Learning-Based Safety-Stability-Driven Control for Safety-Critical Systems under Model Uncertainties

Lei Zheng[1], Rui Yang[2], Jiesen Pan[2], Hui Cheng[2*], and Haifeng Hu[1]

[1] School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China

[2] School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China

Email: {zhenglei5, yangr38}@mail2.sysu.edu.cn, panjs3@mail3.sysu.edu.cn, {chengh9, huhaif}@mail.sysu.edu.cn

Abstract—**Safety and tracking stability are crucial for safety-critical systems such as self-driving cars, autonomous mobile robots, and industrial manipulators. To efficiently control safety-critical systems to ensure their safety and achieve tracking stability, accurate system dynamic models are usually required. However, accurate system models are not always available in practice. In this paper, a learning-based safety-stability-driven control (LBSC) algorithm is presented to guarantee the safety and tracking stability for nonlinear safety-critical systems subject to control input constraints under model uncertainties. Gaussian Processes (GPs) are employed to learn the model error between the nominal model and the actual system dynamics, and the estimated mean and variance of the model error are used to quantify a high-confidence uncertainty bound. Using this estimated uncertainty bound, a safety barrier constraint is devised to ensure safety, and a stability constraint is developed to achieve rapid and accurate tracking. Then the proposed LBSC method is formulated as a quadratic program incorporating the safety barrier, the stability constraint, and the control constraints. The effectiveness of the LBSC method is illustrated on the safety-critical connected cruise control (CCC) system simulator under model uncertainties.**

Index Terms—**Safety, tracking stability, uncertainty, safety-critical systems**

I. INTRODUCTION

Safety is a fundamental issue for safety-critical systems [1], such as self-driving cars, autonomous mobile robots, industrial manipulators, chemical reactors. Safety-critical systems may operate in unsafe states due to model inaccuracies, external disturbances, and environmental changes. It will lead to unexpected failures and severe damage to machines, environments, and even human life [2].

On the other hand, achieving stable tracking performance with respect to tracking accuracy and convergence rate is crucial for dynamical control systems to accurately follow the desired states. However, there is a tradeoff between safety and tracking performance, and it is difficult for safety-critical systems with control input constraints to simultaneously satisfy safety and stabilization constraints in practical applications. For example, it is challenging for a self-driving car to rapidly converge to the desired speed and maintain it while avoiding unexpected obstacles. For a team of safety-critical vehicles, each vehicle keeps tracking its front vehicle at a desired constant speed while maintaining a safe following distance with it in normal situations. However, when a vehicle decelerates

urgently in unexpected situations, the vehicle behind has to violate its stability constraint and reduce its speed to avoid collision with the front vehicle. In these cases, there exists a conflict between safety and stable high-performance tracking. For safety-critical systems, safety must not be violated and the tracking errors should be kept as small as possible. Hence, it is important to develop efficient methods for safety-critical systems to mediate the tradeoff between safety and tracking performance.

Accurate system models are usually required to achieve safety and accurate tracking for dynamical control systems. However, exact system models are hard to obtain or even unavailable in practical applications. To address this challenging problem, Gaussian Processes (GPs) have been incorporated into Model Predictive Control (MPC) to account for model uncertainties [3], [4]. However, it is nontrivial to specify a proper cost function [5] to handle safety and tracking performance tradeoff.

Alternatively, the CBF-CLF-QPs approaches [6] incorporating control barrier functions (CBFs) and control Lyapunov functions (CLFs) based on quadratic programs (QPs) are promising to mediate the tradeoff between safety and tracking stability, with the safety being guaranteed. The CBF-CLF-QPs framework has achieved great success in varieties of safety-critical systems such as adaptive cruise control systems [7], [8], freedom bipedal robots [9], and quadrotor systems [10]. Nevertheless, in the presence of model uncertainties, the safety constraints and tracking stability constraints may be violated.

Considering the limitations of current learning-based MPC approaches and CBF-CLF-QPs methods, learning-based approaches are desirable to be developed to ensure both safety and tracking stability for the nonlinear safety-critical systems with model uncertainties. Particularly, as safety must not be violated, it should mediate the tradeoff between safety and stabilization objective when there exist conflicts between them.

In this paper, drawing inspiration from the CBF-CLF-QPs framework, we propose a learning-based algorithm utilizing GPs to model the uncertain dynamical system (inaccurate system parameters, continuous road grade changes, etc) online. The predicted mean and variance of GPs are used to quantify a high confidence interval for uncertainty. With the estimated uncertainty bounds, a safety barrier represented by CBFs is devised to enforce safety, and a stability constraint based on CLFs is formulated to achieve high-performance
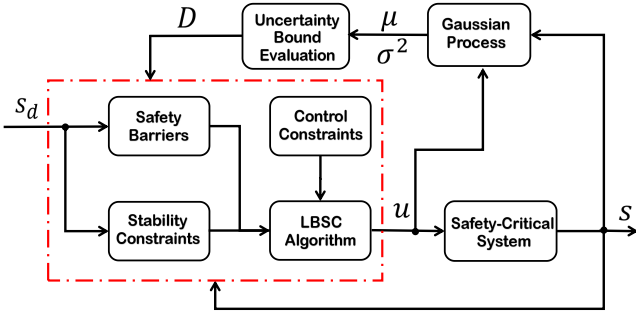
*Corresponding author is Hui Cheng[2].

Fig. 1. **Block diagram of our proposed strategy.** GPs are utilized to learn the model errors of the safety-critical systems. A high confidence interval D with respect to the uncertainty bound can be evaluated based on the predicted mean μ and variance σ^2 of the model errors. Using the high confidence interval D, the LBSC is formulated unifying the safety barrier, the stability constraint, and the control constraints. The symbol s, s_d and u denote the actual system state, the desired system state, and the control input, respectively.

tracking. Considering input constraints, the learning-based safety-stability-driven control (LBSC) algorithm for safety-critical systems is formulated in a standard QP formulation incorporating the safety barrier and the stability constraint as shown in Fig. I.

The **main contributions** of this paper are presented as follows:

- We present a novel learning-based algorithm LBSC to achieve safety guarantees and stabilization objectives for the nonlinear safety-critical systems with model uncertainties. The LBSC method is formulated as a constrained QP incorporating safety (represented by CBFs), stability (represented by CLFs), and control constraints to mediate the tradeoff between safety guarantees and tracking stability.
- We extend the CBFs and CLFs to the uncertain safety-critical systems, which explicitly uses the estimated uncertainties via GPs to guarantee safety and tracking stability.
- We validate the effectiveness of the LBSC algorithm via numerical simulations on the high-speed safety-critical connected cruise control system under model uncertainties.

This paper is organized as follows: Section II presents the preliminaries used in this paper. The problem statement is introduced in Section III. The proposed LBSC algorithm is presented in Section IV with necessary derivations. Simulation results are shown in Section V on a connected cruise control safety-critical system to validate the LBSC approach. Finally, conclusions are drawn in Section VI.

II. PRELIMINARIES

In this section, a brief review of Control Barrier Functions (CBFs) and Gaussian process (GPs) are presented.

Consider a nonlinear affine control system

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

where $x \in \mathscr{X} \subseteq \mathbb{R}^n$, $u \in \mathscr{U} \subseteq \mathbb{R}^m$ denote the states and the control input of the system, and the function $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n \times m}$ are Lipschitz continuous.

A. Control Barrier Functions (CBFs)

CBFs have been widely used in control systems to enforce safety constraints [11]. A *safety set* $\mathscr{S}$ [11] is considered to motivate a formulation of the CBFs, which is defined by

$$\mathscr{S} := \{x \in \mathbb{R}^n | h(x) \geq 0\}, \tag{2}$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function.

Definition 1 ([12])**.** The set $\mathscr{S}$ is called *forward invariant*, if for every $x_0 \in \mathscr{S}$, $x(t, x_0) \in \mathscr{S}$ for all $t \in \mathbb{R}_0^+$.

To ensure forward invariance of the set $\mathscr{S}$, e.g. mobile robots stay in the collision-free safety set at all times, we consider the following definition.

Definition 2 ([7])**.** For the dynamical system (1), given a set $\mathscr{S} \subset \mathbb{R}^n$ defined by (2) for a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$, the function h is called a *Zeroing Control Barrier Function* (*ZCBF*) defined on the set $\mathscr{D}$ with $\mathscr{S} \subseteq \mathscr{D} \subset \mathbb{R}^n$, if there exists an extended class $\mathscr{K}$ function [12] α such that

$$\sup_{u \in \mathscr{U}} [L_f h(x) + L_g h(x)u + \alpha(h(x))] \geq 0, \forall x \in \mathscr{D}, \tag{3}$$

where L represents the Lie derivatives. To be more specific:

$$L_f h(x) = \frac{\partial h(x)}{\partial x} f(x), \; L_g h(x) = \frac{\partial h(x)}{\partial x} g(x). \tag{4}$$

ZCBF is a special control barrier function that comes with asymptotic stability [8]. The existence of a ZCBF implies the asymptotic stability and forward invariance of $\mathscr{S}$ as proved in [8].

B. Gaussian Processes

In this paper, we consider an uncertain control affine system with partially uncertain dynamics:

$$\dot{x} = f(x) + g(x)u + d(x), \tag{5}$$

where $x \in \mathscr{X} \subseteq \mathbb{R}^n$ denotes the system state, $u \in \mathscr{U} \subseteq \mathbb{R}^m$ is the control input, functions $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n \times m}$ compose a prior model representing our knowledge of the actual system, and model error $d : \mathbb{R}^n \to \mathbb{R}^n$ represents uncertain discrepancies between the prior model and the actual system. Similar to [13], assume that the functions $f(x)$, $g(x)$ and $d(x)$ in (5) are Lipschitz continuous to generalize the dynamics to the unexplored states.

A GP is an efficiently nonparametric regression method to estimate complex functions and their uncertain distribution [14]. GPs can be used to learn the model error $d(x)$ using the collected data from the system during operation. To make the problem tractable, similar to Assumption 1 in [15], the following regularity assumption is considered.

Assumption. The unknown model error $d(x)$ has a bounded norm in the associated Reproducing Kernel Hilbert Space (RKHS) [16], corresponding to a differentiable kernel k.

This assumption can be interpreted as a requirement on the smoothness of the model error $d(x)$ (e.g. inaccurate system parameters, continuous road grade changes for a car). Also, its boundedness implies that $d(x)$ is regular with respect to the kernel [17]. In this study, GPs [14] are used to learn the model uncertainty $d(x)$ in (5) with the regularity assumption.

Given n observations $\mathbf{D}_n := \{x_i, \hat{d}(x_i)\}_{i=1}^n$, the mean and variance of $d(x_*)$ at the query state x_* can be given by

$$\mu(x_*) = \mathbf{k}_n^T (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \hat{\mathbf{d}}_n, \qquad (6)$$

$$\sigma^2(x_*) = k(x_*, x_*) - \mathbf{k}_n^T (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{k}_n, \qquad (7)$$

respectively, where $\hat{\mathbf{d}}_n = [\hat{d}(x_1), \hat{d}(x_2), ..., \hat{d}(x_n)]$ is the observed vector subject to a zero mean Gaussian noise $\omega \sim \mathcal{N}(0, \sigma^2)$. $\mathbf{K} \in \mathbb{R}^{n \times n}$ is the covariance matrix with entries, where $[\mathbf{K}]_{(i,j)} = k(x_i, x_j)$, $i, j \in \{1, ..., n\}$, and $k(x_i, x_j)$ is the kernel function. $\mathbf{k}_n = [k(x_1, x_*), k(x_2, x_*), ..., k(x_n, x_*)]$, and $\mathbf{I} \in \mathbb{R}^{n \times n}$ is the identity matrix.

With the system model error $d(x)$ learned by the GPs, a high probability confidence interval $\mathscr{D}(x)$ on the uncertain dynamics $d(x)$ can be obtained to enforce its uncertainty bound by designing the constant c_δ [15].

$$\mathscr{D}(x) = \{d \mid \mu(x) - c_\delta \sigma(x) \le d \le \mu(x) + c_\delta \sigma(x)\}. \qquad (8)$$

where c_δ is a design parameter to get $(1 - \delta)$ confidence, $\delta \in (0, 1)$.

For instance, 95.5% and 99.7% confidence of the uncertainty bound can be achieved at $c_\delta = 2$ and $c_\delta = 3$, respectively.

III. PROBLEM STATEMENT

Consider the uncertain nonlinear control affine system (5) with a given initial state. We aim to design a learning-based controller to track the desired trajectory. The controller must satisfy both the initial state constraints and control input constraints. Meanwhile, it should rapidly drive the system (5) to the target state with safety guarantees under model uncertainties. Specifically, the following desired objectives need to be satisfied:

- **Safety**: The control scheme should guarantee the safety of the uncertain safety-critical systems under model uncertainties with a probability that can be made arbitrarily high through the controller design.

- **Tracking stability**: For feasible desired trajectories, the tracking errors can rapidly converge to a sufficiently small neighborhood of the desired state with a probability that can be made arbitrarily high through the controller design.

- **Adaptability**: The controller should enable the closed-loop control system to continuously adapt to online changes of system parameters as well as continuous environment disturbances.

- **The tradeoff between safety and stability constraints**: The controller can mediate the tradeoff between safety guarantees and high-performance tracking with respect to tracking accuracy and convergence rate enforced by stability constraints.

IV. LEARNING-BASED SAFETY-STABILITY-DRIVEN CONTROL (LBSC) APPROACH

In this section, inspired by the CBF-CLF-QPs [6], an LBSC controller is proposed based on GPs under the Lipschitz continuous assumption for the system (5) and the **Assumption** in II-B. The performance of the LBSC algorithm is analyzed with respect to four control objectives addressed in III. Moreover, the safety guarantees and tracking stability of the LBSC algorithm will be formulated mathematically in detail.

A. Safety Barrier

Based on the ZCBF defined in **Definition 2**, a safety barrier for safety-critical systems can be constructed. Concretely, we aim to design a safety barrier for the uncertain system to keep the state x in the forward invariant safety set.

As the model error $d(x)$ is unknown in prior, GPs are employed to estimate the model uncertainties in terms of the predicted mean $\mu(x)$ and variance $\sigma(x)$ of the model error $d(x)$ through (6) and (7).

Using the definition of the ZCBF and the high confidence interval $D(x)$ defined by (8) with respect to model uncertainties, for all $x \in \mathscr{S}$, the following safe control space K_{rzbf} is formulated for the uncertain dynamical system (5).

$$K_{rzbf}(x) = \{u \in \mathscr{U} \mid \inf_{d \in \mathscr{D}(x)} [\dot{h}(x) + \alpha(h(x))] \ge 0\}. \qquad (9)$$

where $h(x)$ is a ZCBF, $\dot{h}(x) = \frac{\partial h(x)}{\partial x} \dot{x} = L_f h(x) + L_g h(x) u + L_d h(x)$, and $L_d h(x)$ denotes the Lie derivatives of h with respect to the model error $d(x)$.

Lemma 1. Given a set $\mathscr{S} \subset \mathbb{R}^n$ defined by (2) with an associated ZCBF $h(x)$, the control input $u \in K_{rzbf}$ has a probability of at least $(1 - \delta)$, $\delta \in (0, 1)$, to guarantee the forward invariance of the set $\mathscr{S}$ for the uncertain dynamical system (5)

Proof: From (8), there is a probability of at least $(1 - \delta)$ such that the bounded model uncertainty $d(x) \in \mathscr{D}(x)$ for all $x \in \mathscr{X}$. Since the control input $u \in \mathscr{U}$ of the safe control space K_{rzbf} satisfies the constraint in (9), the following result holds with a probability of at least $(1 - \delta)$:

$$\dot{h}(x) + \alpha(h(x)) \ge 0, \forall x \in \mathscr{S}. \qquad (10)$$

As a result, the control input $u \in \mathscr{U}$ of the safe control space K_{rzbf} has a probability of at least $(1 - \delta)$ to guarantee the forward invariance of the set $\mathscr{S}$ for the uncertain dynamical system (5) as proven in [7]. $\blacksquare$

For convenience, the constraint in (9) can be equivalently expressed as

$$L_f h(x) + L_g h(x) u + L_\mu h(x) - c_\delta |L_\sigma h(x)| \ge -\alpha(h(x)), \quad (11)$$

where $L_\mu h(x)$ and $L_\sigma h(x)$ denote the Lie derivatives of $h(x)$ with respect to μ and σ, respectively.

Remark 1. Using more informative data collected for the system dynamics, the bounded uncertainty σ will gradually decrease. Thus, the probability rendering $\mathscr{S}$ forward invariant is much higher than $(1 - \delta)$ in most cases.

B. Stability Constraint

In this study, a stability constraint is developed to enforce rapidly tracking stability with respect to exponential stability [12] in the presence of model uncertainties.

The desired stable high-performance tracking with respect to convergence rate and tracking accuracy can be captured by an *exponentially stabilizing control Lyapunov function* $(ES - CLF)$ [18]. Based on an ES-CLF and the high confidence interval $D(x)$ defined by (8) with respect to model uncertainties, for all $x \in \mathscr{S}$, we consider the following admissible stabilizing control space to ensure stable high tracking performance for the uncertain dynamical system (5)

$$K_{rclf}(x) = \{u \in \mathcal{U} \mid \sup_{d \in \mathcal{D}(x)} [\dot{V}(x) + cV(x)] \leq 0\}, \quad (12)$$

where $V(x)$ is an ES-CLF, $\dot{V}(x) = \frac{\partial V(x)}{\partial x}\dot{x} = L_f V(x) + L_g V(x)u + L_d V(x)$, and c is a positive constant.

Similar to part IV-A, the uncertainty bound of $d(x)$ in (5) can be estimated via GPs.

Lemma 2. Given a set $\mathscr{S} \subset \mathbb{R}^n$ defined by (2) with an associated ES-CLF, the control input $u \in K_{rclf}$ has a probability of at least $(1-\delta)$, $\delta \in (0,1)$, to exponentially stabilize the uncertain dynamical system (5).

Proof: From (8), there is a probability of at least $(1-\delta)$ such that the bounded model uncertainty $d(x) \in \mathscr{D}(x)$ for all $x \in \mathscr{X}$. Since the control input $u \in \mathcal{U}$ of the stabilizing control space K_{rclf} satisfies the constraint in (12), the following result holds with a probability of at least $(1-\delta)$:

$$\dot{V}(x) + cV(x) \leq 0, \forall x \in \mathscr{S}. \quad (13)$$

Hence, the control input $u \in \mathcal{U}$ of the control space K_{rclf} has a probability of at least $(1-\delta)$ to exponentially stabilize the uncertain dynamical system (5) as shown in [18]. ∎

For convenience, the constraints in (12) can be simplified as follows,

$$L_f V(x) + L_g V(x)u + L_\mu V(x) + c_\delta |L_\sigma V(x)| \leq -cV(x). \quad (14)$$

With an associated ES-CLF, we can enforce the state of the uncertain dynamical system (5) to rapidly converge to the desired state with a probability larger than $(1-\delta)$ based on (14).

Remark 2. With more informative data collected for the system dynamics, the bounded uncertainty σ will gradually decrease. Thus, the probability of the control input $u \in \mathcal{U} \in K_{rclf}$ to enforce tracking stability with respect to exponential stability for the uncertain dynamical system (5) is much higher than $(1-\delta)$ in most cases.

C. Learning-based Safety-Stability-Driven Controller (LBSC)

In practical applications, the safety-critical systems are usually subject to control input constraints, such as torque saturation constraints. In this study, using the estimated uncertainty bound estimated by GPs, the LBSC approach is formulated as a quadratic program (QP) controller incorporating the safety barrier (represented by CBF) in (11), stabilization objective (represented by ES-CLF) in (14) and control constraints.

Furthermore, the tradeoff of safety and tracking stability is mediated by setting relaxation variables for safety and stability constraints. The proposed LBSC controller is then formulated as follows:

$$u^*(x) = \underset{u \in \mathbb{R}^m, (\varepsilon, \eta) \in \mathbb{R}}{\arg\min} \frac{1}{2}u^T H(x)u + K_\varepsilon \varepsilon^2 + K_\eta \eta^2, \quad (15)$$

$$\text{s.t.} \quad A_{rzbf}u + b_{rzcbf} \leq \varepsilon, \quad \text{(Safety)}$$
$$A_{rclf}u + b_{rclf} \leq \eta, \quad \text{(Stability)}$$
$$u_{min} \leq u \leq u_{max}, \quad \text{(Control Constraints)}$$

where

$$A_{rzcbf} = -L_g h(x),$$
$$b_{rzcbf} = -L_f h(x) - L_\mu h(x) + c_\delta |L_\sigma h(x)| - \alpha(h(x)),$$
$$A_{rclf} = L_g V(x),$$
$$b_{rclf} = L_f V(x) + L_\mu V(x) + c_\delta |L_\sigma V(x)| + cV(x),$$

$u_{min} \in \mathcal{U}$ and $u_{max} \in \mathcal{U}$ are the lower bound and the upper bound of the control inputs, respectively; $H(x) \in \mathbb{R}^{m \times m}$ is positive definite; c is a positive constant; ε and η are relaxation variables for safety and stability constraints, respectively; K_ε and K_η are two positive constants to penalize safety and tracking stability violation, respectively.

The solution $u^*(x)$ to the QP problem in (15) is always feasible because the relaxation variables ε and η can ensure no conflict among the safety, stability, and control input constraints. Similar to Theorem 3 in [8], it can be proved that $u^*(x)$ is Lipschitz continuous. Moreover, the optimization is not sensitive to the parameters K_ε and K_η as long as they are large enough (e.g. $K_\varepsilon = 10^{30}$, $K_\eta = 10^{20}$). In this way, the safety and stability constraints violations will be penalized heavily when the parameters are set large values.

Remark 3. Note that K_ε (related to safety) is set extremely larger than K_η (related to stabilization) to make the safety guarantees much stricter than the stabilization objective of tracking. As a result, the QP formation in (15) can handle the tradeoff between safety and tracking stability. If no control inputs are satisfying both safety and tracking stability constraints, it would sacrifice tracking performance to guarantee safety constraints.

V. EXPERIMENTS

In this section, the connected cruise control (CCC) system, a typical nonlinear safety-critical system, is examined to illustrate the performance of the proposed LBSC approach. For a CCC system consisting of a team of cars (one controlled car among several human-driven cars), urgent deceleration of the lead car in unexpected situations may cause the car behind to collide with the front car. Moreover, the self-driving car in the CCC system may drive in unsafe states due to model uncertainties. It poses a critical challenge to achieve accurate tracking and safety guarantees for the CCC system in these situations. The LBSC approach is investigated by comparing it against the other three baselines in simulations for the CCC system.

- **GP-based Adaptive sampling (GPAS) Method** [3]: An adaptive sampling based MPC strategy through GPs and cross-entropy. The GPAS is performed for the CCC system by elaborating a cost function measuring the tradeoff between safety and tracking performance.
- **Control barrier function and control Lyapunov function based quadratic programs (CBF-CLF-QPs)** [7]: The ZCBF is utilized for the CBF-CLF-QPs [7] as it provides robustness property under model perturbations as investigated in [8].
- **LBSC-N**: An ablation version of the proposed LBSC algorithm in (15). In the LBSC-N the value of the weight

K_{ε} for safety constraints is equal to the weight K_{η} for stability constraints, i.e., $K_{\eta} = K_{\varepsilon} = 10^{30}$.

A. Experimental Setup

For the CCC system, we consider a chain of five cars following in order on a straight road with uncertain slope or grade, including autonomous car 4 and four human-driven cars including cars 1, 2, 3, and 5. Car 1 leads the chain of five cars with a series of aggressive accelerations and urgent braking events. To form the fleet of five cars, the space headway between a car and its front car is required to range from the minimum safe distance to the maximum tracking distance. With this constraint of the space headway, a car can guarantee safety while rapidly tracking the car in front of it in the presence of urgent deceleration and acceleration. In the CCC system, assume that each car can access its front cars' positions, velocities, and accelerations via vehicle-to-vehicle (V2V) communication [19].

For the CCC system, its states are denoted as $q = [p_1, v_1, a_1, p_2, v_2, a_2, p_3, v_3, a_3, p_4, v_4, a_4, p_5, v_5, a_5]$, where p_i, v_i, and a_i are the position(in m), velocity (in m/s), acceleration (in m/s^2) of the ith car, respectively; and the control inputs are represented as $u = [u_1, u_2, u_3, u_4, u_5]$, where the control input u_i (in Newtons) of the ith car is the total wheel force.

The car dynamics are inspired by work [8]:

$$\begin{bmatrix} \dot{p}_i \\ \dot{v}_i \end{bmatrix} = \begin{bmatrix} v_i \\ a_i \end{bmatrix} = \begin{bmatrix} v_i \\ \frac{-F_r - F_f}{M} \end{bmatrix} + \begin{bmatrix} 0 \\ g\Delta\theta \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix} u_i, \tag{16}$$

where $F_r = f_0 + f_1 v_f + f_2 v_f^2$ is the aerodynamic drag (in Newtons) with constants f_0, f_1, and f_2; $F_f = f_f M g$ is the rolling resistance (in Newtons), and f_f is the rolling resistance coefficient determined empirically; M is the mass of the car, and g is the gravitational acceleration; $\Delta\theta$ is a perturbation to v_i (reflecting unmodeled road grade).

TABLE I
ACCURATE SYSTEM PARAMETER VALUES IN SIMULATIONS

M	1650 kg	g	9.81 m/s^2
k_b	30	k_p	2000
f_f	0.015	f_0	0.1 N
f_1	5 $N \cdot s/m$	f_2	0.25 $N \cdot s^2/m^2$
v_{max}	40 m/s	a_{max}	0.3×9.81 m/s^2
c_a	0.3	c_d	0.3

On the other hand, the control input u_i considering in (16) for ith human-driven cars are described as follows [19]:

$$u_i(t) = k_b(V_i(p_{i+1}(t) - p_i(t) - l_i) - v_i(t)) + k_p(v_{i+1}(t) - v_i), \tag{17}$$

$$V_i(B_i) = \begin{cases} 0 & \text{if } B_i \le B_{st,i} \\ k_i(B_i - B_{st,i}) & \text{if } B_{st,i} < B_i < B_{go,i} , \\ v_{max} & B_i \ge B_{go,i} \end{cases} \tag{18}$$

where k_b and k_p denote two control gains; $B_i = p_{i+1} - p_i - l_i$, and l_i denotes the length of ith car; $B_{st,i}$ and $B_{go,i}$ denote coefficients of the control law used by the ith human-driven car, $i = 2, 3, 5$. For a small following distance $(B_i < B_{go,i})$,
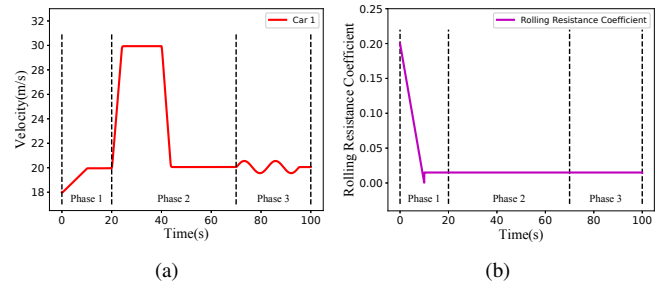

Fig. 2. (a) The velocity of the lead car in three phases. (b) The rolling resistance coefficient of the road.

the car intends to stop, while for a large following distance $(B_i \ge B_{go,i})$, it intends to travel with the speed limit v_{max}. In our experiments, $B_{st,i} = 25m$ and $B_{go,i} = 100m$ are determined empirically.

Other parameter values of the CCC systems are set referred to [11] and [19].

In the simulations, the accurate system parameters are given in Table I. The initial states of the CCC system are set as $q = [\ 0\ m, 18\ m/s, 0m/s^2, 60\ m, 18\ m/s, 0m/s^2, 120\ m, 18\ m/s, 0\ m/s^2, 180\ m, 18m/s, 0m/s^2, 240\ m, 18\ m/s, 0\ m/s^2\ ]$.

In the presence of model uncertainties (i.e., uncertain rolling resistance, aerodynamic drag, and uncertain slope or grade), we aim to control the input of the autonomous car 4 between the human-driven car 3 and car 5 to enable the autonomous car 4 to rapidly converge to a fixed cruising velocity, maintain it and meanwhile keep a safe space headway (between 25 m and 100 m) with its front human-driven car 3 in the chain of five cars.

To study the tracking performance of the LBSC controller in various situations, the velocity of the lead car varies according to the three phases as shown in Fig. 2(a).

Phase 1 ($0s \sim 20s$). The lead car 1 accelerates from $18m/s$ to $20m/s$ with a fixed acceleration and then drives smoothly at a constant speed of v_d.

Phase 2 ($20s \sim 70s$). The lead car 1 rapidly accelerates to reach the velocity of $30m/s$ and then maintains a fixed speed until the $40s$. After that, it urgently decelerates back to the velocity of $20m/s$.

Phase 3 ($70s \sim 100s$). The velocity of the lead car 1 is set to slightly and continuously varying as a sin function.

Road Disturbances. In each phase, the rolling resistance coefficient f_f of the road is shown in Fig. 2(b). In phase 3, A grade perturbation is given by $g\Delta\theta = 2.5sin(0.5t)$ when the simulation time $t \ge 70s$.

Model Uncertainties. Assume that the autonomous car 4 has prior knowledge of nominal models of its dynamics and the other human-driven cars. To illustrate the discrepancies between the nominal and the accurate models, the parameter values of the nominal models known by car 4 in prior are set differently from the accurate values.

- The parameters of the crude model of car 4 in (16) are set as $f_f = 0.2$, $F_r = 0$, $\Delta\theta = 0$.
- The parameters of the other human-driven cars in (18) and (16) are set as $k_b = 20$, $k_p = 1000$, $f_f = 0.2$, $F_r = 0$, $\Delta\theta = 0$.

Safety Constraints. The position of the autonomous car 4 is constrained within an interval between the $B_{st,4}$ and $B_{go,4}$.

$$B_{st,4} \leq p_3 - p_4 \leq B_{go,4}. \tag{19}$$

This safety range aims to maintain a safe following distance as specified by a space headway. In the experiments, we set $B_{st,4} = 25m$, $B_{go,4} = 100m$. The space headway hence ranges from $25m$ to $100m$. The barrier function $h_1(x)$ and $h_2(x)$ can be parameterized as

$$
\begin{aligned}
h_1(x) &= p_4 - p_3 - B_{st,4}, \\
h_2(x) &= -p_4 + p_3 + B_{go,4},
\end{aligned}
\tag{20}
$$

respectively. In the following experiments, we set $B_{st,4} = 25m$, $B_{go,4} = 100m$.

Tracking Performance. The LBSC controller aims to rapidly drive the autonomous car at the desired cruising velocity with high tracking accuracy. This stabilization objective is encoded as an ES-CLF in a standard quadratic form as:

$$V(x) = \frac{1}{2}||v_4 - v_{des4}||^2. \tag{21}$$

In each phase, the tracking errors are evaluated via Mean Absolute Error (MAE) between the desired and tracked velocity of the autonomous car 4:

$$MAE = \frac{1}{T} \sum_{t=1}^{T} || v_4(t) - v_{des}(t) || \tag{22}$$

where T is the number of samples in each phase.

Control Constraints. Similarly to [11], the control input set considered in the CCC system is defined by:

$$U_{CCC} = [u_{min}, \ u_{max}] = [-c_d Mg, \ c_a Mg], \tag{23}$$

where u_{min} and u_{max} are the maximum control value for deceleration and acceleration, respectively. c_d and c_a denote deceleration and acceleration coefficients, respectively.

In the following experiments, each GPs model uses the past $T_s = 30$ observations collected at $50\ Hz$ to predict model uncertainties and road perturbations online. To generate high confidence intervals in (8), we use $\mathscr{D}(x) = \{d \mid \mu(x) - c_\delta \sigma(x) \leq d \leq \mu(x) + c_\delta \sigma(x)\}$, where $c_\delta = 3$. The GPs are implemented and tuned using the Python library scikit-learn [20].

The feedback controller $u(x)$ for the autonomous car 4 can then be obtained by the following QP problem:

$$u^*(x) = \underset{u \in \mathbb{R}^1, (\varepsilon,\eta) \in \mathbb{R}}{\arg\min} \frac{1}{2} u^T H(x) u + K_\varepsilon \varepsilon^2 + K_\eta \eta^2, \tag{24}$$

$$
\begin{aligned}
\text{s.t.} \quad & A_1 u + b_1 \leq \varepsilon, \ A_2 u + b_2 \leq \varepsilon, && \text{(Safety)} \\
& A_3 u + b_3 \leq \eta, && \text{(Stability)} \\
& u_{min} \leq u \leq u_{max}, && \text{(Control Constraints)}
\end{aligned}
$$

where u denotes the control inputs. $H(x) = M^{-2}$, $K_\varepsilon = 10^{30}$, $K_\eta = 10^{20}$, $c_\delta = 3$.

$$
\begin{aligned}
A_1 &= -L_g h_1(x), A_2 = -L_g h_2(x), \\
b_1 &= -L_f h_1(x) - L_\mu h_1(x) + c_\delta |L_\sigma h_1(x)| - \alpha(h_1(x)), \\
b_2 &= -L_f h_2(x) - L_\mu h_2(x) + c_\delta |L_\sigma h_2(x)| - \alpha(h_2(x)), \\
A_3 &= L_g V(x), \\
b_3 &= L_f V(x) + L_\mu V(x) + c_\delta |L_\sigma V(x)| + c V(x),
\end{aligned}
$$

where the corresponding extended class $\mathscr{K}$ function α is simply chosen as $\alpha(h_1) = 5h_1$ and $\alpha(h_2) = 5h_2$, the positive constant is set as $c = 0.6$.
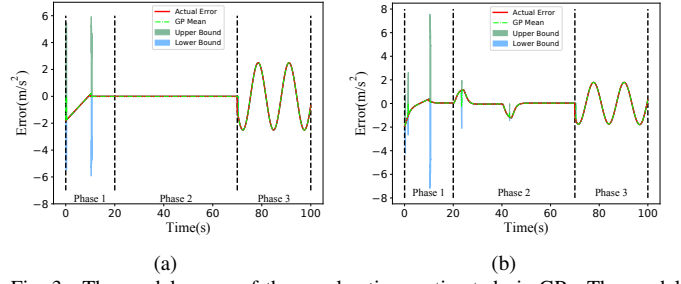

Fig. 3. The model errors of the accelerations estimated via GPs. The model errors of the acceleration of (a) the autonomous car 4, (b) the human-driven car 3.

B. Quantitative Experiments

We build a CCC simulation environment by Python 3.6 to numerically validate the performance of the proposed LBSC controller. The python library CVXOPT is utilized to solve the QP problem in (24), and the average time of solving this QP problem is 2.45 ms. In the experiments, the time of an episode is set as 100 s, and the control frequency is 50 Hz. The average time-consumption of the proposed LBSC algorithm is 49 s for each episode and each control step takes $9.8ms$ on average, which enables the LBSC algorithm to provide real-time control for the autonomous car 4.

In the simulations, the car 4 is controlled to cruise at the desired speed of $v_d = 20m/s$ while keeping a safe space headway ranging from $25m$ to $100\ m$ within its front car 3. The performance of the various algorithms is studied and compared as shown in Fig. 4. The MAE defined in (22) is used to evaluate the tracking errors in each phase.

As shown in Fig. 3, the actual model errors of accelerations of the car 3 and the car 4 both lie within the high confidence uncertainty bounds estimated by GPs in each phase, indicating that GPs can model the unknown discrepancies between the prior model and the actual system. The estimated uncertainties increase sharply at the beginning of simulations and $t = 10s$ when the friction coefficient changes, and decrease quickly with more data gathered. It shows that GPs can learn from experiences and provide online estimations. These estimated errors and uncertainty bounds are incorporated into the LBSC controller to help it adapts to new environment changes and generate smooth controls as depicted in Figs. 4(a).

It can be seen from Fig. 4(b), in **phase 1**, the proposed LBSC approach enables the autonomous car 4 to accelerate to its desired cruising speed of 20 m/s and illustrates faster convergence performance than the GPAS method [3]. Meanwhile, as shown in Fig. 4(c), the distance of the car 4 to its front car 3 is kept within the safe range from 25 m to 100 m.

In **phase 2**, the lead car 1 urgently accelerate then decelerate. In such situations, as illustrated in Fig. 4(c), the LBSC method strictly ensures the specified space headway constraints. On the other hand, as shown in Fig. 4(d), the tracking MAE of the LBSC controller in **phase 2** is much larger than that in **phase 1**. It indicates that the LBSC approach enables the safety to be a higher priority when there is a conflict between safety and tracking stability.
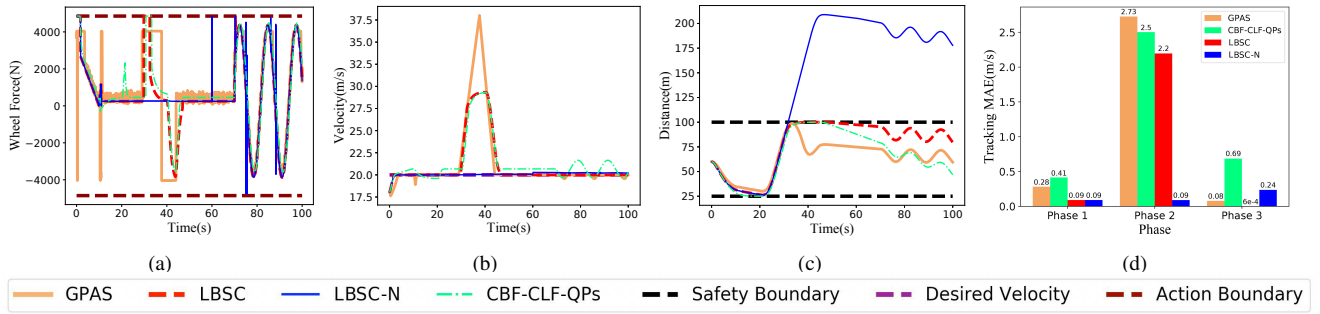
Fig. 4. Simulation results on the CCC system. The goal is to drive the autonomous car 4 to the desired cruising velocity while enforcing the safety constraints that the car 4 keeps a space headway ranging from $25m$ to 100 m with its front car 3. The control input (wheel force) and velocity of the autonomous car 4 are shown in 4(a) and 4(b) , respectively. The space headway between car 4 and car 3 is shown in 4(c). The tracking MAE between the desired and tracked velocity of the car 4 is shown in 4(d).

In **phase 3**, the velocity of the lead car 1 continuously varies. As shown in Fig. 4, the tracking stability and safety of the car 4 are both guaranteed by the LBSC method. On the contrary, it shows from Fig. 4(c) that the LBSC-N algorithm violates the safety constraints in **phase 2** and **phase 3**, although its tracking errors are smaller compared to other methods in **phase 2** as shown in Fig. 4(d).

Furthermore, it can be shown from Figs. 4(a) and 4(b) that the LBSC method generates smoother control input (wheel force) and velocity curves than those of the GPAS method. Besides, as shown in Fig. 4(d), the tracking MAE of the LBSC method is much less than the GPAS and CBF-CLF-QPs methods [7]. It implies that the LBSC controller can achieve better tracking performance.

VI. CONCLUSION

In this paper, a learning-based control algorithm (LBSC) was proposed for nonlinear safety-critical systems subject to control input constraints under model uncertainties. GPs were employed to learn the model errors online between the nominal model and the actual system dynamics. Specifically, using the uncertainty bound estimated via GPs, a safety barrier and a stability constraint were proposed to achieve safety (represented by ZCBF) guarantees and stabilization objective (represented by ES-CLF) for the uncertain safety-critical systems, respectively. Considering control constraints, the safety barrier and stability constraint were unified in a QP to mediate the tradeoff between safety guarantees and tracking performance. The effectiveness of the LBSC algorithm was illustrated on the safety-critical CCC system under the high change rate of the acceleration and model uncertainties. In future work, the experimental validations of the proposed LBSC method will be performed on wheeled and aerial robots.

REFERENCES

[1] N. Hovakimyan, C. Cao, E. Kharisov, E. Xargay, and I. M. Gregory, "L 1 adaptive control for safety-critical systems," *IEEE Control Systems Magazine*, vol. 31, no. 5, pp. 54–104, 2011.

[2] J. C. Knight, "Safety critical systems: challenges and directions," *Proceedings of the 24th International Conference on Software Engineering. ICSE 2002*, pp. 547–550, 2002.

[3] T. Y. Teck, A. Kunapareddy, and M. Kobilarov, "Gaussian process adaptive sampling using the cross-entropy method for environmental sensing and monitoring," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6220–6227, 2018.

[4] C. J. Ostafew, A. P. Schoellig, and T. D. Barfoot, "Robust constrained learning-based nmpc enabling reliable mobile robot path tracking," *I. J. Robotics Res.*, vol. 35, pp. 1547–1563, 2016.

[5] S. Schaal and C. G. Atkeson, "Learning control in robotics," *IEEE Robotics & Automation Magazine*, vol. 17, no. 2, pp. 20–29, 2010.

[6] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," *2019 18th European Control Conference (ECC)*, pp. 3420–3431, 2019.

[7] 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, pp. 3861–3876, 2017.

[8] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames, "Robustness of control barrier functions for safety critical control," in *ADHS*, 2015.

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

[10] G. Wu and K. Sreenath, "Safety-critical control of a 3d quadrotor with range-limited sensing," in *Dynamic Systems and Control Conference*, vol. 50695. American Society of Mechanical Engineers, 2016, p. V001T05A006.

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

[12] H. K. Khalil, "Nonlinear systems," *Upper Saddle River*, 2002.

[13] L. Wang, E. Theodorou, and M. Egerstedt, "Safe learning of quadrotor dynamics using barrier certificates," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2460–2465, 2017.

[14] C. E. Rasmussen and C. K. I. Williams, "Gaussian processes for machine learning," in *Adaptive computation and machine learning*, 2005.

[15] F. Berkenkamp, R. Moriconi, A. P. Schoellig, and A. Krause, "Safe learning of regions of attraction for uncertain, nonlinear systems with gaussian processes," *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 4661–4666, 2016.

[16] B. Schölkopf, A. J. Smola, F. Bach *et al.*, *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[17] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic regret bounds for gaussian process optimization in the bandit setting," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, 2012.

[18] 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.

[19] C. R. He, J. I. Ge, and G. Orosz, "Data-based fuel-economy optimization of connected automated trucks in traffic," *2018 Annual American Control Conference (ACC)*, pp. 5576–5581, 2018.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, G. Louppe, P. Prettenhofer, R. Weiss, V. Dubourg, J. VanderPlas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.