

Article

Event-Triggered and Adaptive ADMM-Based Distributed Model Predictive Control for Vehicle Platoon

Hanzhe Zou ¹, Hongtao Ye ^{1,2,*} , Wenguang Luo ¹ , Xiaohua Zhou ¹ and Jiayan Wen ² 

¹ School of Automation, Guangxi University of Science and Technology, Liuzhou 545036, China; 18913592507@163.com (H.Z.); wgluo@gxust.edu.cn (W.L.); zhxh76@126.com (X.Z.)

² Guangxi Key Laboratory of Automobile Components and Vehicle Technology, Guangxi University of Science and Technology, Liuzhou 545036, China; wenjiayan2012@126.com

* Correspondence: yehongtao@126.com; Tel.: +86-151-7774-1648

Abstract

This paper proposes a distributed model predictive control (DMPC) framework integrating an event-triggered mechanism and an adaptive alternating direction method of multipliers (ADMM) to address the challenges of constrained computational resources and stringent real-time requirements in distributed vehicle platoon control systems. Firstly, the longitudinal dynamic model and communication topology of the vehicle platoon are established. Secondly, under the DMPC framework, a controller integrating residual-based adaptive ADMM and an event-triggered mechanism is designed. The adaptive ADMM dynamically adjusts the penalty parameter by leveraging residual information, which significantly accelerates the solving of the quadratic programming (QP) subproblems of DMPC and ensures the real-time performance of the control system. In order to reduce unnecessary solver invocations, the event-triggered mechanism is employed. Finally, numerical simulations verify that the proposed control strategy significantly reduces both the computation time per optimization and the cumulative optimization instances throughout the process. The proposed approach effectively alleviates the computational burden on onboard resources and enhances the real-time performance of vehicle platoon control.

Keywords: vehicle platoon; distributed model predictive control; alternating direction method of multipliers; event-triggered mechanism



Academic Editor: Julian F Dunne

Received: 1 August 2025

Revised: 29 September 2025

Accepted: 1 October 2025

Published: 3 October 2025

Citation: Zou, H.; Ye, H.; Luo, W.; Zhou, X.; Wen, J. Event-Triggered and Adaptive ADMM-Based Distributed Model Predictive Control for Vehicle Platoon. *Vehicles* **2025**, *7*, 115. <https://doi.org/10.3390/vehicles7040115>

Correction Statement: This article has been republished with a minor change. The change does not affect the scientific content of the article and further details are available within the backmatter of the website version of this article.

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The increasing severity of traffic congestion and vehicle collisions in recent years has placed substantial pressure on modern transportation systems. To address these challenges, intelligent transportation systems (ITS) have been proposed, offering a unified framework that integrates autonomous driving technologies, vehicle-to-everything (V2X) wireless communications, and advanced computing into traffic management. This integration facilitates seamless interaction between vehicles-to-infrastructure (V2I) communication and vehicles-to-vehicles (V2V) communication, thereby enabling the formation of vehicle platoons that maintain desired inter-vehicle spacing and travel velocities.

Compared to traditional single-vehicle control strategies, the adoption of platoon-based control extends the optimization objective to the entire vehicle group. This holistic coordination allows for significantly reduced inter-vehicle gaps, leading to improved traffic throughput, enhanced longitudinal stability, and notable gains in overall energy efficiency for multi-vehicle systems [1–3].

Distributed model predictive control (DMPC) has emerged as a promising control strategy due to its capability to explicitly handle system constraints and address multi-objective optimization problems in a systematic manner. By leveraging future system predictions, DMPC enables comprehensive performance optimization over a receding horizon. As a result, it has been extensively investigated and applied in a wide range of domains, particularly in the cooperative control of vehicle platoons. Zheng et al. [4] established a four-element modeling framework for vehicle platoon control, comprising node dynamics, information flow topology, platoon geometric configuration, and distributed controllers. Concurrently, a distributed model predictive controller suitable for nonlinear heterogeneous platoons was designed. Zhang et al. [5] proposed a self-organizing cooperative control strategy, where merging vehicles form local platoons via a dynamic critical interval algorithm, and a DMPC framework coordinates their integration into the main formation. However, this approach still suffers from a high computational burden due to nonlinear modeling and DMPC optimization, relies on idealized assumptions such as homogeneity and perfect communication, and lacks sufficient validation under realistic conditions. Similarly, Hu et al. [6] proposed a distributed economic MPC strategy with switched feedback gains to improve fuel efficiency in vehicle platoons, and validated its stability and performance through theoretical analysis and simulations. However, the proposed EMPC strategy exhibits limitations in computational efficiency for real-time implementation, relies on idealized communication assumptions, and lacks comprehensive experimental validation under realistic disturbances and heterogeneous vehicle dynamics. Qiang et al. [7] proposed a DMPC scheme for heterogeneous vehicle platoons with inter-vehicle distance constraints. By adopting a parallel optimization framework with alternating updates between even- and odd-indexed vehicles and a tailored terminal inequality constraint, their approach overcomes key limitations of conventional methods, such as infeasibility in the absence of leader information, difficulty in maintaining spacing constraints, and limited adaptability to varying leader speeds. Pi et al. [8] further developed a DMPC-based energy-efficient strategy for electric platoons and integrated braking force distribution with energy recovery, which improved stability, tracking accuracy, and energy efficiency. Cen et al. [9] proposed a DMPC scheme for nonlinear platoon systems that employs a fused state estimator to mitigate missing state information caused by communication delays. However, the proposed framework relies on a linearized vehicle model and lacks a rigorous theoretical analysis of closed-loop and string stability under the combined effects of nonlinear dynamics, communication delays, and state estimation. Xu et al. [10] tackled critical challenges in highway multi-platoon cooperative control, namely poor system architecture scalability, low control accuracy under non-ideal communication conditions, and inadequate leader vehicle control adaptability in mixed traffic environments. Zhao et al. [11] and Zhao et al. [12] presented systematic cloud-based model predictive control schemes with novel delay compensation mechanisms, rigorously guaranteeing both asymptotic and string stability for vehicle platoons under heterogeneous communication delays, thereby enhancing tracking accuracy, fuel efficiency, and practical applicability. Wen et al. [13] proposed a novel linear control law for connected and automated vehicle platoons that incorporates a vehicle speed prediction model to effectively mitigate the detrimental effects of time-varying communication delays, thereby enhancing platoon stability and safety. Their solution comprises a system architecture combining static management with dynamic switching capability, alongside a delay-compensated DMPC method and a robust MPC strategy for the leader vehicle under disturbances from human-driven vehicles. While the aforementioned studies successfully employed DMPC to achieve effective control of vehicle platoon systems in their respective scenarios, the computational burden associated with solving the DMPC optimization problem becomes significant when considering

complex vehicle dynamics characterized by coupled longitudinal-lateral nonlinearities. Consequently, there is a pressing need to investigate computationally efficient solution algorithms to ensure the real-time performance of cooperative platoon control.

The implementation of DMPC requires executing model predictions and solving receding horizon optimization problems at every sampling instant, which imposes a substantial computational burden on vehicle platoon systems. To address this issue, increasing attention has recently been directed toward event-triggered control mechanisms, which aim to reduce unnecessary computations by updating control actions only when certain state-dependent conditions are met. This paradigm offers the potential to alleviate resource consumption while maintaining satisfactory control performance. Zhao et al. [14] proposed an adaptive event-triggered safety control approach that dynamically adjusts the event-triggering threshold based on abrupt changes in the preceding vehicle's behavior to optimize communication bandwidth utilization. By integrating an adaptive cost function and a controller gain optimization algorithm, the method ensures the strict string stability of vehicle platoons under limited bandwidth conditions, thereby enhancing both safety and control performance of the platoon. Luo et al. [15] proposed an event-triggered tube-based DMPC with a shrinking horizon and terminal region to ensure robustness under disturbances and delays. The triggering rule based on control plan variations reduces communication, but may not activate under large errors, posing safety risks. Han [16] proposed a bandwidth-aware scheduling strategy integrated with event-triggered DMPC to alleviate communication and computational load by dynamically adjusting triggering thresholds and optimization frequency. However, the proposed mechanisms lack practical implementation details for bandwidth state estimation, exhibit high parameter dependency, and insufficiently address robustness concerns under communication delays, packet losses, and potential event storms. Similarly, Selvaraj et al. [17] developed a memory-based event-triggered platooning control scheme that utilizes only position measurements and dynamic thresholds to improve communication efficiency while guaranteeing closed-loop stability through Lyapunov-based linear matrix inequality (LMI) formulations. Chen et al. [18] proposed an asynchronous self-triggered stochastic distributed MPC scheme that ensures probabilistic constraint satisfaction and quadratic stability for vehicle platoons under stochastic disturbances, while substantially reducing communication load in vehicular ad hoc networks. Du et al. [19] proposed a hierarchical dynamic event-triggered control protocol that guarantees zero-error tracking for heterogeneous vehicle platoons with actuator uncertainties and a nonzero-input leader, substantially reducing communication costs. However, the analysis and design are conducted under an ideal communication network assumption, leaving the system's stability and performance vulnerable to practical network-induced imperfections such as communication delays and packet dropouts. Wang et al. [20] proposed a novel periodic event-triggered fault detection filter for vehicle platoons, which effectively saves communication resources while ensuring asymptotic stability and H_∞ performance of the residual system under actuator faults and external disturbances. Although event-triggered mechanisms can effectively alleviate computational and communication burdens, their effectiveness may deteriorate in certain application scenarios.

A core advantage of the alternating direction method of multipliers (ADMM) in solving QP problems lies in its powerful decomposition capability. The algorithm decomposes complex problems into computationally tractable subproblems, allowing for parallel processing in large-scale applications. It is particularly effective at decoupling and handling complex constraints while maintaining high memory efficiency and ease of parallelization. Consequently, ADMM serves as a powerful framework for large-scale or structured quadratic programming. Li [21] proposed a heuristic adaptive ADMM approach for solving

MPC problems, effectively enhancing the convergence rate of the algorithm. However, the adaptive parameters still require manual tuning, and improper settings may increase the convergence time. Tang [22] extended Li's heuristic method to a multi-UAV simulation platform, demonstrating its potential and effectiveness for multi-agent systems and providing a novel approach for multi-agent coordinated control. Mallick et al. [23] developed a DMPC strategy based on a switching ADMM framework for piecewise affine (PWA) systems. This approach addresses the inherent nonconvex optimization problem by decomposing it into a sequence of convex subproblems. It concurrently coordinates the consensus of coupled states among subsystems, significantly reducing the computational burden while simultaneously ensuring closed-loop stability and recursive feasibility. Feng [24] addressed longitudinal-lateral coupling and tire nonlinearities using Koopman operator theory. They employed the ADMM to solve the resulting QP problem, achieving longitudinal tracking for vehicle platoons under complex driving conditions. Concurrently, Bai et al. [25] proposed a parallel DMPC framework based on ADMM, which effectively addresses the optimization problem with coupled safety constraints in cooperative control for connected automated vehicle (CAV) platooning, achieving both computational efficiency and robust safety guarantees. However, the framework lacks a thorough analysis of its convergence rate and computational scalability for large-scale platoons under real-world communication delays and dynamic topologies. However, within the aforementioned studies utilizing ADMM, investigations into its convergence properties remain limited. Furthermore, the design of adaptive parameters is often heavily reliant on domain expertise, and suboptimal parameter selection may fail to achieve peak performance.

The main contributions are summarized as follows.

1. To overcome the excessive computation time associated with solving the core QP problem within traditional DMPC, which may result in delayed control actions and potential collisions, we design a residual-feedback-based adaptive ADMM algorithm. This algorithm dynamically adjusts the penalty parameter based on primal and dual residuals observed during the iterative process, thereby significantly enhancing computational efficiency within the control cycle.
2. To further reduce the computational load of the control algorithm and conserve valuable onboard computing resources, an event-triggered mechanism is introduced. This mechanism departs from conventional time-triggered, fixed-period computation paradigms. Instead, the DMPC optimization is initiated only when the system state satisfies predefined triggering conditions.

2. Problem Formulation

2.1. Longitudinal Dynamic System of a Single Vehicle

In order to capture the essential characteristics of longitudinal vehicle motion, it is necessary to establish a nonlinear vehicle dynamics model that considers traction force, aerodynamic drag, and rolling resistance. Such a model provides a more realistic description of the vehicle's motion compared with simplified kinematic representations and serves as the foundation for subsequent controller design. The nonlinear longitudinal dynamics of the vehicle i can be formulated as follows:

$$\begin{cases} \dot{p}_i(t) = v_i(t) \\ \frac{\eta_{T,i}}{r_{w,i}} T_i(t) = m_i \dot{v}_i(t) + C_{A,i}(t) v_i^2(t) + m_i g f_{r,i} \\ \tau T_i(t) + T_i(t) = T_{des,i}(t) \end{cases} \quad (1)$$

where $p_i(t)$ and $v_i(t)$ are the position and velocity of vehicle i ; $T_i(t)$ and $T_{des,i}(t)$ represent the actual and desired driving or braking torque, respectively; m_i is the vehicle mass; $r_{w,i}$ is

the tire radius; $C_{A,i}(t)$ is the lumped air resistance coefficient; $\eta_{T,i}$ and $f_{r,i}$ are the mechanical efficiency and rolling resistance coefficient, respectively; g is the gravity constant; τ is the time lag of the dynamic system.

To facilitate the analysis of vehicle dynamics, we adopt the feedback linearization technique, which has been widely employed in previous studies. The corresponding feedback strategy is expressed as follows:

$$T_{\text{des},i} = \frac{r_{w,i}}{\eta_{T,i}} (C_{A,i}v_i(2\tau v_i + v_i) + m_i g f_{r,i} + m_i u_i) \quad (2)$$

Then, the nonlinear vehicle dynamics model is transformed into a third-order state-space model by applying feedback linearization method. The longitudinal dynamic model of vehicle i , $i \in \{0, 1, \dots, N\}$ can be formulated as follows:

$$\begin{cases} \dot{p}_i = v_i(t) \\ \dot{v}_i = a_i(t) \\ \dot{a}_i = -\frac{1}{\tau_i} a_i(t) + \frac{1}{\tau_i} u_i(t) \end{cases} \quad (3)$$

where $p_i(t)$, $v_i(t)$ and $a_i(t)$ denote the position, velocity, and acceleration of the vehicle i , respectively, and $u_i(t)$ represents the control input for the vehicle i , which represents the desired acceleration. Defining the state vector for the vehicle i at time t as $x_i(t) = [p_i(t), v_i(t), a_i(t)]^T$. The third-order state-space equation can be obtained as follows:

$$\begin{cases} \dot{x}_i(t) = A_{i,c}x_i(t) + B_{i,c}u_i(t) \\ y_i(t) = C_{i,c}x_i(t) \end{cases} \quad (4)$$

$$\text{where } A_{i,c} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1/\tau_i \end{bmatrix}, B_{i,c} = \begin{bmatrix} 0 \\ 0 \\ 1/\tau_i \end{bmatrix}, C_{i,c} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Given that wireless communication inherently enables packet-based data transmission among vehicles, the use of sampled data is more suitable for system analysis and design. Therefore, a sampling period $T > 0$ is defined, and the discrete-time longitudinal dynamics of the vehicle are derived using the forward Euler method. Moreover, the state error model is formulated as follows:

$$\begin{cases} \tilde{x}_i(k+1) = A_{i,d}\tilde{x}_i(k) + B_{i,d}\tilde{u}_i(k) \\ \tilde{y}_i(k) = C_{i,d}\tilde{x}_i(k) \end{cases} \quad (5)$$

$$\text{where } \tilde{x}_i(k) = x_i(k) - x_i^{\text{ref}}(k), \tilde{u}_i(k) = u_i(k) - u_i^{\text{ref}}(k), A_{i,d} = TA_{i,c} + I = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & T \\ 0 & 0 & 1 - T/\tau_i \end{bmatrix},$$

$$B_{i,d} = TB_{i,c} = \begin{bmatrix} 0 \\ 0 \\ T/\tau_i \end{bmatrix}, C_{i,d} = C_{i,c} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The reference signal $x_i^{\text{ref}}(k)$ is directly determined by the adopted communication topology. Specifically, depending on the selected topology, it represents either the state of the leader vehicle or the state of the immediately preceding vehicle.

2.2. Communication Topology

In the cooperative control of vehicle platoons, the information flow topology is crucial for determining system stability and coordination performance. This study utilizes concepts from algebraic graph theory to characterize the information flow among vehicles. The

communication topology of the vehicle platoon is modeled as a directed graph $G = \{V, E\}$, where $V = \{1, 2, \dots, N\}$ denotes the set of vehicles and $E = V \times V$ represents the set of directed communication links. Different information flow topologies have a significant impact on the design of the local objective functions for individual vehicles. Several prevalent communication topologies are illustrated in Figure 1.

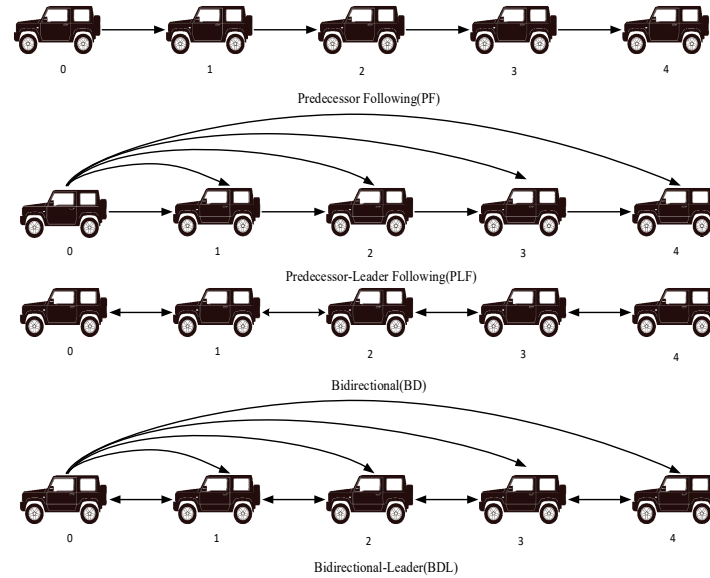


Figure 1. Common Topological Configurations in Vehicle Platooning.

Within the platoon, vehicle 0 denotes the leading vehicle, while vehicles 1 to 4 represent the following vehicles. In principle, the proposed event-triggered DMPC framework can be extended to above topologies with minor modifications in the coupling terms of the cost function and the constraint set.

2.3. Formation Objective

The primary objective of vehicle platooning control is to achieve consensus among vehicles. Consensus refers to the ability of follower vehicles to synchronize their velocity and acceleration with the leader. Meanwhile, each vehicle should maintain a desired inter-vehicle spacing with its adjacent vehicles. The formation objective of platoon control can be defined as follows:

$$\begin{cases} \lim_{t \rightarrow \infty} |p_{i-1}(t) - p_i(t) - d_{i-1,i}| = 0 \\ \lim_{t \rightarrow \infty} |v_i(t) - v_0(t)| = 0 \\ \lim_{t \rightarrow \infty} |a_i(t) - a_0(t)| = 0 \end{cases} \quad (6)$$

where $v_0(t)$ denotes the actual velocity of the leader vehicle. Within the proposed framework, the leader is not subject to control; rather, its velocity is manually specified and broadcast to the follower vehicles through the adopted communication topology. This predefined velocity acts as the external input that governs the platoon dynamics. $d_{i-1,i}$ denotes the desired distance between the vehicle i and its immediate predecessor. The choice of $d_{i-1,i}$ significantly influences the geometric configuration of the vehicle platoon. Common platoon geometries include constant spacing, constant time headway, and non-linear spacing policies. In this study, a constant spacing policy is adopted to enhance the traffic throughput of the platoon:

$$d_{i-1,i} = C \quad (7)$$

where C denotes a constant.

3. Adaptive ADMM-Based Event-Triggered Distributed Model Predictive Control

3.1. Prediction Model

The predictive model estimates future system states over a finite prediction horizon, utilizing the current state and historical data. For the convenience of controller design, we further construct the augmented variable $\xi_i(k) = [\tilde{x}_i(k), \tilde{u}_i(k-1)]$, $\eta_i(k) = \tilde{x}_i(k)$, Equation (5) can be obtained as follows:

$$\begin{cases} \xi_i(k+1) = A_{k,i}\xi_i(k) + B_{k,i}\tilde{u}_i(k) \\ \eta_i(k) = C_{k,i}\xi_i(k) \end{cases} \quad (8)$$

where $A_{k,i} = \begin{bmatrix} A_{i,d} & B_{i,d} \\ 0_{m \times n} & I_m \end{bmatrix}$, $B_{k,i} = \begin{bmatrix} B_{i,d} \\ I_m \end{bmatrix}$, $C_{k,i} = \begin{bmatrix} C_d & 0_n \end{bmatrix}$. Here, $m = 1$ and $n = 3$ denote the dimensions of the control input and the vector state vector, respectively. $\eta_i(k)$ directly reflects the tracking error with respect to the designated reference vehicle, since $\xi_i(k)$ contains the error state $\tilde{x}_i(k)$. Hence, minimizing $\eta_i(k)$ in the cost function is equivalent to minimizing the tracking error.

In the DMPC scheme, the prediction horizon and control horizon are typically denoted as N_p and N_c , respectively. Assuming that the state variable of vehicle i at time step k is represented by $\xi_i(k)$, the following prediction equation can be derived based on Equation (8):

$$Y_i(k) = \psi_i \xi_i(k) + \theta_i \Delta U_i(k) \quad (9)$$

$$\text{where } Y_i(k) = \begin{bmatrix} \eta_i(k+1|k) \\ \eta_i(k+2|k) \\ \vdots \\ \eta_i(k+N_p|k) \end{bmatrix}, \quad \psi_i = \begin{bmatrix} C_{k,i}A_{k,i} \\ \vdots \\ C_{k,i}A_{k,i}^{N_c} \\ \vdots \\ C_{k,i}A_{k,i}^{N_p} \end{bmatrix}, \quad \Delta U_i(k) = \begin{bmatrix} \tilde{u}_i(k+1|k) \\ \tilde{u}_i(k+2|k) \\ \vdots \\ \tilde{u}_i(k+N_c|k) \end{bmatrix},$$

$$\theta_i = \begin{bmatrix} C_{k,i}B_{k,i} & 0 & \dots & 0 \\ C_{k,i}A_{k,i}B_{k,i} & C_{k,i}B_{k,i} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C_{k,i}A_{k,i}^{N_p-1}B_{k,i} & C_{k,i}A_{k,i}^{N_p-2}B_{k,i} & \dots & C_{k,i}A_{k,i}^{N_p-N_c-1}B_{k,i} \end{bmatrix}.$$

3.2. Design of the Cost Function

Building upon the previously described vehicle platoon model, a cost function is defined for each vehicle by incorporating its current state as well as the states of neighboring and leading vehicles. To ensure feasible and stable control performance while avoiding infeasibility issues, a slack variable ε is introduced, which guarantees the existence of a feasible solution at each control step. Then, the objective function is designed as follows:

$$J_{i,k} = \sum_{k=1}^{N_p} \|\eta_i^p(k+j|k) - \eta_{i,des}(k+j|k)\|_{Q_i}^2 + \sum_{k=1}^{N_c} \|\tilde{u}(k+j|k)\|_{R_i}^2 + \sigma\varepsilon^2$$

s.t.

$$u_{\min} \leq \tilde{u}(k+j|k) \leq u_{\max}$$

$$\Delta \tilde{u}_{\min} \leq \Delta \tilde{u}(k+j|k) \leq \Delta \tilde{u}_{\max} \quad (10)$$

where Q_i represents the weight matrix associated with the tracking performance, R_i represents the weighting matrix penalizing the variation in control inputs, and $\eta_{i,des}(k+j|k)$ denotes the desired reference trajectory.

The formulated cost function can be transformed into a standard QP problem for optimization. However, traditional QP solvers such as interior-point methods and active-set methods often fail to satisfy the real-time computational requirements of large-scale, multi-controller systems like vehicle platoons. Therefore, a more efficient optimization approach is required to enable fast and accurate tracking of the leader vehicle by multiple follower vehicles.

3.3. Residual-Based Adaptive ADMM Algorithm

The ADMM is a robust optimization framework that decomposes centralized control problems into smaller subproblems, enabling parallel solutions. It is particularly effective for distributed convex optimization, progressively converging to a globally optimal or acceptable locally optimal solution through iterative updates. ADMM has demonstrated strong performance in solving large-scale constrained optimization problems. It is typically applied to problems formulated in the following standard form:

$$\begin{cases} \min_{x,z} f(x) + g(z) \\ s.t. Cx + Dz = b \end{cases} \quad (11)$$

where f and g represent convex functions, x and z are the variables to be optimized, and $Cx + Dz = b$ denotes the equality constraints to be satisfied.

By introducing the dual variable ω , the augmented Lagrange function corresponding to Equation (12) is constructed as follows:

$$L_\rho(z, v, \omega) = f(x) + g(z) + \omega^T(Cx + Dz - b) + \frac{\rho}{2} \|Cx + Dz - b\|_2^2 \quad (12)$$

In a typical ADMM framework, the iterative process involves successive updates of the primal variables x and z , the dual variable ω and the penalty parameter ρ . To reduce computational complexity and facilitate implementation, the dual variable is often replaced by its scaled form, $\mu = \omega/\rho$, leading to the following update rules.

$$\begin{cases} x^{k+1} = \underset{x}{\operatorname{argmin}} \left\{ f(x) + \frac{\rho}{2} \|Cx + Dz^k - b + \mu^k\|_2^2 \right\} \\ z^{k+1} = \underset{z}{\operatorname{argmin}} \left\{ g(z) + \frac{\rho}{2} \|Cx^{k+1} + Dz - b + \mu^k\|_2^2 \right\} \\ \mu^{k+1} = \mu^k + Cx^{k+1} + Dz^{k+1} - b \end{cases} \quad (13)$$

During the iterative process, the primal variables x and z are updated alternately, which contributes to improved computational efficiency and faster convergence.

Existing adaptive ADMM algorithms typically employ heuristic ratio-based rules to adjust the penalty parameter adaptively at each iteration, based on intermediate solutions. However, as a generalization of the standard ADMM, the performance of such ratio-adaptive ADMM methods is highly sensitive to the proper selection of tuning parameters, which are usually specified through prior calibration by expert users. Improper selection of these adaptive parameters by non-expert users may significantly degrade the convergence speed. To address this limitation, this section proposes a residual-ratio-based adaptive ADMM method, which dynamically adjusts the relative magnitudes of the primal and dual residuals. This enables automatic parameter tuning and accelerates convergence, without requiring any user-defined parameters.

In distributed optimization problems, the primal residual $r^k = \|Ax^k + Bz^k - c\|$ represents the violation of the consensus constraints, while the dual residual $s^k = \|\rho A^T(z^k - z^{k-1})\|$ reflects the convergence behavior of the dual variables. Traditional ADMM employs a fixed penalty parameter ρ , which may lead to an imbalance between these two types of residuals. Specifically, a large ρ tends to reduce the primal residual but significantly increases the dual residual, potentially causing oscillations when $s^k \gg r^k$. Conversely, a small ρ results in a large primal residual and a small dual residual. The exit condition of the algorithm depends on the primal and dual residuals. Throughout the convergence process, it is crucial to ensure that both residuals remain sufficiently small to accelerate convergence and improve overall efficiency.

The algorithm is considered to have converged when both the primal and dual residuals satisfy the following conditions:

$$\begin{cases} \|r^k\|_2 \leq \varepsilon^{prim} \\ \|s^k\|_2 \leq \varepsilon^{dual} \end{cases} \quad (14)$$

where $\varepsilon^{prim} = \sqrt{n}\varepsilon^{abs} + \varepsilon^{rel} \max\{\|Ax^k\|_2, \|z^k\|_2, \|b\|_2\}$, $\varepsilon^{dual} = \sqrt{n}\varepsilon^{abs} + \varepsilon^{rel} \rho \|A^T x^k\|_2$.

Accordingly, a dynamic balancing criterion is established as follows:

$$\frac{\|r^k\|}{\|s^k\|} \approx O(1) \quad (15)$$

To enhance algorithm robustness, the penalty parameter ρ is adaptively adjusted to maintain the primal and dual residuals within the same order of magnitude. Based on the dynamic balancing criterion, a normalized residual ratio is introduced. Specifically, the normalized residual ratio between the primal and dual residuals is defined, and the corresponding expression can be formulated as follows:

$$\tilde{r}^k = \frac{\|r^k\|}{\|r^0\|}, \tilde{s}^k = \frac{\|s^k\|}{\|s^0\|} \quad (16)$$

where r^0, s^0 denote the initial primal and dual residuals, respectively.

The adaptive penalty parameter update rule, based on the geometric mean principle, is formulated as follows:

$$\rho^{k+1} = \rho^k \cdot \sqrt{\frac{\tilde{r}^k}{\tilde{s}^k}} \quad (17)$$

The geometric mean principle ensures that the proposed adaptive rule maintains a balance between global search and local convergence. Specifically, when $\tilde{r}^k > \tilde{s}^k$, the penalty parameter ρ is increased to enforce constraint satisfaction, while it is decreased to mitigate oscillations when the dual residual dominates.

Meanwhile, to ensure the continuity of the Lagrange multiplier $\lambda^k = \rho^k \mu^k$, the dual variables are scaled synchronously as follows:

$$\mu^{k+1} = \mu^k \cdot \frac{\rho^k}{\rho^{k+1}} \quad (18)$$

The cost function described in Equation (9) is reformulated into a QP form, which is expressed as follows:

$$\begin{cases} \min J = \frac{1}{2}x^T Hx + f^T x \\ s.t. Ax \leq b \end{cases} \quad (19)$$

$$\text{where } H = 2 \begin{bmatrix} \theta_i^T Q \theta_i + R & 0 \\ 0 & \rho \end{bmatrix}, f = [2\psi_i \xi_i(k) Q_i \theta_i \quad 0], x = \begin{bmatrix} \Delta u \\ \varepsilon \end{bmatrix}, A = \begin{bmatrix} I & 0 \\ -I & 0 \\ \theta_i & 0 \\ -\theta_i & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} \Delta u_{i,\max} \\ -\Delta u_{i,\min} \\ Y_i(k) - \psi_i \xi_i(k) \\ Y_i(k) + \psi_i \xi_i(k) \end{bmatrix}.$$

Based on the derived standard QP formulation for the longitudinal tracking control problem of the vehicle platoon, dual variables are introduced to reformulate the problem into the canonical ADMM form. The resulting ADMM formulation is given as follows:

$$\begin{cases} \min_{x,z} J = f(x) + g(z) \\ \text{s.t. } Ax - b + z = 0 \end{cases} \quad (20)$$

where $f(x) = \frac{1}{2}x^T Hx + f^T x$, and $g(z)$ is the indicator function. $g(z)$ defined as

$$g(z) = \begin{cases} 0, & \text{if } Az \leq b \\ +\infty, & \text{otherwise} \end{cases}.$$

Based on the Lagrange multiplier method and the preceding ADMM formulation, the augmented Lagrange function is given as follows:

$$L_\rho(x, z, y) = f(x) + g(z) + y^T (Ax - b + z) + \frac{\rho}{2} \|Ax - b + z\|_2^2 \quad (21)$$

Based on Equation (11), the scaled dual variable $\mu = y/\rho$ is introduced. Then, by combining the first-order optimality conditions for the primal variable x and the auxiliary variable z , the iterative update rules are derived as follows:

$$\begin{cases} x^{k+1} = (H + \rho A^T A)^{-1} [-f - \rho A^T (z^k + \mu^k - b)] \\ z^{k+1} = \max\{0, -Ax^k - \mu^k + b\} \\ \mu^{k+1} = \mu^k + Ax^{k+1} - b + z^{k+1} \end{cases} \quad (22)$$

To enhance the convergence rate and computational efficiency of the ADMM algorithm, a relaxation parameter $\alpha \in (1, 2)$ is introduced. Additionally, the solution from the previous iteration is leveraged in the update process to accelerate subsequent iterations. The modified update equations are given as follows:

$$\begin{cases} x^{k+1} = (H + \rho A^T A)^{-1} [-f - \rho A^T (z^k + \mu^k - b)] \\ z^{k+1} = \max\{0, -\alpha(Ax^k - b) + (1 - \alpha)z^k - \mu^k\} \\ \mu^{k+1} = \mu^k + \alpha(Ax^{k+1} - b + z^{k+1}) + (1 - \alpha)(z^{k+1} - z^k) \end{cases} \quad (23)$$

The computational complexity of the proposed ADMM-MPC algorithm is dominated by the x -update, which involves solving a linear system. Let the decision variable dimension be n_x . Each iteration of ADMM requires $O(n_x^3)$ operations, and the total complexity is $O(Kn_x^3)$, where K is the number of iterations. Compared to classical active-set methods (ASM) and interior-point methods (IPM), ADMM exhibits a similar per-iteration cost but typically converges with fewer iterations in practice, particularly when warm-start strategies are employed.

Extending to the distributed case with M vehicles, each vehicle solves its local QP of dimension n at every iteration. The total computational complexity of the multi-vehicle system is therefore $O(KMn_x^3)$.

3.4. Design of Event-Triggered Mechanism

In the development of cooperative control systems for vehicle platoons, the traditional time-triggered mechanism was initially applicable in the early stages when in-vehicle networks were relatively underdeveloped. However, with the evolution of intelligent transportation systems towards large-scale and dynamic deployments, the limitations of time-triggered mechanisms have become more apparent. Specifically, their rigid resource utilization leads to unnecessary computation and communication even when the platoon is in a steady cruising state or the tracking error has converged within an acceptable tolerance, thereby resulting in wasted computational resources.

In contrast, the event-triggered mechanism effectively addresses the trade-off between limited computational resources and control timeliness by transitioning from a time-driven to a state-driven paradigm. It has demonstrated significant advantages in reducing resource consumption, lowering computation frequency, and maintaining system stability. Specifically, within the framework of DMPC, where the controller performs receding horizon optimization to generate the optimal control sequence, the triggering condition is defined as follows:

$$\Delta u_i(k) = \begin{cases} \Delta U_i(1, k), & \text{if case1} \\ \Delta U_i(k - k_t + 1, k_t), & \text{if case2} \end{cases} \quad (24)$$

$$\text{Case1: } \left| \psi_i^{\text{ref}}(k+1) - \psi_i(k+1|k) \right| \geq \omega_i \parallel k - k_t > N_c$$

$$\text{Case2: } \left| \psi_i^{\text{ref}}(k+1) - \psi_i(k+1|k) \right| \leq \omega_i$$

where $\Delta U_i(1, k)$ denotes the control input at the time step k , and k_t represents the most recent triggering instant. The position and velocity components of the output are selected to formulate the triggering condition:

$$\psi_i(k+1|k) = \begin{bmatrix} p_i(k+1|k) \\ v_i(k+1|k) \end{bmatrix}$$

$$\psi_i^{\text{ref}}(k+1) = \begin{bmatrix} p_i^{\text{ref}}(k+1) \\ v_i^{\text{ref}}(k+1) \end{bmatrix} \quad (25)$$

The parameter $\omega_i \in (0, 1)$ serves as a triggering threshold, which is used to regulate the triggering frequency. When $\omega_i = 0$, the event-triggered mechanism reduces to the conventional time-triggered scheme. As ω_i increases, the triggering frequency decreases accordingly.

Following the description of the core algorithmic components, the overall workflow of the proposed event-triggered adaptive ADMM-DMPC framework is illustrated in Figure 2.

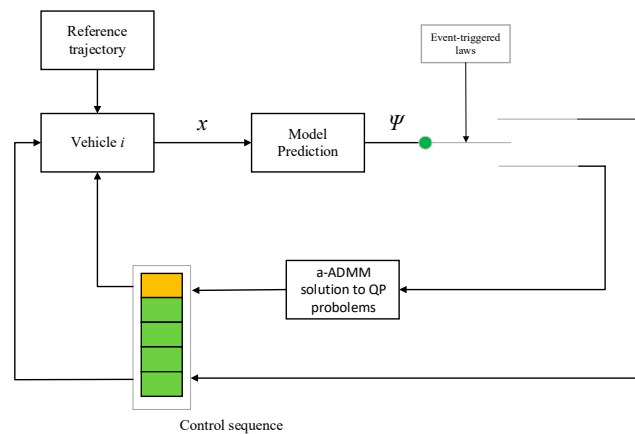


Figure 2. Framework of the Event-triggered and Adaptive ADMM-based DMPC for Vehicle Platoon.

The system accepts a reference trajectory as input, while the event-triggered mechanism determines when local optimization should be executed. Each vehicle formulates and solves its local QP using the a-ADMM algorithm in coordination with its neighbors. This workflow highlights the closed-loop integration from trajectory planning to distributed control execution.

To further clarify the real-time operation of the event-triggered DMPC scheme, the detailed algorithmic steps at each sampling instant are summarized as follows:

1. Check the event-triggering condition (24). If the triggering condition is satisfied, each vehicle initiates the distributed ADMM-based DMPC optimization; otherwise, the vehicles directly reuse the previously computed optimal control sequence and apply the control input corresponding to the current sampling instant, then proceed to step 8.
2. For each vehicle, formulate the local quadratic programming problem by expressing the objective function in the form of (10) according to its own state-space model and the coupling constraints imposed by the communication topology.
3. Reformulate (10) into the decomposed form suitable for ADMM optimization (cf. Equation (20)).
4. Initialize the ADMM solver with the optimal solutions obtained at time $d - 1$
5. Update the primal and dual variables iteratively according to the distributed ADMM updating rules, as given in (23).
6. Adaptively update the penalty parameter ρ based on the normalized residual ratio (16) and the update rule (17), so as to dynamically balance the primal and dual residuals and enhance convergence speed.
7. Check the termination criteria (14). If satisfied, terminate the iteration and transmit the control input corresponding to the current sampling instant from the optimized sequence of each vehicle to the platoon system; otherwise, continue the distributed iteration until the maximum number of iterations is reached.
8. Advance to the next sampling instant $d + 1$ and repeat step 1.

4. Asymptotic Stability

To establish the stability of the vehicle platoon, the cost function is commonly employed as a Lyapunov candidate. Nevertheless, under the event-triggered mechanism, the optimization problem is not solved at every discrete sampling instant, which prevents the monotonic decrease in the Lyapunov function from being directly ensured.

Theorem 1. *The following theorem provides conditions to ensure the Lyapunov stability of the system (8).*

Supposed that:

1. *there exists a stabilizing terminal feedback K such that $A_{cl} = A_{K,i} + B_{K,i}K$ is Schur;*
2. *there exists a positive definite matrix $P > 0$ satisfying the discrete-time Lyapunov inequality.*

$$A_{cl}^T P A_{cl} - P \leq -Q, Q > 0. \quad (26)$$

Then, for sufficiently large $\rho > 0$, the platoon is asymptotically stable.

Proof of Theorem 1. We define the collective Lyapunov function:

$$V(k) = \sum_{i=1}^N (\xi_i(k)^T P \xi_i(k) + \rho \|e_i(k)\|^2) \quad (27)$$

where $\xi_i(k)$ contains the error state $\tilde{x}_i(k)$, $e_i(k) = \xi_i(k) - \xi_{i,p}(k)$.

Substituting Equation (8) into $V_i(k+1) - V_i(k)$, we obtain:

$$\Delta V_i = \xi_i^T (A_{cl}^T P A_{cl} - P) \xi_i - 2\xi_i^T S e_i + e_i^T R_e e_i + \rho (\|e_i(k+1)\|^2 - \|e_i(k)\|^2) \quad (28)$$

where $A_{cl} = A_{K,i} + B_{K,i}K$, $S = A_{cl}^T P B_{K,i}K$, $R_e = K^T B_{K,i}^T P B_{K,i}K$.

From Equation (26), the first term satisfies:

$$\xi_i^T (A_{cl}^T P A_{cl} - P) \xi_i \leq -\xi_i^T Q \xi_i \quad (29)$$

Thus,

$$\Delta V_i \leq -\xi_i^T Q \xi_i - 2\xi_i^T S e_i + e_i^T R_e e_i + \rho (\|e_i(k+1)\|^2 - \|e_i(k)\|^2) \quad (30)$$

By Young's inequality, for any $\varepsilon > 0$, the second term satisfies:

$$-2\xi_i^T S e_i \leq \varepsilon \|\xi_i\|^2 + \varepsilon^{-1} \|S\|^2 \|e_i\|^2 \quad (31)$$

By applying Young's inequality to $\xi_i^T S e_i$, we obtain an upper bound involving only quadratic terms in ξ_i and e_i . Also, the third term satisfies $e_i^T R_e e_i \leq \lambda_{\max}(R_e) \|e_i\|^2$.

Thus,

$$\Delta V_i \leq -\alpha_1 \|\xi_i\|^2 + (\varepsilon^{-1} \|S\|^2 + \lambda_{\max}(R_e)) \|e_i\|^2 + \rho (\|e_i(k+1)\|^2 - \|e_i(k)\|^2), \quad (32)$$

where $\alpha_1 = \lambda_{\min}(Q) - \varepsilon > 0$.

The prediction error evolves as follows:

$$e_i(k+1) = F_e e_i(k), \quad (33)$$

where $F_e = A_{cl}$. Hence,

$$\|e_i(k+1)\|^2 - \|e_i(k)\|^2 = e_i^T (F_e^T F_e - I) e_i \quad (34)$$

Substituting into Equation (32):

$$\Delta V_i \leq -\alpha_1 \|\xi_i\|^2 + e_i^T M_e(\rho) e_i \quad (35)$$

where $M_e(\rho) = \varepsilon^{-1} \|S\|^2 I + \lambda_{\max}(R_e) I + \rho (F_e^T F_e - I)$.

If $\|F_e\| < 1$, then $I - F_e^T F_e > 0$. Choosing

$$\rho \geq \frac{\varepsilon^{-1} \|S\|^2 I + \lambda_{\max}(R_e) + \alpha_2 I}{\lambda_{\min}(I - F_e^T F_e)} \quad (36)$$

Ensures $M_e(\rho) \leq -\alpha_2 I$ for some $\alpha_2 > 0$.

Thus,

$$\Delta V_i \leq -\alpha_1 \|\xi_i\|^2 - \alpha_2 \|e_i\|^2 \quad (37)$$

Since $V(k) = \sum_{i=1}^N V_i(k)$, we obtain

$$\Delta V \leq -\alpha_1 \sum_{i=1}^N \|\xi_i\|^2 - \alpha_2 \sum_{i=1}^N \|e_i\|^2 \quad (38)$$

Then, for sufficiently large ρ satisfying the (36), $V(k)$ decreases strictly unless all $\xi_i = 0$ and $e_i = 0$ the closed-loop vehicle platoon is asymptotically stable. $\square$

5. Simulation and Result Analysis

To verify the effectiveness of the proposed controller, simulation experiments are carried out in MATLABR2022a/Simulink. A platoon of five intelligent connected autonomous vehicles is considered, consisting of 1 leading vehicle and 4 following vehicles. The predecessor-following (PF) topology is adopted as the communication structure. The parameter settings for the cost function are summarized in Table 1.

Table 1. Parameter Settings of the Cost Function for the DMPC.

Parameter	Value
Q_i	10 I
R_i	5 I
ρ	10
T	0.05 s
N_p	60
N_c	30

A total simulation time of 30 s was considered, with a sampling period of $T = 0.05$ s, leading to 600 sampled data points for each vehicle state.

5.1. Scenario-Based Validation

5.1.1. Disturbance Scenario

The disturbance scenario is designed to assess whether vehicles are initially operating from different states. The objective is to determine if they can converge to the desired states specified by the leading vehicle under the proposed controller. The corresponding simulation results are illustrated in Figure 3.

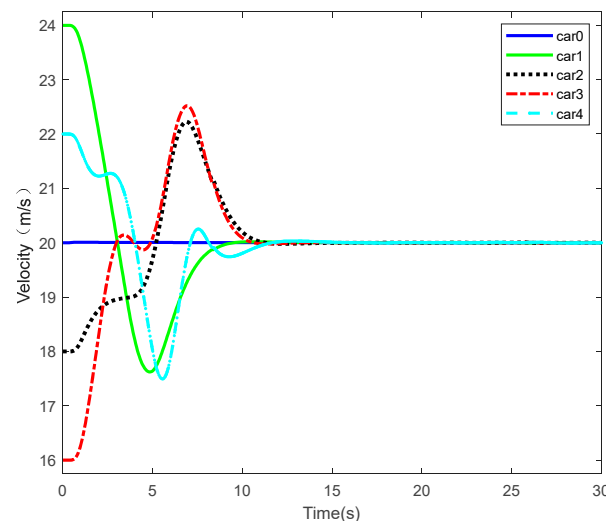


Figure 3. The speed trajectories of each vehicle under the disturbance scenario.

Figure 3 illustrates the platoon configuration, where Car 0 denotes the leader vehicle and Cars 1–4 denote the follower vehicles. The leader vehicle is not equipped with a controller; instead, its velocity profile is predefined and transmitted to the followers through the communication topology. In the disturbance scenario, the leader vehicle maintains a constant velocity of 20 m/s, which is transmitted to the followers through the communication topology. The initial speeds of the following vehicles (ordered front to rear) are set to 24 m/s, 18 m/s, 16 m/s, and 22 m/s, respectively. This heterogeneous configuration places the system in an unstable initial condition. Nevertheless, the simulation result demonstrates that the following vehicles, controlled by the proposed controller, rapidly converge

to the steady state. They achieve consensus with the leader vehicle's speed, matching the desired state. The simulation result robustly validates the designed controller's capability to stabilize the system despite significant initial internal instability. Furthermore, it satisfies the requirements for Lyapunov stability and consensus theory.

5.1.2. Acceleration Scenario

Acceleration represents one of the most fundamental and frequently encountered driving maneuvers in real-world traffic scenarios. Consequently, validating the performance of autonomous vehicle platoon systems under acceleration scenarios is essential.

To this end, we define the following reference velocity signal to verify the platoon's behavior during acceleration.

$$v_0 = \begin{cases} 10 \text{ m/s}, & t < 8 \text{ s} \\ 10 + 2t \text{ m/s}, & 8 \text{ s} \leq t < 13 \text{ s} \\ 20 \text{ m/s}, & t \geq 13 \text{ s} \end{cases} \quad (39)$$

At the commencement of the simulation, all leading and following vehicles within the platoon possess identical initial velocities and exhibit zero position errors. The corresponding simulation results are presented in Figure 4.

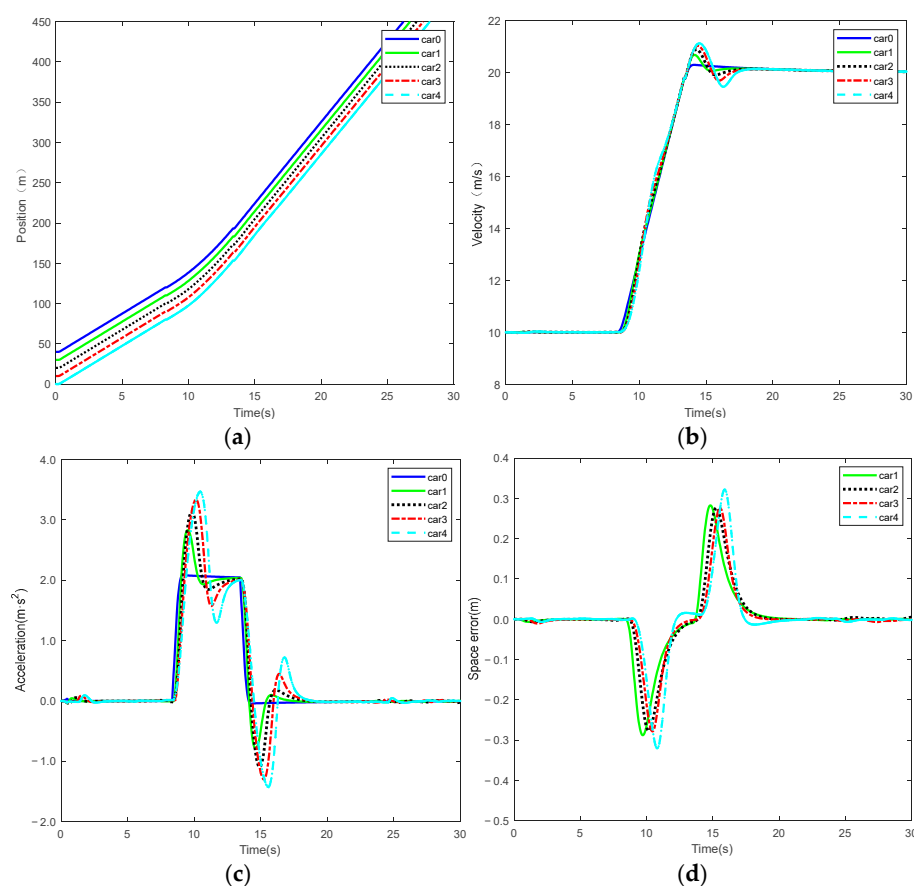


Figure 4. State trajectories of individual vehicles under accelerating scenario. (a) Position state trajectories; (b) velocity state trajectories; (c) acceleration state trajectories; (d) spacing error trajectories.

Figure 4 shows longitudinal position, velocity, acceleration and space error of vehicle platoon. Where Figure 4a shows that the position of each vehicle exhibits linear growth over time with parallel trajectories, indicating stable inter-vehicle spacing and well-maintained formation. Figure 4b,c show that the velocity and acceleration states of the

following vehicles effectively synchronize with variations in the leading vehicle, reflecting coordinated velocity consensus. Regarding space error, Figure 4d shows that the initial error for all vehicles approximates zero. Transient fluctuations are observed during acceleration, subsequently converging to a stable state, confirming the high stability of the overall platoon formation.

5.1.3. Deceleration Scenario

Validating deceleration maneuvers within vehicle platoons is crucial for ensuring road traffic safety and traffic flow stability. Verification under such scenarios ensures the accuracy of coordinated responses across the platoon.

To this end, we define the following reference velocity signal to verify the platoon's behavior during deceleration.

$$v_0 = \begin{cases} 20 \text{ m/s}, & t < 8 \text{ s}, \\ 20 - 2t \text{ m/s}, & 8 \text{ s} \leq t < 13 \text{ s}, \\ 10 \text{ m/s}, & t \geq 13 \text{ s}. \end{cases} \quad (40)$$

The simulation results of the developed vehicle platooning system under the proposed DMPC controller are presented in Figure 5.

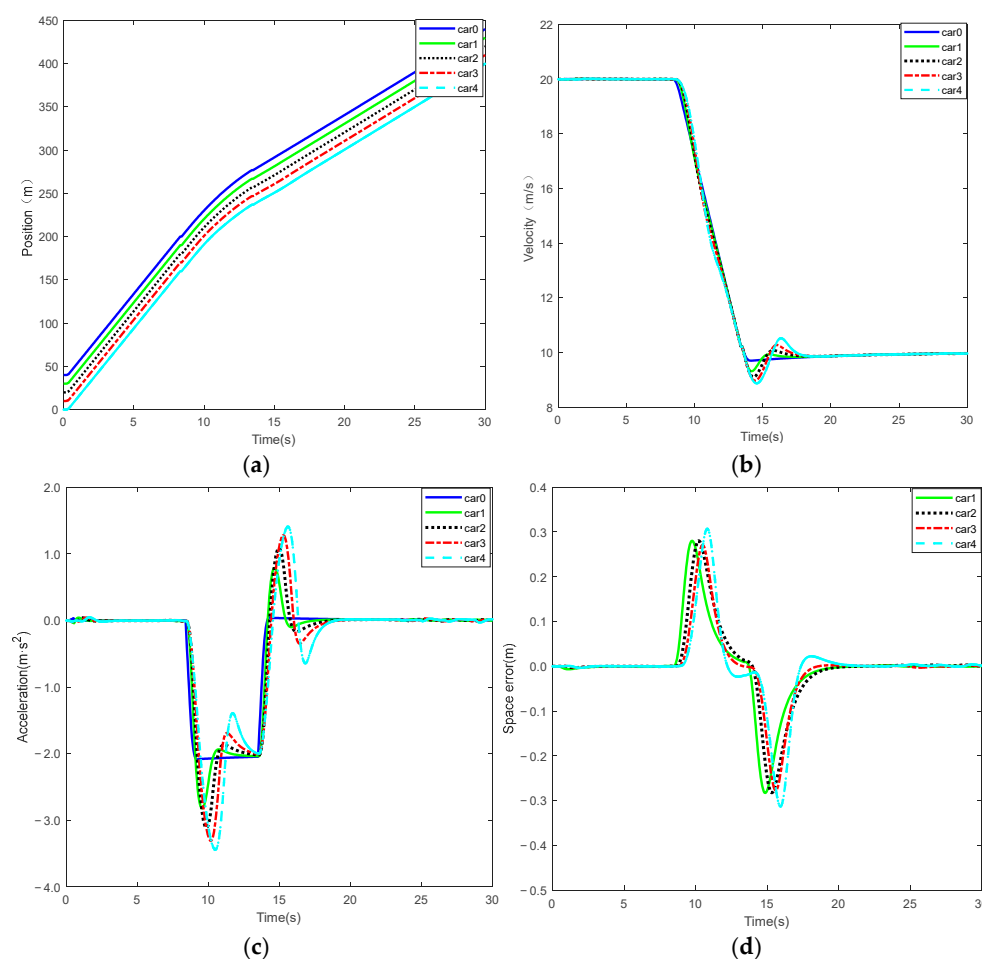


Figure 5. State trajectories of individual vehicles under decelerating scenario. (a) Position state trajectories; (b) velocity state trajectories; (c) acceleration state trajectories; (d) spacing error trajectories.

Figure 5 illustrates the longitudinal position, velocity, acceleration, and space error of the vehicle platoon under deceleration scenarios. Where Figure 5a shows that the position

trajectories of all vehicles remain parallel during the deceleration process, indicating stable inter-vehicle spacing and consistent maintenance of the desired following distance, with no risk of collision. Figure 5b,c demonstrate that the velocity and acceleration profiles of the following vehicles rapidly adapt to the state changes in the leading vehicle, reflecting effective coordination and disturbance rejection within the platoon. Regarding space error, Figure 5d shows that the position errors-measured relative to the leader-exhibit only brief transient deviations during deceleration and quickly converge to a narrow bound, signifying high-precision tracking and strong formation stability. Collectively, the platoon demonstrates synchronized velocity profiles, consistent acceleration patterns, smooth progression of inter-vehicle spacing with constant distance, and well-regulated position errors. These results collectively verify the effectiveness of the proposed control algorithm in enhancing safety and preserving platoon coherence under deceleration conditions.

5.2. Comparison

Based on the simulation results under the aforementioned three scenarios, the proposed DMPC controller achieves comparable tracking performance across all follower vehicles. For clarity and conciseness of comparison, a representative follower vehicle is selected to conduct detailed performance analysis.

5.2.1. Comparison of Triggering Strategies

To validate the effectiveness of the proposed event-triggered mechanism in reducing computational burden while ensuring tracking accuracy, the position errors of the vehicles are illustrated in Figure 6 and compared with those obtained using the velocity-based triggering strategy presented in [16].

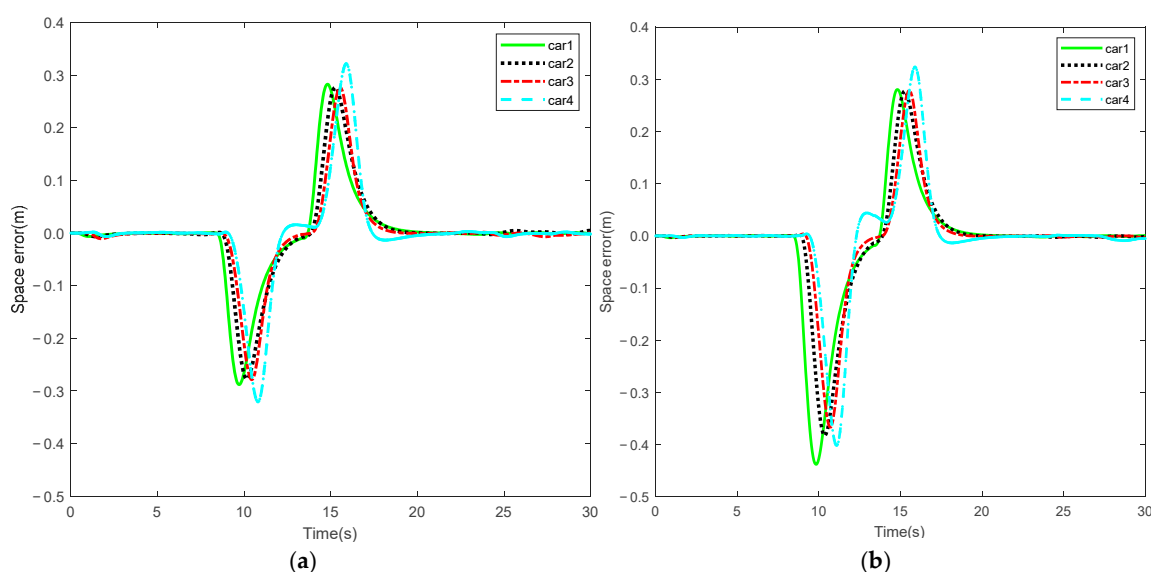


Figure 6. Comparison of tracking accuracy under two event-triggered mechanisms. (a) Space error under the position–velocity trigger; (b) space error under the velocity-based trigger [16].

Compared with the event-triggered strategy presented in [16], the proposed mechanism yields smaller error fluctuations during vehicle operation. This indicates that the proposed approach offers improved control precision and effectively mitigates the growth of position errors.

The corresponding quantitative evaluation results are summarized in Table 2, which presents tracking accuracy metrics under both triggering strategies.

Table 2. Comparison of tracking accuracy parameters for two event-triggered mechanisms.

Method	Average Position Error (m)	Maximum Position Error (m)
Position–Velocity Trigger	0.033	0.321
Velocity–Based Trigger in [16]	0.040	0.437

The simulation results demonstrate that the proposed event-triggered mechanism reduces the average position error by 17.5% and the maximum position error by 26.6%, compared to the velocity-based triggering strategy presented in [16], thereby verifying its effectiveness in suppressing tracking errors. Overall, the proposed position–velocity event-triggered mechanism achieves significant improvements in both average and maximum position errors over the single-variable velocity-based approach.

In addition, the proposed event-triggered mechanism also demonstrates advantages in reducing computational burden. A comparison of the triggering counts for the two event-triggered schemes during simulation is provided in Table 3.

Table 3. Comparison of triggering counts for two event-triggered mechanisms.

Event-Triggered Mechanism	Triggering Counts
Position–Velocity Trigger	382
Velocity–Based Trigger in [16]	482

As shown in Table 3, the proposed event-triggered mechanism significantly reduces the number of triggering events compared with the strategy presented in [16]. Specifically, the proposed method yields 382 triggering counts during the simulation, whereas the method in [16] triggers 482 times, representing a reduction of approximately 20.7%. This result indicates that the proposed mechanism effectively decreases the triggering frequency, thereby reducing both computational burdens.

5.2.2. Performance Comparison of ADMM Variants

As demonstrated in [26], the ADMM-based approach achieves significantly better real-time performance in solving quadratic programming problems compared to classical IPM and ASM. Therefore, these two traditional methods are excluded from the comparison. A qualitative analysis is first conducted by recording the computation time of each ADMM variant at every sampling instant, and the results are visualized in Figure 7.

Algorithm 1 represents Proposed ADMM in this article.

Algorithm 2 represents Heuristic ADMM in [21].

Algorithm 3 represents ADMM in [26].

Subsequently, a quantitative analysis is performed by comparing two performance metrics: the maximum and average computation times, as summarized in Table 4.

Table 4. Comparison of computational time among the three ADMM.

Method	Average Computational Time (s)	Maximum Computational Time (s)
Proposed ADMM	0.0016	0.013
Heuristic ADMM in [21]	0.0023	0.038
ADMM in [26]	0.0043	0.021

A cross-reference comparison of computation time results reported in [21,26] demonstrates that the ADMM-based method proposed in this study offers substantial improvements in computational efficiency. Specifically, the proposed method achieves a reduction

of approximately 30.4% and 62.8% in average computation time compared to the methods in [21] and [26], respectively. In addition, the maximum computation time is also lower, with reductions of about 65.8% and 38.1% relative to [21] and [26], respectively.

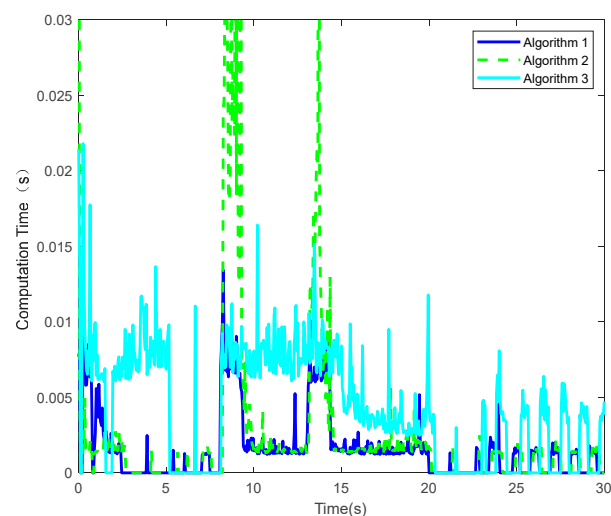


Figure 7. Comparison of computational time among the three ADMM.

To validate the superiority of the proposed adaptive ADMM algorithm, a comparison of tracking accuracy was conducted against two other ADMM variants under identical standardized simulation scenario. The corresponding results are illustrated in Figure 8.

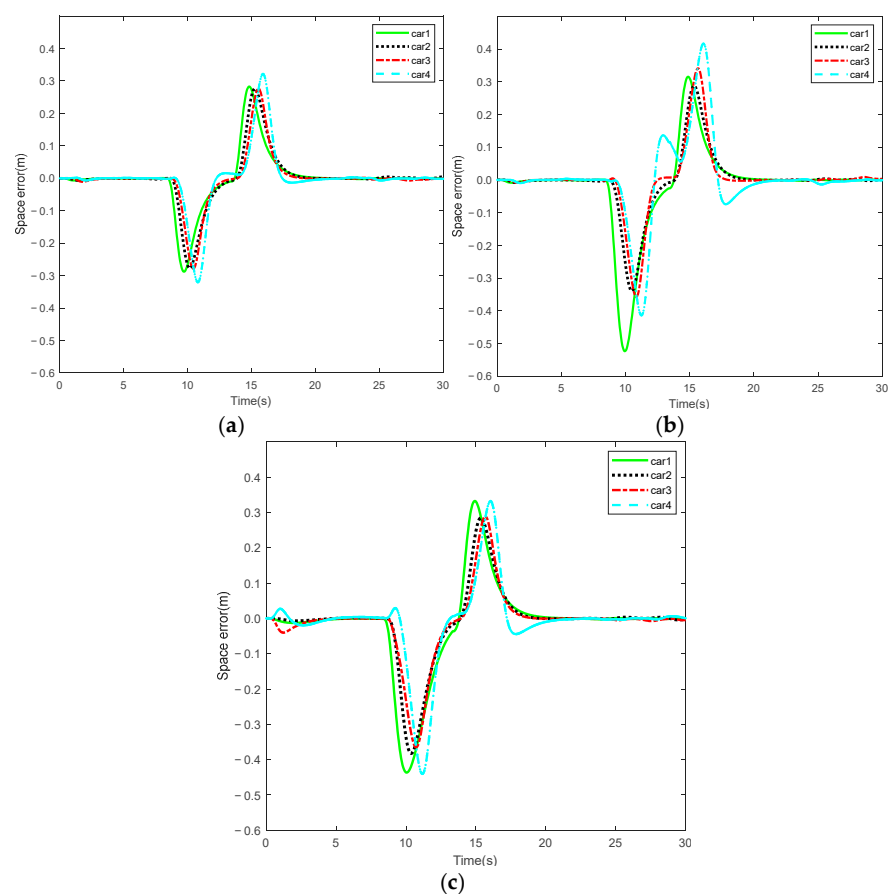


Figure 8. Comparison of tracking accuracy under three ADMM. (a) Space error under the proposed ADMM; (b) space error under the heuristic ADMM in [21]; (c) space error under the ADMM in [26].

As illustrated in Figure 8, the proposed adaptive ADMM algorithm achieves a significantly lower space error compared to the other two ADMM algorithms under identical coordinate scales. Subsequently, a quantitative comparison is performed by comparing two evaluation metrics: average and maximum position errors, as presented in Table 5.

Table 5. Comparison of tracking accuracy parameters for three ADMM.

Method	Average Position Error (m)	Maximum Position Error(m)
Proposed ADMM	0.033	0.321
Heuristic ADMM in [21]	0.047	0.524
ADMM in [26]	0.049	0.440

As shown in Table 5, the proposed ADMM method achieves a clear advantage in terms of position error. Specifically, the average position error achieved by the proposed approach is 0.033 m, which is significantly lower than the results of 0.047 m and 0.049 m reported in [21,26], corresponding to reductions of approximately 29.8% and 32.7%, respectively. Moreover, the maximum position error is also reduced, with a value of 0.321 m compared to 0.524 m in [21] and 0.440 m in [26], representing reductions of around 38.7% and 27%, respectively. These results demonstrate the effectiveness of the proposed method in limiting maximum tracking errors.

Remark 1. *The simulation results under other topology configurations exhibit similar behavior to those of the predecessor-following (PF) structure and lead to the same conclusions. For brevity, the results for these additional topologies are not included in the paper.*

The above results demonstrate that the proposed ADMM method effectively improves positioning accuracy by reducing the maximum tracking error and mitigating error fluctuations, thereby enhancing overall system stability and reliability. In summary, under road-driving scenarios with stringent real-time requirements, the proposed residual-based adaptive ADMM method significantly reduces computation time while improving tracking accuracy. These advantages contribute to enhanced driving safety and system stability, providing a reliable foundation for real-time vehicle control applications.

Moreover, the proposed event-triggered DMPC scheme demonstrates robustness across disturbance, acceleration, and deceleration scenarios, where the position errors consistently converge to zero, validating its capability to cope with dynamic driving conditions. From a computational perspective, each vehicle only solves a local QP through the a-ADMM decomposition, leading to moderate iteration counts (approximately 10–20 per step) and real-time feasibility. Simulations with varying fleet sizes further indicate that the average computation and communication burden per vehicle increases approximately linearly, thereby confirming the scalability of the distributed architecture. Future work will focus on extending the analysis to communication delays, packet losses, and larger-scale fleets.

6. Conclusions

This paper addresses the critical challenges of limited onboard computational resources and stringent real-time control requirements in DMPC-based vehicle platoon systems by presenting an event-triggered adaptive ADMM-DMPC framework. A longitudinal vehicle dynamics model and a communication topology are established to facilitate the implementation of the proposed control strategy under the DMPC scheme. Specifically, a residual-based adaptive ADMM algorithm is developed, which dynamically adjusts the

penalty parameter based on residual scaling. This significantly accelerates the solution of quadratic programming subproblems in DMPC, thereby ensuring real-time performance. Moreover, the proposed adaptive ADMM method enhances tracking accuracy and improves overall control quality. A dual-state event-triggering mechanism is designed, where optimization is triggered only when position or velocity states exceed predefined thresholds. This substantially reduces unnecessary computations and alleviates computational burden during the control process. The proposed strategy is thoroughly validated through numerical simulations.

To validate the effectiveness of the proposed method, we established a simulation platform within the Matlab/Simulink environment and conducted system simulations. Compared with the velocity-based triggering strategy in [16], the event-triggered mechanism reduces the average error, maximum error, and triggering frequency by 17.5%, 26.6%, and 20.7%, respectively. The ADMM-based distributed optimization further demonstrates improved computational efficiency, with reductions in average computation time of 30.4% and 62.8%, and reductions in maximum computation time of 65.8% and 38.1%, relative to [21,26]. In terms of tracking accuracy, the proposed approach achieves average and maximum errors of 0.033 m and 0.321 m, corresponding to improvements of up to 32.7% and 38.7% over existing methods. These results indicate that the method significantly reduces both the computational time per optimization step and the total number of optimizations performed, effectively mitigating computational demands on vehicle-side processors and markedly improving the real-time performance of cooperative vehicle platooning.

For future work, it is of interest to implement the proposed algorithm in cloud computing, which contributes to further saving onboard computational resources. Furthermore, this study does not consider the coupled lateral and longitudinal control, and it is promising to extend the proposed algorithm to the combined horizontal and longitudinal control of vehicle platoons.

Author Contributions: Conceptualization, J.W.; formal analysis, H.Y.; data curation, W.L.; writing—original draft preparation, H.Z.; writing—review and editing, H.Z.; supervision, H.Y. and X.Z.; funding acquisition, H.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Guangxi Key Laboratory of Automobile Components and Vehicle Technology (2023GKLACVTZZ07).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to thank the editor and reviewers for providing valuable review comments.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Agbaje, P.; Anjum, A.; Mitra, A.; Oseghale, E.; Bloom, G.; Olufowobi, H. Survey of Interoperability Challenges in the Internet of Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 22838–22861. [\[CrossRef\]](#)
2. Li, S.B.; Zheng, Y.; Li, K.Q.; Wu, Y. Dynamical Modeling and Distributed Control of Connected and Automated Vehicles: Challenges and Opportunities. *IEEE Intell. Transp. Syst. Mag.* **2017**, *9*, 46–58. [\[CrossRef\]](#)
3. Wu, C.; Cai, Z.; He, Y.; Lu, X. A Review of Vehicle Group Intelligence in a Connected Environment. *IEEE Trans. Intell. Veh.* **2024**, *9*, 1865–1889. [\[CrossRef\]](#)
4. Zheng, Y.; Li, S.E.; Li, K.; Borrelli, F.; Hedrick, J.K. Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies. *IEEE Trans. Control Syst. Technol.* **2017**, *25*, 899–910. [\[CrossRef\]](#)
5. Zhang, M.; Wang, C.; Zhao, W.; Liu, J.; Zhang, Z. A Multi-Vehicle Self-Organized Cooperative Control Strategy for Platoon Formation in Connected Environment. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 4002–4018. [\[CrossRef\]](#)

6. Hu, M.; Li, C.; Bian, Y.; Zhang, H.; Qin, Z.; Xu, B. Fuel Economy-Oriented Vehicle Platoon Control Using Economic Model Predictive Control. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 20836–20849. [[CrossRef](#)]
7. Qiang, Z.; Dai, L.; Chen, B.; Xia, Y. Distributed Model Predictive Control for Heterogeneous Vehicle Platoon With Inter-Vehicular Spacing Constraints. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 3339–3351. [[CrossRef](#)]
8. Pi, D.; Xue, P.; Xie, B.; Wang, H.; Tang, X. A Platoon Control Method Based on DMPC for Connected Energy-Saving Electric Vehicles. *IEEE Trans. Transp. Electrification*. **2022**, *8*, 3219–3235. [[CrossRef](#)]
9. Cen, S.; Bai, Z.; Wang, J.; Luo, X.; Li, X.; Fang, X.; Yuan, H. Distributed MPC for nonlinear networked vehicle platoon system with communication delays. In Proceedings of the 2024 China Automation Congress (CAC), Qingdao, China, 2–3 November 2024; pp. 1576–1581.
10. Xu, M.; Luo, Y.; Yang, G.; Kong, W.; Li, K. Dynamic Cooperative Automated Lane-Change Maneuver Based on Minimum Safety Spacing Model. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 1537–1544.
11. Zhao, J.; Ma, Y.; Dai, L.; Sun, Z.; Chen, H.; Xia, Y. Distributed Cloud Model Predictive Control With Delay Compensation for Heterogeneous Vehicle Platoons. *IEEE Trans. Veh. Technol.* **2025**, *74*, 11793–11805. [[CrossRef](#)]
12. Zhao, F.; Li, H.; Wang, J.; Li, J. String Stability Based Cloud Predictive Control of Vehicle Platoon With Random Time Delay. *IEEE Trans. Veh. Technol.* **2025**, *74*, 8784–8796. [[CrossRef](#)]
13. Wen, J.; Wang, S.; Dai, M.; Lyu, N. A New Longitudinal Speed Control Method for Connected and Automated Vehicle Platooning Under the Influence of Communication Delay. *IEEE Trans. Veh. Technol.* **2025**, *74*, 1–15. [[CrossRef](#)]
14. Zhao, H.; Li, Z.C.; She, D.S. Adaptive Event-Triggered Safety Control for String Stability of Vehicular Networked Systems. *IEEE Trans. Veh. Technol.* **2025**, *74*, 126–139. [[CrossRef](#)]
15. Luo, Q.Y.; Lam, J. Event-Triggered Tube-DMPC With Shrinking Ingredients for Vehicle Platoon Under Disturbance and Communication Delay. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 8467–8480. [[CrossRef](#)]
16. Han, Q.; Cheng, G.; Yang, H.; Zuo, Z. Bandwidth-aware transmission scheduling and event-triggered distributed MPC for vehicle platoons. In Proceedings of the 2022 41st Chinese Control Conference (CCC), Hefei, China, 25–27 July 2022; pp. 5532–5538.
17. Selvaraj, P.; Sakthivel, R.; Kwon, O.-M.; Sakthivel, R. Event-Triggered Position Scheduling Based Platooning Control Design for Automated Vehicles. *IEEE Trans. Intell. Veh.* **2024**, *9*, 6926–6935. [[CrossRef](#)]
18. Chen, J.; Wei, H.; Zhang, H.; Shi, Y. Asynchronous Self-Triggered Stochastic Distributed MPC for Cooperative Vehicle Platooning Over Vehicular Ad-Hoc Networks. *IEEE Trans. Veh. Technol.* **2023**, *72*, 14061–14073. [[CrossRef](#)]
19. Du, C.; Bian, Y.; Li, Z.; Liu, H.; Yu, S.; Shi, P. Hierarchical Event-Triggered Platoon Control for Heterogeneous Connected Vehicles Subject to Actuator Uncertainties and Non-Zero Inputs. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 235–253. [[CrossRef](#)]
20. Wang, L.; Zhang, Y.; Chen, X.; Li, H.; Zhou, M.; Sun, F. Periodic Event-Triggered Fault Detection for Safe Platooning Control of Intelligent and Connected Vehicles. *IEEE Trans. Veh. Technol.* **2024**, *73*, 5064–5077. [[CrossRef](#)]
21. Li, Y. Research on Model Predictive Control Approach Based on Alternating Direction Method of Multipliers. Master's Thesis, Anhui University of Technology, Ma'anshan, China, 2020.
22. Tang, M.M. Research on Distributed Predictive Control of Multi-UAV and Its Algorithm. Master's Thesis, Nanchang Hangkong University, Nanchang, China, 2022.
23. Mallick, S.; Dabiri, A.; De, B. Distributed model predictive control for piecewise affine systems based on switching ADMM. *IEEE Trans. Autom. Control* **2025**, *70*, 3727–3741. [[CrossRef](#)]
24. Feng, Y.Y. Distributed Model Predictive Control of Truck Platoons Under Intelligent Networked Environment. Ph.D. Thesis, Jilin University, Changchun, China, 2024.
25. Bai, W.Q.; Xu, B.; Liu, H.; Qin, Y.C.; Xiang, C.L. Coordinated Control of CAVs for Platooning Under a Parallel Distributed Model Predictive Control Framework. In Proceedings of the 2022 41st Chinese Control Conference (CCC), Hefei, China, 25–27 July 2022; pp. 5377–5382.
26. Dong, D.; Ye, H.; Luo, W.; Wen, J.; Huang, D. Fast Trajectory Tracking Control Algorithm for Autonomous Vehicles Based on the Alternating Direction Multiplier Method (ADMM) to the Receding Optimization of Model Predictive Control (MPC). *Sensors* **2023**, *23*, 8391. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.