Journal of Systems Architecture 125 (2022) 102432

Contents lists available at ScienceDirect

Journal of Systems Architecture

journal homepage: www.elsevier.com/locate/sysarc

A hybrid controller for safe and efficient longitudinal collision avoidance control

Qiang Wang [a], Xinlei Zheng [b], Jiyong Zhang [b,*], Joseph Sifakis [c]

[a] National Key Laboratory of Science and Technology on Information System Security, Institute of System Engineering, Chinese Academy of Military Science, Beijing, China
[b] School of Automation, Hangzhou Dianzi University, Hangzhou, China
[c] Verimag, Université Grenoble Alpes, France

ARTICLE INFO

Keywords:
Collision avoidance
Model-based design
Model predictive control
Safety Architecture
Autonomous vehicles

ABSTRACT

We design and experimentally evaluate a hybrid safe-by-construction longitudinal collision avoidance controller for autonomous vehicles. The controller combines into a single architecture the respective advantages of a model predictive controller and a discrete safe controller. The model predictive controller is used to achieve optimal efficiency in nominal conditions. The safe controller avoids collision by applying two different policies, for nominal and out-of-nominal conditions, respectively. We present design principles for both controllers and show how each one can contribute in the hybrid architecture to improve performance, road occupancy and passenger comfort while preserving safety. The experimental results confirm the feasibility of the approach and the practical relevance of hybrid controllers for safe and efficient driving.

1. Introduction

It is widely believed that the deployment of autonomous vehicles can improve not only the traffic efficiency, but also its safety. Longitudinal collision avoidance, as a fundamental safety requirement for autonomous vehicle control, plays a crucial role in guaranteeing traffic safety and reducing the number of vehicle crashes, given the fact that more than 50 percent of the total amount of vehicle crashes are rear-end collisions.[1]

A variety of approaches and frameworks has been investigated for longitudinal collision avoidance control. The underlying assumptions vary largely with the level of modeling of the vehicle dynamics and the nature of the controller stimuli. Control-based techniques typically focus on collision avoidance for adaptive cruise control [1,2] taking into account the impact of perception uncertainty and accuracy of vehicle models [3,4]. They allow achieving optimality for specific tasks or scenarios without providing strict safety guarantees. Model Predictive Control (MPC) [2,5], as a prominent optimal control approach has been widely used for vehicle control because it allows handling multiple constraints in a receding horizon. Nonetheless, MPC relies on the use of optimization algorithms for predicting vehicle states and depending on the optimization algorithms and the dynamic model of the vehicles, it may produce no solutions, or result in high computational complexity because of heavy iterative calculations. Furthermore, the MPC controller is designed based on certain assumptions about the vehicle dynamics, and by its nature can hardly guarantee safety, namely collision avoidance.

A different line of works focus on safety using formal methods. These apply a variety of techniques including reachability analysis [6,7], Responsibility Sensitive Safety model [8], logic-based controller synthesis [9,10], as well as the design of safety supervision mechanisms for specific scenarios [11,12]. The basic principle of safe longitudinal collision avoidance control, as formalized and implemented in our previous work [13], is to keep a safe distance with the preceding vehicles such that in any case the ego vehicle has enough space to brake and avoid collisions. Although these results can guarantee correctness by construction, they lead to solutions that privilege strict safety at the expense of efficiency. Designing a collision avoidance controller for autonomous vehicles that meets both efficiency and safety requirements remains a non-trivial problem. The two types of requirements are antagonistic as efficiency implies conflicting properties such as performance (i.e., maximization of the average speed), road occupancy (i.e., keeping the inter-vehicle distance as small as possible) and comfort (i.e., no sudden speed changes).

In search of solutions seeking compromises between efficiency and safety, a few works adopt a hybrid approach combining continuous and discrete control dynamics. The continuous controller is supervised by an automaton that takes over to handle critical situations.

* Corresponding author.
E-mail address: jzhang@hdu.edu.cn (J. Zhang).

[1] http://www.ntsb.gov/safety/safetystudies/Documents/SIR1501.pdf.

https://doi.org/10.1016/j.sysarc.2022.102432
Received 25 September 2021; Received in revised form 28 January 2022; Accepted 5 February 2022
Available online 24 February 2022

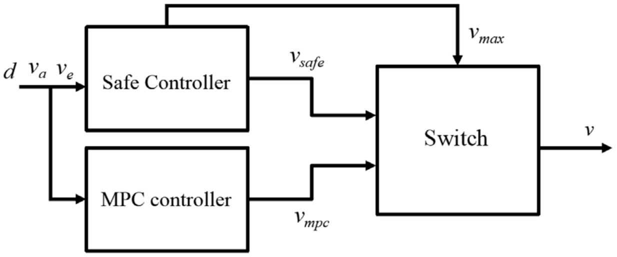


Fig. 1. The architecture of the Hybrid Controller.

Hybrid approaches rely on a principle of "division of roles" often applied in systems engineering that distinguishes between nominal operating conditions and out-of-nominal ones [14]. The continuous controller has parameters tuned to achieve given goals for nominal operation while the discrete controller deals with out-of-nominal situations. For example, [15,16] propose a switch-control approach, where a MPC Controller is safeguarded by an emergency maneuver activated to avoid collision when the ego vehicle is in a critical situation. Nonetheless, the emergency maneuver only takes care of safety without investigating possible trade-offs between safety and efficiency of the switch-control policy. Our work also differs from the hybrid control paradigm, e.g. in [17], which does not address safety, but partitions the vehicle states into a set of linear modes in order to approximate the nonlinear vehicle dynamics.

In this work we design and experimentally evaluate a safe-by-construction Hybrid Controller that proves to be efficient for the three mentioned criteria. The architecture of the Hybrid Controller is shown in Fig. 1. It results from the integration into a single architecture of a nominal MPC Controller and the discrete Safe Controller presented in [13]. The two controllers running in parallel receive the speed v_e and v_a of the ego vehicle and the vehicle ahead respectively, as well as their distance d, and compute the target speeds v_{mpc} and v_{safe} respectively. The control policies for computing v_{mpc} and v_{safe} adopt nominal conditions. In particular, v_{safe} is a safe speed under the assumption that the speed of the vehicle ahead is a continuous function and the deceleration does not exceed some limit corresponding to normal driving conditions. In addition to v_{safe}, the Safe Controller provides a speed v_{max} that is the maximal safe speed for out-of-nominal conditions when the vehicle ahead suddenly stops, e.g. in case of accident. This speed is computed as a function of the relative distance between the ego vehicle and the car ahead with the maximum deceleration rate of the ego vehicle. The Hybrid Controller uses a Switch selecting between the three speeds v_{mpc}, v_{safe} and v_{max} to optimize efficiency criteria while preventing the speed of the ego vehicle to exceed v_{max}. We show that the combined use of v_{mpc}, v_{safe} and v_{max} ensures both efficiency and safety in nominal conditions and moreover safety is preserved in out-of-nominal situations.

Our solution is inspired by the Simplex architecture principle [18], for runtime assurance of safety-critical systems. The architecture uses a Decision Module that switches control from a high-performance but unverified (hence potentially unsafe) Advanced Controller to a verified-safe Baseline Controller if some safety violation is imminent. The idea of Simplex architecture has been extensively applied in the design of safe autonomous systems. In [19,20] the authors have shown the effectiveness of the Simplex architecture for bounding the behavior of an autonomous aircraft taxiing system in order to maintain the safety requirements. In [21], Simplex architecture has been applied to build intelligent and safe unmanned aerial vehicles. In [22], a language support for Simplex architecture design has been proposed for programming safe robotic systems. In [23], a similar architecture called safety cage has been proposed for building safe automotive software.

Nonetheless, in our solution the Safe Controller contributes not only to out-of-nominal situations but also to some nominal situations where it proves to be more efficient than the MPC Controller. Hence, not only the Hybrid Controller is safe but also efficiency gains from the synergistic collaboration are substantial. Of course, the alternation of roles between the MPC and the Safe Controller should be implemented so as to avoid sudden changes of the kinematic state of the vehicle. In particular, care should be taken to avoid jerk (i.e., abrupt changes of acceleration) that might cause passenger discomfort. Additionally, the paper proposes a pragmatic methodology for the comparative evaluation of the three controllers for two types of scenarios: (1) nominal scenarios where the speed of the vehicle ahead is a known continuous function; and (2) out-of-nominal scenarios where the vehicle ahead abruptly brakes.

For nominal scenarios, the three controllers are evaluated against three efficiency criteria.

- The first criterion is performance that measures how much close the speed of the ego vehicle can get to the speed of the vehicle ahead. For a period of time it can be defined as the ratio of the average speed of the ego vehicle with respect to the average speed of the vehicle ahead. This ratio is less than one if the distance between the two vehicles is initially zero.
- The second criterion is road occupancy that measures how much close can get the ego vehicle to the vehicle ahead in collision-free scenarios.
- The third criterion measures passenger comfort that decreases as the standard deviation of the acceleration increases.

The paper is organized as follows. Section 2 presents the design principles for the MPC Controller and the Safe Controller, as well as a comparative study for the two types of scenarios and the three efficiency criteria using the Carla simulator. Section 3 presents the design and the implementation of the Hybrid Controller and its experimental evaluation. Section 4 emphasizes the feasibility and the practical relevance of hybrid controllers for safe and efficient driving and then outlines directions for future work.

2. A comparative study of the two control approaches

2.1. The MPC controller

The MPC paradigm combines three key components. The first is a dynamic model of the ego vehicle, allowing the MPC Controller to predict the vehicle states in a given horizon for changing inputs. The vehicle state is denoted by the vector $x = [p, v, a]^T$, where p is the vehicle position, v is the speed and a is the acceleration. Similarly, the state of vehicle ahead is $x_a = [p_a, v_a, a_a]^T$. The relative distance between the two vehicles is then $d = p_a - p$. We further require that the longitudinal acceleration is buffered as follows:

$$\dot{a} = \frac{u - a}{\tau} \tag{1}$$

where the control stimulus u is the desired acceleration, and τ is the time constant of the actuator lag that captures the inertial characteristics of the vehicle actuator. The vehicle dynamics model is described by the following equation.

$$\dot{x} = A_\tau \cdot x + B_\tau \cdot u \tag{2}$$

where

$$A_\tau = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau} \end{bmatrix}, B_\tau = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{bmatrix} \tag{3}$$

In order to enhance the stability of the system under the constant time sampling control method, we discretize the vehicle dynamics model [24]. If Δt is the discretization pace, the discrete longitudinal dynamics model of the ego vehicle at time instant t_k is given as follows:

$$x(t_{k+1}) = A_d \cdot x(t_k) + B_d \cdot u(t_k) \tag{4}$$

where

$$A_d = e^{A_\tau \Delta t} = \begin{bmatrix} 1 & \Delta t & \tau^2\left(e^{-\frac{\Delta t}{\tau}} - 1\right) + \Delta t\tau \\ 0 & 1 & \tau\left(1 - e^{-\frac{\Delta t}{\tau}}\right) \\ 0 & 0 & e^{-\frac{\Delta t}{\tau}} \end{bmatrix}$$
$$B_d = \int_0^{\Delta t} e^{A_\tau t} dt \cdot B_\tau = \begin{bmatrix} \tau^2\left(1 - e^{-\frac{\Delta t}{\tau}}\right) + \frac{\Delta t^2}{2} - \Delta t\tau \\ \tau\left(e^{-\frac{\Delta t}{\tau}} - 1\right) + \Delta t \\ 1 - e^{-\frac{\Delta t}{\tau}} \end{bmatrix} \tag{5}$$

In addition to the state of the ego vehicle, the MPC Controller also estimates the position of the vehicle ahead in order to compute predictions. We assume that in each MPC prediction horizon, the vehicle ahead decelerates with the constant deceleration a_a. Thus, the position of the vehicle ahead before stopping can be estimated as follows.

$$p_a(t_k) = p_a(t_0) + v_a(t_0) \cdot (t_k - t_0) + 1/2 \cdot a_a \cdot (t_k - t_0)^2 \tag{6}$$

The second key component is a cost function, which describes the expected behavior of the ego vehicle, in order to minimize the relative distance. Optimization consists in finding the best possible inputs that minimize the cost function. The cost function for time horizon h is modeled as a standard quadratic function and the optimization problem is formulated as follows:

$$argmin(u^*(\cdot), \sum_{k=1}^{h}(x_{opt}^T Q x_{opt} + ru^2)) \tag{7}$$

where Q is the weighting matrix for the state vector and r is the weight for the control stimulus. The new state variable x_{opt} is used to get the relative distance as close as possible to the constant d_c and the speed of the ego vehicle to v_a. It is defined by

$$x_{opt} = x_a - x - [d_c, 0, 0]^T \tag{8}$$

The weighting matrix is a diagonal matrix Q = diag $[q_p, q_v, q_a]^T$, where q_p, q_v and q_a are weighting parameters for vehicle position, speed and acceleration respectively. By adjusting the values of these weighting parameters, we can configure preference of the MPC control tendency. For instance, by enlarging the value of q_p we force MPC to drive the ego vehicle closer to the vehicle ahead, so that to reduce the relative distance. In this work, we make use of a constant distance d_c to demonstrate the feasibility and safety of our Hybrid Controller. However, d_c can also be a velocity dependent distance, e.g., by taking into account the velocity of the ego vehicle or the vehicle ahead.

Additionally while performing the optimization, the MPC Controller enforces the following constraints on the minimum or maximum values of speed and acceleration of the vehicle:

$$\begin{cases} v_{min} \le v \le v_{max} \\ u_{min} \le u \le u_{max} \end{cases} \tag{9}$$

where u_{min}, u_{max} (and v_{min}, v_{max}) are user-specified parameters for control stimulus and speed, respectively.

The third component of the MPC Controller is the optimization algorithm for solving this linear quadratic programming problem. For this purpose, we use the open source Python library, cvxopt [25].

Fig. 2 shows the architecture of the MPC Controller. The controller executes an iterative process optimizing the predictions of vehicle states while manipulating inputs for a given horizon. The predictions are based on the specified kinematic model of the vehicle. For each control cycle at time t_k, the controller takes as input the current states of the ego vehicle and of the vehicle ahead, and computes the future states of the ego vehicle to predict the optimal control stimuli u^* minimizing the cost function in the interval $[t_k, t_{k+h}]$, where h is the prediction horizon. The MPC Controller chooses the first element in the sequence as the control stimulus for the ego vehicle, and repeats the cycle at time t_{k+1}. A key advantage of MPC policy is flexibility in achieving complex objectives and implementing multiple constraints when performing optimizations.

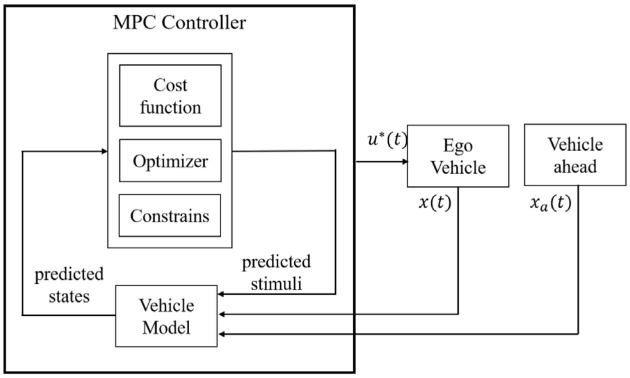


Fig. 2. Architecture of the MPC controller.

2.2. The safe controller

We leverage on a recent work [13] for the Safe Controller, where we have proposed a correct-by-design safe and efficient controller for autonomous vehicles. The controller minimizes the distance between the ego vehicle and the vehicle ahead while preserving safety for both nominal and out-of-nominal conditions. For nominal conditions, the Safe Controller is based on the relative speed $v_e - v_a$ between the ego vehicle and the vehicle ahead. It computes the target speed v_{safe} for moderate nominal acceleration and deceleration rates to enhance passenger comfort. For out-of-nominal conditions, the Safe Controller computes a target speed v_{max} taking into account only the speed of the ego vehicle and the needed braking distance. The braking distance is computed for a maximal deceleration rate that is much larger than the nominal deceleration to cope with dangerous situations, e.g., sudden stops of the vehicle ahead caused by accidents. The controller always keeps the speed $v_{safe} \le v_{max}$ to make sure that in all circumstances safety is preserved.

We briefly review the general design principle that we specialize for nominal and out-of-nominal conditions. The safe control policy relies on the following three functions.

- The function $d(t)$ gives the relative distance at time t between the ego vehicle and the vehicle ahead, which is either stopped or moving in the same direction.
- The braking function $B(v, v')$ gives the distance traveled by the ego vehicle, when braking from initial speed v to target speed v'. When the target speed is $v' = 0$ (i.e, the ego vehicle brakes to a stop), this function is abbreviated as $B(v)$ for simplicity.
- The accelerating function $A(v, v')$ gives the distance traveled by the ego vehicle, when accelerating from initial speed v to target speed v'.

We make no specific assumptions about the implementation of the accelerating and braking functions, e.g., whether acceleration is constant or variable. We simply require that the following properties hold:

- $B(v, v') = 0$ and $A(v, v') = 0$ if and only if $v = v'$.
- Additivity property: $B(v, v_1) + B(v_1, v_2) = B(v, v_2)$ and $A(v, v_1) + A(v_1, v_2) = A(v, v_2)$.
- Strict monotonicity: $B(v, v_1) < B(v, v_2)$ and $A(v, v_1) < A(v, v_2)$ if $v_1 < v_2$.

The basic idea for avoiding collision is to moderate the speed of the vehicle and anticipate the changes of the relative distance so as to have enough space and time to adjust and brake. For any time t, the vehicle only needs to keep track of the distance $d(t)$ and check in real-time whether $d(t)$ is greater than the minimal safe braking distance

$B(v_t)$ for the current speed v_t. It starts braking as soon as $d(t)$ reaches the minimal safe braking distance. In this way, it is guaranteed that if the obstacles ahead do not move in the opposite direction, no collision would happen. The Theorem below formalizes this idea.

Theorem 1. *If at time t the speed v_t of the vehicle is safe with respect to $d(t)$, i.e., $B(v_t) \leq d(t)$ and for any time $t + \vartriangle t$ it is possible to set the speed to a value $v_{t+\Delta t}$ such that the condition $d(t) - d(t + \vartriangle t) \leq B(v_t) - B(v_{t+\Delta t})$ holds, then the vehicle is always safe.*

Proof. The condition $d(t) - d(t + \vartriangle t) \leq B(v_t) - B(v_{t+\Delta t})$ relates changes of $d(t)$ to the changes of speed v. It simply says that the free space ahead does not change faster than the distance that the vehicle travels in some interval $\vartriangle t$. It can be deduced from the safety assumption $0 \leq d(t) - B(v_t)$ and from the condition that $0 \leq d(t) - B(v_t) \leq d(t + \vartriangle t) - B(v_{t+\Delta t})$.

Notice as an application of the above theorem, that if the vehicle brakes from speed V_t and the obstacles ahead do not move in the opposite direction, then the condition $d(t) - d(t + \vartriangle t) \leq B(v_t) - B(v_{t+\Delta t})$ trivially holds. In fact, when the vehicle brakes from v_t for time $\vartriangle t$, it will reach the speed $v_{t+\Delta t} < v_t$ and it will have traveled the distance $B(v_t, v_{t+\Delta t}) = B(v_t) - B(v_{t+\Delta t})$, by application of the additivity property. Then we have that $d(t) - (B(v_t) - B(v_{t+\Delta t}))$ is the distance ahead at time $t + \vartriangle t$ for the controlled vehicle. By the assumption that the obstacles are moving forward or stopped, we have that $d(t) - (B(v_t) - B(v_{t+\Delta t})) \leq d(t + \vartriangle t)$. Thus, Theorem 1 can trivially be applied if obstacles ahead do not move in the opposite direction.

This theorem suggests a simple and safe control policy that ensures collision freedom. For any time t, the vehicle only needs to keep track of the free distance ahead $d(t)$ and check in real-time whether $d(t)$ is greater than the minimal safe braking distance $B(v_t)$ for the current speed v_t. It starts braking as soon as $d(t)$ reaches the minimal safe braking distance. In this way, it is guaranteed that if the obstacles ahead do not move in the opposite direction, no collision would happen.

The above result provides a basis for ensuring collision freedom. Nonetheless, it leaves open the question of how the vehicle can efficiently use the available distance ahead by minimizing the traveling time. What would be an efficient driving policy when the free headway distance is greater than the minimal safe braking distance? We consider that a policy defines the speed function $v(t)$ in response to a free distance $d(t)$. An Accelerating/Braking policy (A/B policy) is a policy of accelerating first to some speed and then braking. Similarly, an Braking/Accelerating policy (B/A policy) is the policy of braking first to some speed and then accelerating. A Constant speed/Braking policy (C/B policy) is the policy of moving at constant speed and then braking. A policy is safe if the relative distance between the controlled vehicle and the obstacle ahead is positive. It is efficient if exceeding the speed enforced by the policy at any point would compromise safety.

The problem is to minimize the traveling time for a given distance, which implies to maximize the average speed. Consider the scenario where the speed of the vehicle is v and there is a stationary obstacle ahead at distance d, which is greater than the braking distance $B(v)$. The application of an A/B policy consists in computing an appropriate target speed v' such that $v < v' \leq v_L$, accelerating the vehicle to v' and then braking to full stop, where v_L is the speed limit. To ensure collision freedom, the total traveled distance must be such that $A(v, v') + B(v') \leq d$. The maximal target speed is given by the condition $v_M = \max\{v' \mid d \geq A(v, v') + B(v')\}$. Such a speed exists as both acceleration and braking functions are monotonically increasing with respect to the target speed v'. Notice that either $v_M \leq v_L$ and $d = A(v, v_M) + B(v_M)$ or $v_M = v_L$ and $d > A(v, v_M) + B(v_M)$.

As an example, for motion at constant acceleration and deceleration (a and b, respectively), we have $A(v, v') = v' * (v' - v)/a + (v' - v)^2/2 * a$ and $B(v') = (v')^2/2 * b$. Then the safety condition becomes $d \geq v' * (v' - v)/a + (v' - v)^2/2 * a + (v')^2/2 * b$, from which we can deduce $v' \leq \sqrt{(2 * a * b * d + b * v^2)/(a + b)}$. Thus the maximal target speed $v_M = \sqrt{(2 * a * b * d + b * v^2)/(a + b)}$. As we require that $v' \geq v$, we have $d \geq v^2/2 * b = B(v)$ and thus the maximal target speed always exists. Let v_d denote the speed reached by accelerating along distance $d(t)$, i.e., $v_d^2 - v^2 = 2 * d(t) * a$, then the formula can be simplified as $v_M = v_d * \sqrt{b/(a + b)}$.

The following theorem shows that for the given free distance d, the A/B policy is the most efficient and that from the given initial speed there is a maximal speed that minimizes the travel time of d.

Theorem 2. *If the speed v of the vehicle is safe with respect to the free distance d, i.e., $B(v) \leq d$, then the A/B policy is always safe and efficient for d.*

Proof. The safety proof is given by the arguments following Theorem 1. To prove efficiency, we consider three basic driving policies: the A/B policy, the B/A policy and the C/B policy. The other possible policies, such as accelerating, driving at constant speed, accelerating and then braking, can be obtained as combinations of the three basic ones. We show that the A/B policy yields the minimal traveling time.

We decompose the free distance ahead d into two segments: one segment of length $D = d - B(v)$ and one segment of length $B(v)$. Due to the additivity property, the distance $B(v)$ is always required regardless of the applied polices in order to brake safely from speed v. So the policies may differ only in the time needed to travel distance D. In the A/B policy, the vehicle travels distance D by first accelerating to the maximal target speed v_M and then braking from v_M to v. We denote by t_A the time needed to accelerate from v to v_M and by t_B the time needed to brake from v_M to v. In the C/B policy, the vehicle first moves at constant speed v for the distance D and then brakes from v for the remaining distance $B(v)$. We denote by t_D the time needed to travel D with constant speed v. We show that t_D is greater than $t_A + t_B$. We denote the speed function during acceleration by $v'(t)$ and the speed function during deceleration by $v''(t)$. Then we have $v'(t) > v$ for $t_A > t > 0$ and $v''(t) > v$ for $t_B > t > 0$. Since $D = v * t_D = \int_0^{t_A} v'(t)dt + \int_0^{t_B} v''(t)dt > \int_0^{t_A} vdt + \int_0^{t_B} vdt = v * (t_A + t_B)$, we have $t_D > t_A + t_B$. Thus the A/B policy takes less time and it is more efficient than the C/B policy.

In the B/A policy, the vehicle first brakes and accelerates for the distance D and then brakes for the remaining distance $B(v)$. We denote by $t'_B + t'_A$ the traveling time of D by braking and accelerating. By applying a similar reasoning, we can show that $t'_B + t_A > t_D$. Thus the C/B policy is more efficient than the B/A policy. This concludes the proof.

Based on the above results, we study a control principle for collision avoidance. We consider that the vehicle speed can change between a finite set of increasing levels $v_0, v_1, \dots, v_n$, where n is a constant, $v_0 = 0$ and v_n is the limit speed of the vehicle. We apply A/B policy to trigger the acceleration and braking from one level to another according to the free distance ahead and the bounds computed as follows, for each speed level $v_i, i \in [1, n]$,

- $B_i = B(v_i)$ is the minimal safe braking distance needed for the vehicle to fully stop from speed v_i;
- $D_i = A(v_{i-1}, v_i) + B(v_i)$ is the minimal safe distance needed for the vehicle to accelerate from speed v_{i-1} to v_i and then brake from v_i to stop.

The highest safe speed level v can be specified as a function of the current speed v_t of the vehicle and the distance d, provided that their initial values v_0 and d_0 satisfy the condition $B(v_0) \leq d_0$.

$$v = \begin{cases} v_{i+1} & \text{when} \quad v_t = v_i \wedge d = D_{i+1} \\ v_{i-1} & \text{when} \quad v_t = v_i \wedge d = B_i \\ v_i & \text{when} \quad v_t = v_i \wedge D_{i+1} > d > B_i \end{cases} \tag{10}$$

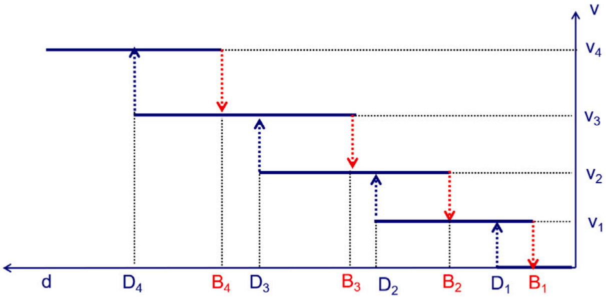


Fig. 3. Illustration of the collision avoidance principle for $n = 4$.

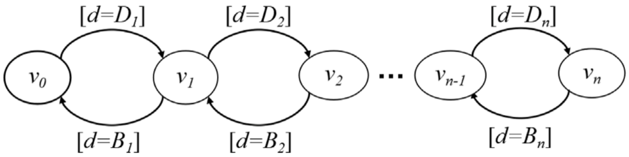


Fig. 4. Automaton modeling the collision avoidance principle.

Fig. 3 illustrates the principle for $n = 4$ speed levels. As the value of d increases, the speed of the vehicle climbs up levels. Safety is preserved by construction. The vehicle can accelerate to a higher level, if it can safely and efficiently use the available distance by combining acceleration and deceleration; in particular braking to a lower level if the distance reaches the bound for safe braking.

Fig. 4 provides a scheme for the computation of the safe speed level in the form of an extended automaton. The control states correspond to speed levels $v_0, \dots, v_n$. The transitions model instantaneous acceleration and braking steps triggered by conditions involving the distance d and the pre-computed bounds B_i and D_i. If the control location is v_i and the distance is equal to the minimal safe acceleration distance (i.e., $d = D_{i+1}$), then the automaton moves to location v_{i+1} after the speed is increased to v_{i+1}. If the distance reaches the minimal safe braking distance (i.e., $d = B_i$), then the automaton moves to location v_{i-1} after the speed is decreased to v_{i-1}. Given that $B_i = B_{i-1} + B(v_i, v_{i-1})$, after braking to v_{i-1} there is still enough space for safe braking. If none of the triggering conditions holds, then the distance d is such that $B_i < d < D_{i+1}$. The automaton stays at location v_i and the speed remains unchanged.

In our context, the speed v_{safe} is computed for nominal conditions considering that $v_t = v_e$ - v_a and that the braking and acceleration functions are defined for moderate rates. For out-of-nominal conditions, the maximal speed v_{max} of the ego vehicle is $v_{max} = \text{Max}\{\ v \mid B_{max}(v_e) \leq d\ \}$, where B_{max} is the deceleration function for some maximal deceleration rate. To make sure that collision is avoided in any case, the controller compares the speeds v_{safe} and v_{max}. When v_{max} is reached, a command is issued for emergency braking.

Compared with the MPC Controller, the Safe Controller can guarantee safety and it is more computational efficient since at any time it chooses the speed minimizing the relative distance depending on simple criteria. Thus, the Safe Controller can be easily implemented in real time without additional costs. On the contrary, the MPC Controller applies more involved computation trying to estimate the future states of both vehicles according to the kinematic model of the vehicle, which often requires computationally expensive optimization techniques.

2.3. Evaluation of the two control approaches

2.3.1. Experimental setting and evaluation criteria

We implement the MPC Controller and the Safe Controller in the Carla (version 0.9.8) simulator [26]. In the experimental evaluations, we consider constant acceleration/deceleration without considering the road friction conditions. For the Safe Controller, the acceleration/deceleration in the nominal setting is taken 3 m/s^2 and the deceleration rate in the out-of-nominal setting (e.g., a burst accident) is taken 12 m/s^2. The speed levels are from the set $\{0, 4, 8, 12, 16, 20, 24, 28, 32\}$ (m/s). The control policy for the computation of safe speed v_{safe} is based on the relative speed v_e - v_a. For the MPC Controller, the prediction horizon is set to 10 steps. The constant d_c in Eq. (8) is 20 m. The weighting matrix Q is set to $diag[50, 400, 1]^T$. Time lag τ is 0.3, and control stimulus weight r is set to 1.

To evaluate the efficiency of the two controllers, we carry out a set of comprehensive experiments performed on a Windows 10 PC with AMD R5 3500 and NVIDIA GTX 1660 SUPER. We consider both nominal and out-of-nominal scenarios.

- In a nominal scenario, the speed of the vehicle ahead is described by the function $v_a(t) = A\sin(\frac{2\pi}{T}t) + v_{a,0}$, where $v_{a,0}$ is taken equal to 12 m/s. In the experiments, we consider three different values of A, i.e., $\{6, 9, 12\}$ m/s, and three different values of T, i.e., $\{10, 20, 30\}$ s.
- In an out-of-nominal scenario, the vehicle ahead brakes suddenly and stops. We assume that the ego vehicle does not know when the sudden braking may occur.

In all scenarios the ego vehicle and the vehicle ahead move in the same lane in the same direction. The initial speed of the ego vehicle is set to 0 m/s. The initial relative distance between the two vehicles is set to 10 m.

We check whether the MPC Controller violates safety for nominal and out-of-nominal scenarios. For nominal scenarios, we evaluate the efficiency of the two controllers with respect to the following three criteria defined for simulation time t_{sim}.

- Performance is measured as the ratio of average speed of the ego vehicle with respect to that of the vehicle ahead, i.e.,

$$\mathcal{M}_p = \frac{\int_0^{t_{sim}} v_e(t)dt}{\int_0^{t_{sim}} v_a(t)dt} \tag{11}$$

  where v_e is the speed of the ego vehicle and v_a is the speed of the vehicle ahead.
- Road occupancy is defined as the ratio of the space occupied by the vehicles over the total available space. For our case with two vehicles, we consider $1/d$ as a measure of the occupancy, where d is their relative distance. For a simulation in time interval $[0, t_{sim}]$, it is given by the formula:

$$\mathcal{M}_o = \frac{1}{t_{sim}} \int_0^{t_{sim}} 1/d(t)dt \tag{12}$$

  The higher the value $\mathcal{M}_o$, the higher the occupancy. Note that the measure uses d, the distance between the two vehicles without taking into account any safety margin.
- Comfort means that the variations of acceleration are close to its average value. We consider that it is measured as the reciprocal of the acceleration variance, i.e.,

$$\mathcal{M}_c = (\frac{1}{t_{sim}} \int_0^{t_{sim}} (a(t) - \bar{a})^2 dt)^{-1} \tag{13}$$

  where a is the acceleration of the ego vehicle, and $\bar{a} = \frac{1}{t_{sim}} \int_0^{t_{sim}} a(t)dt$. Note that the higher the value $\mathcal{M}_c$, the higher the comfort level.

2.3.2. Evaluation of the two controllers

Nominal scenarios. Fig. 5 shows the speed of the ego vehicle for the two controllers and the maximal safe speed v_{max} for the nominal scenarios. Note that the MPC Controller closely follows the speed of the vehicle ahead, in particular when the period of the speed function increases, e.g.,$T = 30$ s. However, the speed of the MPC Controller

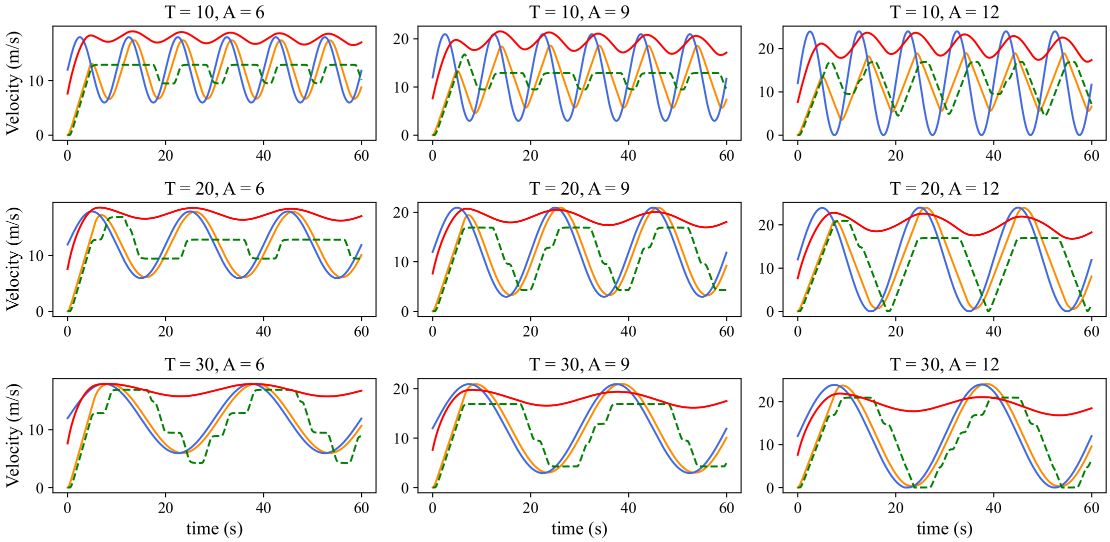


Fig. 5. Speed for the Safe Controller (in green), the MPC Controller (in orange) and the maximal speed v_{max} (in red) for nominal scenarios. The speed of the vehicle ahead is shown in blue. Note the unsafe situations where $v_{mpc} > v_{max}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

cannot avoid unsafe situations as shown in Fig. 5 when the orange line (v_{mpc}) crosses the red line (v_{max}). On the contrary, the Safe Controller is less sensitive to speed changes of the vehicle ahead and allows smaller speed variation due to safety constraints. Despite these constraints, the performance measured as the average speed has not been sacrificed. The upmost part of Table 1 provides performance metrics showing that in most cases, the Safe Controller produces higher average speeds than the MPC Controller and thus maintains slightly higher ratios. Nonetheless, the differences increase when the amplitude of the speed function becomes larger, e.g., A = 12.

Fig. 6 provides the relative distance for the two controllers. It shows that the amplitude variation for the MPC Controller is much smaller, especially when the period and the amplitude of the speed function are larger. This is because the MPC Controller favors speed tracking. On the contrary, the Safe Controller maintains a smaller relative distance on average than the MPC Controller, since its control policy focuses on distance minimization. This observation is confirmed by the higher occupancy metrics for the Safe Controller provided in Table 1.

The comfort metrics provided by Table 1 show that comfort for the Safe Controller is much higher than for the MPC Controller when the period and the amplitude of the speed function are small. The reason is that the Safe Controller is less sensitive to the speed changes of the vehicle ahead and avoids alternating changes of acceleration and deceleration. Nonetheless, for larger periods, e.g. $T = 30$, the Safe Controller is less comfortable.

Out-of-nominal scenarios. Fig. 7 and Fig. 8 show the speed and the relative distance for the emergency scenarios, respectively. Note that both controllers react to a sudden brake of the vehicle ahead. The Safe Controller always maintains a safe distance between the two vehicles, thus avoiding collision. On the contrary, the MPC Controller is unsafe in two out of nine cases, in particular, when the period of the speed function becomes large. For instance, in Fig. 8 we can see that a collision occurs after 40 s when $T = 30$ s and A = 12 m/s (the blue dashed line marks the beginning of the braking).

3. Hybrid collision avoidance control

3.1. Design and implementation of the hybrid controller

The above comparative evaluation confirms that in most cases the MPC Controller is slightly better in terms of performance and occupancy, while it fails to be safe for out-of-nominal scenarios when the vehicle ahead abruptly stops. As expected the Safe Controller preserves safety in all scenarios and assures more comfortable driving.

Table 1
Efficiency metrics for the Safe Controller and the MPC Controller for nominal scenarios when the initial relative distance is 10 m.

Performance	A = 6		A = 9		A = 12	
	MPC	Safe	MPC	Safe	MPC	Safe
T = 10	0.946	0.963	0.945	0.965	0.943	0.974
T = 20	0.945	0.965	0.937	0.968	0.936	0.971
T = 30	0.948	0.954	0.942	0.963	0.934	0.963
Occupancy	A = 6		A = 9		A = 12	
	MPC	Safe	MPC	Safe	MPC	Safe
T = 10	0.022	0.025	0.019	0.024	0.017	0.026
T = 20	0.023	0.025	0.02	0.027	0.018	0.03
T = 30	0.025	0.026	0.022	0.029	0.019	0.034
Comfort	A = 6		A = 9		A = 12	
	MPC	Safe	MPC	Safe	MPC	Safe
T = 10	0.182	0.587	0.141	0.423	0.133	0.168
T = 20	0.437	0.652	0.234	0.284	0.155	0.19
T = 30	0.698	0.411	0.411	0.415	0.267	0.252

We have explained the design principle of the Hybrid Controller in the Introduction (Fig. 1). It consists of the Safe Controller and the MPC Controller running in parallel and a Switch deciding which one of the control stimuli takes effect. The Switch produces the target speed of the ego vehicle taking care that it never exceeds the maximal safe speed v_{max} computed as a function of the distance d and the maximal deceleration of the ego vehicle. For deceleration a_{max}, we have $v_{max} = (2 \cdot a_{max} \cdot d)^{1/2}$.

In the Hybrid Controller under the constraint $v \leq v_{max}$, the Switch selects the highest speed between v_{mpc} and v_{safe} so as to achieve the best performance. Hence, it continuously applies the following rules to select the target speed v of the Hybrid Controller:

```
if $v_{safe} \leq v_{mpc} \leq v_{max}$ then $v := v_{mpc}$
else if $v_{mpc} \leq v_{safe}$ then $v := v_{safe}$
else if $v_{max} \leq v_{mpc}$ then $v := v_{max}$
```

Notice that in all cases the rules prevent the target speed v from exceeding v_{max}. In nominal conditions it can happen that $v_{mpc} \leq v_{safe}$. This is the case when the distance d is large enough and the speed of the vehicle ahead is decreasing. The speed v_{safe} can exceed v_{mpc} as the

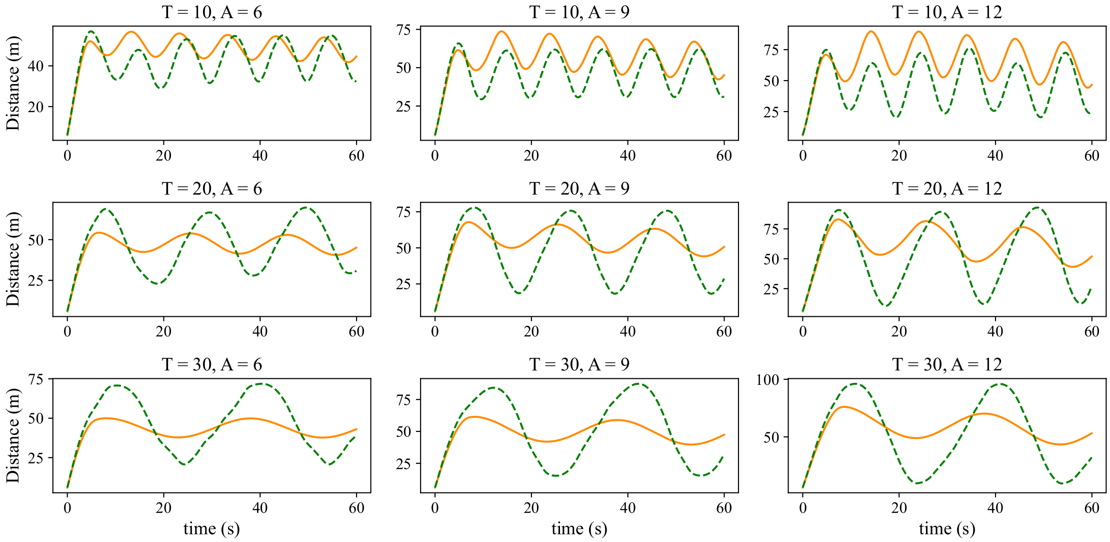


Fig. 6. Relative distance for the Safe Controller (in green) and the MPC Controller (in orange) for nominal scenarios. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

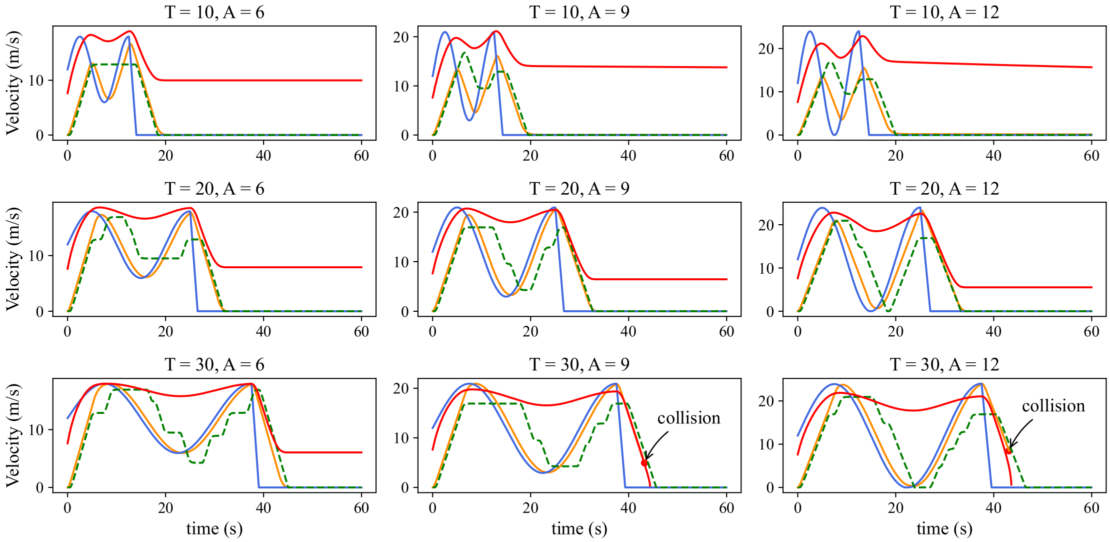


Fig. 7. Speed for the Safe Controller (in green), the MPC Controller (in orange) and the maximal speed v_{max} (in red) in out-of-nominal scenarios when the vehicle ahead suddenly brakes. The speed of the vehicle ahead is shown in blue. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Safe Controller focuses on minimizing the relative distance while the MPC Controller tracks the speed of the vehicle ahead.

An important difference from similar works dealing with hybrid controllers [15,16] is that our Safe Controller contributes not only to safety but also to a large extent to performance and comfort and even in some cases to improve occupancy. This observation is confirmed by experimental results provided in the next section. Notice that this hybrid control principle can be applied by replacing in our architecture the MPC Controller by other learning-based controllers [27].

3.2. Evaluation of the hybrid controller

We consider both nominal and out-of-nominal scenarios as before and adopt the same experimental settings. For nominal scenarios, we compare the efficiency of the three controllers. Furthermore, for out-of-nominal scenarios we consider additional braking rates of the vehicle ahead.

Nominal scenarios. Fig. 9 depicts the speed of the ego vehicle for the three controllers in nominal scenarios. It shows that the Hybrid Controller can also track the speed of the vehicle ahead closely, taking the best from the two controllers. Notice that the purple line can be above the orange line, for instance during simulation time around 20 s when $T = 30$ and $A = 9$. The upmost part of Table 2 provides results comparing the performance of the three controllers. Note that the Hybrid Controller outperforms the two other controllers.

Fig. 10 depicts the relative distance for the three controllers. As expected the distance maintained by the Hybrid Controller is in general smaller taking advantage of the strength of the Safe Controller for minimizing the relative distance. This is also shown in the middle part of Table 2, which compares the occupancy for the three controllers. The Hybrid Controller achieves higher occupancy than the other two controllers.

The comfort metrics provided in the bottom part of Table 2, show that when the period and the amplitude of the speed function are small, the Safe Controller produces the most comfortable driving policies. While when they become larger, the Hybrid Controller is better than the Safe Controller and is slightly outperformed by the MPC Controller.

Table 3 shows time percentages corresponding to the application by the Hybrid Controller of the MPC policy, the nominal safe policy

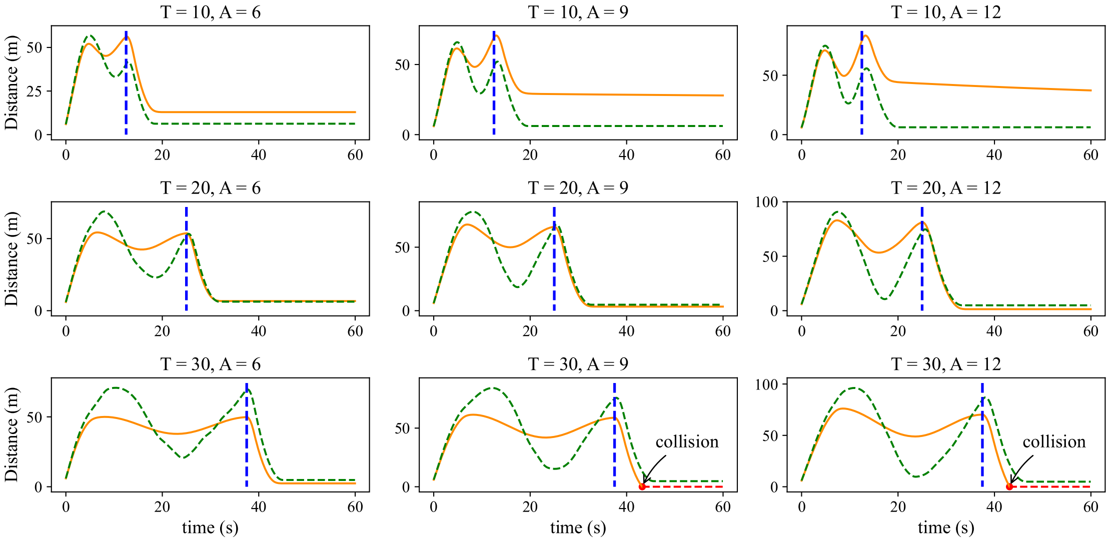


Fig. 8. Relative distance for the Safe Controller (in green) and the MPC Controller (in orange) when the vehicle ahead suddenly brakes at the moment indicated by the dashed blue line. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

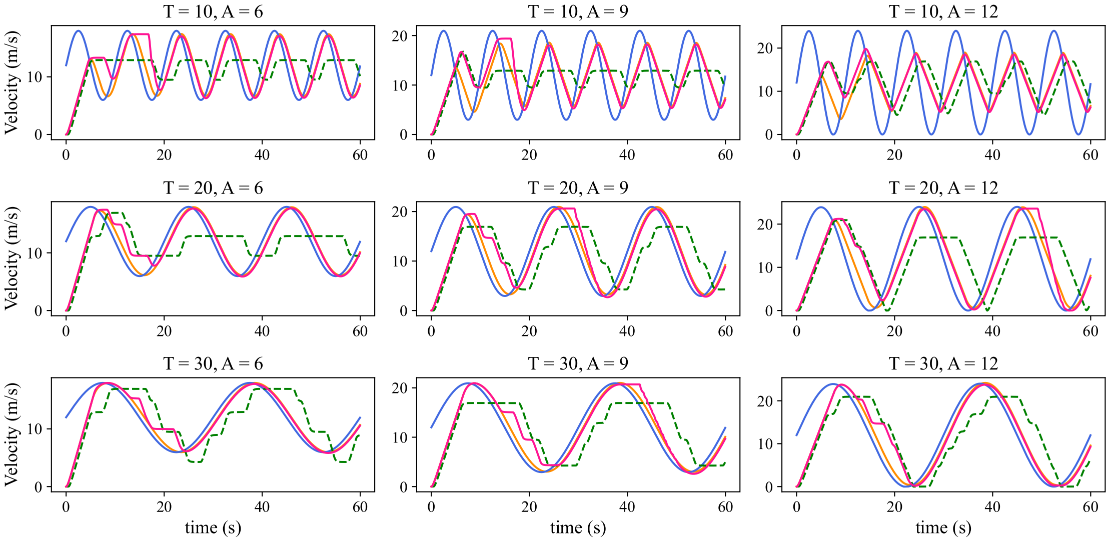


Fig. 9. Speed for the Hybrid Controller (in purple), the MPC Controller (in orange), the Safe Controller (in green) when the speed of the vehicle ahead is a sinusoidal function (in blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and the out-of-nominal safe policy. We can see that the MPC Controller contributes more than the other two, while the contribution of the Safe Controller is non-negligible. The maximal safe speed v_{max} is applied to a very small percentage of cases to ensure safety.

Out-of-nominal scenarios. Figs. 11 and 12 provide results for the three controllers in the out-of-nominal scenarios where the vehicle ahead suddenly brakes with rate 12 m/s^2. Note that the MPC Controller becomes unsafe for increasing amplitude and period of the speed of the vehicle ahead. There are two out of nine settings where the MPC control policy results in collision. In Appendix, additional experiments for braking rates 4 m/s^2 and 8 m/s^2 are provided. They show that the MPC Controller is safe for all scenarios with braking rate 4 m/s^2 while it is fails to be safe in one out of nine scenarios with braking rate 8 m/s^2.

4. Conclusion and discussion

We propose a method for building a hybrid safe-by-construction and efficient collision avoidance controller. The controller integrates a MPC Controller, a discrete Safe Controller and a Switch that combines the outputs of the two controllers to generate stimuli that are safe and efficient. We show experimentally that the Hybrid Controller besides guaranteeing safety, ensures high efficiency because it “takes the best” from each one of the integrated controllers. The MPC Controller seeks policies that reduce both the relative speed and the relative distance while in nominal scenarios the Safe Controller seeks minimization of the relative distance. We show that this hybrid control policy ensures a very good efficiency measured by three criteria: performance, road occupancy and comfort.

We adopt a pragmatic and progressive methodology based on the comparative evaluation of the two constituent controllers for both nominal and out-of-nominal scenarios. The evaluation provides a good insight on the merits of the respective control principles which motivates the design of the Hybrid Controller. The experimental results confirm the feasibility and the practical relevance of hybrid controllers for safe and efficient driving.

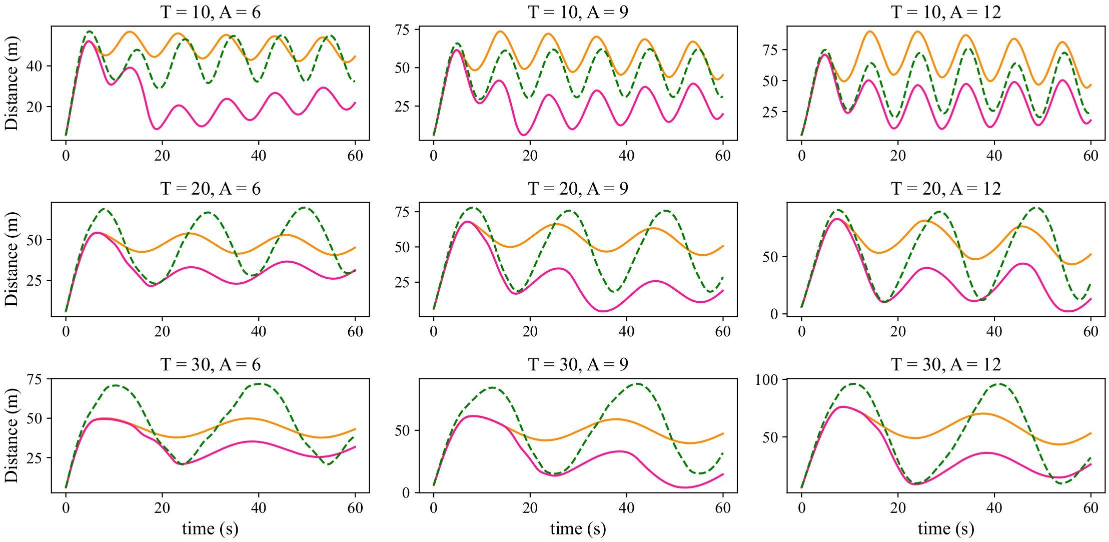


Fig. 10. Relative distance for the Hybrid Controller (in purple), the MPC Controller (in orange) and the Safe Controller (in green) when the speed of the vehicle ahead is a sinusoidal function. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

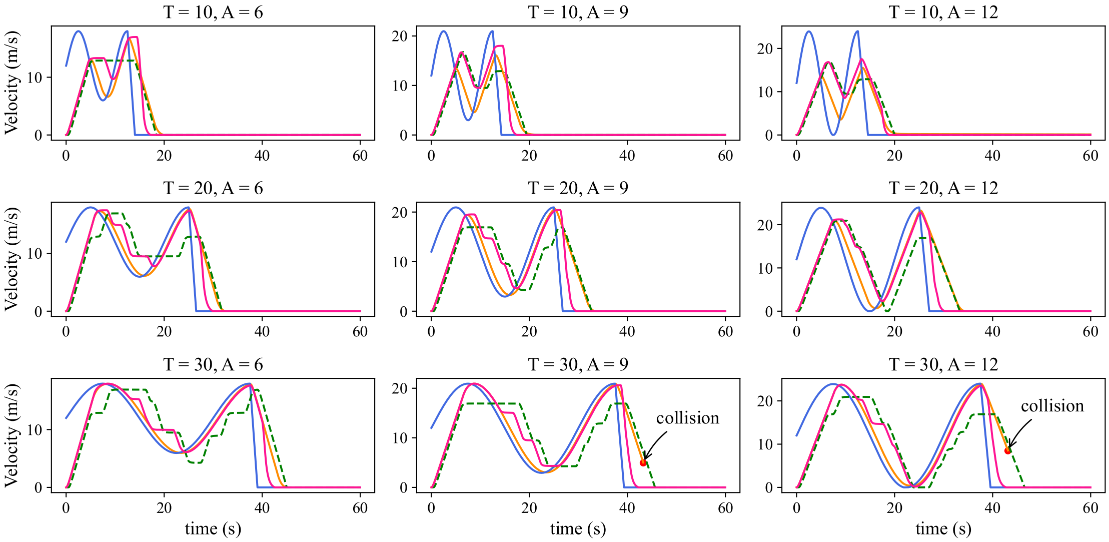


Fig. 11. Speed of the ego vehicle for the Hybrid Controller (in purple), the MPC Controller (in orange) and the Safe Controller (in green) when the vehicle ahead brakes with rate 12 m/s^2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Efficiency metrics for the Hybrid Controller, the Safe Controller and the MPC Controller for nominal scenarios.

Performance	A = 6			A = 9			A = 12		
	MPC	Safe	Hybrid	MPC	Safe	Hybrid	MPC	Safe	Hybrid
T = 10	0.946	0.963	0.978	0.945	0.965	0.981	0.943	0.974	0.983
T = 20	0.945	0.965	0.965	0.937	0.968	0.982	0.936	0.971	0.99
T = 30	0.948	0.954	0.964	0.942	0.963	0.988	0.934	0.963	0.972
Occupancy	A = 6			A = 9			A = 12		
	MPC	Safe	Hybrid	MPC	Safe	Hybrid	MPC	Safe	Hybrid
T = 10	0.022	0.025	0.05	0.019	0.024	0.047	0.017	0.026	0.038
T = 20	0.023	0.025	0.034	0.02	0.027	0.061	0.018	0.03	0.069
T = 30	0.025	0.026	0.033	0.022	0.029	0.065	0.019	0.034	0.043
Comfort	A = 6			A = 9			A = 12		
	MPC	Safe	Hybrid	MPC	Safe	Hybrid	MPC	Safe	Hybrid
T = 10	0.182	0.587	0.177	0.141	0.423	0.123	0.133	0.168	0.134
T = 20	0.437	0.652	0.429	0.234	0.284	0.214	0.155	0.19	0.142
T = 30	0.698	0.411	0.632	0.411	0.415	0.357	0.267	0.252	0.258

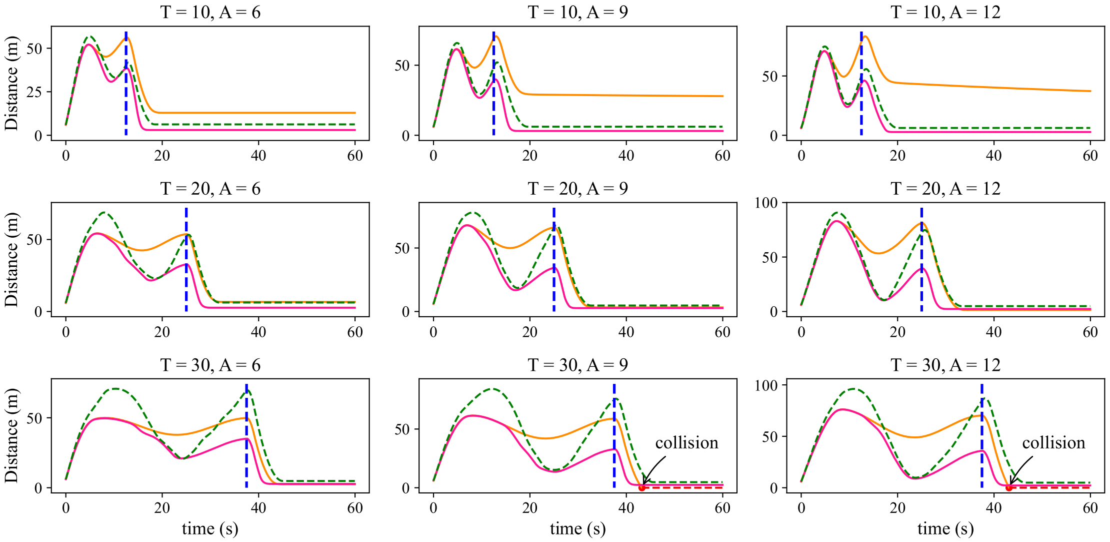


Fig. 12. Relative distance for the Hybrid Controller (in purple), the MPC Controller (in orange) and the Safe Controller (in green) when the vehicle ahead brakes with rate 12 m/s^2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Percentage of time in use for the Safe Controller, the MPC Controller and the maximal safe speed v_{max} for nominal scenarios.

Percentage of time (%)	A = 6			A = 9			A = 12		
	MPC	Safe	MAX	MPC	Safe	MAX	MPC	Safe	MAX
T = 10	69.567	29.367	1.067	84.467	15.533	0	87.8	12.2	0
T = 20	63.267	35.667	1.067	59.567	38.3	2.133	69.367	28	2.633
T = 30	68.067	31.933	0	49.267	48.6	2.133	71.033	28.967	0

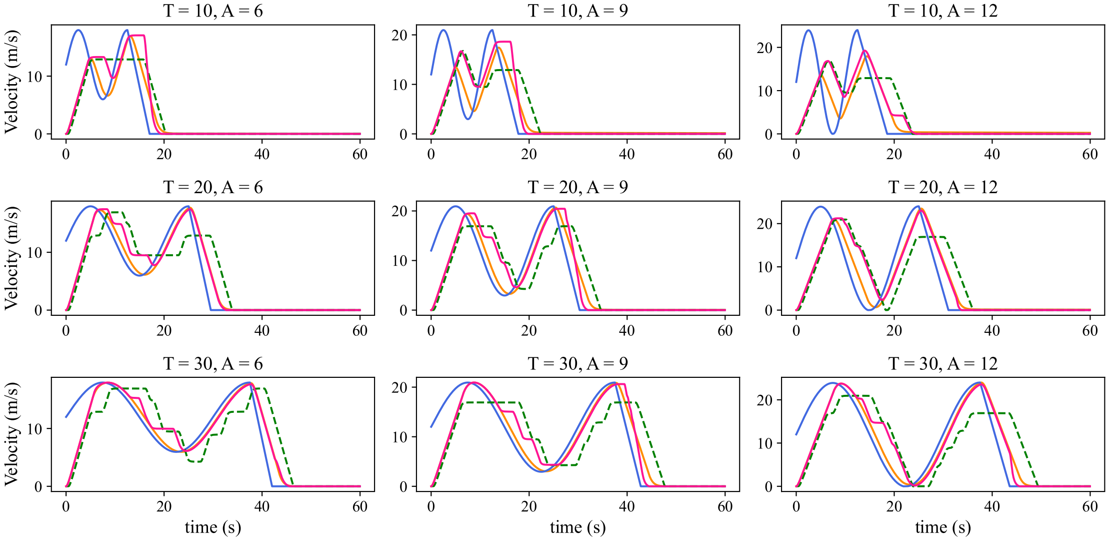


Fig. 13. Speed of the ego vehicle for the Hybrid Controller (in purple), the MPC Controller (in orange) and the Safe Controller (in green) when the vehicle ahead brakes with rate 4 m/s^2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

A key and original lesson from our results is that the Safe Controller is not simply a monitor that takes over in critical situations. It also significantly contributes to efficiency applying a control policy that nicely complements the MPC policy. The experimental results show that the interplay between the dynamics of discrete and continuous controllers pursuing complementary objectives can be surprisingly rich. Its study may lead to more elaborated and enhanced hybrid policies. In future work we will investigate our hybrid control principle by replacing the MPC Controller with other types of adaptive controllers, e.g., machine-learning-based controllers [27].

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the Zhejiang Province Nature Science Foundation of China under Grant LZ22F020003, and in part by the HDU-CECDATA Joint Research Center of Big Data Technologies under Grant KYH063120009.

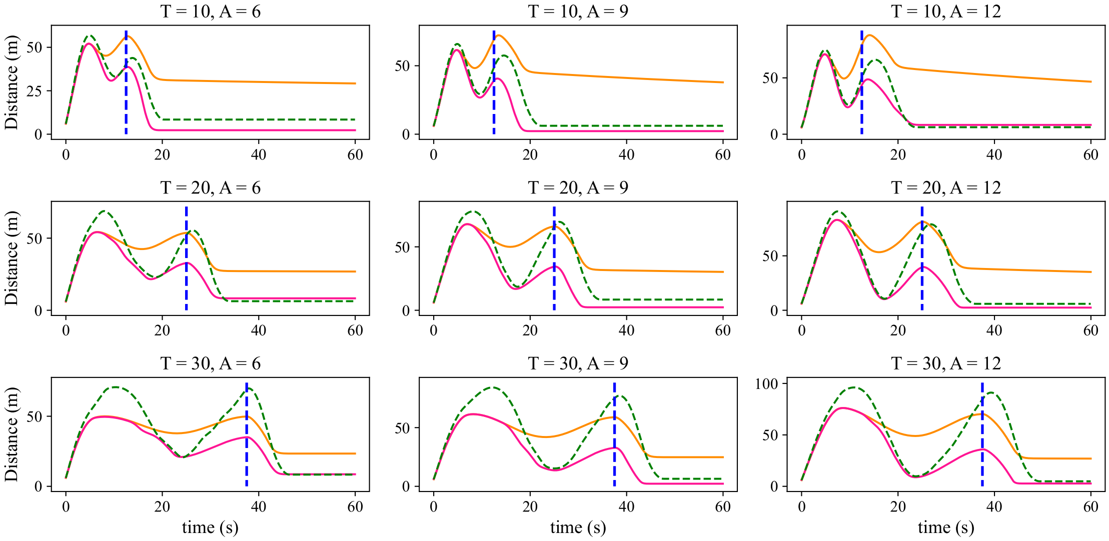


Fig. 14. Relative distance for the Hybrid Controller (in purple), the MPC Controller (in orange) and the Safe Controller (in green) when the vehicle ahead brakes with rate 4 m/s^2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

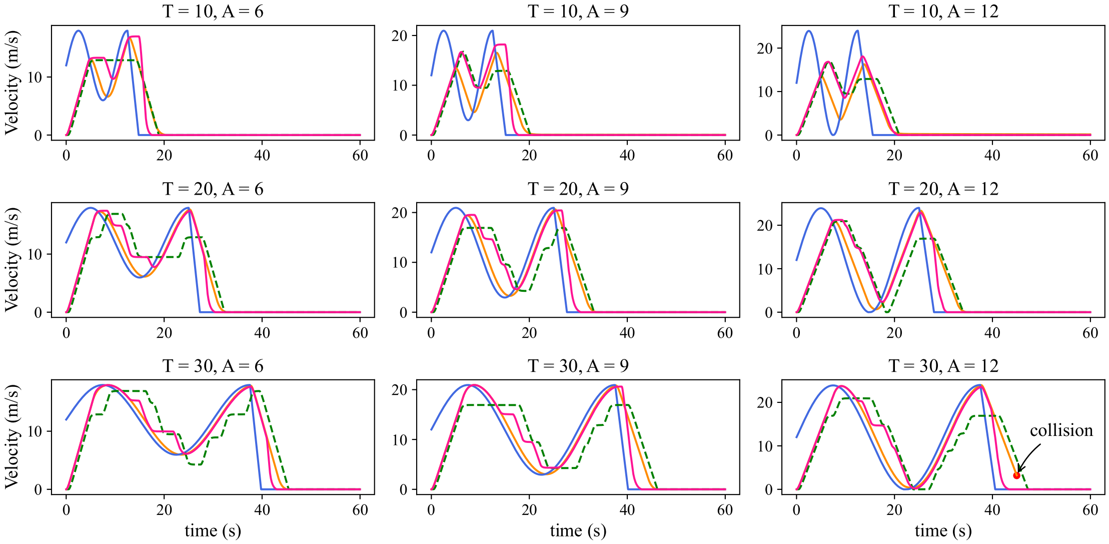


Fig. 15. Speed of the ego vehicle for the Hybrid Controller (in purple), the MPC Controller (in orange) and the Safe Controller (in green) when the vehicle ahead brakes with rate 8 m/s^2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

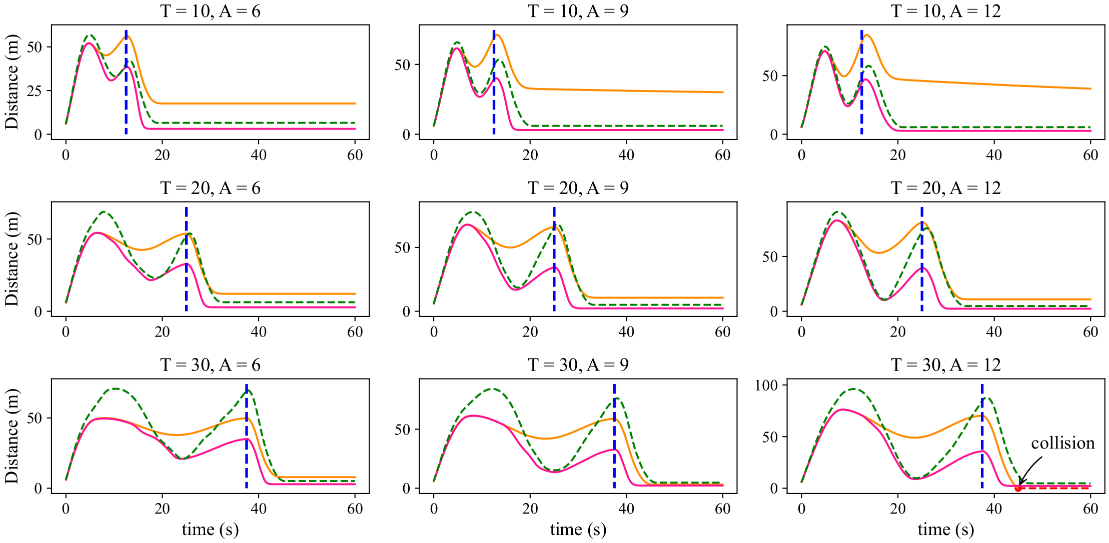


Fig. 16. Relative distance for the Hybrid Controller (in purple), the MPC Controller (in orange) and the Safe Controller (in green) when the vehicle ahead brakes with rate 8 m/s^2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Appendix

See Figs. 13–16.

References

[1] J. Park, D. Kim, Y. Yoon, H. Kim, K. Yi, Obstacle avoidance of autonomous vehicles based on model predictive control, Proc. Inst. Mech. Eng. D.

[2] D.Q. Mayne, Model predictive control: Recent developments and future promise, Automatica 50 (12) (2014) 2967–2986, http://dx.doi.org/10.1016/j.automatica.2014.10.128, http://www.sciencedirect.com/science/article/pii/S0005109814005160.

[3] H. Li, Y. Shi, Distributed model predictive control of constrained nonlinear systems with communication delays, Systems Control Lett. 62 (10) (2013) 819–826, http://dx.doi.org/10.1016/j.sysconle.2013.05.012.

[4] H. Li, Y. Shi, Robust distributed model predictive control of constrained continuous-time nonlinear systems: A robustness constraint approach, IEEE Trans. Automat. Control 59 (6) (2014) 1673–1678, http://dx.doi.org/10.1109/TAC.2013.2294618.

[5] D. Mayne, J. Rawlings, C. Rao, P. Scokaert, Constrained model predictive control: Stability and optimality, Automatica 36 (6) (2000) 789–814, http://dx.doi.org/10.1016/S0005-1098(99)00214-9.

[6] S.M. Loos, A. Platzer, L. Nistor, Adaptive cruise control: Hybrid, distributed, and now formally verified, in: International Symposium on Formal Methods, Springer, 2011.

[7] A. Rizaldi, F. Immler, B. Schürmann, M. Althoff, A formally verified motion planner for autonomous vehicles, in: Automated Technology for Verification and Analysis, Springer, 2018.

[8] S. Shalev-Shwartz, S. Shammah, A. Shashua, On a formal model of safe and scalable self-driving cars, CoRR abs/1708.06374 arXiv:1708.06374.

[9] P. Nilsson, O. Hussien, A. Balkan, Y. Chen, A.D. Ames, J.W. Grizzle, N. Ozay, H. Peng, P. Tabuada, Correct-by-construction adaptive cruise control: Two approaches, IEEE Trans. Control Syst. Technol. (2015).

[10] S. Sadraddini, S. Sivaranjani, V. Gupta, C. Belta, Provably safe cruise control of vehicular platoons, IEEE Control Syst. Lett. (2017).

[11] T. Korssen, V. Dolk, J. van de Mortel-Fronczak, M. Reniers, M. Heemels, Systematic model-based design and implementation of supervisors for advanced driver assistance systems, IEEE Trans. Intell. Transp. Syst. (2017).

[12] J. Krook, L. Svensson, Y. Li, L. Feng, M. Fabian, Design and formal verification of a safe stop supervisor for an automated vehicle, in: 2019 International Conference on Robotics and Automation, 2019.

[13] Q. Wang, D. Li, J. Sifakis, Safe and efficient collision avoidance control for autonomous vehicles, in: MEMOCODE, 2020.

[14] J.R. Mayo, R.C. Armstrong, G.C. Hulette, M. Salloum, A.M. Smith, Robust digital computation in the physical world, Cyber-Phys. Syst. Secur. (2018).

[15] M. Althoff, S. Maierhofer, C. Pek, Provably-correct and comfortable adaptive cruise control, IEEE Trans. Intell. Veh. (2020) 1, http://dx.doi.org/10.1109/TIV.2020.2991953.

[16] S. Magdici, M. Althoff, Adaptive cruise control with safety guarantees for autonomous vehicles, Proc World Congr. Int. Fed. Automatic Control (2017) 5774–5781.

[17] D. Jagga, M. Lv, S. Baldi, Hybrid adaptive chassis control for vehicle lateral stability in the presence of uncertainty, in: 2018 26th Mediterranean Conference on Control and Automation, MED, 2018, pp. 1–6, http://dx.doi.org/10.1109/MED.2018.8442921.

[18] L. Sha, Using simplicity to control complexity, IEEE Softw. 18 (4) (2001) 20–28, http://dx.doi.org/10.1109/MS.2001.936213.

[19] D. Cofer, I. Amundson, R. Sattigeri, A. Passi, S. Rayadurgam, Run-time assurance for learning-enabled systems, in: NASA Formal Methods, 2020.

[20] D. Cofer, I. Amundson, R. Sattigeri, A. Passi, S. Rayadurgam, Run-time assurance for learning-based aircraft taxiing, in: 2020 IEEE/AIAA 39th Digital Avionics Systems Conference, DASC, 2020.

[21] P. Vivekanandan, G. Garcia, H. Yun, S. Keshmiri, A simplex architecture for intelligent and safe unmanned aerial vehicles, in: 2016 IEEE 22nd International Conference on Embedded and Real-Time Computing Systems and Applications, RTCSA, 2016, pp. 69–75, http://dx.doi.org/10.1109/RTCSA.2016.17.

[22] A. Desai, S. Ghosh, S.A. Seshia, N. Shankar, A. Tiwari, Soter: A runtime assurance framework for programming safe robotics systems, in: 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN, 2019, pp. 138–150, http://dx.doi.org/10.1109/DSN.2019.00027.

[23] K. Heckemann, M. Gesell, T. Pfister, K. Berns, K. Schneider, M. Trapp, Safe automotive software, in: Proceedings of the 15th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems - Volume Part IV, KES'11, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 167–176.

[24] R.C. Dorf, R.H. Bishop, Modern Control Systems, Pearson, 2011.

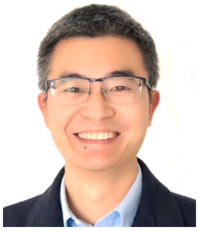
[25] M. Andersen, J. Dahl, Z. Liu, L. Vandenberghe, S. Sra, S. Nowozin, S. Wright, Interior-point methods for large-scale cone programming, Optim. Mach. Learn. 5583.

[26] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, V. Koltun, CARLA: An open urban driving simulator, in: Proceedings of the 1st Annual Conference on Robot Learning, 2017, pp. 1–16.

[27] M. Bojarski, D.D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L.D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, K. Zieba, End to end learning for self-driving cars, 2016, CoRR abs/1604.07316 arXiv:1604.07316.

Qiang Wang received his bachelor and master degree from the National University of Defense Technology, China in 2010 and 2012, respectively. He obtained his Ph.D. from EPFL, Switzerland in 2017, where he has been a key person in the development of the component-based system design and model checking framework named BIP. Currently, his research interests focus on the formal safety analysis and verification techniques for autonomous systems.

Xinlei Zheng received the B.S. degree from Hangzhou Dianzi University, Hangzhou, China, in 2019. He is currently pursuing the Master Degree in Hangzhou Dianzi University, Hangzhou, China. His main research interests include autonomous vehicle control and cooperative control.



Jiyong Zhang received his Ph.D. degree in Computer Science from Swiss Federal Institute of Technology at Lausanne (EPFL) in 2008. He received the B.S. degree and the M.S. degree in Computer Science from Tsinghua University in 1999 and 2001. He is currently a distinguished professor in Hangzhou Dianzi University. His research interests include intelligent information processing, machine learning, data sciences and recommender systems.

Joseph Sifakis received his bachelor's degree in electrical engineering from the Technical University of Athens in 1969 and his Ph.D. in computer science from the University of Grenoble, France, in 1974. He was elected member of the French Academy of Engineering and the European Academy of Sciences in 2008, member of the French Academy of Sciences in 2010, member of the American Academy of Humanities and Sciences in 2015, and foreign member of the American Academy of Engineering in 2017. He is currently the Director Researcher of the French National Science Center. His main research areas are model detection and embedded system design and verification.