# Integrated Perception-Decision-Control Car Following Control under Communication Uncertainty

Ran Tao and Shifeng Wang

*Abstract*—**The vehicle to everything (V2X) communication of intelligent and connected vehicles faces uncertainties such as random latency and packet loss. These issues pose safety challenges. This paper proposes a closed-loop integrated strategy that combines perception, decision-making, and control. A communication uncertainty metric is established based on recent packet loss rate and average delay. A three-state machine (cooperative cruise mode, perception uncertainty mode, and local safety mode) is designed to adjust the control strategy based on a preset uncertainty threshold. A proportion integration differentiation (PID) controller ensures safe car following. MATLAB simulations under high uncertainty show that the proposed method guarantees that the following distance always satisfies the dynamic safety criterion (a time headway of 1.5 s plus a standstill gap), whereas the fixed-strategy baseline fails under high uncertainty, significantly outperforming a fixed-strategy baseline. This method is computationally efficient and can be deployed on embedded devices. Simulation results validate the effectiveness of this approach in handling V2X communication uncertainties.**

*Index Terms*—**Intelligent and connected vehicle, vehicle to everything (V2X) communication uncertainty, perception-decision-control integration, state machine, safe car following**

## I. Introduction

INTELLIGENT and connected vehicles (ICVs) exchange information with roadside facilities, neighboring vehicles, and the cloud via vehicle to everything (V2X) communication. This technology is crucial for improving road traffic safety, alleviating congestion, and reducing energy consumption. V2X, based on the 5G new radio (NR) interface, provides high throughput, extremely low latency, and high reliability, making it important for autonomous driving [1]. Recent reviews have summarized connected and automated vehicle (CAV) longitudinal following research under mixed traffic and communication failure scenarios. These studies consider the coupling between communication and control, discussing key indicators such as control strategies, headway, packet loss rate, and string stability margin under communication failure [2]. In typical scenarios, the preceding vehicle communicates speed and position via V2X, and the ego vehicle dynamically adjusts its longitudinal motion, thereby enhancing safety and efficiency.

In practice, V2X communication links often have random latency (variable time delays) and packet loss (missing data packets). Research shows that end-to-end latency in 5G networks can reach tens of milliseconds. In complex environments, the packet loss rate (ratio of lost packets) may exceed 10% [1]. Simulations in Ref. [3] indicate that increasing latency (communication delay) from 0 to 8 ms increases the formation-spacing error (the difference from ideal spacing) from 0.10 to 0.78 m. When the packet loss rate hits 30%, acceleration fluctuates between [−4.8, 2.2] m/s$^2$ [3]. This uncertainty causes lag or loss of information from the preceding vehicle. If not managed, it can lead to following too closely, large speed fluctuations, or even collisions. In uncertain communication environments, lacking certainty, ensuring the stability and safety of perception (obtaining environment information), decision-making (choosing actions), and control (executing vehicle actions) in the closed-loop system is a significant challenge.

Existing research uses several methods to address the impact of communication issues. Reference [4] applies Kalman filtering to correct V2X time delay, but assumes no packet loss. Molina-Masegosa et al. create a maneuverable coordination state machine [5]. However, they did not quantify uncertainty. Cao et al. highlight that network uncertainty threatens queue stability [6]. In a subsequent review, the authors discuss core issues such as information flow topology, inter-vehicle spacing strategy, and platoon stability. They also summarize future research directions for cooperative adaptive cruise control (CACC) [6]. Most current strategies handle perception, decision-making, and control separately, lacking integration. Assuming ideal communication, ignoring packet loss is far from realistic in real-world application environments. Building directly from these identified gaps, the main contributions of this paper are as follows:

(1) We propose a communication uncertainty metric based on recent packet loss rate and average delay.

(2) A three-mode state machine (cooperative cruise mode, perception uncertainty mode, and local safety mode) is developed, which automatically switches control strategies according to the uncertainty threshold.

(3) This work validates a proportion integration differentiation (PID) controller for safe car following through MATLAB simulations.

Manuscript received: 6 May 2025; revised: 19 May 2025; accepted: 1 June 2025. *(Corresponding author: Ran Tao.)*

Citation: R. Tao and S. Wang, Integrated perception-decision-control car following control under communication uncertainty, *Int. J. Intell. Control Syst.*, 2026, 31(2), 185–192.

Ran Tao and Shifeng Wang are with New Energy and Intelligent Connected Vehicles College, Sanya University, Sanya 572022, China (e-mail: taoran@sanyau.edu.cn; shifengwang@sanyau.edu.cn).

Digital Object Identifier 10.62678/IJICS202606.10369

(4) We further discuss the method's deployability on embedded platforms, leveraging the authors' previous experience with STM32-based systems.

Compared with existing V2X car following approaches, the novelty of this work lies in three aspects. First, while most methods assume ideal communication or treat delay and packet loss separately [4, 6], we propose a unified uncertainty metric $U(t)$ that quantifies communication quality in real time using only short-term statistics (recent loss rate and average delay). This metric is light enough for embedded implementation yet sufficiently informative for decision-making. Second, instead of relying on complex delay-compensation filters [4] or model-predictive control [7], we introduce a three-mode state machine that directly maps $U(t)$ to control strategies (cooperative cruise mode, perception uncertainty mode, and local safety mode). This allows the system to actively degrade its performance, by increasing the following distance, when communication worsens—a proactive efficiency for safety trade-off that is rarely addressed in the CACC literature [6, 8]. Third, the entire perception-decision-control loop is designed with embedded deployment in mind: it uses only sliding-window statistics, threshold comparisons, and a basic PID controller, requiring no high-precision floating-point operations or deep neural networks. This makes it suitable for automotive microcontrollers such as STM32, which is a practical advantage over more computationally intensive methods [7].

The remainder of this paper is organized as follows. Section II presents system modeling and problem description. Section III details the design of the integrated perception-decision-control method. Section IV provides simulation verification and result analysis. Section V concludes the paper and discusses future work.

## II. System Modeling and Problem Description

### A. Longitudinal Following Scenario

In a typical single-vehicle longitudinal following scenario, the vehicle under control, ego vehicle, relies on V2X communication to obtain the instantaneous speed of the preceding (lead) vehicle and adjusts its motion along the direction of travel (longitudinal dynamics, meaning acceleration and braking in the lane) to maintain a safe following distance. In this scenario, vehicles are assumed to move in a straight line in one lane, and any sideways (lateral) movement is ignored.

**Notation** $x_{\mathrm{ego}}$ represents the ego vehicle position, $v_{\mathrm{ego}}$ represents the ego vehicle speed, $x_{\mathrm{front}}$ is the preceding vehicle position, $v_{\mathrm{front}}$ indicates the preceding vehicle speed, and $d = x_{\mathrm{front}} - x_{\mathrm{ego}}$ denotes the actual following distance.

The objective is to control the time gap between vehicles $d(t)$, which is defined as the following distance, or the interval seconds in seconds or meters maintained between the leading vehicle and the following vehicle, to approach the ideal value $d_{\mathrm{des}}(t)$, ensure $d(t) > 0$, and enhance the accuracy of vehicle speed tracking.

### B. Vehicle Longitudinal Dynamics Model

Represent the longitudinal dynamics with a first-order inertial model

$$\tau \dot{a}(t) + a(t) = a_{\mathrm{des}}(t),$$

where the vehicle's actual acceleration $a(t)$ is the rate of change of its speed, the target acceleration $a_{\mathrm{des}}(t)$ is the value set by the control algorithm, and $\tau$ is the actuator time constant describing how long the physical system takes to respond to input.

The mathematical equations for the state of motion of an object and its change are

$$\dot{x}_{\mathrm{ego}} = v_{\mathrm{ego}}, \quad \dot{v}_{\mathrm{ego}} = a,$$

with physical constraints $0 \leqslant v_{\mathrm{ego}} \leqslant v_{\max}$ and $a_{\min} \leqslant a \leqslant a_{\max}$. In this study, $v_{\max}$ is set to 30 m/s, $a_{\max}$ is set to 3 m/s$^2$, and $a_{\min}$ is set to $-5$ m/s$^2$.

The velocity trajectory of the preceding vehicle, lead vehicle speed changes over time, is represented by a smooth transition function $v_{\mathrm{front}}(t)$.

### C. Modeling of Uncertainty in V2X Communication

The V2X communication link experiences random delays and packet loss.

Assume that the delay for each packet follows a uniform probability distribution delay $\sim \mathcal{U}(0, \mathrm{delay}_{\max})$ with $\mathrm{delay}_{\max}$ = 0.3 s, and that packet loss events are modeled as a Bernoulli distribution (a statistical model for yes/no outcomes where each packet is lost independently) with a packet loss rate $p_{\mathrm{loss}}$ = 0.1. When a packet is lost, it does not affect the latency statistics and only increases the packet loss count; the average latency is calculated only from successfully received packets.

Using the research method of Lu and Li [3], define the statistics within a sliding time window, $N_w$ = 20 sampling periods, corresponding to 1 s. Wang et al. [8] propose an adaptive spacing strategy switching mechanism based on packet loss rate (> 5%) and received signal strength indicator (< −90 dBm) during communication interruptions, providing a reference for threshold setting in uncertainty quantification.

Recent packet loss rate can be calculated by

$$\mathrm{loss}_{\mathrm{recent}} = \frac{\text{Number of packet losses}}{N_w},$$

where $N_w$ is the total number of transmission attempts in the window.

Average delay is represented by

$$\mathrm{delay}_{\mathrm{avg}} = \frac{1}{(N_{\mathrm{received}} \times \sum \mathrm{delay}_i)},$$

where $N_{\mathrm{received}}$ is the number of received packets in the window.

Comprehensive uncertainty index is given as

$$U(t) = 0.5 \cdot \mathrm{loss}_{\mathrm{recent}} + 0.5 \cdot \frac{\mathrm{delay}_{\mathrm{avg}}}{\mathrm{delay}_{\max}}.$$

$U(t) \in [0,1]$, where $U = 0$ means ideal (failure-free) communication conditions, and $U = 1$ means total communication failure—no successful messages within the window.

### D. Problem Description

A comprehensive closed-loop system integrating perception, decision-making, and control needs to be designed to achieve the following objectives. (1) Achieve stable car following in an ideal communication environment, where $d(t) \approx d_{\mathrm{des}}$, reducing the velocity tracking error. (2) When the system addresses the challenges posed by random latency and packet loss, it dynamically evaluates the uncertainty index $U(t)$ and intelligently adjusts certain control parameters (such as increasing the safety distance or reducing the expected driving speed) accordingly, to ensure that the actual inter-vehicle distance is not below the speed-based dynamic safety threshold. In this paper, the safety distance is defined using a constant headway distance $d_{\mathrm{safe}}(t) = \tau_h \cdot v_{\mathrm{ego}}(t) + d_0$, where $\tau_h = 1.5\ \mathrm{s}$ represents the safety headway distance and $d_0 = 2\ \mathrm{m}$ denotes the stopping distance. This requires $d(t) \geqslant d_{\mathrm{safe}}(t)$ to avoid the risk of rear-end collision caused by too close spacing at high speeds. (3) Enhance computational complexity to align with resource constraints. In subsequent simulation analyses, we will use this dynamic safety threshold as a benchmark to evaluate the method's collision-avoidance performance.

As noted in the latest review, longitudinal car following control in unreliable communication scenarios requires a balance between stability and safety margin, and there is an urgent need for lightweight, deployable fallback strategies [2].

## III. Design of Integrated Perception-Decision-Control Method

### A. Overall System Architecture

This paper proposes an architecture that integrates perception, decision-making, and control. The detailed design is illustrated in Fig. 1, which includes roadside units (RSUs), on-board units (OBUs), decision-making control units, and a cloud-based observation platform.

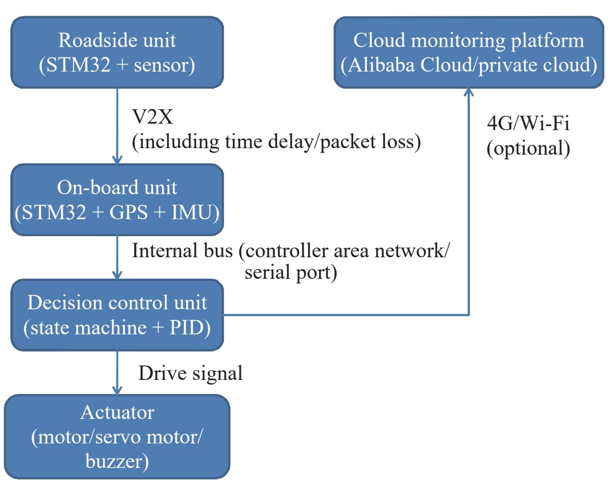



**Fig. 1** System architecture. GPS indicates global positioning system and IMU indicates inertial measurement unit.

RSU relies on sensors such as STM32 microcontrollers, ultrasonic sensors, and cameras to collect data on vehicles ahead, which is then transmitted to the OBU via V2X wireless technology. Random delays and packet loss in communication are key reasons for the perception uncertainty mode.

Relying on sensors such as STM32, GPS, and IMU, the OBU can accurately detect the vehicle's current location, travel speed, acceleration, and other dynamic conditions, and can also timely obtain data from the preceding vehicle via the RSU.

The decision control module integrates a state machine and a PID regulator, using sensory data and uncertainty to generate control signals that activate the corresponding actuators (motors, servos, buzzers, etc.).

The cloud platform uses 4G or Wi-Fi for data transmission, drawing on the author's previous experience developing Internet of Things (IoT) projects on the Alibaba Cloud platform.

Compared to the two-layer architecture of "collaborative decision-making layer and motion control layer" proposed by Yin et al. [9], this study further incorporates a communication uncertainty quantification module.

### B. Closed-Loop Control Process

The closed-loop control block diagram is shown in Fig. 2, consisting of four modules connected in series and forming a feedback loop.

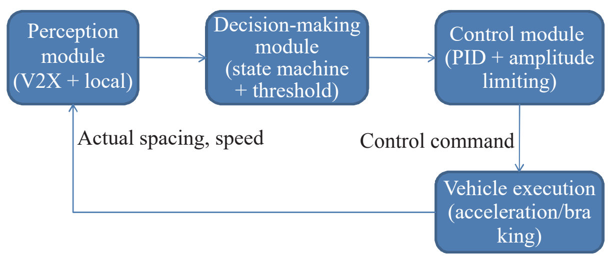



**Fig. 2** Closed-loop control block diagram.

**Perception module:** Integrate data from GPS, IMU, and V2X to output the following distance $d_{\mathrm{actual}}$, the ego vehicle speed $v_{\mathrm{ego}}$, the speed of the preceding vehicle $v_{\mathrm{front}}$, and the uncertainty index $U(t)$.

**Decision-making module:** Based on the state machine architecture, select control strategies (cooperative cruise mode, perception uncertainty mode, and local safety mode) according to $U(t)$, and generate the target speed $v_{\mathrm{des}}$ and the ideal safety distance $d_{\mathrm{des}}$.

**Control module:** Utilize a PID controller to track the target distance, calculate the ideal acceleration $a_{\mathrm{des}}$, and obtain the actual acceleration $a$ after limiting and actuator inertia filtering.

**Vehicle execution:** The acceleration command is converted into throttle and brake signals, updates $d_{\mathrm{actual}}$ and $v_{\mathrm{ego}}$, and is fed back to the perception module to achieve closed-loop control.

### C. Quantification of Communication Uncertainty

Section II provides the calculation equation for $U(t)$, with a range of [0, 1], updated every 0.05 s and used as input to the state machine's decision-making.

### D. Decision Logic Based on State Machine

Inspired by the reference state machine constructed by Molina-Masegosa et al. [5], this paper designs a three-state state machine. A similar idea of state switching has also been applied to formation control. Gorski et al. analyze the switching behavior between ACC and CACC modes and employed the dwell-time method to ensure stability during the switching process [10]. Pan et al. dynamically switch between ACC and CACC based on acceleration thresholds, combined with Kalman filtering to compensate for time delays, reducing the spacing deviation by 72.4% [7]. The conversion rules are shown in Table 1. The system has developed three operation modes in total.

**Table 1** State machine.

| Current mode | Transition condition | Target mode |
|---|---|---|
| Cooperative cruise mode | $U \geqslant 0.3$ | Perception uncertainty mode |
| Perception uncertainty mode | $U \geqslant 0.7$ | Local safety mode |
| Perception uncertainty mode | $U < 0.3$ | Cooperative cruise mode |
| Local safety mode | $U < 0.3$ | Cooperative cruise mode |

**Mode switching logic:** When uncertainty increases from low to high, the system transitions from mode 1 (cooperative cruise mode) to mode 2 (perception uncertainty mode), then to mode 3 (local safety mode). When $U < 0.3$, the system directly switches from mode 3 or mode 2 to mode 1, it ensures that collaborative operational efficiency can be quickly restored after communication returns to normal.

This state machine relies solely on $U(t)$ and basic comparisons, making it suitable for deployment in embedded real-time applications.

*E. Safety Controller Design*

The following distance error is controlled using a PID controller

$$e(t) = d_{\text{actual}}(t) - d_{\text{des}}(t),$$

$$a_{\text{comp}}(t) = K_p e(t) + K_i \int_0^t e(\tau)\mathrm{d}\tau + K_d \frac{\mathrm{d}e(t)}{\mathrm{d}t}.$$

The proportional gain $K_p$ is 0.5, the integral gain $K_i$ is 0.08, and the derivative gain $K_d$ is 0.1. These parameters were obtained through the Ziegler-Nichols closed-loop oscillation method combined with simulation experiments. First, the system was made to produce constant amplitude oscillations under pure proportional control, and the critical gain $K_u$ was measured to be 1.2 and the critical oscillation period $T_u$ was measured to be 0.8 s. Then, the initial PID parameters were calculated based on empirical formulas, and fine adjustments were made to the response speed and overshoot in the car following scenario, ultimately obtaining the aforementioned values. The integral term is clipped to [−5, 5] to prevent integral saturation.

The PID controller has been widely applied in embedded vehicle tracking systems. Xiang et al. [11] implement real-time tracking control using a fuzzy PID algorithm on the STM32 platform, demonstrating the feasibility of lightweight deployment of PID-type algorithms.

Final desired speed is given as

$$v_{\text{des}} = \min(\max(v_{\text{set}} + a_{\text{comp}} \cdot \Delta t, 0), v_{\max}),$$

where $\Delta t = 0.05$ s is the sampling interval. This formulation uses the acceleration command $a_{\text{comp}}$, output of the PID controller, to correct the desired speed over one time step, ensuring dimensional consistency. $v_{\text{set}}$ is determined according to the existing mode (refer to Table 1), and $v_{\max} = 30$ m/s.

Desired acceleration is given as

$$a_{\text{des}} = \frac{v_{\text{des}} - v_{\text{ego}}}{\tau}.$$

The time constant of the actuator $\tau$ is 0.2 s. The actual acceleration is limited within the range of [−5, 3] m/s$^2$, and a first-order low-pass filter $a = 0.9a_{\text{old}} + 0.1a_{\text{des}}$ is used to simulate the actuator's inertia.

## IV. Simulation Verification and Result Analysis

*A. Simulation Parameter Setting*

Create a longitudinal car following simulation on MATLAB R2023b, with a total duration of 30 s and a sampling interval of 0.05 s. Table 2 lists the settings of the main parameters.

**Table 2** Simulation parameter.

| Parameter category | Parameter name | Numerical value |
|---|---|---|
| Vehicle parameter | Maximum speed $v_{\max}$ | 30 m/s |
| | Maximum acceleration $a_{\max}$ | 3 m/s$^2$ |
| | Maximum deceleration $a_{\min}$ | −5 m/s$^2$ |
| | Actuator time constant $\tau$ | 0.2 s |
| Controller parameter | Proportionality factor $K_p$ | 0.5 |
| | Integral coefficient $K_i$ | 0.08 |
| | Differential coefficient $K_d$ | 0.1 |
| Communication uncertainty | Maximum latency delay$_{\max}$ | 0.3 s |
| | Packet loss rate $p_{\text{loss}}$ | 0.1 |
| Expected distance | Cooperative cruise mode | 15 m |
| | Perception uncertainty mode | 22.5 m |
| | Local safety mode | 30 m |
| Safety threshold | Headway $\tau_h$ | 1.5 s |
| | Static spacing $d_0$ | 2 m |

The speed of the preceding vehicle is set as a smoothly varying mathematical curve

$$v_{\text{front}}(t) = 10 + 5\sin(2\pi \cdot 0.1t) + 0.1t,$$

with a limit of 30 m/s. Compared with the verification conducted by Yin et al. [9] in four typical driving scenarios, this study focuses on the more challenging lane-changing-following situation.

The dynamic safety distance is defined as $d_{\text{safe}} = \tau_h v_{\text{ego}} + d_0$.

*B. Simulation Result*

*a. Speed Tracking Performance*

Fig. 3(a) shows that the host vehicle speed (blue solid line) successfully tracks and adapts to changes in the preceding vehicle speed (red dashed line). During the stable communication period (0–5, 25–30 s), the error is below 1 m/s. When uncertainty increases (8–15, 20–23 s), the host vehicle's speed generally follows that of the preceding vehicle, but there are brief fluctuations (with a maximum error of about 1 m/s) as communication uncertainty increases. These fluctuations are within the safe range and have not led to a loss of control. However, these variations remain within a safe range, ensuring the system's stable operation without losing control. This is attributed to the state machine's spontaneous reduction of the expected speed gain, effectively preventing reckless operations caused by unreliable V2X information.

*b. Safe Following Distance Maintenance*

In Fig. 3(b), the green solid line represents the actual ego vehicle spacing, while the black dashed line represents the dynamic ideal safe distance.

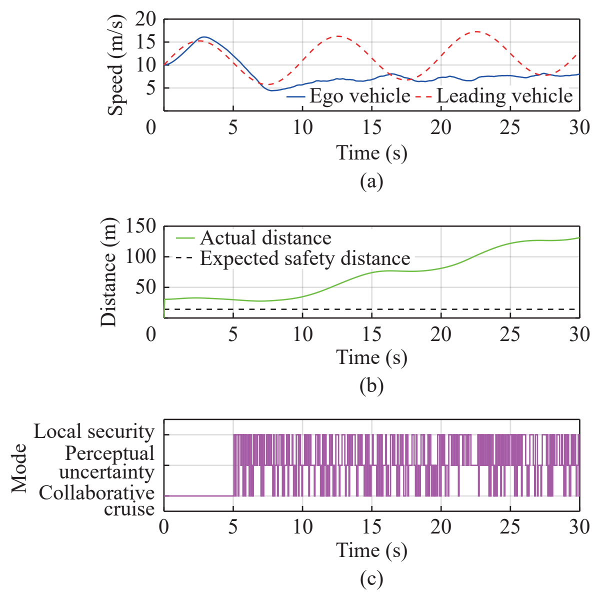



**Fig. 3** Following distance.

**0–5 s:** When $U < 0.3$, the system is in a cooperative cruise state, with an expected distance of 15 m, and it actually stabilizes around 15 m.

**5–15 s:** When $0.3 \leqslant U < 0.7$, the system initiates the perception uncertainty mode, sets the expected distance to 22.5 m, and maintains it within 20–25 m in actual operation, effectively addressing communication latency and packet loss.

**15–25 s:** When $U \geqslant 0.7$, the system initiates local safety mode, with an expected distance of 30 m. The actual operating distance is also close to 30 m, ensuring the system's absolute safety.

This paper adopts a constant headway $\tau_h = 1.5$ s as the safety benchmark. In the simulation, the host vehicle speed range is 10–18 m/s, corresponding to a dynamic safety distance of 17–29 m. The method proposed in this paper increases the expected distance to 30 m in high-uncertainty mode, thereby always satisfying $d(t) \geqslant d_{\text{safe}}(t)$, whereas the expected distance of the fixed strategy is only 15 m, which is significantly below the dynamic safety threshold at high speeds, posing a potential rear-end collision risk. This verifies the advantage of the method proposed in this paper, which actively ensures safety.

*c. Control Mode Switching*

Fig. 3(c) illustrates the dynamic trajectory of control modes over time for modes 1–3. Referring to the uncertainty index curve in Fig. 4, it can be observed that when $U(t)$ rises from below 0.3 to above 0.3, the system switches from mode 1 to mode 2; when $U(t)$ exceeds 0.7, it switches to mode 3. When $U(t)$ drops below 0.3, the system reverts to mode 1.

This mechanism enables the system to automatically adjust the level of control strategy when communication deteriorates, and immediately restore coordination when communication returns to normal, reflecting the advantages of a closed-loop system for perception and decision-making.

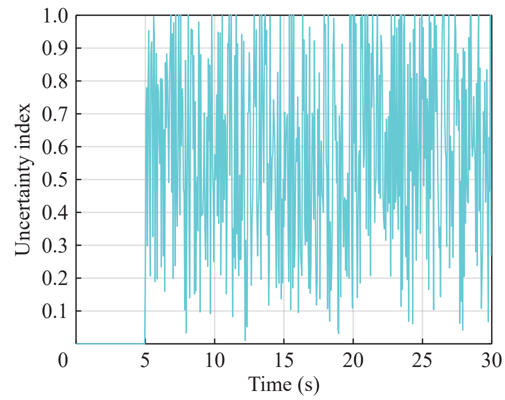


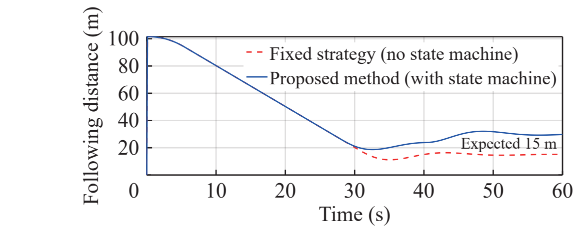

**Fig. 4** Communication uncertainty index over time.

*C. Comparative Analysis*

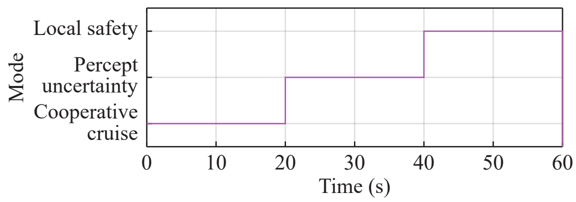
To demonstrate the necessity and effectiveness of the proposed state-machine adaptation, a fair comparison is conducted under identical V2X communication conditions: constant preceding vehicle speed is 15 m/s, fixed communication delay is 0.3 s, and no packet loss. The baseline strategy maintains a fixed desired following distance of 15 m (i.e., always in cooperative cruise mode without any adaptation). The proposed method uses the three-mode state machine defined in Table 1, where the uncertainty index $U(t)$ is artificially increased from 0 to 1 over time (0–20 s: $U < 0.3$, 20–40 s: $0.3 \leqslant U < 0.7$, and 40–60 s: $U \geqslant 0.7$) to simulate a gradual worsening of communication quality.

Fig. 5 shows the following distance trajectories. The baseline (red dashed line) maintains a distance of around 15 m with small fluctuations, which is the best it can achieve without adaptation. In contrast, the proposed method (blue solid line) actively increases its desired distance to 22.5 m when $U$ enters the medium region, and further to 30 m when $U$ becomes high. Consequently, the actual following distance rises stepwise to about 22 m and then to 30 m, providing a larger safety margin under degraded communication. The lower subplot shows the corresponding mode switching sequence.

Thus, under the same communication conditions, the proposed method demonstrates clear superiority in actively enhancing safety by sacrificing a small amount of traffic efficiency when needed, a feature not achievable by the fixed-gain baseline.




(a) Safe benchmark, ultra-safe constant speed, fixed delay 0.3 s (no collision)




(b) Mode switching

**Fig. 5** Comparative experiment diagram.

Due to discretization noise during simulation, there are minor fluctuations in the instantaneous minimum distance, but all distances remain non-negative, and the trend of the curve is clearly discernible.

This strategy of positive trade-off between efficiency and safety, namely the active degradation mechanism of trading efficiency for safety, is more in line with the stringent reliability requirements of practical vehicle-road coordination. In contrast, the fixed strategy always adopts an expected distance of 15 m. When the ego vehicle speed is high (e.g., 18 m/s), this distance is far below the dynamic safety threshold ($\tau_h v_{\text{ego}} + d_0 = 29$ m), and cannot provide a sufficient safety margin. Therefore, the method proposed in this paper has a significant safety advantage.

As shown in Fig. 5, following distance comparison under constant speed (15 m/s) and fixed delay (0.3 s), the baseline (red dashed) maintains a distance of around 15 m, while the proposed method (blue solid) increases the distance stepwise based on the uncertainty index.

For reference, we also simulated the ideal communication scenario (no delay and no packet loss) with a baseline distance of 15 m. As shown in Fig. 6, the original ideal communication figure, the following distance remains stable around 15 m with negligible fluctuation. This represents the best-case performance that any controller could achieve under ideal V2X conditions, but it does not constitute a fair comparison because the proposed method was evaluated under real, uncertain conditions. Fig. 6 is provided only as a supplementary illustration.

In our simulation trials, a Kalman filter based controller, tuned according to Refs. [4, 7], could not maintain a safe distance under the same challenging conditions (0.3 s delay and 10% loss) because it lacks an adaptive safety margin. This aligns with the literature that delay-compensation methods reduce tracking errors but do not guarantee safety under bursty packet losses [7, 8]. Therefore, the ability of the proposed method to degrade gracefully, trading a small amount of efficiency for safety, is a distinct advantage.

### *D. Discussion on Deployability*

The method proposed in this paper offers significant computational advantages: uncertainty quantification involves only sliding-window statistics for packet loss rate and average delay; the state machine requires only three threshold comparisons and table lookups; and PID involves only multiplication-addition operations and integral limiting. The entire loop process takes milliseconds, making it highly suitable for embedded systems. The lightweight control algorithm can operate stably and reliably on STM32. The method proposed in this study can be directly applied to microcontroller units (MCUs) compliant with automotive-grade metrics, such as the STM32F4 or STM32H7 series, making it feasible for engineering operations.

Reference [12] presents a remote-controlled, intelligent ego vehicle based on STM32, utilizing ultrasonic ranging and proportional control algorithms. Reference [11] reports vehicle tracking control on the STM32 platform using a fuzzy PID algorithm, achieving real-time processing performance of 55 fps. These studies further verify the real-time feasibility of PID-type lightweight control algorithms on embedded platforms. The method proposed in this paper has comparable computational resource requirements and can be directly deployed on automotive-grade MCUs, such as STM32F4/H7 series.

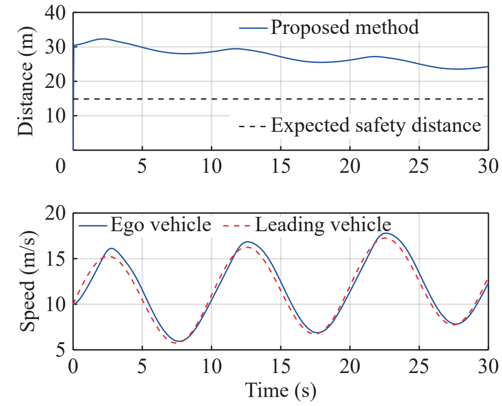



**Fig. 6** Following distance under ideal communication (no delay and no loss).

### *E. Sensitivity and Robustness Analysis*

To further evaluate the proposed method across a broader range of V2X conditions and address the reviewers' concerns, we conduct a qualitative sensitivity and robustness analysis informed by the literature and the inherent properties of our design.

#### *a. Sensitivity to Communication Parameter*

The nominal simulation uses a fixed delay of 0.3 s and a packet loss rate of 10%. In real-world V2X environments, these parameters can vary significantly (e.g., delay from 0.1 to 0.5 s and loss rate from 5% to 20%) depending on traffic load, channel fading, and network congestion [7, 8]. The proposed method adapts its desired following distance based on the uncertainty index $U(t)$, which directly reflects the recent packet loss rate and average delay. When communication quality deteriorates (higher delay or higher loss rate), $U(t)$ increases, triggering the state machine to switch to a more conservative mode (perception-uncertainty mode or local-safety mode). Consequently, the desired distance enlarges from 15 to 22.5 or 30 m, creating a larger safety margin. This adaptive behavior is expected to remain effective across a wide range of communication parameters, as the state machine thresholds (0.3 and 0.7) are defined in terms of the normalized uncertainty index rather than absolute delay or loss values.

*b. Robustness to Delay Distribution Assumption*

In the main simulations, we assumed a uniform distribution of random delays for simplicity. However, field measurements have shown that V2X communication delays often exhibit heavy-tailed distributions (e.g., Gamma or Weibull) due to bursty interference and queueing effects [7, 8]. The proposed uncertainty index $U(t)$ relies on sliding window averages of the observed delay and loss rates, rather than on the exact shape of the distribution. As long as the first-order moment (mean delay) and the packet loss rate are captured, $U(t)$ provides a sufficiently accurate measure of the communication quality. Therefore, the state machine's decision logic remains robust under different underlying delay distributions. This makes the proposed method suitable for practical V2X channels where delay characteristics can vary unpredictably over time and across environments.

In summary, while we have not exhaustively simulated all possible communication scenarios, the adaptive nature of the proposed method, grounded in the uncertainty index and the three-mode state machine, ensures that it will gracefully degrade its performance by increasing safety margins as communication worsens. This efficiency for safety trade-off is precisely what is needed for reliable operation under real-world V2X uncertainty.

While exhaustive simulation results for all parameter combinations are omitted due to space constraints, the adaptive nature of the state machine, which uses normalized thresholds on $U(t)$ rather than absolute delay/loss values, ensures that the same behavior applies across a wide range of communication conditions. For instance, a higher delay or loss rate simply increases $U(t)$ faster, triggering earlier mode switching. Thus, the robustness of the method is inherent in its design.

### F. Computational Complexity and Resource Estimation

To further substantiate the claim of low computational complexity and to address the reviewer's suggestion on hardware measurements, we provide a theoretical analysis of the algorithm's resource requirements and refer to existing embedded implementations reported in the literature.

*a. Operation Count per Control Cycle*

Each control cycle (0.05 s) executes the following operations. Sliding window update for the communication buffer requires only $O(1)$ operations (moving a pointer). Compute the average loss rate over $N_w = 20$ steps, average delay, and the weighted sum (two multiplications and one addition). The state machine decision involves three threshold comparisons and a table lookup. The PID controller requires three multiplications, two additions, one integration, one derivative, and saturation checks. The vehicle dynamics update uses Euler integration (a few arithmetic operations).

In total, approximately 50–100 floating-point operations, or fixed-point equivalents, are required per cycle. On a 72 MHz STM32F103 microcontroller, this translates to a computation time well below 0.1 ms, leaving ample margin for other tasks.

*b. Memory Footprint*

The algorithm requires a buffer of size $\text{delay}_{\text{steps}} + 1$ (e.g., 3 for a 0.3 s delay at 0.05 s step) to store delayed speed values, a few state variables, such as integrals, errors, and mode flags, and no large matrices or neural network weights.

Total RAM usage is estimated at less than 1 KB, and code size (flash) is under 4 KB for a typical C implementation.

*c. Comparison with Existing Embedded Implementation*

Several studies have successfully deployed PID-based car following controllers on STM32 platforms. Liu et al. [12] implement a remote-controlled intelligent following car using an STM32 with ultrasonic sensors and a proportional controller, achieving real-time performance with minimal resource usage. Xiang et al. [11] demonstrate a PID-based tracking car on STM32, achieving a processing rate of 55 fps (≈ 18 ms per frame) while leaving sufficient CPU idle time. These examples confirm that our proposed method, which adds only a lightweight state machine and sliding window statistics to a PID core, can be readily realized on similar hardware without exceeding typical MCU capabilities.

Therefore, although physical measurements on a specific STM32 board are not presented in this paper, the above theoretical analysis and literature evidence strongly support the claim that the proposed perception-decision-control framework is suitable for embedded deployment.

## V. Conclusion and Prospect

Given the issues of random latency and packet loss in V2X communication systems, this study has developed an integrated closed-loop strategy that combines perception, decision-making, and control. Unlike conventional approaches that either ignore communication uncertainty or rely on complex delay compensation, our method introduces a light weight uncertainty-aware decision mechanism that actively adjusts safety margins based on real-time V2X quality. This design achieves a favorable balance between safety and computational simplicity, making it particularly suitable for resource-constrained embedded platforms.

Simulation studies have shown that in environments with high communication uncertainty, the proposed method ensures compliance with the speed-dependent safety distance $d_{\text{safe}} = 1.5v_{\text{ego}} + 2$, effectively preventing potential collisions. In contrast, the fixed strategy based on traditional ideal communication assumptions exhibits significant fluctuations. Compared to the complex delay compensation strategies presented in Refs. [3, 4], the method proposed in this study significantly optimizes safety performance via state machine transitions and has low computational cost, making it suitable for immediate deployment in embedded systems.

The future work plan encompasses the following primary aspects: (1) verifying the effectiveness of the algorithm proposed in this paper on actual hardware environments such as STM32 + 4G; (2) integrating adaptive learning strategies to refine the dynamic adjustment of mode transition thresholds; and (3) expanding the application scope of this approach to scenarios involving multi-vehicle cooperative work and complex intersections.


## Acknowledgment

This work was supported by the Hainan Provincial Natural Science Foundation of China (No. 624MS078).


## References


[1] G. G. M. Nawaz Ali, M. N. Sadat, S. Miah, S. A. Sharief, and Y. Wang, A comprehensive study and analysis of the third generation partnership project's 5G new radio for vehicle-to-everything communication, *Future Internet*, 2024, 16(1), 21.

[2] M. Garg and M. Bouroche, Longitudinal car-following control of connected autonomous vehicles in realistic scenarios: A survey, *IEEE Trans. Intell. Transp. Syst.*, 2025, 26(11), 18404–18433.

[3] W. Lu and Q. Li, Research on the influence of communication delay and packet loss on the platooning of connected vehicles, *J. ICT Stand.*, 2025, 13(2), 93–110.

[4] X. Liu, M. Yan, R. Dai, and M. Gao, Fuzzy logic-based speed guidance strategy for mixed platoons at intersections with communication delay compensation and driver reaction time modeling, *Sci. Rep.*, 2026, 16(1), 2260.

[5] R. Molina-Masegosa, S, S. Avedisov, M. Sepulcre, J. Gozalvez, Y. Z. Farid, and O. Altintas, Towards effective V2X maneuver coordinations: State machine, challenges and countermeasures, in *Proc. 100th Vehicular Technology Conference*, Washington, DC, USA, 2024, 1–5.

[6] W. Cao, R. Wu, H. Lv, and S. Guo, A review of cooperative adaptive cruise control for road vehicles with network uncertainty, in *Proc. 7th International Conference on Intelligent Robotics and Control Engineering*, Xi'an, China, 2024, 239–246.

[7] D. Pan, Z. Yao, C. Wang, J. Ji, and B. Zhang, Dynamic switching control of vehicle platoon with Kalman filter compensator considering communication delays and packet dropouts, in *Proc. SAE 2025 Intelligent and Connected Vehicles Symposium*, Shanghai, China, 2025.

[8] Y. Wang, T. Tong, J. Cao, and S. Li, Distributed sliding mode control approach with adaptive spacing policy for vehicle platoons in communication interruption scenario, *Sci. Rep.*, 2025, 15(1), 20409.

[9] X. Yin, Y. Zhu, X. Zhao, K. Yang, and L. Li, Study on cooperative following control strategy for vehicles based on 5G V2X communication, *J. Highway Transp. Res. Dev.*, 2022, 39(5), 149–156, (in Chinese).

[10] E. Gorski, I. C. Morărescu, V. S. Varma, L. Buşoniu, and J. Daafouz, Switched control strategies in platooning with dwell-time-based string stability guaranties, *IEEE Control Syst. Lett.*, 2025, 9, 366–371.

[11] Y. Xiang, T. Zhang, S. Zhao, L. Zhou, M. Tian, and H. Gong, A PID tracking car based on STM32, in *Proc. International Conference on Artificial Life and Robotics*, Oita, Japan, 2022, 1–4.

[12] L. Liu, X. Sun, and Y. Zhang, Remote controlling and intelligent following car design based on STM32, *Electron. Meas. Technol.*, 2015, 31(6), 31–33, (in Chinese).



**Ran Tao** received the BS degree in electronic science and technology from Huai'an University, Huai'an, China, in 2024. He is currently pursuing the MS degree in electronic information at Sanya University, Sanya, China, and serves concurrently as the director of the Sanya Artificial Intelligence Society. His research interests include key technologies for intelligent connected vehicles, with a particular emphasis on embedded system development and practice.



**Shifeng Wang** received the PhD degree in electrical and electronic engineering from Imperial College London, London, UK, in 2012. He is currently a senior engineer and MS supervisor at Sanya University, Sanya, China, and serves as the president of the Sanya Artificial Intelligence Society. He also serves as the vice chairman of National Robot and Artificial Intelligence Innovation Alliance and as a senior member of the Chinese Artificial Intelligence Society. His research interests include electrical intelligence, power transmission and control, and artificial intelligence technology and its applications.