

Implementation Resource Allocation for Collision-Avoidance Assistance Systems Considering Driver Capabilities

Zejiang Wang[✉], Graduate Student Member, IEEE, Adrian Cosio, and Junmin Wang[✉], Senior Member, IEEE

Abstract—Various collision-avoidance assistance (CAA) systems, such as automatic emergency braking (AEB) and lane-keeping assistance (LKA), have been developed in the last decades to enhance the active safety of ground vehicles. Meanwhile, more electronic computing units (ECUs) have been embedded inside a vehicle to support the diversified CAA systems, which complicate the automotive electrical/electronic architecture and increase the cost. Instead of adding *extra* ECUs, we propose to allocate the *existing* implementation resources, i.e., the available processor time and memory space to the CAA systems, per individual driver's maneuver capabilities. As an illustrative example, we first show that two drivers can exhibit distinct maneuvers in a pre-crash situation on highway, according to which they can be classified as either steering-oriented or braking-oriented. Then, we design two CAA systems: an AEB and an LKA, based on the ultra-local model predictive control method. Furthermore, we show that by adjusting the prediction horizons of the two controllers, the implementation resources can be allocated to the two CAA systems in different fashions, which yields three control modes: standard mode, steering-enhanced mode, and braking-enhanced mode. Finally, by comparing the control performance of each driver-type/control-mode pair through both CarSim-Simulink joint simulations and driver-in-the-loop simulator experiments, we demonstrate that by allocating more resources to compensate for the weakness of a driver's maneuver, the CAA systems can provide enhanced driving safety by consuming the same overall amount of the implementation resources.

Index Terms—Collision avoidance, model-free control, model predictive control, resources allocation.

I. INTRODUCTION

VARIOUS collision avoidance assistance (CAA) systems for ground vehicles have been developed in the last three decades [1]. Noteworthy examples include forward collision warning (FCW) [2], automatic emergency braking

(AEB) [3], [4], and lane-keeping assistance (LKA) system [5]. Model predictive control (MPC) has been extensively applied for CAA systems to treat the safety constraints and the nonlinearities of the driver-vehicle system under emergencies. For instance, authors in [6] propose separately an LKA system considering the ranges of tire sideslip angles and an adaptive cruise controller respecting acceleration and inter-vehicle distance constraints. Authors in [7] design an integrated braking and steering controller for LKA while satisfying lateral offset constraints. The work in [8] contributes a shared steering controller helping a driver to stay on road while following driver's inherent intention.

Although the effectiveness of MPC for collision avoidance and lane-keeping assistance is widely recognized, MPC incurs heavy online computational load [9] and occupies more memory space than non-optimization-based control strategies [10]. Embedding more electronic computing units (ECUs) into a passenger car can meet the excessive hardware requirements of MPC. However, they complicate the automotive electronic architecture, degrade system reliability, and increase cost [11].

From an ergonomic perspective, CAA systems should adapt their assistance to compensate for the weakness of a driver's maneuver while limiting its intrusion on tasks that the driver is inherently competent to fulfill [12]. Instead of introducing *extra* ECUs, we propose to allocate the *existing* implementation resources, including the available processor time and memory space, to different CAA systems per driver's capabilities.

Capability-based CAA systems design [13] typically includes three major steps. Firstly, a driver model is derived from the historical driving data [14], [15]. Secondly, the driver's longitudinal and lateral control capabilities are evaluated based on the fitted driver model parameters. For instance, authors in [16] quantify drivers' steering skills according to the highest speed at which a driver can pass a lane-change test without colliding any cones. Moreover, experimental results indicate a strong correlation between the identified preview time/neuromuscular lag and the steering skill level. In parallel, authors in [17] define driver's braking skill according to the minimum time headway to the front car, and conclude that the control gains of the inter-vehicle distance and relative speed can reflect the braking skill. Thirdly, CAA systems adapt to the evaluated driving capabilities. For example, authors in [18] propose an FCW system by first classifying a driver as either reckless, general, or cautious,

Manuscript received September 20, 2020; revised December 10, 2020, March 3, 2021, May 12, 2021, and September 20, 2021; accepted October 1, 2021. This work was supported in part by the National Science Foundation under Award 1901632. The Associate Editor for this article was K. Yi. (Corresponding author: Junmin Wang.)

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Institutional Review Board (IRB) at The University of Texas at Austin.

Zejiang Wang and Junmin Wang are with Walker Department of Mechanical Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: wangzejiang@utexas.edu; jwang@austin.utexas.edu).

Adrian Cosio is with Sandia National Laboratories, Albuquerque, NM 87185 USA (e-mail: adriancosio14@gmail.com).

Digital Object Identifier 10.1109/TITS.2021.3117918

and then adapting the warning threshold accordingly. Authors in [19] design a model predictive lane change system by including a driver's steering ability as an additional constraint.

However, the implementation resource consumption is rarely considered in the existing capability-based CAA systems. On the other hand, resource-aware automotive control, which either maximizes the control performance within the given implementation resources or minimizes the resource consumptions with guaranteed control performance, has recently been proposed [20]–[22]. Nevertheless, a driver's capabilities are seldom considered in scheduling the resources. Filling the gap between capability-based CAA systems and resource-aware automotive control, we propose to allocate the existing implementation resources to CAA systems according to the driver's capabilities. To the best of the authors' knowledge, no previous work explicitly showed the feasibility and safety benefits of this idea.

As an illustrative example, we first show in Section II that two drivers can behave differently in a highway pre-crash scenario. Then, we illustrate in Section III that the discrepancies in driver maneuver can be exploited to classify a driver as either steering-oriented or braking-oriented. Section IV formulates an AEB system and an LKA system based on the ultra-local model predictive control (ULMPC) method, and illustrates how to create three control modes: standard mode, steering-enhanced mode, and braking-enhanced mode by adjusting the prediction horizons of the two predictive controllers. Section V compares each driver-type/control-mode pair (steering-oriented driver plus standard control mode, steering-oriented driver plus steering-enhanced control mode, etc.) via CarSim-Simulink joint simulation and driver-in-the-loop driving simulator experiments. Simulation and experimental results demonstrate the feasibility and benefits of the proposed resources allocation philosophy based on driver capabilities. Finally, Section VI concludes this paper.

The contribution of this paper is threefold. Firstly, we demonstrate that a driver can be classified as either braking-oriented or steering-oriented via a support vector machine model. Secondly, we formulate two ULMPC-based CAA systems for preventing rear-end and run-off-road collisions. Thirdly, through simulations and driving simulator experiments, we demonstrate the feasibility and safety benefits of allocating the implementation resources to CAA systems per driver's capabilities.

II. BEHAVIORS OF DRIVERS IN A PRE-CRASH SCENARIO

In this section, we first describe a representative pre-crash scenario. Then, we use driver model simulations and driver-in-the-loop experiments to demonstrate that two drivers can manifest different maneuvers in this critical setting.

A. Representative Pre-Crash Scenario

Fig. 1 shows the pre-crash scenario under investigation. The subject vehicle (SV) follows a front vehicle (FV) in the left lane. Meanwhile, another target vehicle (TV) proceeds relatively slower in the right lane. When the distance between FV and SV decreases below a threshold, FV yields a brake

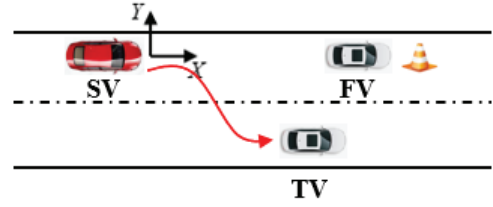


Fig. 1. Representative pre-crash scenario.

in front of an obstacle. To escape FV, the SV driver initiates a right turn and a brake at the same time. SV must always remain on the road and avoid rear-ending TV after the lane-change.

We study this specific scenario for three reasons. Firstly, the given scenario requires both steering and braking maneuvers from the SV driver, so that CAA's assistance in both longitudinal and lateral control is required. Secondly, the authors' previous work [14], [15] utilized such a scenario to validate driver models. Lastly, the designed scenario covers two of the most frequent types of traffic accidents in the U.S., namely, the rear-end collision and the run-off-road collision [23]. This critical scenario has been extensively studied for collision analysis and prevention [24], [25].

B. Maneuver Discrepancies Demonstrated in Simulation

We first use the longitudinal and lateral driver models validated in [14] and [15] to simulate the SV driver's steering and braking maneuvers in the critical scenario in Fig. 1. Special attention will be paid to revealing maneuver discrepancies between two human drivers.

The longitudinal driver model in [14] describes the desired acceleration during car-following, as:

$$a_{xr}(t) = C_1 [\Delta X(t - T_d) - X_r(t)] - C_2 [\Delta v_x(t - T_d)], \quad (1)$$

with $\Delta X = X_f - X$, $\Delta v_x = v_x - v_{xf}$. In (1), X and v_x are the longitudinal position and speed of SV. X_f , v_{xf} represent the counterparts of the vehicle in front of SV. $X_r(t)$ shows the preferred inter-vehicle distance of the SV driver. Positive parameters: C_1 , C_2 , and T_d are the distance-error gain, speed-error gain, and human response delay.

As shown in Fig. 1, the car immediately in front of SV would gradually shift from FV to TV during the lane change. Hence, we introduce the attention weights validated in [24]:

$$\begin{cases} \alpha_{TV}(t) = \frac{\tan(y_n(t)/Y)}{\tan(1)}, \\ \alpha_{FV}(t) = 1 - \alpha_{TV}(t). \end{cases} \quad (2)$$

In (2), $0 \leq \alpha_{TV}(t)$, $\alpha_{FV}(t) \leq 1$, are the attention weights that the SV driver assigns to TV and FV, respectively. $y_n(t)$ is the lateral offset of SV w.r.t the original path before changing lane, and $Y = 3.66$ is the U.S. standard highway lane width in meter.

Inspired by [24], we formulate the SV driver's desired longitudinal acceleration under the influence of

TABLE I
DRIVER MODEL PARAMETER SETS

	Lateral Parameters				Longitudinal Parameters		
	G_h	T_h	T_p	K_{ff}	C_1	C_2	T_d
Driver1	0.59	0.32	0.75	0.22	0.07	0.26	0.8
Driver15	0.3	0.05	0.85	0.2	0.005	0.02	1.3

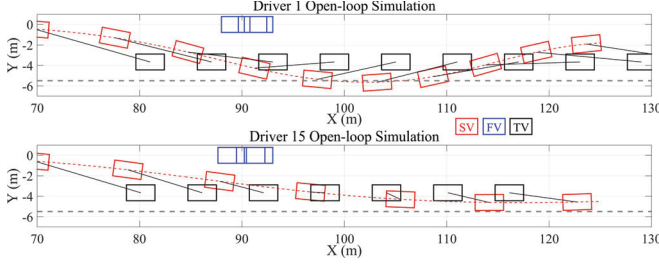


Fig. 2. Open-loop vehicle trajectories comparison.

both FV and TV as:

$$a_{xr}^*(t) = a_{FV}(t) a_{xr}^{FV}(t) + a_{TV}(t) a_{xr}^{TV}(t), \quad (3)$$

where $a_{xr}^{FV}(t)$ and $a_{xr}^{TV}(t)$ are the SV driver's desired longitudinal accelerations induced by FV and TV, respectively.

In parallel, the lateral driver model reads as [15]:

$$\delta_{sw}(t) = \frac{G_h}{1 + T_h s} [Y_{des}(t) - (Y(t) + v_x(t) \sin \psi(t) T_p)] + K_{ff} \gamma_{des}(t). \quad (4)$$

In (4), $\delta_{sw}(t)$ is the simulated steering wheel angle, G_h indicates the feedback gain, T_h is the neuromuscular lag, $Y_{des}(t)$ is the targeted lateral position, $Y(t) + v_x(t) \sin \psi(t) T_p$ represents the predicted lateral position in T_p seconds along the current vehicle heading angle $\psi(t)$, K_{ff} is the feedforward gain, and $\gamma_{des}(t)$ is the feedforward vehicle yaw rate.

The longitudinal and lateral driver models were validated in [14], [15] with human subject experiments. Fitting the recorded maneuver data with the model outputs yielded the drivers' parameter sets. For instance, the model parameters of Driver1 and Driver15 are replicated in Table I. Per [16], [17], the more significant error-feedback gains $C_{1,2}$, and the shorter reaction delay T_d imply that Driver1 is more skilled in longitudinal control than Driver15. Meanwhile, the longer preview time T_p and the smaller neuromuscular lag T_h suggest that Driver15 is more adept in lateral control than Driver1.

To confirm the maneuver differences between Driver1 and Driver15, we conduct CarSim-Simulink joint simulation. We first use a lookup table to convert $a_{xr}^*(t)$ in (3) into the driver-induced master cylinder pressure P_{MC}^d . Then, we inject the simulated P_{MC}^d and δ_{sw} in (4) into the CarSim model, where the vehicle configurations can be found in [9]. The simulated vehicle trajectories and driver maneuvers are compared in Fig. 2 and Fig. 3. In Fig. 2, the red, blue, and black blocks represent the SV, FV, and TV. A solid line connects the centers of SV and TV. The dashed line indicates the road edge.

Therefore, the simulated Driver1 yields adequate brake control and avoids rear-ending FV and TV. However, it also

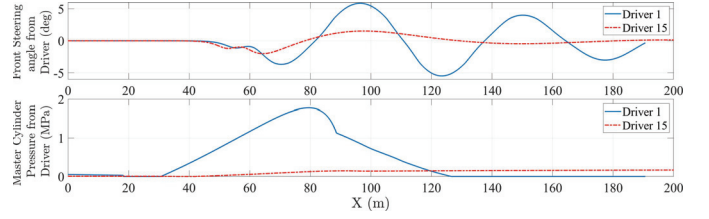


Fig. 3. Open-loop driver maneuvers comparison.

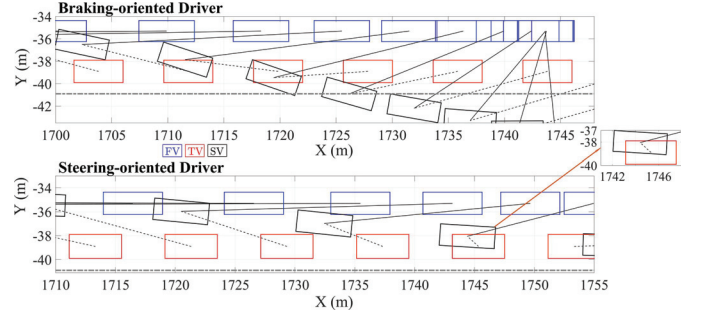


Fig. 4. Representative vehicle trajectories from driving simulator experiments. The blue, red, and black blocks indicate the FV, TV, and SV, respectively. A solid line connects the centers of FV and SV, while a dashed line connects the centers of SV and TV. A dash-dotted line indicates the road edge.

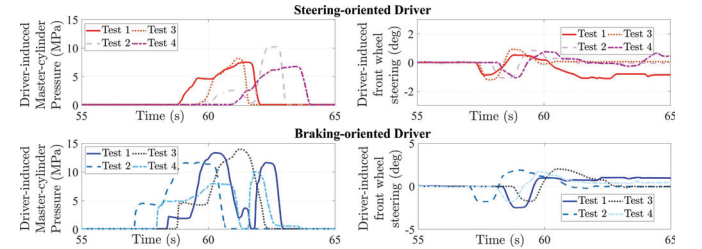


Fig. 5. Four representative maneuver records from driving simulator experiments. The upper subplot exhibits the master-cylinder pressure and the front wheel steering from a steering-oriented driver. The lower subplot shows the counterparts of a braking-oriented driver.

produces oscillating steering wheel angles, forcing SV to run off-road. In contrast, the simulated Driver15 is more capable of steering control but has a deficient braking control, resulting in rear-end collision with TV. Note that in Fig. 2 and Fig. 3, solely the simulated driver models provide the braking and steering commands while the CAA systems are not activated. Hereafter, we denote Driver1 and Driver15 as the braking-oriented and the steering-oriented driver, respectively.

C. Maneuver Discrepancies Shown in Simulator Experiments

In addition to CarSim simulations, driving simulator experiments were also performed to demonstrate maneuver differences between two drivers. Two drivers repeatedly operated a driving simulator several times. Representative vehicle trajectories and maneuvers, without CAA system involvement, are compared in Fig. 4 and Fig. 5, respectively. The simulator setup is explained in Section V.

As shown in Fig. 4, one of the two drivers could keep the inter-vehicle distance between the SV and the TV but went

across the road boundary. In contrast, another driver could maintain the SV on the road, but rear-ended the slowly moving TV. Fig. 5 further illustrated the underlying reasons: one driver yielded fast-rising and adequate master cylinder pressure to decelerate the SV. Meanwhile, he also produced a large and oscillatory steering wheel angle, which pushed SV out of the lane. On the contrary, another driver issued a steady steering command but an insufficient braking command. In summary, Fig. 4 and Fig. 5 reveal again that two drivers can manifest different maneuvers in this critical scenario, because of their unbalanced steering and braking control capabilities.

III. DRIVER-TYPE CLASSIFICATION

This section shows that the driver maneuver discrepancies revealed in Section II can be exploited to classify a driver as either steering-oriented or braking-oriented in the pre-crash scenario in Fig. 1. Driver-type classification is generally conducted via machine-learning (ML) approaches, such as decision tree, neural network, support vector machine (SVM), and fuzzy logic [26]–[29]. Among all the methods, SVM maintains two particular advantages: it finds a global solution and escapes local minima [28], and it is more robust to sample outliers [26]. Interestingly, SVM was typically used to classify driving characteristics in only one dimension: either longitudinal or lateral. For instance, authors in [26] identified a driver as normal or skilled by observing the steering maneuvers. Authors in [27] classified a driver as aggressive or normal based on the longitudinal speed and throttle input. As an exception, ten attributes, including the steering maneuver for lateral control and the throttle input for longitudinal control, were considered in [29]. However, the purpose was to evaluate the *overall* driving skill without differentiating the lateral and the longitudinal control capabilities.

Here, we employ SVM to categorize a driver as either steering-oriented or braking-oriented. Representative attributes for classification are selected as: a) the rate of change of the master-cylinder pressure when a driver initiated a brake action and b) the most negative front (road) wheel steering angle during the lane-change process. As shown in Fig. 5, a braking-oriented driver shall produce attributes with large magnitudes, whereas a steering-oriented driver shall yield substantially smaller ones.

Six participants were recruited, with each operating a driving simulator several times. In total, 484 pairs of feature samples were recorded. We labeled the six drivers as either steering-oriented or braking-oriented according to their *overall* performance during the driving simulator studies. Drivers who frequently collided with TV but could remain SV on the road were labeled as steering-oriented. In contrast, drivers often went across the road boundary but did not rear-end TV were labeled as braking-oriented. Because of driver behavior variations, one steering-oriented driver could sometimes act like a braking-oriented driver and vice versa. However, we ignored the variations and consistently labeled each driver according to their typical behaviors. Similar simplification was also adopted in [26], [29] to reduce the labeling efforts.

SVM classification was performed via MATLAB Classification Learner. The tunable hyperparameters include the kernel

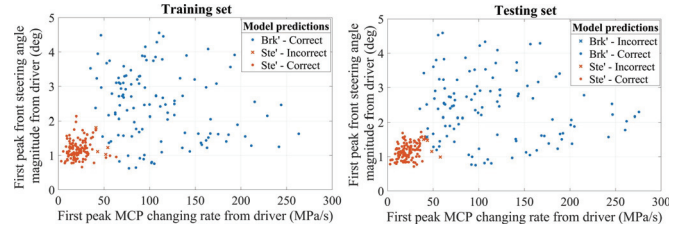


Fig. 6. Driver type classification via SVM.

function, the box constraint level, and the option for data standardization. A prescreening of applicable kernel functions reveals that the easiest-to-interpret linear kernel overperforms the more complicated quadratic, cubic, or RBF kernels. Then, as the two attributes have different scales, we enabled data standardization to improve the fit. Finally, the box constraint level, which corresponds to the soft margin penalty c in SVM's primal equation, was tuned by MATLAB SVM hyperparameter optimizer.

We randomly selected 50% of the data samples for SVM training, and the rest was used for SVM testing. The classification results of both the training dataset and the testing dataset are demonstrated in Fig. 6. In the training dataset, four samples from the braking-oriented drivers were mistakenly recognized as the steering-oriented data points, while in the testing dataset, seven samples from the braking-oriented drivers were wrongly acknowledged as the steering-oriented data points, and one sample from the steering-oriented drivers was erroneously identified as the braking-oriented data point. The recognition accuracy reached 98.3% and 96.7% for the training and testing datasets, respectively.

IV. ULMPC DRIVING-ASSISTING SYSTEM

To assist the SV driver in the critical scenario in Fig. 1, we develop two CAA systems: a master-cylinder pressure controller for braking assistance and an LKA system for steering assistance. Moreover, we demonstrate how to allocate the implementation resources to the two CAA systems via parameter tunings, which yield three control modes: standard mode, steering-enhanced mode, and braking-enhance mode.

We utilize ultra-local model predictive control (ULMPC) for CAA systems design. ULMPC was recently proposed in [30]. We adopt this novel algorithm for three reasons: Firstly, it explicitly decouples the driver-vehicle dynamics into a longitudinal and a lateral ultra-local model, which facilitates demonstrating the dilemma between the control assistance and the implementation resources consumption. Secondly, its model-free nature [31] provides inherent robustness against modeling errors and external disturbances. Thirdly, its control performance has been compared with a model-based predictive controller [30], and the results were promising.

A. Braking-Assistance System Design

The braking-assistance system provides extra master-cylinder pressure to maintain the time-to-collision (TTC) of the SV to the TV above a given threshold. We hereafter denote

the TTC as κ . The longitudinal ultra-local model reads:

$$\dot{\kappa}(t) = f_{\kappa}(t) + \alpha_{\kappa} P_{MC}^c(t). \quad (5)$$

In (5), P_{MC}^c is the assisted master-cylinder pressure in MPa. Constant α_{κ} is tuned to make the magnitudes of $\dot{\kappa}(t)$ and $\alpha_{\kappa} P_{MC}^c(t)$ close [31]. The offset term $f_{\kappa}(t)$ is assumed piecewise constant and updated at each sampling step. With $\hat{\kappa}(t)$ estimated by the algebraic differentiator in [32], the offset term $f_{\kappa}(t)$ in (5) can be approximated as:

$$f_{\kappa}(t) \approx \hat{f}_{\kappa}(t) = \hat{\kappa}(t) - \alpha_{\kappa} P_{MC}^c(t - T_s^{\kappa}), \quad (6)$$

where $P_{MC}^c(t - T_s^{\kappa})$ is the assisted master-cylinder pressure at the previous sampling step.

Combining (6) with (5) leads to:

$$\begin{aligned} \dot{\kappa}(t) &\approx \hat{\kappa}(t) - \alpha_{\kappa} P_{MC}^c(t - T_s^{\kappa}) \\ &\quad + \alpha_{\kappa} [P_{MC}^c(t - T_s^{\kappa}) + \Delta P_{MC}^c(t)] \\ &= \hat{\kappa}(t) + \alpha_{\kappa} \Delta P_{MC}^c(t). \end{aligned} \quad (7)$$

Equation (7) is then discretized with the sampling period T_s^{κ} :

$$\kappa(k+1) = \kappa(k) + \alpha_{\kappa} T_s^{\kappa} \Delta P_{MC}^c(k) + T_s^{\kappa} \hat{\kappa}(k). \quad (8)$$

With $x^{mc} := \kappa$, $u^{mc} := \Delta P_{MC}^c$, Eq. (8) is restated as:

$$x_{k+1|k}^{mc} = x_{k|k}^{mc} + B u_{k|k}^{mc} + d^{mc}(k). \quad (9)$$

where $x_{k|k}^{mc}$ indicates the TTC at the current step, $x_{k+1|k}^{mc}$ is the one-step predicted TTC, and $u_{k|k}^{mc}$ represents the assisted pressure increment at the current step.

Then, the cost function to minimize is formulated as:

$$\begin{aligned} J_{mc} = & \sum_{i=0}^{H_c-1} \|u_{k+i|k}^{mc}\|_{R_{\Delta p}}^2 + \|P_{MC}^c(k-1) + u_{k|k}^{mc}\|_{R_p}^2 + \dots \\ & + \|P_{MC}^c(k-1) + u_{k|k}^{mc} + u_{k+1|k}^{mc} + \dots + u_{k+H_c-1|k}^{mc}\|_{R_p}^2 \\ & + \rho_{\kappa} \zeta_{\kappa}. \end{aligned} \quad (10)$$

In (10), the first term restricts the control fluctuation within the control horizon H_c . The middle terms penalize the control intervention, with $P_{MC}^c(k-1)$ as the assisted command at the last step. The last term is the violation penalty w.r.t. the TTC threshold, with ρ_{κ} as the weight and ζ_{κ} as the slack variable.

Finally, the constrained optimization problem formulates as:

$$\begin{aligned} u_{k|k}^{mc*} = & \min J_{mc} \\ s.t. & x_{k+i+1|k}^{mc} = x_{k+i|k}^{mc} + B u_{k+i|k}^{mc} + d^{mc}(k), \\ & i = 0 \dots H_p, \\ & -u_{\max}^{mc} \leq u_{k+i|k}^{mc} \leq u_{\max}^{mc}, \\ & i = 0 \dots H_c - 1, \\ & u_{k+i|k}^{mc} = 0, i = H_c \dots H_p, \\ & 0 \leq P_{MC}(k-1) + \Delta P_{MC}^d(k) \\ & + \sum_{j=0}^{j_+} u_{k+j|k}^{mc} \leq P_{MC}^{\max}, \\ & j_+ = 0 \dots H_c - 1, \\ & \kappa_{\min} - \zeta_{\kappa} \leq x_{k+i|k}^{mc}, \end{aligned} \quad (11.a) \quad (11.b) \quad (11.c) \quad (11.d)$$

$$i = 1 \dots 1 + H_p. \quad (11.e)$$

(11)

Eq. (11.a) indicates the longitudinal ultra-local model. Eqs. (11.b) and (11.c) limit the increments of the assisted master-cylinder pressure within the prediction horizon H_p , where $u_{\max}^{mc}$ is the maximum increment value. Eq. (11.d) constrains the overall master cylinder pressure (from both driver and control assistance), with $P_{MC}^{\max}$ as the overall pressure threshold. In (11.d), $P_{MC}(k-1) = P_{MC}^d(k-1) + P_{MC}^c(k-1)$ is the total master cylinder pressure at the last step. $\Delta P_{MC}^d(k) \approx \hat{P}_{MC}^d(k) T_s^{\kappa}$ is the pressure increment from a driver at the current step, where $\hat{P}_{MC}^d(k)$ can be algebraically estimated from [32]. Finally, (11.e) restricts the predicted TTC within the prediction horizon, with $\kappa_{\min}$ as the minimum TTC that SV should maintain. We implement soft constraints to guarantee iterative feasibility.

Solving (11) yields the optimal control increment $u_{k|k}^{mc*}$, and the optimally assisted master cylinder pressure becomes:

$$P_{MC}^{c*}(k) = P_{MC}^c(k-1) + u_{k|k}^{mc*}. \quad (12)$$

Problem formulation in (11) is similar to [7], where the predicted TTC triggers the control intervention.

B. Steering-Assistance System Design

The LKA system corrects the driver's front steering angle to maintain the SV on the road. Seeing SV as a rectangle, we define $(Y(t) := \min(Y_{fl}, Y_{fr}, Y_{rl}, Y_{rr}))$, with $Y_{fl}, Y_{fr}, Y_{rl}, Y_{rr}$ as the lateral coordinates of SV's four vertices. The lateral ultra-local model [33] reads:

$$\ddot{Y}(t) = f_Y(t) + \alpha_Y \delta_f^c(t). \quad (13)$$

In (13), $\delta_f^c(t)$ is the assisted steering angle. The constant α_Y is tuned to make the magnitudes of $\ddot{Y}(t)$ and $\alpha_Y \delta_f^c(t)$ close. The piecewise constant $f_Y(t)$, similar to $f_{\kappa}(t)$ in (5), is periodically updated. The nonlinear coupling between the longitudinal and lateral dynamics of the vehicle is implicitly maintained in the offset terms $f_{\kappa}(t)$ and $f_Y(t)$.

With $\ddot{Y}(t)$ being algebraically estimated online [30], the offset term $f_Y(t)$ in (13) is approximated as:

$$f_Y(t) \approx \hat{f}_Y(t) = \hat{\ddot{Y}}(t) - \alpha_Y \delta_f^c(t - T_s^Y), \quad (14)$$

where $\delta_f^c(t - T_s^Y)$ is the steering correction at the last step, T_s^Y is the steering-system sampling period. T_s^Y is in general different from T_s^{κ} in (7). Substituting (14) back into (13) yields:

$$\begin{aligned} \ddot{Y}(t) &\approx \hat{\ddot{Y}}(t) - \alpha_Y \delta_f^c(t - T_s^Y) + \alpha_Y \delta_f^c(t) \\ &= \hat{\ddot{Y}}(t) + \alpha_Y \Delta \delta_f^c(t). \end{aligned} \quad (15)$$

Discretizing (15) with T_s^Y leads to:

$$\begin{aligned} \begin{bmatrix} Y(k+1) \\ \dot{Y}(k+1) \end{bmatrix} &= \begin{bmatrix} 1 & T_s^Y \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Y(k) \\ \dot{Y}(k) \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 \\ \alpha_Y T_s^Y \end{bmatrix} \Delta \delta_f^c(k) + \begin{bmatrix} 0 \\ \hat{\ddot{Y}}(k) T_s^Y \end{bmatrix}. \end{aligned} \quad (16)$$

With $x^{st} := [Y(t), \dot{Y}(t)]^T$, $u^{st} := \Delta \delta_f^c(t)$, Eq. (16) is reformulated as:

$$x_{k+1|k}^{st} = Ax_{k|k}^{st} + Bu_{k|k}^{st} + d^{st}(k). \quad (17)$$

In (17), $x_{k|k}^{st}$ indicates the lateral states at the current step, $x_{k+1|k}^{st}$ is the one-step predicted lateral states, and $u_{k|k}^{st}$ represents the steering control increment at the current step.

The cost function for steering assistance is formulated as:

$$J_{st} = \sum_{i=0}^{H_c-1} \left\| u_{k+i|k}^{st} \right\|_{R_{\Delta\delta}}^2 + \left\| \delta_f^c(k-1) + u_{k|k}^{st} \right\|_{R_\delta}^2 + \dots + \left\| \delta_f^c(k-1) + u_{k|k}^{st} + u_{k+1|k}^{st} + \dots + u_{k+H_c-1|k}^{st} \right\|_{R_\delta}^2 + \rho_Y \zeta_Y. \quad (18)$$

Similar to (10), the first term limits the control fluctuation. The middle terms penalize the control intervention, with $\delta_f^c(k-1)$ as the assisted command at the previous step. The last term is the violation penalty w.r.t. the threshold on $Y(t)$, with ρ_Y as the penalty weight and ζ_Y as the slack variable.

Finally, the steering assistance problem is formulated as (19), shown at the bottom of the page.

Equation (19.a) indicates the lateral ultra-local model dynamics. Equations (19.b) and (19.c) limit the assisted steering increments within the prediction horizon H_p , where $u_{\max}^{st}$ is the maximum steering increment. Equation (19.d) limits the overall front steering angle (from driver and control intervention). $\delta_f^{\max}(k)$ is the maximum steering angle within the stability envelope [34]. $\delta_f(k-1) = \delta_f^d(k-1) + \delta_f^c(k-1)$ is the overall front steering angle at the last step. $\Delta \delta_f^d(k) \approx \hat{\delta}_f^d(k) T_s^Y$ indicates the steering increment from the driver at the current step, where $\hat{\delta}_f^d(k)$ is algebraically estimated from [32]. Finally, Eq. (19.e) restricts the predicted $Y(t)$ within the prediction horizon, with $Y_{\min}$ as the minimum lateral coordinate. We implement soft constraints to guarantee iterative feasibility. Solving (19) yields the optimal steering angle increment $u_{k|k}^{st*}$, and the final steering intervention at the step k becomes:

$$\delta_f^{c*}(k) = \delta_f^c(k-1) + u_{k|k}^{st*}. \quad (20)$$

Note that the proposed CAA systems are different from the controllers in [30]. In [30], ULMPC was employed for

automated vehicle path-tracking without human intervention, whereas the CAA systems work in parallel with driver manipulations. Besides, the cost functions in [30] focused on minimizing the path-tracking errors, but the cost functions (10) and (18) penalize the potential collision hazards. To the best of our knowledge, this is the first paper to employ ULMPC for CAA system design.

C. Resources Allocation via Parameter Tuning

The two CAA systems consume implementation resources, e.g., the available processor time and memory space. By tuning the prediction horizons H_p in (11) and (19), we can allocate the resources to the two CAA systems in different ways, ultimately creating three control modes. For instance, we have: a) Standard mode: $H_p^{steer} = 45$, $H_p^{brake} = 45$; b) Braking-enhanced mode: $H_p^{steer} = 40$, $H_p^{brake} = 50$; and c) Steering-enhanced mode: $H_p^{steer} = 50$, $H_p^{brake} = 40$. The names of the three control modes come from the fact that an extended preview horizon can enhance the assistance level while also consuming more implementation resources. For simplicity, the control horizons H_c in the two CAA systems are uniformly fixed as one, and the sampling periods T_s^x , T_s^Y are tuned as 0.008s and 0.016s, respectively.

We first show the resource consumptions of the three control modes in Matlab. Define the computational load [35] as:

$$L_{cpu} = T_{WCET} / T_s, \quad (21)$$

with T_{WCET} as the worst-case execution time and T_s as the control sampling period. Then, we summarize in Table II the computational loads of the two CAA systems in different control modes. STE and BRK stand for the steering- and the braking-assistance system, respectively. 40, 45, and 50 reflect the prediction steps.

From Table II, we can conclude that the three control modes entail almost the same overall processor load, but different modes alter the relative ratios. The standard control mode allots nearly equal computational resources to the two CAA systems. However, the steering-enhanced mode would grant more computational resources to the LKA system, and vice versa. Then, the occupied memory spaces of the three control modes, gauged by the *MEX* file sizes, are compared in Table III. Again, the overall memory consumptions of the three control modes remain practically the same, but

$$u_{k|k}^{st*} = \min J_{st} \quad (19)$$

$$s.t. \quad \begin{cases} x_{k+i+1|k}^{st} = Ax_{k+i|k}^{st} + Bu_{k+i|k}^{st} + d^{st}(k), & i = 0 \dots H_p, & (19.a) \\ -u_{\max}^{st} \leq u_{k+i|k}^{st} \leq u_{\max}^{st}, & i = 0 \dots H_c - 1, & (19.b) \\ u_{k+i|k}^{st} = 0, & i = H_c \dots H_p, & (19.c) \\ -\delta_f^{\max}(k) \leq \delta_f(k-1) + \Delta \delta_f^d(k) \\ + \sum_{j=0}^{j+} u_{k+j|k}^{st} \leq \delta_f^{\max}(k), & j+ = 0 \dots H_c - 1, & (19.d) \\ Y_{\min} - \zeta_Y \leq Y_{k+i|k}, & i = 1 \dots 1 + H_p. & (19.e) \end{cases}$$

TABLE II
ULMPC COMPUTATIONAL LOAD ON MATLAB

Control Mode	Computational Load	Overall Load	Ratio
Steering-enhanced	STE50 0.0362	0.0639	56.75%
	BRK40 0.0276		43.25%
Standard	STE45 0.0331	0.0638	51.86%
	BRK45 0.0307		48.14%
Braking-enhanced	STE40 0.0291	0.0639	45.50%
	BRK50 0.0348		54.50%

TABLE III
ULMPC ALGORITHM MEX-FILE SIZE ON MATLAB

Control Mode	MEX File Size	Overall Size	Ratio
Steering-enhanced	STE 50 1251 KB	1785KB	70.08%
	BRK40 534 KB		29.92%
Standard	STE45 1137 KB	1724KB	65.95%
	BRK45 587 KB		34.05%
Braking-enhanced	STE40 1023 KB	1663KB	61.52%
	BRK50 640 KB		38.48%

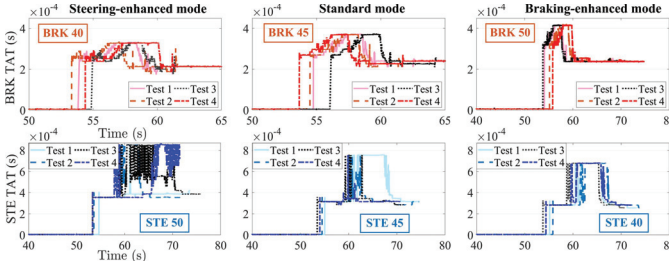


Fig. 7. Representative task turnaround times on Scalexio.

TABLE IV
ULMPC COMPUTATIONAL LOAD ON SCALEXIO

Control Mode	Computational Load	Overall Load	Ratio
Steering-enhanced	STE50 0.0542	0.0961	56.40%
	BRK40 0.0419		43.60%
Standard	STE45 0.0480	0.0961	50.05%
	BRK45 0.0481		49.95%
Braking-enhanced	STE40 0.0430	0.0960	55.21%
	BRK50 0.0530		44.79%

the relative ratio of the memory allocation changes per the different control modes.

Afterward, we show the resource consumptions of the three control modes in the driving simulator system. The two CAA systems were implemented in a dSPACE Scalexio real-time computing unit, and the task turnaround times (TATs) of the two CAA systems under the three control modes were recorded. Representative TATs are shown in Fig. 7.

Fig. 7 reflects that the braking-assistance system's maximum TAT increased as we shifted from the steering-enhanced mode to the braking-enhanced mode and vice versa. From Fig. 7, we can summarize the computational loads of the two CAA systems in the three control modes in Table IV, which yielded similar conclusions from Table II again.

Then, the occupied memory spaces of the three control modes on the Scalexio platform, quantified by the *rta* file

TABLE V
ULMPC ALGORITHM RTA-FILE SIZES ON SCALEXIO

Control Mode	MEX File Size	Overall Size	Ratio
Steering-enhanced	STE50 156 KB	251 KB	62.15%
	BRK40 95 KB		37.85%
Standard	STE45 148 KB	247 KB	59.92%
	BRK45 99 KB		40.08%
Braking-enhanced	STE40 139 KB	242 KB	57.44%
	BRK50 103 KB		42.56%

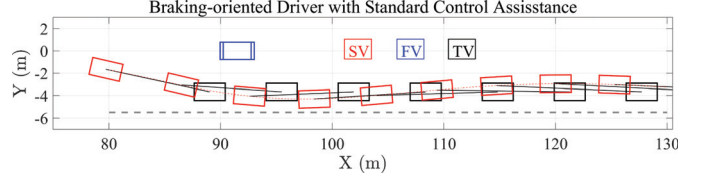


Fig. 8. Vehicle trajectory of Driver1 with the standard-mode CAA systems.

sizes, are compared in Table V. Similar conclusions from Table III can be drawn again from Table V. Tables II-V show the feasibility of resource allocation via parameter tuning.

Remark: Note that the purpose of this paper is to demonstrate the *feasibility* and *safety benefits* of allocating the implementation resources to CAA systems per driver capabilities. To this end, three groups of hard-coded prediction horizons were tuned as examples. However, *designing* driver-capabilities-based CAA systems must consider the driver behavior variations and adaptively regulate the parameters of CAA systems. For instance, based on the online monitored driver braking and steering outputs, a trained logistic regression model can periodically update the probability that a driver belongs to a specific class (e.g., the probability that the driver is steering-oriented is 60%). Note that the regression output is a number between zero and one, whereas the *classification* results in Section III are only binary. The online updated logistic regression output can then be exploited to proportionally allocate the available implementation resources to the CAA systems to compensate for a driver's (relatively) deficient maneuver. Finally, the available resources can be translated into the online calibrated MPC parameters, e.g., the prediction horizon, the control horizon, and the sampling period, via lookup tables [9]. Designing resource-aware capabilities-based CAA systems will be studied in the next step.

V. SIMULATION AND EXPERIMENTAL RESULTS

This section is devoted to verifying this paper's key idea: By allocating more implementation resources to compensate for a driver's relatively deficient maneuver, the CAA systems can provide enhanced safety by consuming practically the same total amount of implementation resources. Control results of each driver-type/control-mode pair are compared through both CarSim-Simulink joint simulations and driver-in-the-loop experiments.

A. CarSim-Simulink Joint Simulations

We first show that the simulated driver models in Fig. 2 can avoid the rear-end and the run-off-road collisions with the

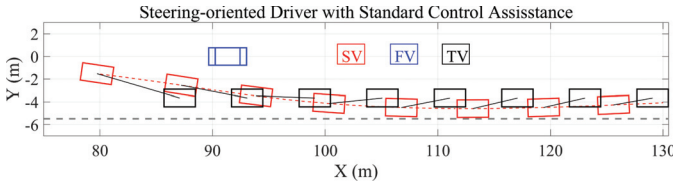


Fig. 9. Vehicle trajectory of Driver15 with the standard-mode CAA systems.

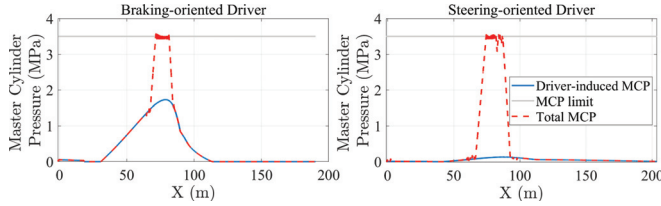


Fig. 10. Master-cylinder pressure assistance to Driver1 and Driver15.

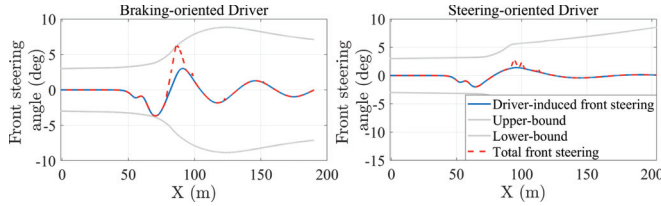


Fig. 11. Front steering assistance to Driver1 and Driver15.

standard mode CAA systems. The thresholds in (11), (19) are set as: $Y_{\min} = -5.49m$, $\kappa_{\min} = 0.5s$. With the standard mode CAA systems being enabled, SV trajectories of the simulated Driver1 model (braking-oriented) and the Driver15 model (steering-oriented) are depicted in Fig. 8 and Fig. 9, respectively. Therefore, the simulated Driver1 can now stay on the road, while the simulated Driver15 can avoid rear-ending the slowly moving TV.

For both driver models, the standard mode control assistances, i.e., P_{MC}^* in (12) and δ_f^{C*} in (20), are shown in Fig. 10 and Fig. 11, respectively.

In Fig. 10 and Fig. 11, the solid lines indicate the original commands from the simulated driver models, while the dashed lines show the overall commands with control interventions, which consistently satisfy the actuator constraints. Moreover, the control intervention demonstrates its adaptive nature. The steering-oriented driver model receives a marginal steering correction and a sufficiently high master-cylinder pressure support. Instead, a noticeable steering correction and moderate braking support are given to the braking-oriented driver model.

Further, we show that the CAA systems provide enhanced safety benefits if more resources are allocated to compensate for the weakness of driver maneuvers. We compare the SV's TTC and $Y(t)$ in Fig. 12.

As shown in Fig. 12, for the steering-oriented driver model, the three control modes yield only marginal discrepancies in $Y(t)$. However, the braking-enhanced mode noticeably outperforms the others in enlarging the minimum TTC. Meanwhile, for the braking-oriented driver model, no substantial

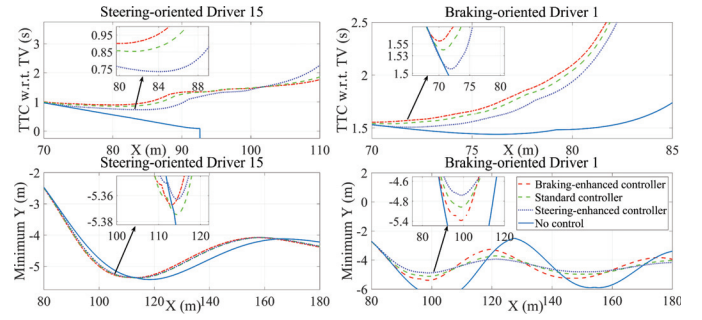


Fig. 12. Safety enhancement via resources allocation.

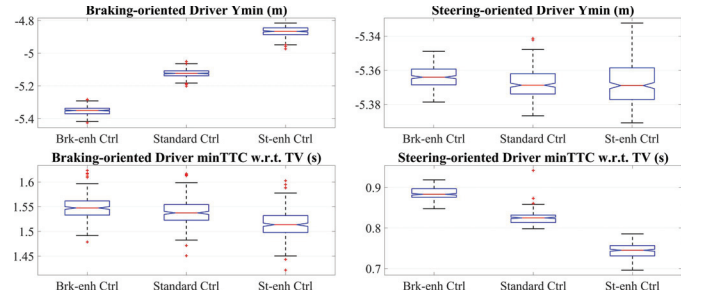


Fig. 13. Minimum Y and TTC with three control modes. On each box, the center mark represents the median, while the bottom and top edges of the box represent the 25th and 75th percentiles, respectively. The whiskers extend to the most extreme data points that are not considered outliers, and the outliers are drawn separately using the “+” sign.

differences in the minimum TTC can be observed from the three control modes, but the steering-enhanced mode produces the highest $Y(t)$.

Constant parameters in Table I cannot reflect the driver behavior variations in reality. To compensate, we ran 250 simulations for each driver-type/control-mode pair, with the driver model parameters being randomly generated from a Gaussian distribution. Parameter means were set equal to the nominal values in Table I, and the standard deviations were 10% of the mean values. Simulation results with smaller standard deviations can be found in [36]. Fig. 13 shows the boxplots of SV's minimum TTC and $Y(t)$ from the two simulated driver models.

For the braking-oriented driver models, $Y(t)$ blocks' notches under the three control modes do not overlap with each other. Therefore, we can conclude with 95% confidence that allocating more resources to the steering assistance CAA system would substantially improve the lateral control performance. On the contrary, the distributions of the minimum TTC are barely influenced by control mode shifting. This is expected because braking-oriented drivers can properly apply sufficient brake control, trivializing the intervention from the braking-assistance CAA system. Likewise, as the steering correction remains slight for the steering-oriented driver models, the distributions of $Y(t)$ remain almost identical regardless of the control mode. However, when more resources are allocated to compensate for the driver models' deficient braking, the CAA systems effectively uplift the minimum TTC.

TABLE VI
MULTI-SIMULATION COSTS FROM THREE CONTROL MODES

	Braking-oriented driver models			Steering-oriented driver models		
	Br-Enh	Standard	St-Enh	Br-Enh	Standard	St-Enh
J_{TTC}	0.8003	0.8094	0.8345	2.5413	3.0196	4.0360
$J_{Y_{min}}$	7.1608	2.6227	1.5594	7.9085	8.0619	8.3673
<i>Sum</i>	7.961	3.432	2.394	10.450	11.082	12.403

To quantify the safety enhancement via resources allocation, we introduce two hyperbolic cosecant barrier functions [37]:

$$J_{TTC}(\kappa) = \frac{2e^{\kappa - \kappa_{min}}}{e^{2(\kappa - \kappa_{min})} - 1}, \quad J_{Y_{min}}(Y) = \frac{2e^{Y - Y_{min}}}{e^{2(Y - Y_{min})} - 1}. \quad (22)$$

Substituting the medians in Fig. 13 to (22), we summarize the cost functions in Table VI. The abbreviations ‘Br/St-Enh’ indicate ‘braking/steering-enhanced’, respectively.

Therefore, switching from the Standard control mode to the St-Enh mode can effectively improve the lateral control performance (+39%) with a negligibly sacrificed (−3.1%) TTC control lost for the braking-oriented driver models. Instead, if more resources are wrongly allotted to the braking-assistance CAA system, a marginal TTC control enhancement (+1.1%) along with a significantly degraded road-keeping performance (−173%) will be yielded. Comparable conclusions can be drawn for the steering-oriented driver models. The sum of the two costs, defined as the safety metric, validates the safety benefits of allocating more implementation resources to compensate for a driver’s (relatively) deficient maneuver.

B. Driver-in-the-Loop Experiments

1) *Experimental Setup*: The driving simulator consists of a six-degree-of-freedom motion base, a visual/audio system, and a dSPACE Scalexio real-time computing unit. Driver inputs, i.e., steering wheel angle, brake pedal position, and throttle pedal position, are transferred to Scalexio as the inputs to ASM, which is an industrial-proven vehicle dynamics and traffic simulation software. Meanwhile, simulated vehicle states and surrounding traffic from ASM are sent back to the motion base and the visual/audio system to synchronize the motion base movement, the projected scenario animation, and the background sounds. The CAA systems in (11) and (19) were implemented via CVXGEN [38].

Two drivers participated in the driving tests. They held valid driver licenses and had self-reported annual mileages above 4000 miles. They stayed in good mental status during the tests. When the experiment kicked off, the ASM cruise control was automatically triggered to maintain SV’s longitudinal speed at 108km/h. The driver only focused on lane-keeping via steering control. Once the distance between SV and FV went below 25m, FV initiated a full brake, forcing the driver to simultaneously brake and change to the right lane.

Each driver operated the simulator sixty times, with the three control modes executed twenty times each, in a random order. During the experiments, drivers were not informed of

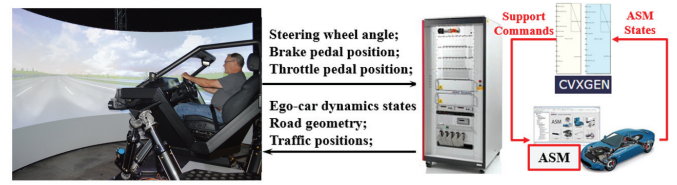


Fig. 14. Driving simulator platform.

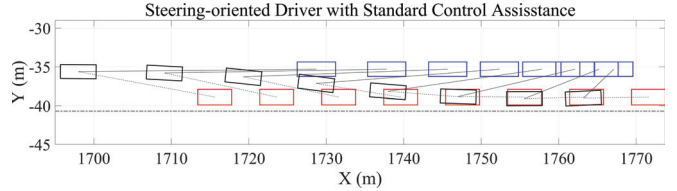


Fig. 15. Trajectory of the steering-oriented driver with standard-mode control.

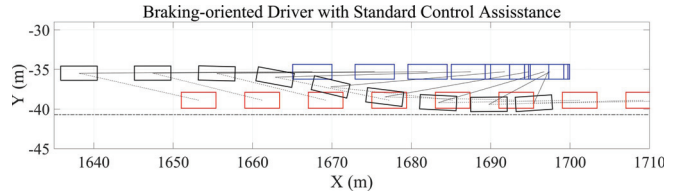


Fig. 16. Trajectory of the braking-oriented driver with standard-mode control.

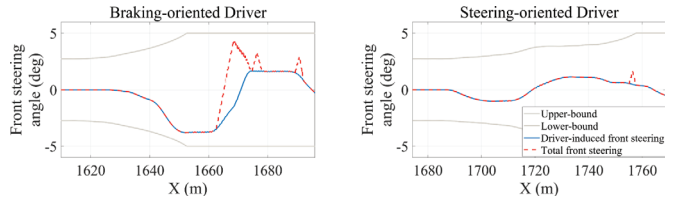


Fig. 17. Front steering from the CAA systems.

the existence of the CAA systems nor the control mode. For memory washing and workload reduction, experiments were executed on three separate days. Before the actual tests, several ‘familiarization’ sessions were given to each driver. The protocol of this study was approved by the Institutional Review Board (IRB).

2) *Experimental Results*: Echoing Fig. 4, representative SV trajectories from the two drivers with the standard-mode CAA systems enabled are depicted in Fig. 15 and Fig. 16. The black, blue, and red blocks represent SV, FV, and TV. Note that due to map transfer, the lateral position threshold in (19) was reset as: $Y_{min} = -40.9m$, and the minimum TTC in (11) was set as $\kappa_{min} = 1.0s$.

We can conclude from Fig. 15 and Fig. 16 that the standard mode CAA systems help both drivers to avoid the rear-end and the run-off-road collisions.

Then, the representative standard-mode control interventions for the two drivers are drawn in Fig. 17 and Fig. 18. Like Fig. 10 and Fig. 11, the CAA systems adapt the control intervention per the driver characteristics, with the actuator constraints always respected.

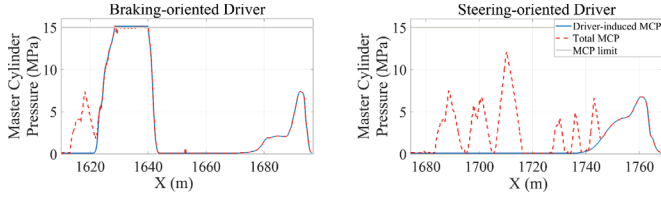


Fig. 18. Master-cylinder pressure from the CAA systems.

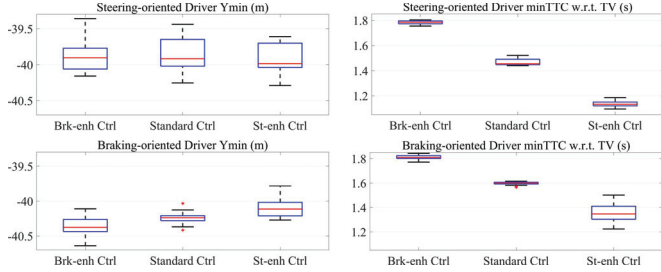


Fig. 19. Minimum Y and TTC of SV under three control modes.

TABLE VII
EXPERIMENTAL COSTS FROM THREE CONTROL MODES

	Steering-oriented Driver			Braking-oriented Driver		
	Br-Enh	Standard	St-Enh	Br-Enh	Standard	St-Enh
J_{TTC}	0.6211	0.9222	1.5834	0.6085	0.7660	1.2675
$J_{Y_{min}}$	1.7584	2.1700	2.3738	16.4102	2.9630	2.2616
Sum	2.3795	3.0922	3.9572	17.0187	3.7290	3.5291

Finally, the box plots of the minimum TTC and the minimum lateral position of SV under the different control modes are compared in Fig. 19.

Like Fig. 13, the different control modes only marginally influence the distribution of SV's minimum lateral position for the steering-oriented driver. Instead, the steering-enhanced control mode effectively improves the lateral control performance for the braking-oriented driver. Interestingly, for both drivers, the three control modes consistently yield distinct distributions of the minimum TTC. Fig. 18 reveals that the master-cylinder pressure control was triggered before human intervention, even for the braking-oriented driver. Therefore, the minimum TTC becomes somehow control-dominated instead of driver-dominated.

The safety metrics from the driving simulator experiments are summarized in Table VII. As the size of the dataset is relatively small, we utilize each boxplot's minimum data point to calculate the worst-case results. As demonstrated in Table VII, the Br (St)-Enh mode yields the highest safety metric to the steering (braking)-oriented driver. Instead, if more implementation resources were wrongly allocated to support the driver's inherent strength, it would negatively affect the driving safety of the overall driver-vehicle system.

VI. CONCLUSION

We propose that the limited implementation resources should be allocated to CAA systems per the driver's maneuver skills. By allocating more implementation resources to

compensate for a driver's (relatively) deficient maneuver, the CAA systems can provide enhanced protection by consuming practically the same overall amount of implementation resources. CarSim-Simulink joint simulations and driving simulator experiments demonstrate the feasibility and the effectiveness of the approach. Capability-based resource-aware CAA systems design and verification under more complex driving scenarios will be studied in the next step.

REFERENCES

- [1] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 4, pp. 6–22, Oct. 2014, doi: [10.1109/ITS.2014.2336271](https://doi.org/10.1109/ITS.2014.2336271).
- [2] K. D. Kusano and H. C. Gabler, "Safety benefits of forward collision warning, brake assist, and autonomous braking systems in rear-end collisions," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1546–1555, Dec. 2012, doi: [10.1109/ITITS.2012.2191542](https://doi.org/10.1109/ITITS.2012.2191542).
- [3] A. Vahidi and A. Eskandarian, "Research advances in intelligent collision avoidance and adaptive cruise control," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 3, pp. 143–153, Sep. 2003, doi: [10.1109/ITITS.2003.821292](https://doi.org/10.1109/ITITS.2003.821292).
- [4] Z. Wang, Y. Bai, J. Zha, J. Wang, and X. Wang, "Connected vehicle driving safety enhancement via dynamic communication channel selection," *Mechatronics*, vol. 74, Apr. 2021, Art. no. 102512, doi: [10.1016/j.mechatronics.2021.102512](https://doi.org/10.1016/j.mechatronics.2021.102512).
- [5] M. Schorn, U. Stahlin, A. Khanafer, and R. Isermann, "Nonlinear trajectory following control for automatic steering of a collision avoiding vehicle," in *Proc. Amer. Control Conf. (ACC)*, Minneapolis, MN, USA, 2006, pp. 1–6, doi: [10.1109/acc.2006.1657656](https://doi.org/10.1109/acc.2006.1657656).
- [6] S. Lefèvre, A. Carvalho, Y. Gao, H. E. Tseng, and F. Borrelli, "Driver models for personalised driving assistance," *Veh. Syst. Dyn.*, vol. 53, no. 12, pp. 1705–1720, 2015, doi: [10.1080/00423114.2015.1062899](https://doi.org/10.1080/00423114.2015.1062899).
- [7] A. Gray, M. Ali, Y. Gao, J. K. Hedrick, and F. Borrelli, "Integrated threat assessment and control design for roadway departure avoidance," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Anchorage, AK, USA, Sep. 2012, pp. 1714–1719, doi: [10.1109/ITSC.2012.6338781](https://doi.org/10.1109/ITSC.2012.6338781).
- [8] S. M. Erlien, S. Fujita, and J. C. Gerdes, "Shared steering control using safe envelopes for obstacle avoidance and vehicle stability," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 441–451, Feb. 2016, doi: [10.1109/ITITS.2015.2453404](https://doi.org/10.1109/ITITS.2015.2453404).
- [9] Z. Wang, Y. Bai, J. Wang, and X. Wang, "Vehicle path-tracking linear-time-varying model predictive control controller parameter selection considering central process unit computational load," *J. Dyn. Syst., Meas., Control*, vol. 141, no. 5, May 2019, Art. no. 051004, doi: [10.1115/1.4042196](https://doi.org/10.1115/1.4042196).
- [10] Z. Wang, J. Zha, and J. Wang, "Autonomous vehicle trajectory following: A flatness model predictive control approach with hardware-in-the-loop verification," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 9, pp. 5613–5623, Sep. 2021, doi: [10.1109/ITITS.2020.2987987](https://doi.org/10.1109/ITITS.2020.2987987).
- [11] P. Leteinturier, S. Brewerton, and K. Scheibert, "Multicore benefits and challenges for automotive applications," SAE Tech. Paper 2008-01-0989, 2008, doi: [10.4271/2008-01-0989](https://doi.org/10.4271/2008-01-0989).
- [12] F. Mars, M. Deroo, and J.-M. Hoc, "Analysis of human-machine cooperation when driving with different degrees of haptic shared control," *IEEE Trans. Haptics*, vol. 7, no. 3, pp. 324–333, Jul. 2014, doi: [10.1109/TOH.2013.2295095](https://doi.org/10.1109/TOH.2013.2295095).
- [13] M. Hasenjager, M. Heckmann, and H. Wersing, "A survey of personalization for advanced driver assistance systems," *IEEE Trans. Intell. Veh.*, vol. 5, no. 2, pp. 335–344, Jun. 2020, doi: [10.1109/TIV.2019.2955910](https://doi.org/10.1109/TIV.2019.2955910).
- [14] S. Schnelle, J. Wang, R. Jagacinski, and H.-J. Su, "A feedforward and feedback integrated lateral and longitudinal driver model for personalized advanced driver assistance systems," *Mechatronics*, vol. 50, pp. 177–188, Apr. 2018, doi: [10.1016/j.mechatronics.2018.02.007](https://doi.org/10.1016/j.mechatronics.2018.02.007).
- [15] S. Schnelle, J. Wang, H.-J. Su, and R. Jagacinski, "A personalizable driver steering model capable of predicting driver behaviors in vehicle collision avoidance maneuvers," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 5, pp. 625–635, Oct. 2017, doi: [10.1109/THMS.2016.2608930](https://doi.org/10.1109/THMS.2016.2608930).
- [16] A. Erseus, L. Drugge, and A. S. Trigell, "A path tracking driver model with representation of driving skill," *Int. J. Veh. Syst. Model. Test.*, vol. 6, no. 2, pp. 145–186, Sep. 2011, doi: [10.1504/IJVSMT.2011.042394](https://doi.org/10.1504/IJVSMT.2011.042394).

- [17] M. Saffarian, J. C. F. de Winter, and R. Happee, "Enhancing driver car-following performance with a distance and acceleration display," *IEEE Trans. Human-Mach. Syst.*, vol. 43, no. 1, pp. 8–16, Jan. 2013, doi: [10.1109/TSMCA.2012.2207105](https://doi.org/10.1109/TSMCA.2012.2207105).
- [18] C. Su, W. Deng, H. Sun, J. Wu, B. Sun, and S. Yang, "Forward collision avoidance systems considering driver's driving behavior recognized by Gaussian mixture model," in *Proc. IEEE Intell. Veh. Symp.*, Los Angeles, CA, USA, Jun. 2017, pp. 535–540, doi: [10.1109/IVS.2017.7995773](https://doi.org/10.1109/IVS.2017.7995773).
- [19] J. Wang, J. Wang, R. Wang, and C. Hu, "A framework of vehicle trajectory replanning in lane exchanging with considerations of driver characteristics," *IEEE Trans. Veh. Technol.*, vol. 66, no. 5, pp. 3583–3596, May 2017, doi: [10.1109/TVT.2016.2609154](https://doi.org/10.1109/TVT.2016.2609154).
- [20] L. Maldonado, W. Chang, D. Roy, A. Annaswamy, D. Goswami, and S. Chakraborty, "Exploiting system dynamics for resource-efficient automotive CPS design," in *Proc. Design Automat. Test Eur. Conf. Exhib.*, Florence, Italy, 2019, pp. 234–239, doi: [10.23919/DAT.2019.8715176](https://doi.org/10.23919/DAT.2019.8715176).
- [21] Y. Bai, Z. Wang, X. Wang, and J. Wang, "AutoE2E: End-to-end real-time middleware for autonomous driving control," in *Proc. Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Singapore, 2020, pp. 1101–1111, doi: [10.1109/ICDCS47774.2020.00092](https://doi.org/10.1109/ICDCS47774.2020.00092).
- [22] X. Dai, W. Chang, S. Zhao, and A. Burns, "A dual-mode strategy for performance-maximisation and resource-efficient CPS design," *ACM Trans. Embedded Comput. Syst.*, vol. 18, no. 5s, pp. 1–20, Oct. 2019, doi: [10.1145/3358213](https://doi.org/10.1145/3358213).
- [23] S. E. Lee, E. Llaneras, S. G. Klauer, and J. Sudweeks, "Analyses of rear-end crashes and near-crashes in the 100-car naturalistic driving study to support rear-signaling countermeasure development," Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. HS 810-846, Oct. 2007.
- [24] D. Yang, L. Zhu, B. Ran, Y. Pu, and P. Hui, "Modeling and analysis of the lane-changing execution in longitudinal direction," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2984–2992, Oct. 2016, doi: [10.1109/TITS.2016.2542109](https://doi.org/10.1109/TITS.2016.2542109).
- [25] H. Jula, E. B. Kosmatopoulos, and P. A. Ioannou, "Collision avoidance analysis for lane changing and merging," *IEEE Trans. Veh. Technol.*, vol. 49, no. 6, pp. 2295–2308, Nov. 2000, doi: [10.1109/25.901899](https://doi.org/10.1109/25.901899).
- [26] Y. Zhang, W. C. Lin, and Y.-K. S. Chin, "A pattern-recognition approach for driving skill characterization," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 4, pp. 905–916, Dec. 2010, doi: [10.1109/TITS.2010.2055239](https://doi.org/10.1109/TITS.2010.2055239).
- [27] W. Wang, J. Xi, A. Chong, and L. Li, "Driving style classification using a semisupervised support vector machine," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 5, pp. 650–660, Oct. 2017, doi: [10.1109/THMS.2017.2736948](https://doi.org/10.1109/THMS.2017.2736948).
- [28] G. S. Aoude, V. R. Desaraju, L. H. Stephens, and J. P. How, "Driver behavior classification at intersections and validation on large naturalistic data set," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 724–736, Jun. 2012, doi: [10.1109/TITS.2011.2179537](https://doi.org/10.1109/TITS.2011.2179537).
- [29] N. P. Chandrasiri, K. Nawa, and A. Ishii, "Driving skill classification in curve driving scenes using machine learning," *J. Mod. Transp.*, vol. 24, no. 3, pp. 196–206, Sep. 2016, doi: [10.1007/S40534-016-0098-2](https://doi.org/10.1007/S40534-016-0098-2).
- [30] Z. Wang and J. Wang, "Ultra-local model predictive control: A model-free approach and its application on automated vehicle trajectory tracking," *Control Eng. Pract.*, vol. 101, Aug. 2020, Art. no. 104482, doi: [10.1016/j.conengprac.2020.104482](https://doi.org/10.1016/j.conengprac.2020.104482).
- [31] M. Fliess and C. Join, "Model-free control," *Int. J. Control*, vol. 86, no. 12, pp. 2228–2252, 2013, doi: [10.1080/00207179.2013.810345](https://doi.org/10.1080/00207179.2013.810345).
- [32] M. Mboup, C. Join, and M. Fliess, "Numerical differentiation with annihilators in noisy environment," *Numer. Algorithms*, vol. 50, no. 4, pp. 439–467, 2009, doi: [10.1007/s11075-008-9236-1](https://doi.org/10.1007/s11075-008-9236-1).
- [33] L. Menhour, B. D'Andréa-Novet, M. Fliess, D. Gruyer, and H. Mounier, "An efficient model-free setting for longitudinal and lateral vehicle control: Validation through the interconnected pro-SiVIC/RTMaps prototyping platform," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 461–475, Feb. 2018, doi: [10.1109/TITS.2017.2699283](https://doi.org/10.1109/TITS.2017.2699283).
- [34] D. Tan, W. Chen, H. Wang, and Z. Gao, "Shared control for lane departure prevention based on the safe envelope of steering wheel angle," *Control Eng. Pract.*, vol. 64, pp. 15–26, Jul. 2017, doi: [10.1016/j.conengprac.2017.04.010](https://doi.org/10.1016/j.conengprac.2017.04.010).
- [35] C. Wanli and C. Samarjit, "Resource-aware automotive control systems design: A cyber-physical systems approach," *Found. Trends Electron. Des. Autom.*, vol. 10, no. 4, pp. 249–369, 2016, doi: [10.1561/10000000045](https://doi.org/10.1561/10000000045).
- [36] Z. Wang and J. Wang, "Personalized ground vehicle collision avoidance system: From a computational resource re-allocation perspective," in *Proc. IEEE Intell. Veh. Symp.*, Las Vegas, NV, USA, Oct./Nov. 2020, pp. 598–603, doi: [10.1109/IV47402.2020.9304754](https://doi.org/10.1109/IV47402.2020.9304754).
- [37] E. Clemente, M. Meza-Sánchez, E. Bugarin, and A. Y. Aguilar-Bustos, "Adaptive behaviors in autonomous navigation with collision avoidance and bounded velocity of an omnidirectional mobile robot: A control theory with genetic programming approach," *J. Intell. Robot. Syst.*, vol. 92, no. 2, pp. 359–380, Oct. 2018, doi: [10.1007/s10846-017-0751-y](https://doi.org/10.1007/s10846-017-0751-y).
- [38] J. Mattingley and S. Boyd, "CVXGEN: A code generator for embedded convex optimization," *Optim. Eng.*, vol. 13, no. 1, pp. 1–27, 2012, doi: [10.1007/s11081-011-9176-9](https://doi.org/10.1007/s11081-011-9176-9).



Zejiang Wang (Graduate Student Member, IEEE) received the B.E. degree (Hons.) in mechanical engineering and automation from Southeast University, Nanjing, China, in 2014, the Dipl.-Ing. degree from ENSTA Paris, Palaiseau, France, and the double M.S. degrees in design, modeling, and architecture of complex industrial systems from École Polytechnique, University of Paris-Saclay, Palaiseau, both in 2017. He is currently pursuing the Ph.D. degree with Walker Department of Mechanical Engineering, The University of Texas at Austin, Austin, TX, USA.

His research interests include system dynamics and control, intelligent transportation systems, and cyber-physical systems.



Adrian Cosio received the B.S. degrees in mechanical engineering and physics from The University of Texas at El Paso in 2018 and the M.S. degree in mechanical engineering from Walker Department of Mechanical Engineering, The University of Texas at Austin. He is currently a Research and Development Aeronautical Engineer with Sandia National Laboratories, Albuquerque, NM, USA, focusing on navigation, guidance, and control. He also has a research background working with material science and engineering, mainly dealing with ferromagnetic materials. His research interests include the simulation and modeling of dynamic systems and control, specifically in vehicular and aeronautical systems.



Junmin Wang (Senior Member, IEEE) received the B.E. degree in automotive engineering and the first M.S. degree in power machinery and engineering from Tsinghua University, Beijing, China, in 1997 and 2000, respectively, the second and third M.S. degrees in electrical engineering and mechanical engineering from the University of Minnesota, Twin Cities, Minneapolis, MN, USA, in 2003, and the Ph.D. degree in mechanical engineering from The University of Texas at Austin, Austin, TX, USA, in 2007. He is the Lee Norris & Linda Steen Norris Endowed Professor with Walker Department of Mechanical Engineering, The University of Texas at Austin. He is the author or coauthor of more than 350 peer-reviewed publications, including 180 journal articles and 13 U.S. patents. He is an IEEE Vehicular Technology Society Distinguished Lecturer, SAE Fellow, and ASME Fellow.