subsequently, the set of all control inputs u which satisfy (4) at a given state $\mathbf{x}$ can be written as such:

$$\begin{aligned}\mathcal{K}_{\text{safe}}(\mathbf{x}) = \\ \{u : \langle \nabla h(\mathbf{x}), f(\mathbf{x})\rangle + \langle \nabla h(\mathbf{x}), g(\mathbf{x})u\rangle \geq -\alpha(h(\mathbf{x}))\}\end{aligned} \tag{7}$$

Where $\alpha(\cdot)$ is a chosen class-$\mathcal{K}$ function. $\mathcal{K}_{\text{safe}}(\mathbf{x})$ then defines a set, known as the *safety kernel*, of control inputs, which at a given state $\mathbf{x}$ will ensure (4) holds, and thus $\mathcal{C}_h$ is forward-invariant.

B. zCBFs with low relative degree

If it is the case that $\langle \nabla h(\mathbf{x}), g(\mathbf{x})u\rangle = 0$ then $\dot{h}(\mathbf{x})$ is not directly influenced by u. This may happen when a given property is of lower relative degree than the control input to the system.

A possible way to account for this is through the use of *higher-order control barrier functions* (HO-CBFS). From an initial lower order CBF, the corresponding HO-CBF can be constructed as follows. First, (4) is rewritten as

$$h_1(\mathbf{x}) := \dot{h}(\mathbf{x}) + \alpha_0(h(\mathbf{x})) \geq 0 \tag{8}$$

where $h_1(\mathbf{x}) \geq 0$ is a new property to be made forward-invariant, and $\alpha_0(\cdot)$ is an initial class-$\mathcal{K}$ function.

Re-applying (4) to (8) yields the following:

$$\dot{h}_1(\mathbf{x}) \geq -\alpha_1(h_1(\mathbf{x})), \forall \mathbf{x} \in \mathcal{C}_h \tag{9}$$

where $\alpha_1(\cdot)$ is a secondary class-$\mathcal{K}$ function. This can be rewritten in terms of $h(\mathbf{x})$ as follows:

$$\begin{aligned}&\dot{h}_1(\mathbf{x}) + \alpha_1(h_1(\mathbf{x})) = \\ &\ddot{h}(\mathbf{x}) + \alpha_0(\dot{h}(\mathbf{x})) + \alpha_1(\dot{h}(\mathbf{x}) + \alpha_0(h(\mathbf{x}))) \geq 0\end{aligned} \tag{10}$$

If subsequently $\ddot{h}(\mathbf{x})$ is influenced by the choice of control input then (10) may be enforced. Note that

$$h_1(\mathbf{x}(t)) \geq 0, \forall (t > 0) \implies h(\mathbf{x}(t)) \geq 0, \forall (t > 0) \tag{11}$$

meaning that if (10) is met then $\mathcal{C}_h$ is forward-invariant. This technique was presented in different forms in [18], [19].

III. zCBFs for Control of Automated Vehicles

In this section the use of ZCBFs for automated vehicle (AV) speed control is presented. First a working system model is introduced. Next, relevant safety properties are introduced and the conditions for their forward-invariance are derived.

A. Automated vehicle control model

Let the following control affine system describe how an AV can be controlled in the context of car-following:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})u = \begin{bmatrix}\dot{v}_f \\ \dot{v}_l \\ \dot{s}\end{bmatrix} = \begin{bmatrix}0 \\ a_l \\ v_l - v_f\end{bmatrix} + \begin{bmatrix}1 \\ 0 \\ 0\end{bmatrix} u \tag{12}$$

where $\mathbf{x} = [v_f, v_l, s] \in \mathbb{R}^3$ is the system state, v_f is the speed of the ego vehicle (i.e., the following AV to be controlled), v_l is the speed of a leading vehicle, s is the space gap between the vehicles, a_l is the acceleration of the leading vehicle, and $u \in \mathbb{R}^1$ is a control acceleration input for the AV.

B. Safety properties

We consider two safety properties for AVs, *i*) maintaining a minimum time gap, and *ii*) collision avoidance. Minimum time-gap can be formulated as a safety property in the following way:

$$h_{TG}(\mathbf{x}) = s - t_{\min} v_f \geq 0 \tag{13}$$

where $h_{TG}(\mathbf{x}) \geq 0$ is the minimum time-gap property the AV seeks to satisfy, and t_{min} is the minimum allowable time-gap difference between the control vehicle and the lead vehicle. The adaptation of this safety constraint via CBFs was first proposed in [15].

The relevant Lie derivatives needed to find the corresponding safety-kernel are as follows:

$$\begin{aligned} L_f h_{TG}(\mathbf{x}) &= v_l - v_f \\ L_g h_{TG}(\mathbf{x})u &= -t_{\text{min}} u \\ \dot{h}_{TG}(\mathbf{x}) &= v_l - v_f - t_{\text{min}} u \end{aligned} \tag{14}$$

If the class-κ function is chosen as $\alpha(h(\mathbf{x})) = kh(\mathbf{x})$ where k is a positive scalar, then the set of safe control inputs (the safety kernel) at a given state can be written as

$$\begin{gathered} \mathcal{K}_{\text{safe},TG}(\mathbf{x}) = \\ \{u : v_l - v_f - t_{\text{min}} u \geq -k\left(s - t_{\text{min}} v_f\right)\} \end{gathered} \tag{15}$$

where $\mathcal{K}_{\text{safe},TG}(\mathbf{x})$ is the safety kernel. Rearranging gives the following constraint on u:

$$u \leq \left(\frac{1}{t_{\text{min}}}\right)(v_l - v_f) + \left(\frac{k}{t_{\text{min}}}\right)(s - t_{\text{min}} v_f) \tag{16}$$

If u satisfies (16) (i.e. is chosen from $\mathcal{K}_{\text{safe},TG}(\mathbf{x})$) for each $\mathbf{x}$, then the set of states satisfying (13) is forward-invariant.

Collision avoidance can be written as a safety property as follows:

$$h_{CA}(\mathbf{x}) = s \geq 0, \tag{17}$$

which can trivially be extended to account for larger minimum spacings if desired. The relevant derivatives are

$$\begin{aligned} L_f h_{CA}(\mathbf{x}) &= v_l - v_f \\ L_g h_{CA}(\mathbf{x})u &= 0 \\ \dot{h}_{CA}(\mathbf{x}) &= v_l - v_f \end{aligned} \tag{18}$$

Notice that this because this property is of lower degree than the control of the system $L_g h_{CA}(\mathbf{x})u = \langle \nabla h_{CA}(\mathbf{x}), g(\mathbf{x})u \rangle = 0$, meaning choices of u do not directly effect $\dot{h}_{CA}(\mathbf{x})$. To account for this, the approach in II-B can be employed to derive a HO-CBF. The second time-derivative of $h_{CA}(\mathbf{x})$ is

$$\ddot{h}_{CA}(\mathbf{x}) = a_l - u \tag{19}$$

which is directly effected by the choice of control input. When paired with (10) gives the following inequality:

$$\begin{aligned} &\ddot{h}_{CA}(\mathbf{x}) + k_1 \dot{h}_{CA}(\mathbf{x}) + k_1(\dot{h}_{CA}(\mathbf{x}) + k_0 h_{CA}(\mathbf{x})) = \\ &a_l - u + (k_0 + k_1)(v_l - v_f) + (k_0 k_1)s \geq 0 \end{aligned} \tag{20}$$

Where k_0 and k_1 are positive scalars corresponding to choices for class-$\mathcal{K}$ functions. This can be rewritten as an explicit inequality on u as follows:

$$u \leq a_l + (k_0 + k_1)(v_l - v_f) + (k_0 k_1)s \tag{21}$$

Selection of u satisfying (21) for a given $\mathbf{x}$ will assure the set of $\mathbf{x}$ satisfying (17) is forward-invariant.

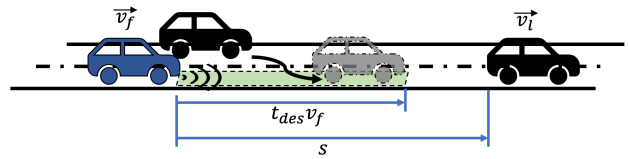

Fig. 1. A vehicle in the adjacent lane (black) merges in front of the ego vehicle (blue) in such a way that it violates the minimum time-gap property (green).

IV. Responding to cut-in events

Here we explicitly consider the scenario in which a cut-in event occurs. First we discuss how a cut-in is a disturbance to the system representing an abrupt change in the system state, and how this can lead to violation of (13). Next, we discuss how zCBFs can recover back to safety after a violation. Finally, we adapt this to consider design of zCBFs for AVs to explicitly handle cut-in events.

A. Cut-in events are disturbances

Cut-ins occur when a vehicle in an adjacent lane merges in front of the AV. Such events cause an abrupt decrease in the spacing (s) and a possible change in the leading speed (v_l). If the decrease in spacing is enough (the vehicle merges in close to the AV), then this can also represent a violation of (13) which cannot be avoided.

A pictorial example of this is shown in Figure 1 in which an AV running a minimum time-gap zCBF maintains a minimum time-gap (green). Since the AV cannot control cut-ins, an important design consideration then is how the AV responds to cut-ins which violate (13).

B. zCBF response to violation of a safety property

In (1) a model was assumed for how system dynamics change over time, and how control inputs can affect that evolution. In reality, it is likely the system will be subject to non-modeled disturbances. To describe this we can write the dynamics with disturbances as follows:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})u + \hat{\mathbf{d}} \tag{22}$$

where $\hat{\mathbf{d}} \in \mathbb{R}^n$ represents non-modeled disturbances. If $\hat{\mathbf{d}}$ is large enough, it may move $\mathbf{x}$ outside of $\mathcal{C}_h$.

Let $\partial\mathcal{C}_h$ refer to the boundary of $\mathcal{C}_h$, which is where $h(\mathbf{x}) = 0$. In the event of a violation of a given $h(\mathbf{x})$, it is the case that $\mathbf{x} \in \setminus\mathcal{C}_h$. For $\mathbf{x} \in \setminus\mathcal{C}_h$ it is also the case that $-\alpha(h(\mathbf{x})) > 0$, since $h(\mathbf{x}) < 0$ and $\alpha(\cdot) \in$ class-$\mathcal{K}$. If u is then chosen such that (4) still holds (i.e. from the safety kernel) then

$$\dot{h}(\mathbf{x}) \geq -\alpha(h(\mathbf{x})) > 0 \tag{23}$$

If $\hat{\mathbf{d}}$ is a *vanishing* disturbance, rather than a *persistent* disturbance, then inputs satisfying (23) will asymptotically return $\mathbf{x}$ to $\partial\mathcal{C}_h$, meaning that safety will be recovered. This result was shown in [14].

C. zCBF design to handle cut-ins

Cut-in events are vanishing disturbances to the state of the controlled AV. If a minimum time-gap zCBF is used then choices of u from the corresponding safety-kernel will converge back to safety asymptotically, as per (23). In the design of a zCBF a class-κ function is employed, but any such function which satisfies (4) is appropriate. When a cut-in occurs, the choice of class-κ function used now effects how the system returns back to safety.

To show this concretely, consider the inequality proposed in (16). In the event of cut-in (16) will still define a safety-kernel of control inputs. Selecting the largest in value control from this kernel will yield the following control law:

$$\begin{aligned} u_{TG}(\mathbf{x}) &= \left(\tfrac{1}{t_{\text{min}}}\right)(v_l - v_f) + \left(\tfrac{k}{t_{\text{min}}}\right)(s - t_{\text{min}} v_f) \\ &= \left(\tfrac{1}{t_{\text{min}}}\right)(v_l - v_f) + \left(\tfrac{k}{t_{\text{min}}}\right) s - k v_f \end{aligned} \tag{24}$$

which is linear controller with 2 parameters t_{min} and k, and an equilibrium for the system when $v_l - v_f = 0$ and $s = t_{\text{min}} v_f$, which is the boundary of the set of $\mathbf{x}$ satisfying (16). For this controller, we identify three possible design criteria that could be used to inform choices of t_{min} and k: pole-placement, string-stability, and collision avoidance.

Pole-placement. For the control law in (24), the vehicle-to-vehicle space-gap transfer function can be calculated as [21]:

$$\Gamma(z) = \frac{z\left(\frac{1}{t_{\text{min}}}\right) + \left(\frac{k}{t_{\text{min}}}\right)}{z^2 + z\left(\left(\frac{1}{t_{\text{min}}}\right) + k\right) + \left(\frac{k}{t_{\text{min}}}\right)} \tag{25}$$

where $z := j\omega$ and $\omega \geq 0$ is a frequency. The poles for this system are:

$$-\frac{1}{2}\left(\frac{1}{t_{\text{min}}} + k\right) \pm \sqrt{\frac{-1}{4t_{\text{min}}^2} - \frac{k}{2t_{\text{min}}} + \frac{k^2}{4}} \tag{26}$$

Different choices of k and t_{min} will affect how the system recovers in the event of a cut-in.

String-stability. The transfer function in (25) has the following norm:

$$|\Gamma(j\omega)| = \sqrt{\frac{\omega^2 \frac{1}{t_{\text{min}}^2} + \frac{k^2}{t_{\text{min}}^2}}{\left(\frac{k}{t_{\text{min}}} - \omega^2\right)^2 + \omega^2\left(\frac{1}{t_{\text{min}}} + k\right)^2}} \tag{27}$$

To achieve *string-stability* of the controller means that $|\Gamma(j\omega)| \leq 1, \forall \omega$. This is achieved if the following condition is met [22]:

$$\left(\frac{1}{t_{\text{min}}} + k\right)^2 - \left(\frac{1}{t_{\text{min}}}\right)^2 - \frac{2k}{t_{\text{min}}} \geq 0 \tag{28}$$

This condition simplifies to

$$k^2 \geq 0 \tag{29}$$

which is true for all $k \geq 0$ (which is required for $\alpha(h(\mathbf{x})) = k(h(\mathbf{x})) \in \kappa$), meaning control from (24) will result in string-stable driving.

Collision avoidance. Without external disturbances if control inputs satisfy (24), then collision avoidance is also achieved for positive speeds. In the event of a cut-in, control inputs that from (24) (rather than (16)) will asymptotically recover the minimum time-gap, but it is not guaranteed that this will happen quickly enough to assure collision avoidance. In particular, control inputs which satisfy (16), but not (20), may lead to collisions.

To address this we propose the following switching condition between the two CBFs:

$$u_{\text{combined}}(\mathbf{x}) = \min\{u_{TG}(\mathbf{x}), u_{CA}(\mathbf{x})\} \tag{30}$$

where $u_{CA}(\mathbf{x})$ is the control law that converts (20) to an equality.

This controller has a switching condition at

$$\begin{aligned} \left(\tfrac{1}{t_{\text{min}}}\right)(v_l - v_f) + \left(\tfrac{k}{t_{\text{min}}}\right)(s - t_{\text{min}} v_f) = \\ a_l + (k_0 + k_1)(v_l - v_f) + (k_0 k_1) s \end{aligned} \tag{31}$$

which is equivalent to

$$\begin{aligned} a_l + \left(k_0 + k_1 - \tfrac{1}{t_{\text{min}}}\right)(v_l - v_f) \\ + \left(k_0 k_1 - \tfrac{k}{t_{\text{min}}}\right) s + k v_f &= 0 \end{aligned} \tag{32}$$

If the left-hand side quantity is negative then control from u_{CA} is chosen, and if positive u_{TG} is chosen. Choices of all 4 parameters, k, k_0, k_1, and t_{min} effect where this switching occurs in state space.

The switching control law in (30) is guaranteed to command control inputs that satisfy both (16) and (20). To see this, we note that since $u_{TG}(\mathbf{x})$ is the largest in value control input that satisfies (16), and since $u(t) \in \mathbb{R}^1$, any control input lower in value will also satisfy (16). The same is true for $u_{CA}(\mathbf{x})$ and (20). As a result, the minimum of either is guaranteed to satisfy both constraints.

V. Experimental Results

Here numerical experiments are shown demonstrating the use of the design considerations presented in Section IV. To simulate a cut-in event an AV is simulated starting from an initial set of $v_f, s,$ and v_l. In all experiments considered these initial conditions do not satisfy (16), thus representing cut-ins that violate the minimum time-gap property.

In Figure 2 the effect that changing k has on how (24) recovers safety is shown. The cut-in reduces s to $10m$, and subsequently the lead vehicles drives constantly at a speed of $10\frac{m}{s}$, which is the same as the initial v_f. Each simulated controller has $t_{\text{min}} = 2.0$, while the response for 3 different choices of k are shown. Larger k causes the vehicle to react more strongly, return to safety more quickly.

In Figure 3 a platoon of AVs each running (24) with $t_{\text{min}} = 2.0, k = 0.1$ are simulated. The same cut-in event as in 2 performed. The first vehicle has the same response as is in 2, but each subsequent following vehicle slows down progressively less. This is in keeping with the control response being string-stable.

Figure 4 shows the response that the control from each of $u_{TG}(x)$, $u_{CA}(x)$, and $u_{\text{combined}}(x)$ to a cut-in event. The cut-in event in question causes $s(0) = 5m, v_f(0) =$

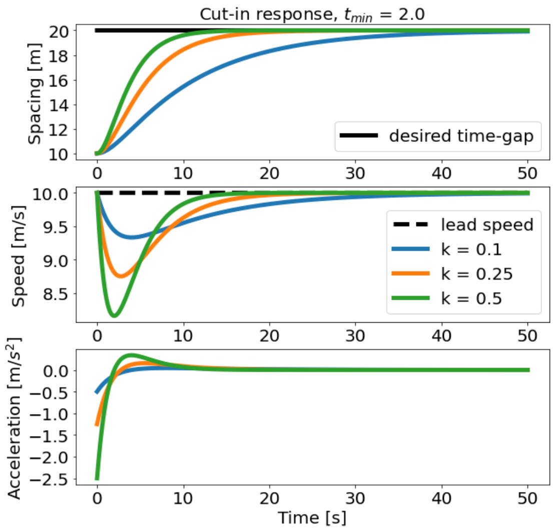

Fig. 2. Response from the minimum time-gap zCBF control law with $t_{\min} = 2.0$ across varying choices of k.

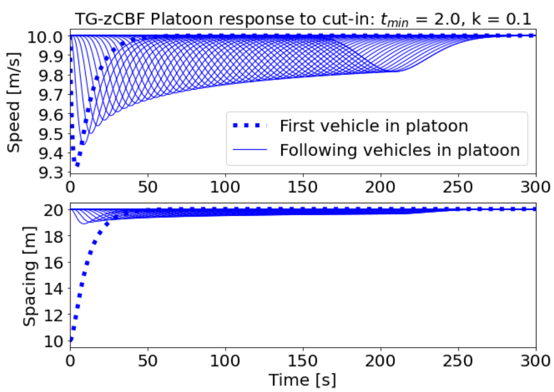

Fig. 3. A platoon of vehicles all running the same minimum time-gap zCBF control with $t_{\min} = 2.0$ and $k = 0.25$

$10\frac{m}{s}$, and$v_l(0) = 5\frac{m}{s}$. Controller parameters used are $k_0 = 1.5$,$k_1 = 1.5$,$k = 0.1$ and $t_{\min} = 2.0$.

The response to the cut-in from u_{TG} leads to a collision, marked as a black 'x'. While in reality a collision would mean an immediate stoppage of control, we continue the simulation to show that u_{TG} eventually returns the AC towards a $2.0s$ time-gap. In comparison, u_{CA} successfully prevents a collision, but allows the spacing to decrease asymptotically close to 0 where it stays. u_{combined} is able to both prevent a collision from occurring, and then begin to return the AV back towards the desired minimum time-gap.

In Figure 5 a set of several cut-in events are shown all in the $\Delta v - s$ plane ($\Delta v = v_l - v_f$). Cut-ins are simulated across different initial v_f and s while for each $v_l(t) = 5\frac{m}{s}$ for the whole simulation. The initial values of $v_f(0)$ are $7.5, 10.0,$ and 12.5, while the cut-in reduces $s(0)$ to $2.5, 5.0,$ and 7.5.

In the top subplot results of control using u_{TG} are shown. While some of the events are handled without collision, those with lower spacings and lower speed differences can lead to collisions. In the bottom subplot control using u_{CA} are shown, where all collisions are prevented, but the state converges to zero spacing over time. In the middle subplot control from u_{combined} is shown, with the division between control inputs shaded in the state space. Collisions are avoided and the state returns back to the desired minimum time-gap over time.

VI. Conclusions and Future Work

This work explored the use of zeroing control barrier functions for safe control of automated vehicles in the presence of cut in events. A cut-in event may violate a minimum time-gap constraint, a corresponding zCBF will yield a control law that asymptotically stabilizes the system back to safety. We derive conditions for pole-placement and string-stability for this controller. Under some cut-in events the minimum time-gap zCBF will not prevent a collision with the leading vehicle. By adding a second a zCBF designed to prevent collisions we find that collisions in those cut-in events can be prevented, and the system subsequently returned to a minimum time-gap. Numerical experiments are presented which validate the proposed methodologies and techniques.

In future work we will consider how these control schemes behave under physical bounds on possible control inputs. We will explore experimentation on live vehicles.

VII. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant CNS-1837652, and work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Vehicle Technologies Office award number CID DE-EE0008872. The views expressed herein do not necessarily represent the views of the U.S. Department of Energy or the United States Government. George Gunter is supported by a National Science Foundation Graduate Research Fellowship.

References

[1] R. E. Stern, S. Cui, M. L. Delle Monache, R. Bhadani, M. Bunting, M. Churchill, N. Hamilton, H. Pohlmann, F. Wu, B. Piccoli *et al.*, "Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments," *Transportation Research Part C: Emerging Technologies*, vol. 89, pp. 205–221, 2018.

[2] A. Talebpour and H. S. Mahmassani, "Influence of connected and autonomous vehicles on traffic flow stability and throughput," *Transportation Research Part C: Emerging Technologies*, vol. 71, pp. 143–163, 2016.

[3] A. R. Kreidieh, C. Wu, and A. M. Bayen, "Dissipating stop-and-go waves in closed and open networks via deep reinforcement learning," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 1475–1480.

[4] C. Wu, A. Kreidieh, K. Parvate, E. Vinitsky, and A. M. Bayen, "Flow: Architecture and benchmarking for reinforcement learning in traffic control," *arXiv preprint arXiv:1710.05465*, vol. 10, 2017.

[5] Y. Wang, Z. Wang, K. Han, P. Tiwari, and D. B. Work, "Personalized adaptive cruise control via gaussian process regression," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 1496–1502.

[6] W. Gao, Z.-P. Jiang, and K. Ozbay, "Data-driven adaptive optimal control of connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1122–1133, 2016.

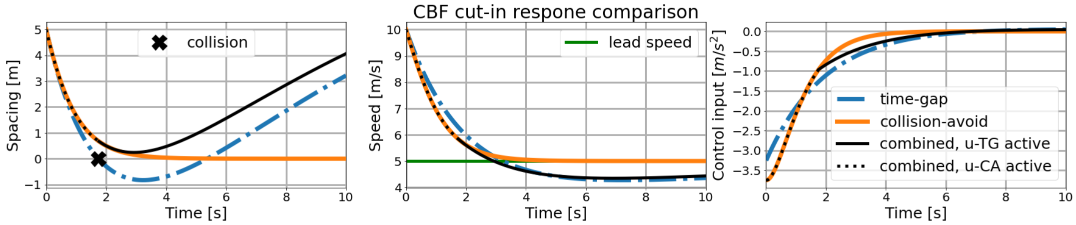

Fig. 4. Responses by different control schemes to a cut-in event. The minimum time-gap control alone is unable to prevent a collision with the vehicle in front.

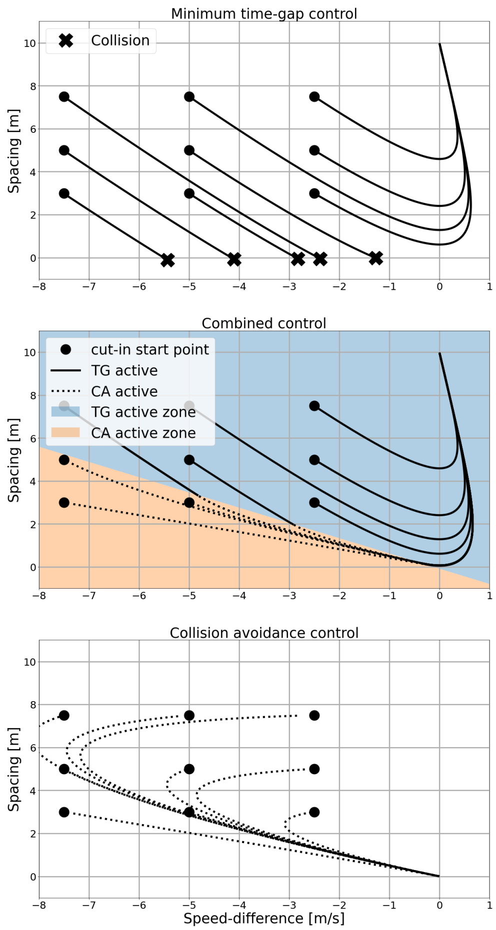

Fig. 5. A comparison of how the different controllers handle a variety of cut-in events.

[7] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, 2014.

[8] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on automatic control*, vol. 50, no. 7, pp. 947–957, 2005.

[9] M. Althoff, O. Stursberg, and M. Buss, "Model-based probabilistic collision detection in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 299–310, 2009.

[10] Q. Lin, S. Verwer, and J. Dolan, "Safety verification of a data-driven adaptive cruise controller," in *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2020, pp. 2146–2151.

[11] W. Xiang, D. M. Lopez, P. Musau, and T. T. Johnson, "Reachable set estimation and verification for neural network models of nonlinear dynamic systems," in *Safe, autonomous and intelligent vehicles*. Springer, 2019, pp. 123–144.

[12] 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, 2016.

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

[14] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames, "Robustness of control barrier functions for safety critical control," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 54–61, 2015.

[15] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 6271–6278.

[16] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick, "End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3387–3395.

[17] J. Choi, F. Castaneda, C. J. Tomlin, and K. Sreenath, "Reinforcement learning for safety-critical control under model uncertainty, using control lyapunov functions and control barrier functions," *arXiv preprint arXiv:2004.07584*, 2020.

[18] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 322–328.

[19] W. Xiao and C. Belta, "Control barrier functions for systems with high relative degree," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 474–479.

[20] F. Blanchini and S. Miani, *Set-theoretic methods in control*. Springer, 2008.

[21] J. Monteil, M. Bouroche, and D. J. Leith, "$\mathcal{L}_2$ and $\mathcal{L}_\infty$ stability analysis of heterogeneous traffic with application to parameter optimization for the control of automated vehicles," *IEEE Transactions on Control Systems Technology*, 2018.

[22] S. Cui, B. Seibold, R. Stern, and D. B. Work, "Stabilizing traffic flow via a single autonomous vehicle: Possibilities and limitations," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 1336–1341.