automation

MDPI

Review

A Review of Control Solutions for Vehicle Platooning via Network Synchronisation Methods

Omar Hanif [1], Patrick Gruber [1], Aldo Sorniotti [2] and Umberto Montanaro [1,*]

[1] Centre for Aerodynamics, Aerospace and Automotive Engineering, University of Surrey, Guildford GU2 7XH, UK; o.hanif@surrey.ac.uk (O.H.); p.gruber@surrey.ac.uk (P.G.)
[2] Department of Mechanical and Aerospace Engineering, Politecnico di Torino, 10129 Torino, Italy; aldo.sorniotti@polito.it
* Correspondence: u.montanaro@surrey.ac.uk

Abstract

Vehicle platooning is a cooperative driving scenario in which a set of consecutive, connected and autonomous vehicles travel at the same speed while controlling their inter-vehicular distance. Organising traffic in platoons of vehicles can mitigate issues in road transport by improving safety, energy efficiency, and road usage. Vehicle platooning scenarios are enabled by communication across the fleet, allowing the design of distributed controllers to impose cooperative vehicle motion. In contrast to initial control strategies tailored for specific network topologies, the last decade has witnessed a substantial increase in vehicle platooning control solutions that treat the cooperative platoon motion as the synchronisation of a network of dynamic systems, thereby enabling their use across a wider range of topologies. Despite numerous publications in recent years, the literature lacks a comprehensive survey of network synchronisation methods for vehicle platooning. To fill this gap, this paper aims to review network synchronisation strategies proposed for controlling the longitudinal motion of vehicle platoons over the period 2013–2025, with particular focus on contributions from 2018 onwards. The literature on network-synchronisation-based vehicle platooning methods is reviewed within a four-component framework. Then, the most widely used families of distributed consensus controllers are analysed, and the ways in which heterogeneity, nonlinearities, delays, packet drops, external disturbances, and cyber attacks are accounted for and mitigated are examined, along with different types of closed-loop stability. The review also surveys approaches from the literature for validating and assessing synchronisation algorithms in vehicle platoons, covering both experimental and simulation studies, as well as the related simulation platforms. The review paper concludes by presenting research trends and gaps, as well as potential future directions.

Keywords: vehicle platooning; distributed controller; networked control systems; cybersecurity

Check for updates

Academic Editor: Yi-Xin Su

Received: 4 December 2025
Revised: 11 February 2026
Accepted: 12 February 2026
Published: 22 February 2026

Copyright: © 2026 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.

1. Introduction

The transportation system has undergone rapid change over the past couple of decades, but these advancements have introduced new challenges. On the one hand, travel has become more efficient, leading to decreases in both travel times and costs. On the other hand, there has been a subsequent increase in the number of road incidents, mismanagement of fuel, and high concentrations of pollution [1]. The need of the hour is to devise a solution concerning road safety and environmental challenges. A review of modern statistics underscores this necessity. In the United States, the National Highway Traffic

Safety Administration's survey for 2021 places the number of vehicle collisions that led to fatalities at 39,508 and the number of fatalities at 42,939 [2]; in the United Kingdom, the Department for Transport counted 1766 road casualties in 2022 [3]. Representing the environmental dimension, the U.S. Energy Information Administration (EIA) reported in 2022 that the transportation sector emitted a cumulative 1488 million metric tons of CO_2 [4]. There is a similar estimate by the U.S. Environmental Protection Agency, which estimates that transport accounts for 29% of national greenhouse gas emissions [4]. Correspondingly, the UK Department of Transport assigns 26% of the country's total greenhouse gas emissions to transportation, equivalent to 427 million tonnes of carbon dioxide in 2021 [5]. These numbers motivate us to pursue more eco-friendly technologies that enhance road safety while maintaining transportation efficiency. Given the aforementioned conditions, it is clear that an advanced solution is necessary to meet both the increasing safety and efficiency requirements. The vehicle platoon is the necessary solution. In the platoon, vehicles coordinate their movement by synchronising with the lead vehicle, using sensors and vehicle-to-vehicle (V2V) communication systems to maintain a uniform speed while maintaining safe spacing. Mathematically, this coordination can be viewed as a synchronisation problem in multi-agent systems, where each agent adjusts its states through local interactions and shared information to achieve a common objective.

The benefits of vehicle platooning in practice are numerous: (i) The collisions are greatly reduced because autonomous vehicles have reaction times of 10–100 ms, which is faster than the typical human response lag time of 1–2 s [6]. (ii) Maintaining a safe distance between vehicles not only increases safety but also reduces traffic bottlenecks and increases road capacity [7]. (iii) There are environmental benefits gained from the aerodynamic drag caused by air flowing over a platoon; this results in payable annual savings in fuel usage as well as reduced greenhouse emissions and thereby contributes to sustainability goals [8,9]. (iv) The uniform acceleration profile of the platoon copes with abrupt physical forces and thereby improves passenger comfort by reducing jerks [10,11]. Understanding the mechanism that yields these benefits requires studying the architecture of the control framework that enables vehicle platooning. The success of platooning depends on the design of an effective control system. Accordingly, this discussion continues with a description of the four-component framework proposed in [8] that underlies modern platooning control systems, as shown in Figure 1.

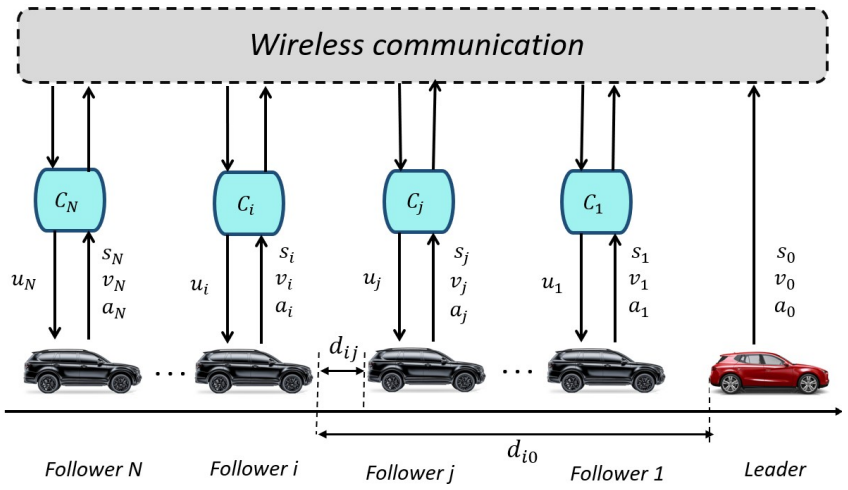


Figure 1. The four-component framework for platooning systems.

(i) Node dynamics: In the case of a vehicle platoon, every vehicle is considered a node. The node modelling must be accurate to ensure that the vehicle's simulated

dynamics reproduce realistic performance [8]. (ii) Information flow topology (IFT): This describes how the information (i.e., the vehicle states) is transmitted over the network to neighbouring vehicles in the platoon. Most commonly used IFT configurations include (a) Predecessor Follower (PF), (b) Predecessor Follower with Leader (PFL), (c) Bidirectional (BD), (d). Bidirectional with Leader (BDL), (e) Two Predecessor Follower (TPF), (f) Two Predecessor Follower with Leader (TPFL), (g) Bidirectional Predecessor Following (BPF), and (h) Bidirectional Predecessor Leader Following (BFL) [8]. (iii). Formation geometry (FG): This denotes the arrangement of the vehicles in the platoon [8]. (iv) Distributed controller (DC): The DCs are localised at each vehicle and perform independent control actions using state information from neighbouring vehicles [8]. The vehicle platoon's control system is hierarchical, consisting of two levels, as shown in Figure 2. The upper-level controller is distributed and responsible for cooperative adaptive cruise control by processing information from surrounding vehicles to determine the desired acceleration or torque. This desired value serves as a reference for the lower-level controller. Located on each vehicle, the lower-level controller executes throttle or brake commands to follow the reference provided by the upper-level controller.

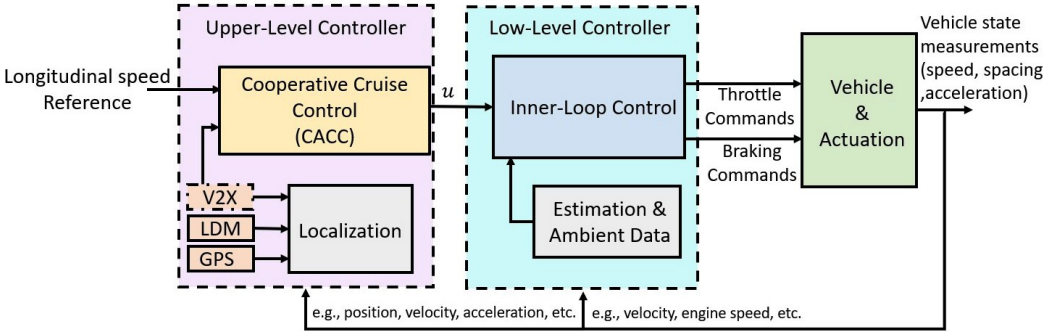


Figure 2. Hierarchical control architecture.

1.1. Survey Methodology

For a comprehensive analysis and generating the statistical trends as shown in Figure 3, a systematic literature survey was conducted in the following manner:

(i). A systematic search was conducted across primary academic databases, including IEEE Xplore, Scopus, Web of Science, and Google Scholar, using keywords such as "vehicle platoon," "formation control of vehicles in platoon," "distributed control of vehicle platoons," and "consensus control of vehicle platoons."

(ii). The resulting literature was further classified broadly into two main categories based on the control structure: (i) control law related to a specific topology, e.g., control law catering to vehicles communicating in a PF, BD, etc., topology; (ii) control law addressing general topologies (GTs). The control law designed to make consensus of the vehicles' states, thereby converging for a wide range of topologies, is known as the control law for general or generic topologies (GTs).This term has been coined in several publications, including [12–18], and refers to control laws that support platoon formation with vehicles communicating over a broad range of topologies. Figure 3 is plotted based on this classification.

(iii). While Figure 3 is obtained from papers published from 2013 to 2025, the rest of the review paper reports the more recent work from 2018 to 2025, pertaining to GTs.

(iv). The focus of the review is on the topic of distributed longitudinal control of vehicle platoons; hence, it excludes other areas of platooning, such as lateral control, two-dimensional control, traffic flow analysis, or human–vehicle interactions.

(v). The final selection of the literature is peer-reviewed journal articles and conference proceedings published only in the English language between January 2013 and October 2025.

From Figure 3, it can be inferred that the literature published between 2013 and 2015 mostly focused on developing the distributed controller for a particular topology. Over the years, a shift has occurred toward the development of distributed control algorithms that are not limited to a particular topology and can accommodate multiple topologies. This review paper examines the literature on vehicle platoons, focusing on communication in GT by breaking it down into four key components: node dynamics, information flow topologies, formation geometry, and distributed controllers. By focusing on a variety of distributed control approaches applicable to generic topologies, this review paper is novel in comparison to other literature surveys on vehicle platoons, which either did not cover a wide range of distributed controllers or did not specifically focus on generic topologies. Some of the literature survey papers are reviewed in Section 1.2.

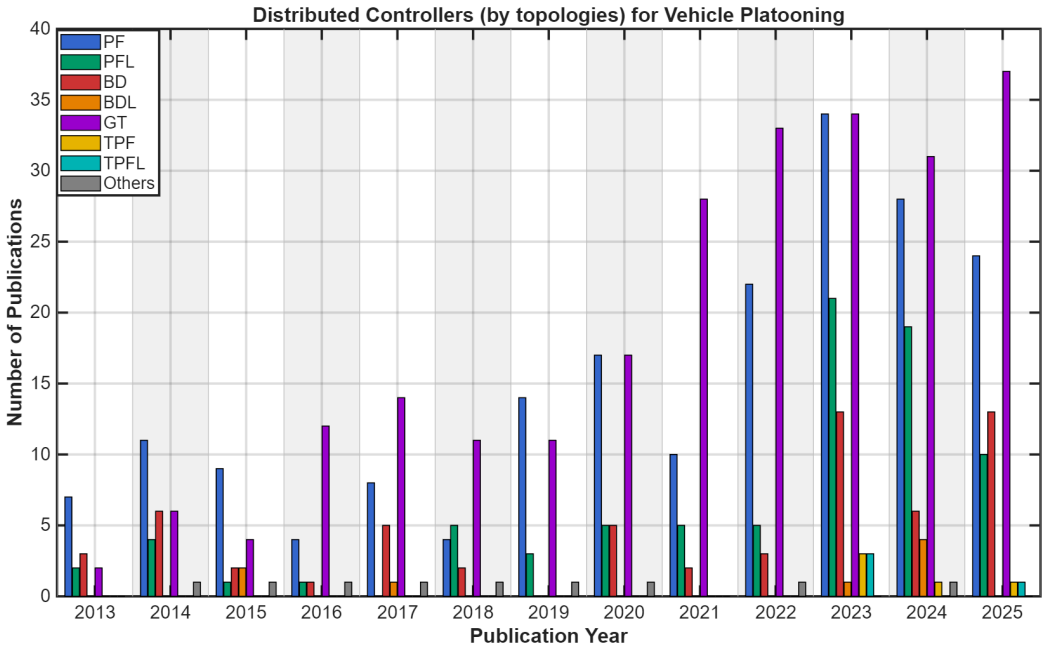


Figure 3. Bar chart of publications by topology type.

1.2. Previous Surveys on Distributed Control of Vehicles in a Platoon

This subsection surveys the selected literature review papers on vehicle platooning. It provides insights into how the selection is made, the main themes that literature reviews have covered, and, finally, the reason for this review paper on vehicle platoon control.

1.2.1. Methods of Selection

The selection is based on peer-reviewed survey papers on vehicle platooning, published in English only. Emphasis is placed squarely on surveys that include some discussion of platoon control, whether distributed, cooperative, or formation. Review articles that deal with vehicle platoons but do not address control are systematically excluded. The search strategy is used to canvass such databases as the web address of the Institute of Electrical and Electronics Engineers (IEEE) Research Library, presumed to be an electronic database of articles released as of 2013, such as the Science Direct database and the Association for Computing Machinery (ACM) Digital Library, as well as Google Scholar. If a paper is duplicated across multiple databases, it is manually deleted from each database. In cases where topics overlap with other surveys, each survey is tagged with multiple tags

summarising all relevant overlapping attributes, as listed in Table 1. The surveys below are categorised based on the major theme they focus on. Note that only this section, Section 1.2, contains the survey of reviews on vehicle platoons. Meanwhile, the other sections of our review paper present a comprehensive survey of technical papers on the longitudinal control of vehicle platoon formations communicating in the GT network.

Table 1. Summary of literature surveys on vehicle platooning.

Attributes	Literature Subjects Discussed	References
Node Dynamics	Modelling of individual vehicle dynamics in the platoon	[8,11,19–28]
Communication networks	Topologies between vehicles	[8,19,20,22–32]
Distributed Controllers	Distributed algorithms used for platoon control, e.g., MPC, SMC, etc.	[8,11,19–28,31,33]
Formation Geometry	How vehicles maintain their positions relative to each other	[8,19–25,27,28,30,31]
Performance and Stability metrics	Performance criteria such as stability (internal, string, robust), stability margin, coherence behaviour	[8,11,19–28,30–32]
Platoon Projects	Projects related to platoon	[21,26,27,29,30,32,33]
System Uncertainty and Robustness	Analysis and mitigation of factors like delays, packet loss, cyber attacks, and model uncertainties	[8,19,20,22–29,31–33]
Manoeuvres	Operations like merging, lane changing, etc.	[11,21,23,26–34]
Communication Architecture	Vehicular networking, V2V, V2I communication	[11,21–23,25,26,31,33]
Cybersecurity	Mitigation strategies regarding cyberfaults, DoS, etc.	[28,31]
Energy-based	Energy optimisation and all literature related to energy saving, fuel saving	[21–24,26,27,29–33]
Traffic Impact and Environmental Impacts	Effects on traffic flow, safety, environmental benefits	[21–34]

1.2.2. Overview of Existing Surveys in Vehicle Platoon

This section reviews the key themes that each survey focuses on.

(i). Surveys centric to the four components of platoon

Early surveys on vehicle platoon control organised the literature around four core components: node modelling, information flow topology, formation geometry, and distributed control [8,19]. They provided the other foundational concepts, such as different types of stability; however, the development of the distributed controller for GT lacked sufficient depth, as did the discussion of the practical challenges of vehicle platooning.

(ii). Surveys centric to stability

While most of the review papers slightly touched the stability part, including the definitions and types (also shown in Table 1), ref. [20] fully focused on string stability, which is a type of platoon stability dealing with the amplification of error in states of vehicles down the string. It then compared different definitions of string stability and analysed tools for computing across the time and frequency domains [20].

(iii). Surveys centric to Coordination, formation, and manoeuvres

This category of surveys emphasises the high-level organisational aspects of a platoon, outlining how the individual vehicles are grouped, coordinated, and controlled during complex operational manoeuvres. The main theme includes formation and coordination architectures, distributed controller algorithms, communication protocols, cooperative planning of trajectories and practices of joining, leaving, merging, and splitting operations [11,21–23,33]; while these review papers offer important insights into the logic behind decision making at the formation level, they are limited almost exclusively to a set of distributed longitudinal controllers and cover little on stability issues across generic network topologies, which is essential to the present review.

(iv). Surveys centric to influencing factors and consensus in uncertainties

Advancements in vehicle platoon technology have led to increased focus on identifying the sources of uncertainty that affect the platoon's coordinated behaviour and stability. The work in [24] explicitly compared factors such as vehicle heterogeneity, longitudinal disturbances, communication time delays, and network connectivity disconnections and reported their contributions to stability in terms of amplification of errors down the string (also known as string stability) and tracking errors. It concluded that network disconnections are the most detrimental cause of string instability. In another review work presented in [25], the authors organised the literature into imperfect transmission (communication time delays and data packet loss), resource constraints, and switching topologies. Further, they focused on measuring resilience, convergence rate, and robustness in the presence of communication network uncertainties. A complementary work regarding factors affecting stability is surveyed in [35], where large-scale cooperative driving scenarios, wireless communication, and stability are identified as the fundamentals of vehicle platooning. The work ensured reliable and secure communication as a means of successful, scalable cooperation. Although these surveys provide an explicit perspective on factors, they do not offer a controller family taxonomy centred on distributed, longitudinal control laws (e.g., sliding mode and model predictive control) and generic information flow topologies, which are the focus of the present review.

(v). Surveys centric to the CAV ecosystem and applications

Within this category, studies address the integration of platooning into the broader cooperative driving and Connected Autonomous Vehicle (CAV) ecosystem. The main theme covers system architectures, practical use cases, deployment considerations, and transport-level ramifications, but it does not go into controller-level stability investigations. For example, overviews of CAV control issues, prospective challenges and use cases are presented in [26,29], and a panoramic overview of freeway platooning and its broader implications is contained in [30]. The concept of collaborative autonomous driving and multi-vehicle cooperative control in transportation systems is surveyed in [27,31]. A complementary survey is given in [34]. It focuses on planning and decision-making for autonomous vehicles, including behaviour-aware planning for traffic interactions, safety and security considerations, and fleet-level operations (e.g., routing and assignment in mobility-on-demand environments). Although platooning is not the primary focus, this work supports the wider CAV ecosystem perspective and the need for critical safety and coordination perspectives, as well as controller-level platoon studies. In summary, these works comprise a critical part of the application contextualisation and platoon system integration; they generally do not provide the depth of controller analysis and the issues of vehicle platooning, including faults and cyber attacks, that our paper aims to provide.

1.2.3. Contribution of This Review Paper

Table 1 summarises the literature reviewed above, and identifies the key topics covered by the respective review paper. As shown in the table, no single survey on platoon control addresses all the themes. Hence, there is a need to cover the themes which neither of the review papers has surveyed. Importantly, they also do not precisely review papers for the synchronisation of states over a wide range of topologies. Lastly, many do not account for the most recent research up to 2025. Hence, there is an immediate need for up-to-date research that integrates all the aforementioned elements. This also includes covering practical challenges in the formation control of vehicle platoons, such as heterogeneity, nonlinearities, communication time delays, packet drops, switching topologies, external disturbances, and cyber-physical threats. Moreover, this review paper surveys studies used to validate platoon-control theory and sheds light on platooning trials and projects underway worldwide.

The remainder of this paper is structured as follows: Section 2 defines the objectives of platoon control. Section 3 presents a mathematical overview of the modelling components, including node dynamics, information flow topologies and formation geometry. Section 4 provides a detailed survey and classification of the distributed control strategies developed for vehicle platoons. Section 6 reviews platoon performance analysis, including stability analysis techniques. Section 7 discusses the simulation platforms and experimental setups presented in the literature. Section 8 discusses about the feasibility of the distributed controller in deployment and the barriers associated with it in light of real world scenarios. Finally, Section 9 concludes the paper, identifies critical research gaps, and proposes future research directions. As this review targets to survey vehicle platooning control solutions where the cooperative platoon motion is formulated and solved via network synchronisation strategies, Appendix A reports the formulation of the synchronisation problem of a network of dynamic systems for the sake of completeness. This is also shown in the roadmap of this review in Figure 4. The centre of the figure illustrates the introduction, followed by the formulation of the control problem. The right side of the figure corresponds to the components required for formulating the control problem. They are node dynamics, formation geometry, information flow topology, distributed control strategies, and fault mitigation. The left side of the figure informs the post-design phases. They include performance analysis, covering aspects of platoon stability, and algorithm validation through simulation and experimentation. Each main block is marked with a number corresponding to the section number, providing an overview of the contributions within each module.

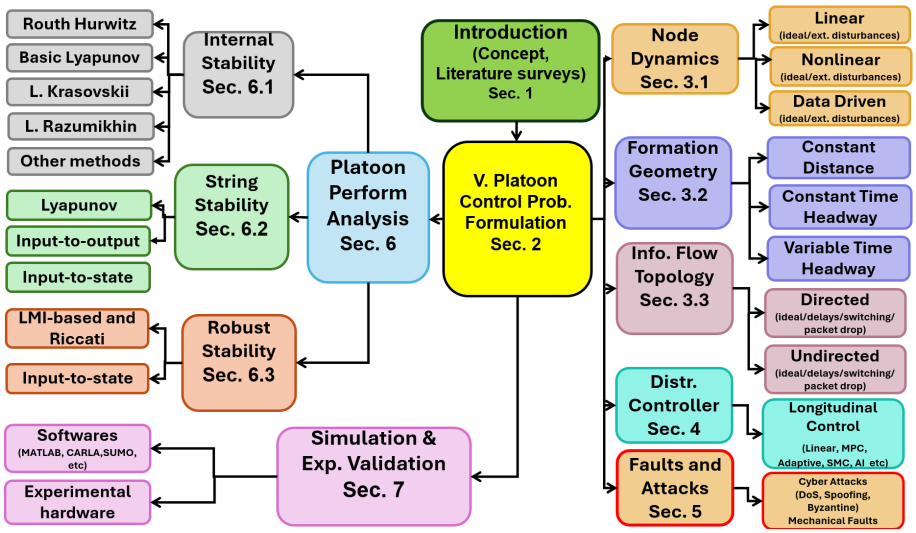


Figure 4. Structure and contributions of the review modules.

2. Platoon Control Problem Formulation

The control strategy for the platoon is devised by minimising a set of tracking errors in the vehicles' states. There are three main error types, which can be categorised as detailed below.

(i). Position error: This error measures how far the vehicle's position is from the desired inter-vehicular spacing. The position error of vehicle i with respect to a neighbouring vehicle j and the leader 0, respectively, is:

$$e_{s_{ij}} = s_i - s_j - d_{ij}^{\text{des}}, \tag{1a}$$

$$e_{s_{i0}} = s_i - s_0 - d_{i0}^{\text{des}}. \tag{1b}$$

The vehicle platoon control objective is to achieve the convergence of the position error to zero as

$$\lim_{t\to+\infty} \left|e_{s_{ij}}(t)\right| = 0, \qquad \lim_{t\to+\infty} \left|e_{s_{i0}}(t)\right| = 0. \tag{2}$$

(ii). Velocity error: This error states the difference of velocities between the vehicles. This can be with respect to the follower or the leader vehicle. Convergence of this error to zero brings the synchronisation of all vehicles of the platoon. The velocity error is defined as

$$e_{v_{ij}} = v_i - v_j, \tag{3a}$$

$$e_{v_{i0}} = v_i - v_0. \tag{3b}$$

The vehicle platoon control objective is to achieve the convergence of the velocity error to zero as $\lim_{t\to+\infty} |e_{v_{ij|}} = 0$ and $\lim_{t\to+\infty} |e_{v_{i0}}| = 0$.

(iii). Acceleration error: This refers to the difference between the acceleration of vehicles, either for the follower or the leader. It is responsible for the smooth and coordinated change of speed in a platoon. The acceleration errors with respect to neighbour j and leader 0 for vehicle i are given by:

$$e_{a_{ij}} = a_i - a_j, \tag{4a}$$

$$e_{a_{i0}} = a_i - a_0. \tag{4b}$$

The vehicle platoon control objective is to achieve the convergence of the velocity error to zero as: $\lim_{t\to+\infty} |e_{a_{ij}}| = 0$, and $\lim_{t\to+\infty} |e_{a_{i0}}| = 0$.

To ensure that the tracking state errors represented in (1), (3), and (4) are zero is to ensure the successful formation of a vehicle platoon, as established in [8]. Platoon control strategy, therefore, aims at two main objectives. (i) Synchronous alignment of state variables, namely velocity and acceleration, in all follower vehicles with respect to the leader vehicle, enforcing a uniform velocity profile throughout the platoon. (ii) Maintaining a predetermined inter-vehicle spacing, d_{ij}^{des}, according to a predetermined spacing policy. For the mathematical formalisation of these objectives, let us consider the ith follower vehicle, where $(i = 1, 2, \ldots, N)$. Its longitudinal position, velocity, and acceleration are denoted by s_i, v_i, and a_i, respectively. The leader's corresponding states are indicated by the subscript 0, and $d_{i,j}^{\text{des}}$ represents the desired separation between vehicle i and a neighbouring vehicle j. Accordingly, the overall platoon control objectives, summarised in the literature [36–42], are as follows:

$$\lim_{t\to+\infty}\left|s_i - s_j - d_{i,j}^{\text{des}}\right| = 0, \tag{5}$$

$$\lim_{t\to+\infty}|v_i - v_0| = 0, \tag{6}$$

$$\lim_{t\to+\infty}|a_i - a_0| = 0. \tag{7}$$

3. Platoon System Modelling

In order to preserve the integrity of the four-component framework introduced in Section 1, it is imperative that the constituent elements be modelled accurately; while the review on the design of distributed controllers is presented in a comprehensive way in Section 4, the contents of this specific section focus on the following: (i) modelling of node dynamics; (ii) modelling of formation geometry; (iii) modelling of the communication network.

3.1. Node Dynamic Modelling

In a platoon, each vehicle is termed as a node; for example, the ith node represents the ith vehicle in the platoon, as shown in Figure 1. The nodes are categorised by order of their dynamics in the following section.

3.1.1. First-Order Node Modelling

The simplest model for node dynamics is the first-order model. This type of model intentionally omits powertrain and chassis dynamics; thus, vehicle velocity is the only control input [43]. The first-order node dynamics model for the ith vehicle are characterised by:

$$\dot{s}_i(t) = u_i(t). \tag{8}$$

Although this representation does not include the dynamics of acceleration, aerodynamic drag, and powertrain lag, it is a powerful tool for exploring theoretical concepts, enabling focus on distributed control strategies, network topology effects, and multiagent system stability without node dynamics [43]. From a practical point of view, this abstraction has been shown to be most suitable when the dynamics of propulsion and braking are sufficiently fast that inertial and powertrain lag are negligible relative to the control update cycle. Such an assumption is often justified for electric powertrains, where rapid torque application allows inertial lag effects in follower vehicles to be ignored [44]. When these lags cannot be dismissed, it becomes necessary to use higher-order longitudinal models that explicitly capture the dynamics of the actuators.

3.1.2. Second-Order Node Modelling

Another type of node model is a second-order model. It exhibits second-order dynamics and may take either linear or nonlinear forms. A typical example of a nonlinear second-order representation [45] is:

$$\begin{cases} \dot{s}_i(t) = v_i(t), \\ \dot{v}_i(t) = \frac{\eta_{Ti}}{R_{Wi}M_i}\,T_{di}(t) - g\,f_i(t) - \frac{C_{Ai}}{M_i}\,v_i(t)^2, \end{cases} \tag{9}$$

where

$$\begin{cases} f_i(t) = \left(\sin\theta_i(t) + \cos\theta_i(t)\right)\dfrac{C_r}{1000}\left(c_1\,v_0(t) + c_2\right), \\ C_{Ai} = \dfrac{\rho}{2}\,C_{Di}\left(d_{i-1,i}(t)\right)\,A_i, \end{cases} \tag{10}$$

where $s_i(t)$ and $v_i(t)$ are the longitudinal position and velocity of the ith vehicle, respectively; $T_{di}(t)$ is the driving/braking torque applied at the wheels; η_{Ti} is the drivetrain efficiency factor; R_{Wi} is the wheel radius; M_i is the vehicle mass; and g is the gravitational acceleration. The term $f_i(t)$ represents the rolling-resistance-related component depending on the road inclination $\theta_i(t)$, where C_r is a rolling resistance coefficient and c_1, c_2 are surface/tyre-dependent coefficients, with $v_0(t)$ denoting the reference (lead-vehicle) speed used in that rolling resistance model. The aerodynamic term is captured by C_{Ai}, where ρ is the air density, A_i is the frontal area of the ith vehicle, $d_{i-1,i}(t)$ is the inter-vehicular distance between vehicles $(i-1)$ and i, and $C_{Di}\big(d_{i-1,i}(t)\big)$ is the drag coefficient modelled as a function of that distance (to capture spacing-dependent aerodynamic effects). Using the principle of state-feedback linearisation, the nonlinear second-order model is transformed into linear second-order form [46,47]:

$$\dot{s}_i(t) = v_i(t), \tag{11a}$$

$$\dot{v}_i(t) = u_i(t). \tag{11b}$$

The linearised form presented in (11) also assumes that the acceleration is attained with a very small actuation lag. This assumption is justified when the traction system provides fast torque tracking (e.g., electric engines) [44]. Where this condition is not satisfied, it is common practice for the dynamics of the actuators to be augmented in the model of the nodes, leading to the third-order representation presented in the subsequent paragraph.

3.1.3. Third-Order Node Modelling

A third-order model of the node dynamics shows more parameters and thus more accurate characteristics than the first- and second-order forms. For example, the third-order nonlinear model accounts for engine, braking, aerodynamic drag, tyre friction, and rolling resistance effects [48]. The third-order nonlinear node model is [49]:

$$\begin{cases} \dot{s}_i(t) = v_i(t), \\ \dot{v}_i(t) = -\frac{C_{Ai}}{M_i}\dot{v}_i(t)^2 - gf_i + \frac{\eta_{Ti}}{M_i R_{Wi}} T_{di}, \\ \dot{T}_i = -\frac{1}{\tau_i} T_i - \frac{1}{\tau_i} T_{di}, \end{cases} \tag{12}$$

where $s_i(t)$ and $v_i(t)$ are position and velocity of the ith node; M_i is vehicle mass; C_{Ai} is drag coefficient; g is gravitational constant; f_i is rolling resistance coefficient; R_{Wi} is tyre radius; η_{Ti} is drivetrain efficiency; T_i is actual driving/braking torque; T_{di} is commanded torque; and τ_i represents the inertial time lag of the powertrain. The third-order nonlinear model in (12) is linearised via state-feedback linearisation [50–52]. The desired torque T_{di} compensates all nonlinear terms:

$$T_{di}(t) = \frac{R_{Wi}}{\eta_{Ti}}\Big(M_i u_i(t) + C_{Ai} v_i^2(t) + 2\tau_i C_{Ai} v_i(t) a_i(t) + g M_i\Big). \tag{13}$$

Hence, after state feedback linearisation, the linear third-order model is:

$$\dot{s}_i(t) = v_i(t), \tag{14a}$$

$$\dot{v}_i(t) = a_i(t), \tag{14b}$$

$$\dot{a}_i(t) = -\frac{1}{\tau_i} a_i(t) + \frac{1}{\tau_i} u_i(t). \tag{14c}$$

3.1.4. Data-Driven and Model-Free Node Modelling

Data-driven or model-free techniques do not employ a first-principles approach to system modelling. These approaches are found to be beneficial when vehicular dynamics are complex, highly nonlinear or contain unknown parameters [53,54]. For example,

ref. [53] uses a reinforcement learning algorithm where the controller incorporates a control policy through its interaction with the environment, thereby eliminating the need for explicit description of the system [53,54]. A further approach uses a linear state-space framework with the system matrices as latent parameters. Within this paradigm, a Q-learning scheme directly recovers the optimal control gain from the system data, eliminating the need for an explicit model of the nodes [55]. Despite the promising possibilities data-driven techniques offer, these approaches still face significant stability issues. Pure reinforcement-learning-based controllers, for instance, have no formal stability guarantees as their optimisation procedures aim to maximise a reward rather than ensuring adherence to Lyapunov stability criteria [56,57]. To overcome these problems, recent approaches have implemented control-theoretic safeguards such as control barrier functions and Lyapunov-based constraints, thereby guaranteeing stability and safety during both the training and deployment phases [56,58].

3.1.5. General Modelling of Multiagent Systems

A significant amount of research on platoon control follows a generalised modelling framework rather than a particular model type [50,59–62]. Rather than committing to specific physical models shown in Sections 3.1.1–3.1.3, this modelling method formulates the vehicle platooning node problem as an abstract multi-agent system (MAS). This abstraction enables greater focus on control-theoretical solutions, such as stability analysis, and allows the developed algorithm to be subsequently validated on second or third-order node dynamics. The general model for the the ith agent is:

$$\dot{x}_i(t) = A_i x_i(t) + B_i u_i(t), \tag{15}$$

where $x_i(t)$ is the state vector, $u_i(t)$ is the control input, and A_i and B_i are the state and input matrices, respectively, [50,61]. For platoons modelled with nonlinear node dynamics, the generic formulation for the ith agent becomes:

$$\dot{x}_i(t) = A_i x_i(t) + B_i u_i(t) + f_i(x_i, t), \tag{16}$$

with the nonlinear term $f_i(x_i, t)$ containing the intrinsic dynamics of each agent [59,60,62]. Controllers based on these generalised structures have broad applicability to platooning scenarios.

3.1.6. Challenges Pertaining to Node Modelling

(i) In terms of disturbances: In the presence of a disturbance acting on the node, it is crucial that the vehicle model at the node level must explicitly account for the accurate disturbance model, as addressed in previous research [63,64]. These disturbances can be caused by wind, friction, model mismatches in feedback linearisation, as mentioned in [17]. Broadly, there are three classes of disturbance models: (i) ideal, or zero disturbance, as presented in [65–67]; (ii) $\mathcal{L}_2$, or finite energy disturbance model, as presented in [68]; (iii) $\mathcal{L}_\infty$, or bounded amplitude disturbance model, as studied in [49,69]. Based on the aforementioned classes, the disturbance function $\epsilon_i(t)$ can collectively be shown by the following mathematical expression:

$$\begin{cases} \epsilon_i(t) = 0 & \text{ideal,} \\ \int_0^\infty \|\epsilon_i(t)\|^2 dt \text{ is finite} & L_2, \\ \|\epsilon_i(t)\| \leq \|\epsilon_i\|_\infty & L_\infty. \end{cases} \tag{17}$$

(ii) In terms of the nature of the platoon: Vehicle platoons are classified as homogeneous or heterogeneous. A homogeneous platoon consists of N nodes with identical model parameters [70] (e.g., $C_{Ai} = C_A$; $M_i = M$; $f_i = f$; $\eta_{Ti} = \eta_T$; $R_{Wi} = R_W$; $\tau_i = \tau$ in Equations (9), (12) and (14)). Conversely, a heterogeneous platoon contains nodes with differing parameters [71,72]. Control strategies must be tailored to this characteristic to ensure effective performance.

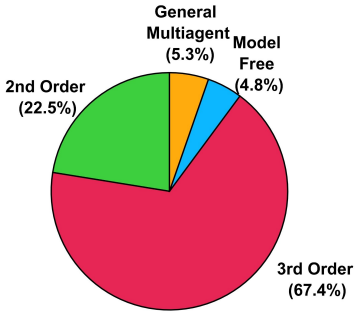
Table 2 gives a summary of the classification of the literature that is based on a systematic selection strategy developed from the definitions expressed in Sections 3.1.1–3.1.5. The distribution of each research follows a hierarchical protocol: Primarily, the modelling framework—a physical, data-driven or general MAS. Secondly, the order of dynamics and the respective linearity are determined by doing a cross-reference to the dynamic formulations used for controller synthesis—for example, feedback-linearised representations (e.g., (11) and (14)) are classified as linear, while those that maintain intrinsic nonlinearities (e.g., (9) and (12)) are specified as nonlinear. Thirdly, the external disturbance is sorted as ideal, $\mathcal{L}_2$ or $\mathcal{L}_\infty$ according to the definitions provided in (17). Finally, the platoon is homogeneous or heterogeneous, depending on the parameter uniformity as described in Section 3.1.6 *(ii)*. It is pertinent to note that this classification is based solely on the model used for the theoretical control development and stability analysis, not on other theoretical models used to test robustness in simulation. Comparative analyses of these approaches are presented in more detail in Tables 3–5. A statistical overview is shown in Figure 5, which indicates that models with third-order dynamics (67.4% of publications) and linear dynamics (75.3% of publications) are commonly used, as they provide a satisfactory trade-off between accuracy and complexity while also simplifying the stability analysis. Moreover, it is shown that nonlinear models represent about a quarter of the recent literature; thus, vehicle platooning control solutions that can be operated without mid-level controllers (e.g., those implementing (13)), needed for linearising vehicle dynamics, are still limited, thus showing potential future research.

Table 2. Classification of literature based on node modelling. Papers which develop controllers for more than one model type are indicated with an asterisk (*).

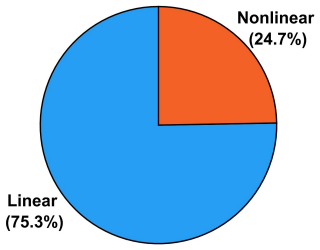
Node States	Node Dynamics	Disturbances	Platoon Nature: Homogeneous	Platoon Nature: Heterogeneous
First and Second Orders	Linear	Ideal	[38,43,64,65,67,71,73–77]	—
		$\mathcal{L}_2$	[78]	[79]
		$\mathcal{L}_\infty$	[47,63,80], [81] *	[72,82]
	Nonlinear	Ideal	[83–87]	[45,51,88–91]
		$\mathcal{L}_2$	—	[92]
		$\mathcal{L}_\infty$	[93,94]	[95–103]
Third	Linear	Ideal	[39–42,52,70,104–113], [114] *, [115–132]	[114] *, [133–155]
		$\mathcal{L}_2$	[14–17,37,156–163]	[164–167]
		$\mathcal{L}_\infty$	[12,36,49,168–185], [81] *, [186–188]	[13,18,189–196]
	Nonlinear	Ideal	[197]	[198–201]
		$\mathcal{L}_2$	—	[202]
		$\mathcal{L}_\infty$	[203]	[204–213]
Model free	—	Ideal	[54,55,214–216]	[53,217,218]
		$\mathcal{L}_\infty$	—	[219]
General Multiagent	Linear	Ideal	[61,66,220]	—
		$\mathcal{L}_\infty$	[221]	[50,62]
	Nonlinear	Ideal	—	[59,222]
		$\mathcal{L}_\infty$	—	[60,223]

Table 3. Comparison of second- and third-order vehicle models.

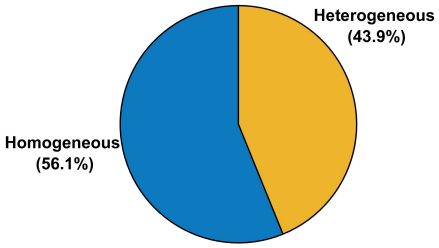
Attribute	2nd Order Model	3rd Order Model
Realism	A simplified model that may not capture all real vehicle dynamics [42,156].	Offers a more realistic representation of longitudinal dynamics [156].
States	Includes position and velocity states to model vehicle dynamics. There is no actuator lag [178].	In addition to position and velocity, acceleration is the third state present to model vehicle dynamics. An actuator lag is also present [138,178].
Passenger Comfort	Not explicitly considered, as the model lacks jerk (change in acceleration) dynamics.	Allows for direct management of jerk to improve passenger comfort [42,156].



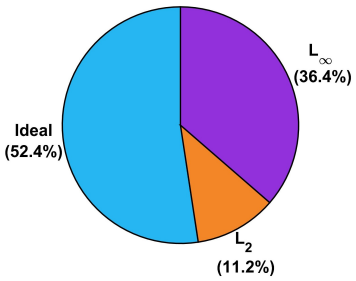
(**a**) Node model type



(**b**) Node behaviour



(**c**) Nature of platoon



(**d**) Type of disturbance acting at node level

Figure 5. Overview of node-level aspects in platoon modelling: (**a**) node model type; (**b**) node behaviour; (**c**) nature of platoon; (**d**) type of disturbance acting at node level.

Table 4. Comparison of linear and nonlinear vehicle models.

Attribute	Nonlinear Model	Linear Model
Realism	A more realistic model depicting a vehicle's inherent dynamics and physics by inherently including complex behaviours. This covers aspects such as drivetrain dynamics, physical input constraints, aerodynamic drag, and rolling resistance [51,54,88].	A simplified model derived from a nonlinear model, which typically omits many of the complexities such as inherent nonlinearities and practical constraints on control inputs of the original dynamics [51,54].
Suitability for Analysis	The complexity makes rigorous theoretical analysis and controller synthesis challenging [16,54].	The simplified model enables straightforward stability proofs and the application of established control theory [51,54].

Table 5. Comparison of physical vs. data-driven modelling approaches.

Attribute	Physical Model	Data-Driven (Model-Free)
Requirement	Requires knowledge of the system dynamics, as the controller design depends on a mathematical model [222].	Does not require a prior mathematical model. The controller design relies solely on the system's input and output data, respectively, [216,222].
Handling Uncertainty	Performance is heavily dependent on the model's accuracy and can be very vulnerable to unmodelled dynamics and external disturbances [216].	Well-suited for systems with unknown or stochastic characteristics, as it learns the behaviour directly from interaction with the system [53].
Controller Design	Consists of a single phase where a controller is designed based on a known system matrix [222].	Involves two main phases: a model learning phase from data, followed by a controller design phase based on the learned model [222].

3.2. Formation Geometry

Formation geometry (FG) is the second of the four components of the vehicle platoon control framework. FG is the desired spatial arrangement or policies between the vehicles in the platoon [8]. Three main spacing policies have been published in the literature as follows: (i) constant distance spacing (CDS) (ii); constant time headway (CTH); (iii) variable time headway (VTH).

3.2.1. Constant Distance Spacing (CDS)

The CDS policy guarantees that the desired spacing between the ith and jth vehicles is maintained regardless of the platoon's speed. This spacing policy avoids increased spacing, resulting in a tighter platoon, thereby reducing fuel usage and utilising road capacity more efficiently than other spacing policies [98,107]. According to the CDS policy, the desired distance between ith and jth vehicle is:

$$d_{i,j}^{\text{des}} = d_{i,j}^{\text{st}}, \tag{18}$$

where $d_{i,j}^{\text{des}}$ is the desired distance between the ith and jth vehicles, and $d_{i,j}^{\text{st}}$ is the constant standstill distance.

3.2.2. Constant Time Headway (CTH)

The CTH policy ensures a constant time interval between two successive vehicles. Under such a policy, the distance between vehicles is adjusted linearly with the platoon velocity. A platoon with a CTH spacing policy is often said to perform better in terms of safety at high speeds due to the larger resultant spacings. However, these larger distances cause a loose platoon formation, and thus a lower throughput on the highway [93,111,158]. According to CTH policy, the distance it wants to keep between the ith and jth vehicle is:

$$d_{i,j}^{\text{des}} = h_{i,j} v_0(t) + d_{i,j}^{\text{st}}, \tag{19}$$

where $h_{i,j}$ is the time headway, $d_{i,j}^{\text{st}}$ is the stand still distance between ith and jth vehicles and $v_0(t)$ is the velocity of the leader vehicle.

3.2.3. Variable Time Headway (VTH)

The variable time headway (VTH) is also characterised as a nonlinear headway policy [224]. In this policy, the desired inter-vehicular spacing between the vehicles varies nonlinearly with the platoon velocity. VTH policy at high speeds permits larger gaps, while at low speeds it allows smaller gaps, thereby increasing road capacity [83,190]. Empirical research indicates that VTH improves safety and performance in complex driving conditions characterised by frequent velocity variations, better than the aforementioned policies (CDS and CTH). In general, the VTH policy, a nonlinear function of platoon velocity, can be expressed as:

$$d_{i,j}^{\text{des}}(t) = d_{i,j}^{\text{st}} + h_{i,j}\big(v_0(t)\big)\, v_0(t). \tag{20}$$

where $h_{i,j}(\cdot)$ is the time-headway. It can be any suitable nonlinear function of the velocity argument. A popular VTH policy is the quadratic function of the platoon speed [83,190]

$$d_{i,j}^{\text{des}}(t) = c_{i,j}\, v_0^2(t) + h_{i,j}\, v_0(t) + d_{i,j}^{\text{st}}, \tag{21}$$

where $c_{i,j} > 0$ and $h_{i,j} > 0$ are design coefficients, and $d_{i,j}^{\text{st}}$ is the standstill distance between the ith and jth vehicles. The time-headway $h_{i,j}(\cdot)$ of the VTH can have other nonlinear forms for (e.g., saturating or piecewise functions) depending on requirements. More formulations can be found in [224,225].

Table 6 contains a detailed summary of the available literature, based on a selection approach from the definitions of spacing policy described in Sections 3.2.1–3.2.3. For each study, the spacing policy has been allocated by cross-referencing the policy concerned implemented in the controller development with the corresponding formalism: CDS (18), CTH (19) or VTH (21). After the policy is identified, the "Typical Values"column records the inter-vehicular parameters reported within the same formalism. Where no numerical value is provided, the study is marked with 'Not Specified'. Importantly, the table reflects the spacing policy adopted for theoretical controller design and stability analysis, rather than any additional policy used only for robustness checks in the simulation. As illustrated in Figure 6, the most common type of spacing policy is CDS. It is found in 78.6% of the surveyed literature. Figure 6b further clarifies the parameter distribution, showing that within CDS implementations, inter-vehicular distances within 5–10 m are most often adopted (28.3% of the surveyed literature) and the 10–20 m distance range is second most commonly adopted (16.6% of the surveyed literature). Regarding the other prominent spacing policy, CTH accounting for 18.2% of the publications, most platooning is performed with small temporal headways of 0–1 s (12.3% of the studies). In contrast, VTH policy is poorly represented at 3.2% of the surveyed literature. This significant asymmetry suggests a clear opportunity for future research into adaptive spacing policies that better account for practical traffic conditions. A comparative analysis of CDS, CTH and VTH, highlighting key advantages, limitations, and the feasibility of these technologies to inform appropriate policy choices, is presented in Table 7.

Table 6. Spacing policies for vehicle platooning. Papers addressing more than one spacing policy simultaneously are denoted with an asterisk (*).

Policy	Typical Values	References
Constant Distance	<5 m	[38,51,62,65,75,78,85,100,155,159,173,177,179,195,206,212,219]
	[5–10] m	[13,14,18,37,39,55,61,73,74,82,98,102–104,106–108,116,119,120,123,125,127,130,141,143,145,150,154,162,164,168,169,171,174,180–185,187,191–193,197,199,201,203,205,207,220,221]
	(10–20] m	[217] *, [41,47,59,63,66,81,84,86,88,90,91,96,101,115,121,124,134–136,139,140,163,186,196,200,213,214,216,218,222]
	>20 m	[15–17,49,52,64,77,117,118,153,157,170,176,211]
	Not Specified	[12,36,40,43,50,54,60,67,70–72,76,92,94,95,97,113,122,126,128,131,132,161,165,172,175,178,198,202,208,210,215,223]
Constant Time Headway	(0–1] s	[217] *, [45,79,80,99,109,111,112,114,129,133,142,144,147–149,158,160,167,188,194,204,209]
	[1–2] s	[42,53,138,151,156,166]
	Not Specified	[87,93,110,137,152]
Variable Time Headway		[83,89,105,146,189,190]

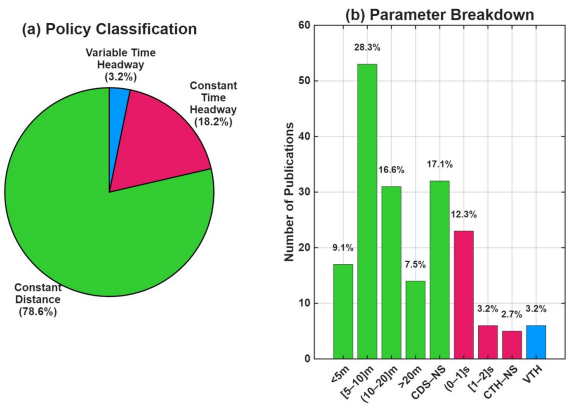


Figure 6. Classification: (**a**) adoption of the spacing policies; (**b**) quantitative distribution of the inter-vehicular distances and temporal headways.

Table 7. Comparison of spacing policies.

Policy	Property	Advantages	Disadvantages	Typical Use Cases and Feasibility
CDS	Fixed inter-vehicular distance, independent of velocity [125,164].	• High traffic capacity with minimal gaps [164]. • Boosts road capacity, cuts fuel use and energy [123,125].	• Unsafe at high speeds: risks hard braking or collisions [125].	• **Operating scenario:** Low to moderate speeds suitable for urban conditions (30–50 km/h) [226]. • **Values:** Short gaps (5–10 m) for light vehicle platoon; longer gaps (10–20 m) for heavy vehicle (trucks) platoons [227].
CTH	Distance increases linearly with platoon velocity [142,159,209].	• More collision resistant, i.e., gap grows with speed [151,188,194]. • Boosts string stability and disturbance rejection [151,159].	• Lower throughput at low speeds due to larger gaps [151].	• **Operating scenario:** Moderate–high speeds suitable for highway conditions (70–100 km/h) [228]. • **Values:** 0.6–1.0 s for homogenous platoons; 1.0–2.0 s for heterogeneous platoons [229,230].
VTH	Nonlinear distance function of speed, often with quadratic terms [83,189].	• More adaptive to changing traffic than CDS and CTH [83,189]. • Balances stability, capacity: small gaps low-speed, large high-speed [83,190].	• Challenge to design a controller due to complex spacing [83].	• **Operating scenario:** Flexible, capable of transitioning safely between urban and highway environments (0–100 km/h) [83]. • **Values:** Vary (dynamic headway), adjusted online for safety under diverse braking capabilities [190].

3.3. Information Flow Topology

In addition to the modelling of nodes and formation geometry, network modelling is the third component of the four-component framework for vehicle platooning. A representation of a vehicle platoon with N followers communicating across the network can be modelled by a graph $\mathcal{G}_N = (\mathcal{V}_N, \mathcal{E}_N)$, where $\mathcal{V}_N$ is the set of vertices, or nodes, and $\mathcal{E}_N$ is the set of directed edges, or arcs, inside the network. If the ith vehicle receives information from the jth vehicle, this is represented by the ordered pair (j, i) in the set $\mathcal{E}_N$. The adjacency matrix, $\mathcal{A}_N \in \mathbb{R}^{N\times N}$, is defined where its entries $\tilde{a}_{ij}$ are given by:

$$\tilde{a}_{ij} = \begin{cases} 1, & \text{if } (j,i) \in \mathcal{E}_N, \\ 0, & \text{otherwise,} \end{cases} \tag{22}$$

thereby depicting information being received by the ith vehicle from the jth vehicle with assumption that there is no self-looping, i.e., $\tilde{a}_{ii} = 0$. To account for communications with the lead vehicle, the graph $\mathcal{G}_N$ is augmented with a node 0 representing the leader. This gives a new augmented graph $\mathcal{G}_{N+1} = (\mathcal{V}_{N+1}, \mathcal{E}_{N+1})$, where $\mathcal{V}_{N+1} = \{0, 1, 2, \ldots, N\}$. The corresponding adjacency matrix is $\mathcal{A}_{N+1} \in \mathbb{R}^{(N+1)\times(N+1)}$. For the leader, the entries are such that $\tilde{a}_{0j} = 0$ for all $j = 0, 1, \ldots, N$, reflecting the assumption that follower vehicles do not transmit information to the leader. On the other hand, the followers receive information from the leader whenever $\tilde{a}_{i0} = 1$, otherwise $\tilde{a}_{i0} = 0$ for $i = 1, 2, \ldots, N$. The set of neighbours of the ith vehicle in $\mathcal{G}_{N+1}$ is denoted by

$$\mathcal{N}_i = \{\, j \in \{0, 1, \ldots, N\} \mid \tilde{a}_{ij} = 1 \,\}. \tag{23}$$

The Laplacian matrix of $\mathcal{G}_N$ is defined component-wise as

$$\mathcal{L}_{ij} = \begin{cases} -\tilde{a}_{ij}, & i \neq j, \\ \sum\limits_{\substack{k=1 \\ k \neq i}}^{N} \tilde{a}_{ik}, & i = j, \end{cases} \qquad i, j \in \{1, \ldots, N\}, \tag{24}$$

or more compactly,

$$\mathcal{L} = D - \mathcal{A}_N, \tag{25}$$

where $D = \text{diag}(d_1, \ldots, d_N)$ denotes the in-degree matrix and $d_i = \sum_{k=1}^{N} \tilde{a}_{ik}$ denotes the in-degree of node i. In addition, the information flow from leader to followers is modelled by a pinning matrix $\mathcal{P} \in \mathbb{R}^{N \times N}$, given by

$$\mathcal{P} = \text{diag}(p_1, p_2, \ldots, p_N), \tag{26}$$

where $p_i = 1$ if the edge $(0, i) \in \mathcal{E}_{N+1}$, meaning the vehicle i is pinned to the leader, and $p_i = 0$ if there is no direct connection. Combining it with the Laplacian, along with the pinning matrix, gives the Information matrix

$$\mathcal{H} = \mathcal{L} + \mathcal{P}. \tag{27}$$

The above information models the network topology for every platoon control literature, as detailed in [12,71,108,135,141,156], and many others. Besides the network modelling, the information flow between vehicles can be classified broadly into two categories: (i) undirected topology; (ii) directed topology.

3.3.1. Undirected Topology

In network modelling, the communication graph $\mathcal{G}_N$ is called undirected if information travels in both directions, i.e., from one node to another and back to the former. In other words, if a node i is able to send information to another node j, then j also should be able to send to i. Accordingly, the entries of the adjacency matrix $\mathcal{A}_N$ are defined as follows: if there is an edge (i,j), there is also an edge if it is reversed, i.e., (j,i). Mathematically, it can be written as:

$$\tilde{a}_{ij} = 1 \text{ and } \tilde{a}_{ji} = 1 \iff (i, j) \in \mathcal{E}_N. \tag{28}$$

It is conventional to assume there are no self-loops, i.e., $\tilde{a}_{ii} = 0$ for all $i \in \{1, \ldots, N\}$. As a result, both $\mathcal{A}_N$ and the corresponding Laplacian matrix $\mathcal{L}$ are symmetric matrices. Some examples of undirected topologies include the Bidirectional (BD) and Bidirectional Leader (BDL) configurations [16].

3.3.2. Directed Topology

A graph is called a directed graph, or digraph, $\mathcal{G}_N$, when the information travelling involves a one-way link. In other words, if a vehicle i can receive information from j, the existence of a return information path from i to j does not exist. As a result, both $\mathcal{A}_N$ and the corresponding Laplacian matrix $\mathcal{L}$ are asymmetric matrices, respectively. An entry $\tilde{a}_{ij} = 1$ means that there exists an edge pointing from node j to node i, while the vice versa is not true [231]. Common topologies for directed communication include Predecessor Follower (PF) and Predecessor Follower with Leader (PFL) [139]. A comparison between the two classes of topologies, i.e., undirected and directed, is given in Table 8.

Table 8. Comparison of directed and undirected topologies based on the provided literature.

Attribute	Directed	Undirected
Characteristics	• Information flow is one-way between vehicles (e.g., vehicle i receives from j, but j does not receive from i [12,72,98,102,107,110,156,174,205]. • Some examples include PF, PFL, TPF, TPFL, etc. [12,107,110,157].	• Information flow is two-way between connected vehicles, i.e., i receives information from j, then j also receives from i [117,128,174]. • Some of the examples include BD, BDL, BPF, BFL [110,128,157].
Mathematical Properties	• The Information matrix $\mathcal{H}$ is asymmetric [12,102,107,171]. • Eigenvalues of $\mathcal{H}$ can be either real and positive with multiplicity or complex in nature [12]. This asymmetry does not always decompose $\mathcal{H}$ into a scalar value [12,160].	• The Information matrix $\mathcal{H}$ is symmetric and positive definite [117,171,174]. • Eigenvalues of $\mathcal{H}$ are real and positive [12,117,174]. This symmetry allows for diagonalisation, resulting in decomposition of $\mathcal{H}$ into scalar value [117].
Scalability and Disturbance Propagation	• Disturbance propagation varies significantly with the type of directed topology. PF topology propagates disturbances downstream, potentially amplifying them. PFL, TPFL propagate disturbances upstream [107]. • Directed topology with leader improves disturbance propagation [107,181]. • Scalability depends on the magnitude of the eigenvalue, with the smallest eigenvalue signifying poor scaling [12]. Hence, PF is the least scalable.	• Disturbance propagation depends on the type of undirected topology. BD topology propagates disturbances downstream only [107]. BDL topology propagates disturbances both upstream and downstream [107]. • Scalability depends on the magnitude of the eigenvalue, with the smallest eigenvalue signifying poor scaling [12]. • Hence, BPF scales poorly, while BPLF performs better [12,42].

3.3.3. Preliminaries on Network Modelling

This subsection outlines the fundamentals of network modelling in vehicle platoon formation control.

(i). Spanning Tree

A spanning tree represents a subgraph which links all vertices without creating cycles [232]. In an undirected graph, the spanning tree has exactly $N-1$ edges for a graph that has N vertices, thus ensuring global connectivity [232]. For a directed graph, the directed spanning tree exists when at least one root node connects to all other nodes via directed paths [233]. The spanning tree is both necessary and sufficient for reaching consensus in distributed control frameworks [114].

(ii). Directed Acyclic Graph (DAG)

A directed acyclic graph or digraph is a connectivity graph that does not contain any cycle; thus, any path of directed edges from the starting vertex does not return to the same vertex [139].

(iii). Importance of Laplacian $(\mathcal{L})$ and Information Matrix $(\mathcal{H})$

In vehicular platooning networks, two matrices, the Laplacian matrix $\mathcal{L}$ and the Information matrix $\mathcal{H}$, play pivotal roles in determining the ability to attenuate the state errors, disturbance and providing the robustness of the platoon. For any connected graph, the smallest eigenvalue of the Laplacian satisfies $\lambda_1(L) = 0$; meanwhile, $\lambda_2(L)$, the second-smallest eigenvalue, is the Fiedler value, widely used as an indicator of algebraic connectivity and robustness of the network. A higher Fiedler value indicates greater connectivity and resilience to fragmentation [234]. The Information matrix $\mathcal{H}$ is related to the Laplacian matrix $\mathcal{L}$, as shown in (27). Analogously, the eigenvalue $\lambda_2(\mathcal{H})$ is an important measure of performance; a greater $\lambda_2(\mathcal{H})$indicates denser coupling, i.e., faster information exchange and a rapid convergence of states [39].

(iv). Influence of Information Flow Topology on Platoon Performance

The information flow topology fundamentally influences platooning performance by affecting the eigenvalue distribution of the closed-loop matrix, thereby impacting four key performance domains [39,233].

(v). a. Stability Margin

The stability margin is determined by the decay rate of the least stable closed-loop eigenvalue. Theoretical analysis shows that, in the case of topologies when all the followers are not connected to the leader such as Bidirectional (BD), the stability margin tends to zero, at a rate of $\mathcal{O}(1/N^2)$, as the platoon size N grows [233]. In contrast, topologies that use all follower leader connection information, such as Bidirectional–Leader (BDL), maintain a stability margin that is bounded away from zero independent of N, which is the number of vehicles, hence providing a higher quality of internal stability [17,233].

(v). b. Scalability

Scalability is related inherently to the algebraic connectivity of the communication graph, i.e., the smallest non-zero eigenvalue $\lambda_{\min}$. Topologies based solely on local interactions, without a leader-to-all-followers structure (e.g., PF or BD), have poor scalability because $\lambda_{\min}$ approaches zero as the fleet size increases, thereby amplifying errors along the string [16,233]. On the contrary, pinning control strategies, which utilise leader-to-all-followers information, maintain a strictly positive $\lambda_{\min}$, effectively inhibiting the propagation of errors and improving scalability [16].

(v). c. Convergence

The rate of convergence of the spacing errors to reach the consensus trajectory is determined by the algebraic connectivity of the communication graph [39]. A topology that has a greater minimum non-zero eigenvalue ($\lambda_{\min}$) will result in a higher rate of convergence. Importantly, it has been demonstrated that utilising information from both directions (as in BD topology) can reduce $\lambda_{\min}$ compared to directed (unidirected) topologies, thereby slowing convergence [39]. However, topologies with pin followers on the leader (such as BDL) have significantly faster convergence than unpinned topologies [17]. Additionally, increasing the number of information inputs, such as in a Two Predecessor Follower (TPF) topology leads to an improved eigenvalue distribution compared to the standard PF topology, thereby accelerating convergence [39].

(v). d. Robustness

Robustness against external disturbances is measured in terms of the *zeta*-gain (and is the $\mathcal{H}_\infty$ norm from disturbances to spacing errors). Analytical derivations have established that the value of zeta is lower bounded by $\zeta \geq 1/(\lambda_{\min}\sqrt{k_p})$ where k_p denotes the position feedback gain [16]. Consequently, topologies with weak connectivity or no leader-to-all-followers connectivity, e.g., BD, suffer from severe disturbance amplification, where the error energy increases quadratically $\mathcal{O}(N^2)$ [16] with platoon size. By pinning the followers to the leader, this amplification is greatly reduced into a linear growth trend ($\mathcal{O}(N)$), indicating direct leader information as a primary cause for the limitation of disturbance propagation [16].

3.3.4. Network Communication Issues in Vehicle Platooning

(i). Ideal Network Communication

Vehicle platooning, if assumed to have ideal network communication, shows the inter-vehicular communication as perfect, and the information exchange is instantaneously devoid of any delay, packet losses or topological changes [87,109]. Consequently, every vehicle gets a full and precise state information from its neighbours [87,109,235].

(ii). Time-Varying Delay

Time-varying delay refers to a latency in the communication network that varies over time. It is caused by network congestion and channel variability [87,235]. These delays are modelled as functions of time and incorporated into the control law to address the lag in retrieving neighbouring-state information [235,236]. When the time-varying delays are heterogeneous in nature, each communication link (j, i) experiences a distinct $\phi_i(t)$ (continuous time) or $\phi_i(k)$ (discrete time) delay which is measured as the delay from vehicle j to vehicle i [236–238]. This heterogeneity arises from different network conditions across communication links [138,237]. On the other hand, a homogeneous time-varying delay implies a uniform delay over all the links (i.e., $\phi_i(t) = \phi_j(t) = \phi(t)$) [13,236].

(iii). Time-Constant Delay

A fixed or time-constant delay represents network latency that is constant and independent of time. These types of delays are modelled as constant ϕ and incorporated in the control law [109,239]. Heterogeneous time-constant delays attribute a distinct constant ϕ_i to each link, thus representing the fact that vehicle i receives information with a link-specific fixed lag [238,240]. On the other hand, a homogeneous time-constant delay ϕ is assumed, that each of the links maintains the same and constant delay across the vehicles of the platoon (i.e., $\phi_i = \phi_j = \phi$) [109,239].

(iv). Topology Disturbance

When the communication topology across the vehicle platoon network is not constant but keeps on changing, this is known as topology disturbance, which is grouped into: (iv.a) switching topology and (iv.b) time-varying topology.

(iv). a. Switching Topology

If the vehicles in a vehicular platoon network experience changes in the communication topology at discrete time instants during communication, this is referred to as a switching topology [241,242]. This phenomenon is also perceived as a disturbance in the topology [70]. The switching topology is modelled as a finite set of topology graphs $G_\emptyset = \{G_1(t), G_2(t), \ldots, G_l(t)\}$. At any time t, the current topology $G(t)$ belongs to $G_\emptyset$. Defining a time function $\rho(t) : [0, +\infty] \to \mathbb{N}$, where $\mathbb{N} = \{1, 2, \ldots, l\}$, the current topology is [93,241]:

$$G(t) = G_{\rho(t)}, \quad G_{\rho(t)} \in G_\emptyset, \tag{29}$$

where

$$G_\emptyset = \bigcup_{i=1}^{l} G_i. \tag{30}$$

This randomness is represented by a Markov chain [209]. A Markov chain assumes that the conditional distribution of the next network state depends only on the current state. Each topology (for example, G_1, G_2) represents a different state of the Markov chain, and transitions are determined by a transition matrix that depends solely on the current topology [209]. However, in practice, the transition probabilities for such Markov chains are often not known a priori, thereby necessitating conservative designs when worst-case scenarios are presumed. To overcome the above conservatism, control schemes have been designed that either allow the use of partly unknown transition matrices or rely on dwell-time conditions rather than explicitly using stochastic models [243,244]. Alternatively, situation-aware or adaptive switching can be implemented, in which the active topology is updated online in response to triggering events, such as cut-ins or changes in communication quality, which may lead to network communication limits arising from under- or over-defined statistical assumptions [241,242].

(iv). b. Continuous Time-Varying Topology

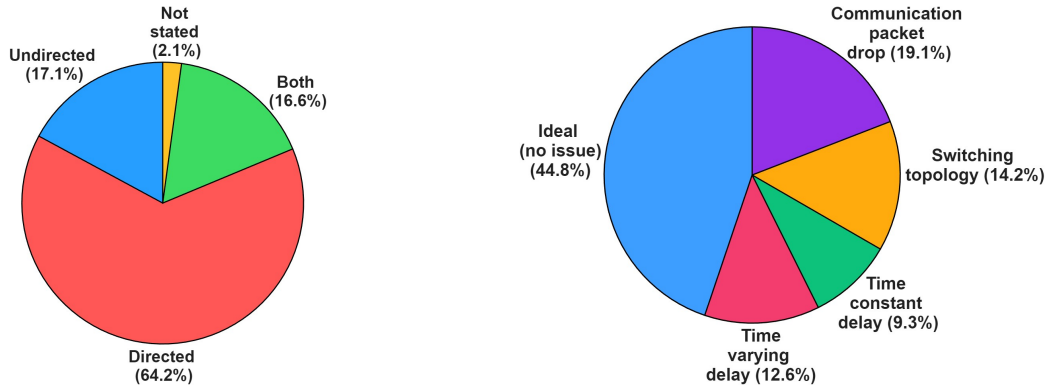
Another topology disturbance is a continuous-time-varying topology. This can be modelled as a network that evolves continuously over time. In this framework, the interactions among vehicles are modelled by a time-varying adjacency matrix $\mathcal{A}_N(t)$, resulting in a time-varying graph Laplacian matrix $\mathcal{L}(t)$ [93,209]. This dynamic change in topologies occurs whenever communication links between vehicles are created or lost as vehicles move in or out of the communication radius [75]. This results in the entries of $\mathcal{A}_N(t)$ changing in response to the real-time position. Moreover, this framework also handles uncertain communication topologies with unknown network structure [209].

(v). Communication Packet Drops

A communication packet drop occurs when data transmissions do not reach, or only partially reach, the target node. Such losses are usually caused by channel fading, external interferences, collision-induced errors and buffer overflows [109,245–248]. Communication packet drop is often modelled by a Bernoulli distribution. A binary random variable $\theta_{ij}(k) \in \{0, 1\}$ denotes the reception status on link (j, i) at time k [246,248,249]. Let $\theta_{ij}(k) = 0$ indicate successful reception with probability p and $\theta_{ij}(k) = 1$ means packet loss with probability $1 - p$. Here, p is assumed to be the packet reception rate [138,247,248]. Hence, the expected packet loss rate is given by $\mathbb{E}(\theta_{ij}(k)) = 1 - p$ [138,249,250].

Table 9 presents a classification of the literature on a systematic selection criteria based on the definitions introduced above in Sections 3.3.1–3.3.4. The taxonomy follows a hierarchical protocol: first, the basic network topology is identified as being either undirected or directed or both (controllers made for both types of topology), according to the graphical properties of the network provided by Sections 3.3.1 and 3.3.2. Second, the literature is subclassified based on the network issues/imperfections incorporated in the synthesis of the controllers. These imperfections include ideal communication (for the sake of classification, this category has been included under imperfections), constant or varying delays, topology disturbances (switching/time-varying), and packet drops, according to the canonical definitions shown in Section 3.3.4. The resulting trends are illustrated in Figure 7. Figure 7a indicates that directed topologies receive the majority of attention (64.2% of the publications), followed by undirected configurations. Figure 7b reveals that although ideal communication assumptions remain prevalent (44.8% of the publications), a significant proportion of research addresses communication time delays (approximately 22% of the publications), followed by packet drops and switching topologies.

The distributed controller classes surveyed in Section 4 for vehicle platoons make use of this synchronisation theory of MAS, which is articulated in Appendix A.



(**a**) Network topologies. (**b**) Challenges related to network topologies.

Figure 7. Overview of platoon communication aspects: (**a**) network topologies; (**b**) challenges related to information flow and network topology.

Table 9. Topology type and communication issues. Publications that involve multiple forms of communication topologies and/or communication issues are marked with an asterisk (*).

Type	Issues		References
Undirected	Ideal (no issues)		[16,54,62,64,99,130,146,164,168,170,176,177,188,195,208,215,219] [71] *
	Time-varying delay	Heterogeneous	[18,161,202]
		Homogeneous	[131,169]
	Time Constant delay	Homogeneous	[40,210] [66] *
	Switching Topology/Time Varying Topology		[93,127]
	Fading channel/Packet drop		[15,17,179,203] [71] *
Directed	Ideal (no issues)		[43,45,47,50,51,59,60,72–74,77,79,82,86,88,90,96,98,100–104,116,123,132,133,139,142,145,152–154,158,172,189,193,194,196,197,200,201,205,206,212,214,216,218,220,222,223]
	Time-varying delay	Heterogeneous	[85,89,106,121,140,147,163,166,167,184,190,192] [84] *, [78] *, [144] *
		Homogeneous	[87,94,105,221] [136] *, [108] *, [109] *, [138] *, [49] *, [83] *, [185] *, [52] *
	Time Constant delay	Heterogeneous	[61,113,165]
		Homogeneous	[38,76,115,135,159,178,211] [173] *, [42] *
	Switching Topology/Time Varying Topology		[14,55,67,70,75,91,92,119,134,137,149,155,162,175,180,187,204,209] [49] *, [217] *, [83] *, [78] *, [144] *
	Fading channel/Packet drop		[36,37,53,63,112,122,125,148,160,182,191] [136] *, [108] *, [109] *, [138] *, [173] *, [217] *, [84] *, [42] *, [185] *, [52] *
Both	Ideal (no issues)		[12,39,81,95,107,128,141,143,171,174,181,183,186,207]
	Time-varying delay	Heterogeneous	[150] [110] *, [41] *, [124] *
		Homogeneous	[13] [156] *
	Time Constant delay	Heterogeneous	[111,151]
		Homogeneous	[114,126] [118] *, [198] *
	Switching Topology/Time Varying Topology		[120,199]
	Fading channel/Packet drop		[117,157,213] [110] *, [118] *, [41] *, [124] *, [198] *, [156] *
Not Stated	—		[65,80,97,129]

4. Distributed Control Strategies

This section reviews distributed control strategies for vehicle platoon formation and stable operation, thereby completing the fourth component of the four-component platoon control framework presented in Section 3. The end of this section, i.e., Section 4.13, presents the discussion which includes the trends in the use of the various distributed controllers discussed, as well as the preferred choice of different controllers for different challenging situations.

Fundamentally distributed control algorithms can be divided according to their dependence on time into the following categories: (i) continuous time; (ii) discrete time; (iii) event-triggered.

4.1. Continuous Time

A continuous-time distributed control law (commonly denoted as $u_i(t)$) describes the control input as a continuous function of time t, where $t \in \mathbb{R}$. An example of one

continuous distributed control law in the area of vehicle platooning can be found in [174]. The others are given in Table 10.

Table 10. Distributed controllers for vehicle platoon formation. Publications with an asterisk (*) indicate the use of a hybrid strategy or multiple control approaches to improve platoon performance.

Control Type	Continuous Time	Discrete Time	Event-Triggered
Linear State Feedback	[12,39,40,70,76,77,87–89,92,107,108,110,111,113–115,119,120,126,128,129,137–139,142,152,158,159,162,165,181], [38,42,61,79,85,116,122,135,145,164,195,197] *	[49,71,109,112,118,136,140,170,186], [42,64,117] *	[36,52,63,78,93,132,161,179,182], [187] *
$\mathcal{H}_\infty$ Control	[164,176] *, [13,16,17,157,184]	[15,156], [121] *	[14,169]
Adaptive Control	[86,121,154,155,178] *, [41,50,59,60,82,124,141,143,144,163,173,174,180,206,208]	[101,202] *	[205] *, [37,160,171,172,212,220,223]
Model Predictive Control	[45]	[18,51,65,66,80,84,90,91,106,123,127,134,177,198,213,222]	[148,185,191,221]
Sliding-Mode Control	[86,95,98,103,196] *, [47,97,99,100,102,146,151,189,190,192,199,204,207]	[188], [200] *	—
Observer Based Controller	[38,61,95,98,103,116,122,135,150,154,155,166,176,178,183,195–197] *, [125]	[42,79,101,117,200,202,203] *	[64,187,211] *
AI Control	[54,62,133,153,193,209]	[67,203] *, [53,55,201,216]	[205] *, [215]
Nonlinear Control	[73–75,83,104,147,149,167], [85,145,150] *	—	[72,131,210]
Other Control approaches	[43,81,105,168], [183] *	[67] *, [217,219]	[94,130], [211] *

4.2. Discrete Time

A discrete-time distributed control law (commonly denoted as $u_i[k]$) describes the control input as a discrete function of time. This means that the control input is computed not continuously but at uniform intervals indexed by k, where $k \in \mathbb{Z}$ and expressed as $u[k] = u(kT_s)$, where $T_s = 1/f_s$ represents the sampling period. An example of a discrete-time distributed control law is presented in [66]; further examples are in Table 10.

4.3. Event-Triggered

Event-triggered control computes control actions only when predefined conditions are met [251]. It does not generate control signals continuously, as in continuous-time control, nor does it sample at fixed time intervals, as in discrete-time control. As a result, it significantly reduces communication load, bandwidth usage, and energy consumption, making it very appropriate for resource-constrained networked environments [251,252]. The main caveat lies in the Zeno phenomenon. In this phenomenon, the control signal is triggered too rapidly over a finite time horizon, which can jeopardise the stability and long-term sustainability of system resources. Hence, the design of triggering policies needs to ensure a strictly positive minimum inter-event interval. This, in turn, prevents superfluous updates [252,253]. The development of event-triggered control is discussed in the subsequent paragraphs.

4.3.1. Design of Event-Triggered Scheme

In the conventional event-triggered control architecture used in vehicular platooning, as shown in Figure 8, the sensors of every vehicle continuously monitor the state variables of the vehicle. An event generator takes this sensory data and determines if a given triggering rule has been met. Typically, such a rule is based on the difference between the vehicle's instantaneous state and its last broadcast state. Upon reaching the criterion, the event generator notifies the relevant vehicles of the updated state info, which, in turn, recompute

their control inputs. Between triggering events, vehicles use local state estimators to predict their neighbours' states, maintaining coordination while reducing communication burden. A typical condition triggering event is formulated [253] as follows.

$$t_{k+1}^{i} = \inf\{\, t > t_{k}^{i} : f_i(e_i(t), x_i(t), t) \geq 0 \,\}, \tag{31}$$

where t_k^i is the time of the latest event for vehicle i. The triggering function $f_i(\cdot)$ depends on the vehicle state $x_i(t)$ and estimation error $e_i(t) = x_i(t) - x_i(t_k^i)$, measuring the difference between the real state and the last communicated state. The function f_i may include performance measures such as error norm, adaptive thresholds, and time-dependent parameters, ensuring triggering occurs only when necessary. Event-triggered strategies are further classified as discussed next.

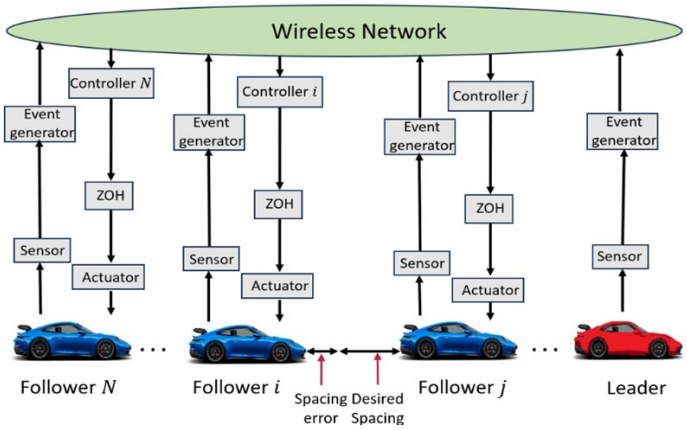


Figure 8. Event-triggered control architecture for vehicular platoon.

4.3.2. Static Event-Triggered Control (SETC)

In static event-triggered control, events are triggered according to predefined criteria. Vehicle i updates its state and sends this information to its neighbours whenever the state error function fulfils a constant threshold, i.e., $\|e_i(t)\| \leq c$ [220]. A standard error function used in the SETC definition is given in (1) [169]. Although SETC is easy to implement, its conservativity may lead to unnecessary communication because it cannot adapt to fluctuations in underlying network conditions.

4.3.3. Dynamic Event-Triggered Control (DETC)

Dynamic event-triggered control (DETC) enables higher-level adaptability in dynamically tuning the triggering threshold conditions based on evolving system states and prevailing network constraints. This is achieved by using a threshold parameter, which is modelled to meet the requirements. In a recent study [169], a DETC approach with a bandwidth-aware threshold parameter, dynamically adjusted in response to real-time network traffic conditions. The threshold parameter is calculated as a weighted sum of two dynamic components: one for low network traffic and one for high network traffic. The weighting factor, which ranges from 0 to 1, is adjusted based on bandwidth availability. This approach significantly reduces the number of data transmissions while ensuring the required control performance.

A systematic categorisation of the literature is presented in Table 10 based on the definitions defined in Sections 4.4–4.12. The classification involves a systematic approach. First, the main controller architecture is determined starting from distributed linear control (DLC) to artificial intelligence (AI)-based methods by consulting the theoretical formulations detailed in the corresponding subsections (for example, Sections 4.4–4.6, etc.). Second, the domain of controller implementation is identified as either continuous-time, discrete-time,

or event-triggered, as per the definitions provided in Sections 4.1–4.3. Publications with an asterisk (*) indicate studies that use a hybrid strategy or multiple control approaches to improve platoon performance. Similarly, all of the 'observer-based controllers' are asterisked. This is because observers are typically implemented as auxiliary state estimators within a control approach. The statistical development of such strategies can be seen in Figure 9. A comparative analysis of the 2013–2020 and 2021–2025 periods shows a major shift in the paradigm, with the use of linear control strategies falling by approximately half, along with a corresponding decrease in H_∞ control. On the contrary, observer-based, MPC, and AI-based controllers have shown significant growth in their share of publications, along with increased interest in adaptive and SMC approaches. The following subsections provide detailed overviews of these strategies, focusing on the specific ways they address node dynamics and information topology issues.

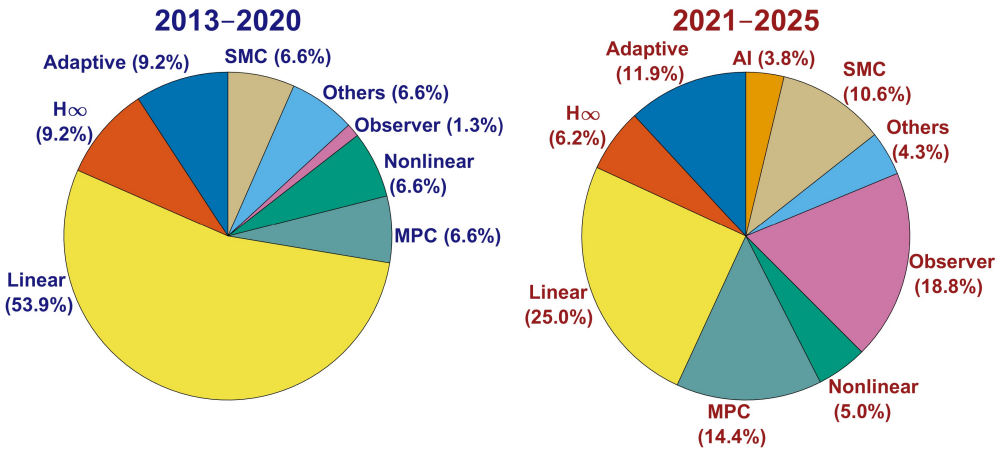


Figure 9. Publication share by controller type: 2013–2020 (**left**) vs. 2021–2025 (**right**).

4.4. Distributed Linear Control (DLC)

Linear control strategies work well for processes whose dynamics can be modelled or approximated by a linear model. In other words, the systems which follow the principles of homogeneity and superposition [254,255]. When this methodology is extended to multiagent systems, distributed linear controllers (DLCs) are obtained. A general form of a distributed feedback linear control law is presented in [12,71,88] as follows:

$$u_i(\cdot) = -\left(\sum_{j=1}^{N_i} \tilde{a}_{ij}\, K\big(x_j(\cdot) - x_i(\cdot) - d_{ij}\big) + p_i\, K\big(x_0(\cdot) - x_i(\cdot) - d_{i0}\big)\right). \tag{32}$$

where K represents a vector of constant gains with the argument $(\cdot)$ signifying that the signals involved can be continuous, discretised or event-triggered. A basic approach to stability analysis involves decomposing the closed-loop controlled dynamics into N characteristic polynomials, where N is the number of vehicles in the platoon. Accordingly, the closed-loop dynamics of the full platoon can be written in a matrix form as: $\dot{X} = (I_N \otimes A - H \otimes BK)X$. For successful decoupling of the system dynamics, the coupling Information matrix $\mathcal{H}$ must have real, distinct eigenvalues. This requirement is commonly met for undirected topologies [71], enabling the system to be reduced to N independent scalar eigenvalues through diagonalisation ($\mathcal{H} = V\Lambda V^{-1}$). If $\mathcal{H}$ has repeated and real eigenvalues or $\mathcal{H}$ is a matrix of the directed topology having complex eigenvalues, then the procedure is detailed in [12]. The following section of DLC is divided into three parts based on the problem it solves: (i) ideal node with ideal communication; (ii); ideal node with communication imperfections; (iii) external disturbance on the node with communication imperfection.

(i). Ideal Node with ideal communication

Early investigations of the DLC, applied to platoons, were conducted under ideal conditions, which means no disturbance to node dynamics, as well as perfect communication. It provided important baseline research. A comparative analysis of six standard topologies using state variable feedback control was performed in [107]. The controller parameters were fine-tuned by using Riccati-based equations. However, the realism of this study was confined to homogeneous platoons. Similarly, ref. [129] performed a simulation-based evaluation of five control algorithms in a platoon simulator, i.e., the Plexe environment, and identified the 'Flatbed' algorithm as the best for disturbance rejection. This work was limited to homogeneous platoons and did not include stability analysis. The authors in [12] derived stability conditions for topologies with arbitrary complex eigenvalues, which allowed analysis of both directed and undirected graphs. They, however, assumed the platoon as homogeneous. In dealing with different topological structures (PF, LPF, TPF, LTPF, BPF and TBPF), ref. [39] designed a modified linear quadratic regulator (LQR) controller for heterogeneous platoons. The work optimised the controller gain as well as the traction matrix for uniform convergence. To account for the heterogeneity of the platoon, a proportional integral (PI) controller was proposed in [88] as a means of mitigating the nonlinear drivetrain dynamics, without requiring feed-forward compensation. Stability conditions were deduced by applying the Routh–Hurwitz criterion. Expanding on the heterogeneous analysis, the work in [139] utilised the structure of directed acyclic graphs (DAGs) topologies to achieve a scalable and decoupled stability analysis based on the internal model principle. Controller gains were calculated from individual Algebraic Riccati Equations (AREs). Further developing DLC for heterogeneous platoons, a predictive spacing strategy was proposed in [142], where the velocity profiles of neighbouring vehicles were used to dynamically adjust inter-vehicular spacing and thereby reduce the risk of collisions. The stability analysis was effectively demonstrated by a Bilinear Matrix Inequality (BMI). The authors in [114] proposed a new dual-mode controller, designed for platoons with velocity constraints, where the operation of the platoon was maintained by using a virtual reference (lead) vehicle. Asymptotic stability was proven using Routh–Hurwitz conditions; however, the analysis assumed a homogeneous platoon. To mitigate the computational cost of event-triggered schemes, ref. [93] proposed a self-triggered strategy that predicts the next update time, thereby eliminating the need for continuous state monitoring.

(ii). Ideal Node with Communication Imperfections

Several research efforts have been targeted at devising controllers that are robust against communication imperfections while not accounting for external disturbances on node dynamics. Based on the Riccati-based techniques, a DLC incorporating heterogeneous communication delays was developed in [113]. The algorithm was scalable and was independent of the platoon size; however, its theoretical delay bound was conservative. In related research, a platoon operating under switching topologies was considered in [70], where stability was guaranteed by using an average dwell-time constraint. The synthesis method also showed platoon-size independence; however, the results were very conservative in the stability condition. Again, when dealing with switching topologies, ref. [119] proposed a dual-objective control approach that minimises energy consumption while maintaining stability. However, the study was limited to homogeneous platoons. Asynchronous control is another key area of research. An asynchronous consensus-based protocol was developed in [140], allowing each vehicle to update its state using a locally generated clock. An important benefit of such a method was its robustness to arbitrarily bounded time-varying delays and switching topologies. Nonetheless, the controller used fixed gains rather than adaptable ones. It constrained the ability to change the performance. Stability thresholds for platoon working in analogue fading channels were theoretically

analysed in [71]. A more realistic scenario was addressed in a study [117]. Here, observers were employed as auxiliary units accompanying linear state feedback communications with non-identical communication packet drops. In [89], a formation control framework for heterogeneous platoons was designed with a nonlinear spacing policy and heterogeneous time-varying communication delays. However, the shortcoming was that the resulting LMI-based stability conditions were conservative. The work in [40] proposed a distributed PID controller that addresses communication and input delays. It computed exact time delay bounds; however, this was again assumed to be a homogeneous platoon. The computational burden of developing the distributed controller was addressed in [126] by developing an analytical method to find the most stringent eigenvalue of the Laplacian matrix, which significantly reduced computation time. In another work, to leverage delay as a control parameter, a proportional retarded (PR) controller was proposed in [115], where delay constructively replaced the derivative term. The work in [110] proposed a control strategy that treated the communication topology as a design variable to address network delays and packet losses. Resource-aware topology switching was addressed in [137], enabling vehicles to adjust their communication links when a state change occurred. The effects of distance-dependence of the packet loss were considered in [138], and probability-based models of the range limitations and packet losses were used in [109,112]. Joint effects of delays and dropouts were studied, and mean-square stability conditions were established in [118]. In another work, feedback linearisation was applied to nonlinear systems, and an adaptive gain was developed to address network imperfections in [136]. Other researchers addressed heterogeneous time delays, speed-prediction models to reduce the impact of delay effects [87], and DLC based on LMI tuning for uncertain links in [108].

(iii). External Disturbance on the node with Communication Imperfection

Addressing the issue of external disturbances acting on the node and network imperfections, the authors in [49] considered sampled-data control with Markovian switching topologies, under the presence of communication delays and external disturbances. They employed the Lyapunov–Krasovskii functional to derive LMI-based stability conditions. To deal with situations where limited state information is available, [170] proposed a static feedback controller based only on sampled position information; however, the analysis was limited to the case of homogeneous dynamics. A new leader-selection strategy was proposed in [158] in order to improve the disturbance rejection ability by enabling the followers to adjust their leader dynamically. This concept was further extended in [165] to deal with the case where vehicles were completely disconnected from the platoon. In an additional piece, for systems characterised by heterogeneous nonlinear dynamics and uncertainties, a robust PID-inspired controller was designed in [92]. It also addressed the issue of switching topologies. Dealing with random additive communication noise, the research in [36] proposed a time-varying consensus gain with an event-triggered mechanism. Further, extending event-triggered control under switching topologies, ref. [78] introduced time-varying delays and external disturbances, utilising a Lyapunov–Krasovskii method to provide less conservative stability conditions. These event-triggered approaches, however, were limited to homogeneous platoons. The work in [126] developed an analytical method for calculating the exact delay margins required to maintain platoon stability. Table 11 categorises the published literature on DLC for vehicle platoons. The table divided the papers by the methodology used to develop the controller parameters. The first category represents methods based on solving the Riccati equation. The second category studies the application of the Routh–Hurwitz criterion to determine the stable bounds of the controller parameters. The third most often cited category comprises approaches that use Lyapunov stability theory to formulate and solve linear matrix inequalities (LMIs). The last cate-

gory collects alternative techniques that do not belong to the above categories, including graphical methods and strategies based on eigenvalue placement.

Table 11. Design of parameters of distributed linear controllers.

Methodology	References
Solution of Riccati Equation	[12,39,70,107,113,116,117,119,139]
Routh–Hurwitz Criterion	[40,71,88,109,114,115,137,159,165]
Lyapunov Stability Solution of LMI	[49,78,87,92,108,110–112,120,136,138,158,170]
Other methods	[36,126,140,142]

4.5. Distributed H_∞ Control (DHC)

The H_∞ control framework is a robust control strategy for the synthesis of controllers that possess the ability to mitigate disturbances and uncertainties [16,256]. Its aim is to minimise the worst-case gain over the whole frequency range, thus ensuring that unwanted excitations, such as sensor noise or external perturbations, are not amplified. This worst-case gain is measured in terms of the H_∞ norm, which for a system whose transfer function is called $G(s)$ is given by its peak gain at all frequencies:

$$\|G(s)\|_\infty = \max_\omega \left[\zeta_{\max}(G(j\omega))\right], \tag{33}$$

where $\zeta_{\max}(G(j\omega))$ represents the maximum singular value at a given frequency ω [16]. Consequently, for any disturbance $\epsilon_i(t)$ with finite energy, the output of the system $g(t)$ is such that:

$$\int_0^\infty |g(t)|^2\, dt \;\le\; \gamma^2 \int_0^\infty |\epsilon_i(t)|^2\, dt, \tag{34}$$

where γ denotes the H_∞-norm. Minimisation of γ therefore ensures better disturbance rejection [16,256]. In the architecture of the distributed H_∞ control (DHC), each vehicle uses a local controller and exchanges the state information with its neighbours. Mathematically, DHC is expressed as:

$$u_i(t) = -\sum_{j=0}^{N} \tilde{a}_{ij}\, K_{H_\infty}\big(x_j(t) - x_i(t) - D_{ij}\big), \tag{35}$$

where $K_{H_\infty} = (K_s, K_v, K_a)$ is the space of H_∞ control gains. Similar to the DLC (in the earlier subsection), the design problem is either decomposed into independent single-vehicle modes (if the eigenvalues of $\mathcal{H}$ are real and distinct) or is treated as the lumped vector of the full closed loop platoon system (if the eigenvalues of $\mathcal{H}$ are real and repetitive or complex in nature) through the spectral analysis of the communication graph matrix. The objective is to find a gain vector K_{H_∞} that minimises the amplification from disturbances ($\epsilon_i(t)$) to tracking errors (e), expressed as $\min_{K_{H_\infty}} \|T_{e,\epsilon}(s)\|_\infty$. This optimisation can be recast into a system of linear matrix inequalities (LMIs), which can be easily solved by standard convex optimisation techniques [16,17,164]. The process of applying DHC to vehicle platoons in the literature can be broadly categorised into two categories: (i) ideal node with communication imperfections; (ii) external disturbance on the node with communication imperfections.

(i). Ideal Node with Communication Imperfection

A DHC approach for handling random packet drops was proposed in [157]. In that work, packet loss was modelled by Bernoulli random variables, and LMI conditions were obtained to ensure H_∞ performance. However, the analysis was restricted to homogeneous platoons as well as to single packet drop events. A later contribution in [15] expanded the discrete-time framework to analytically derive a lower bound for the H_∞-norm to prove the robustness of the platoon system in the presence of random packet drops. Expanding this problem, multiple consecutive packet drops were tackled in the work of [17] using a

Lyapunov–Krasovskii-based approach to LMI. Shifting the focus to the communication resource management, a dynamic event-triggered H_∞ control approach was proposed in [169]. This strategy used a bandwidth-aware mechanism to adapt the data transmission based on the states of the vehicles and network load. Building on the latter work, another study [14] introduced random switching topologies and multi-spacing policies into a dynamic event-triggered framework, utilising Markov chains to account for the randomness of the topology.

(ii). External Disturbance on the node with Communication Imperfection

DHC is especially well-suited to deal with external disturbances (e.g., wind gusts) simultaneously with the network imperfections, as shown in the works [17,156]. The combined issue of external disturbance with packet drop and communication delay was addressed in [156] using an H_∞ consensus controller; however, the analysis used homogeneous platoons. To simultaneously cope with the problems of parameter uncertainties and communication imperfections, a distributed adaptive robust H_∞ control scheme was designed in [184]. This approach successfully dealt with heterogeneous time-varying delays; however, the stability proof of this method was limited to platoons with homogeneous vehicle dynamics. Extending the H_∞ theory to a heterogeneous platoon that has time-varying delays and uncertainties, a fixed-time observer-based sliding-mode controller was proposed in [13]. This design ensures fixed-time convergence and overall robustness irrespective of the initial conditions. Table 12 categorises $\mathcal{H}_\infty$ control methods for vehicle platooning according to the challenges addressed. The first category includes LMI-based $\mathcal{H}_\infty$ designs that guarantee robustness to disturbances and modelling uncertainties, assuming the communication is perfect. The second category includes LMI formulations for a vehicle platoon under network imperfections, primarily random packet drops. The third category focuses on robustness against communication delays combined with external disturbances acting on the nodes. Finally, the last category comprises literature on robust event-triggered control mechanisms.

Table 12. H_∞ control methodologies in vehicle platooning.

Methodology	Focus On	References
LMI-Based H_∞ for Disturbances	Focuses on robustness against model uncertainties and external disturbances, assuming ideal communication.	[16,164]
Stochastic LMI-Based H_∞ for Network Imperfections like packet drops	Explicitly models random network imperfections like packet drops using stochastic variables and aims for H_∞ performance.	[15,17,156,157]
Robust H_∞ for Communication Delays and Uncertainties	Specifically handle significant time-varying delays alongside model uncertainties and disturbances.	[13,184]
Event-Triggered H_∞ Control for Resource Efficiency	Focuses on optimising communication resource usage by deciding when to transmit data.	[14,169]

4.6. Distributed Adaptive Control (DAC)

To mitigate uncertainties in system behaviour, parameter variations, and nonlinear dynamics, adaptive control is employed. This strategy belongs to a class of control strategies that dynamically modify the controller's parameters in real time. The fundamentals of an adaptive control law comprise an adaptive law and an adaptation mechanism. The control law specifies the controller architecture and the interdependencies among its parameters, while the adaptation mechanism recalibrates the parameters to achieve the objective

closed-loop performance. Various adaptation mechanisms exist, including gradient-based methods, least squares algorithms, Lyapunov-based adaptation laws, and neural network techniques. In practice, these mechanisms are often used in adaptive frameworks such as model reference adaptive control (MRAC) or self-tuning regulators (STRs). In such examples, the design of the adaptation law aims to ensure convergence and system stability [257]. For distributed adaptive control (DAC) in vehicle platoons, these adaptive control principles are extended, with each vehicle using a local adaptive controller to address uncertainties introduced by aerodynamic drag, communication delays, nonlinearities, and external disturbances, thereby achieving performance, stability, and fuel economy. Mathematically, the DAC input is given as:

$$u_i(t) = -\Big(\sum_{j \in \mathcal{N}_i} \tilde{a}_{ij} K_{ij}^{\top}(t) \left(x_j(t) - x_i(t)\right) + p_i K_{i0}^{\top}(t) \left(x_0(t) - x_i(t)\right) \Big), \tag{36}$$

where $x_i(t)$, $x_j(t)$, and $x_0(t)$ are the states of the ith, jth, and leader, respectively. The terms $K_{ij}^{\top}(t)$ and $K_{i0}^{\top}(t)$ are the adaptive gain parameters, which are updated with some adaptation mechanisms. The distributed adaptive robust control employed an eigenvalue decomposition method with parameters updated according to an adaptation law proportional to squared tracking errors. The implementation of DAC to vehicle platoons in the literature can be broadly categorised into three categories: (i) ideal node with ideal communication; (ii) ideal node with communication imperfection; (iii) external disturbance on a node with communication imperfection.

(i). Ideal Node with Ideal Communication

This paragraph surveys the paper, focusing on the DAC when there is no disturbance at the node, and communication is considered perfect. A DAC was developed in [220], where the weights of adaptive coupling were updated by squared error-based laws, and the inter-vehicular communication was initiated via an event-triggered mechanism when a certain triggering condition was satisfied. This approach eliminated Zeno behaviour while maintaining system stability. A fully self-organising platoon was demonstrated in [141], where the control gains and communication topology were adapted online. This was accomplished using a dual mechanism of adaptation: control gains were adjusted using an error-driven gradient law coupled with a σ-modification for boundedness, while adjacency matrix weights were based on a potential-field rule to reduce communication load. To handle the heterogeneity in platoon with nodes having nonlinear dynamics, a two-layer adaptive control scheme was presented in [143]. This approach utilised feedback linearisation as the first step to linearise the vehicle's dynamics, followed by the DAC protocol. An adaptation law proportional to the state tracking errors tuned a parameter to compensate for vehicle heterogeneity. Another study on heterogeneous nonlinear platoons in [59] proposed a DAC based on a PID algorithm for leader tracking. It had one main merit: the easy implementation of the adaptive PID without the need for observers. The controller parameters were adjusted using adaptation laws proportional to the squared local consensus errors, with a sigma-modification, thereby ensuring parameter boundedness.

(ii). Ideal Node with Communication Imperfection

DACs are more useful when there is some unwanted element in the system; hence, this paragraph deals with the implementation of DAC in the presence of imperfect communication. A distributed adaptive strategy for coping with multiple heterogeneous time-varying communication delays in vehicle platoons was formulated in [41]. The adaptive gains in this work were adjusted by a law proportional to the square of the errors. A DAC protocol for heterogeneous platoons with both switching topologies and time-varying delays was given in [144]. Its main contribution was a delay-dependent approach. They demonstrated

the stability of the control framework using the Lyapunov–Krasovskii method. To reduce the level of conservatism delays with a lower bound that was not zero was taken into consideration. Instead of relying on an online adaptation law, this protocol utilised fixed controller gains, which were computed offline using an iterative LMI-based algorithm to ensure stability.

(iii). External Disturbance on node with Communication Imperfection

Taking into account both external disturbances and communication imperfections, the authors in [160] modelled the external disturbances at each node and devised a periodic event-triggered mechanism. This method incorporates communication failures to avoid frequent switching between cooperative and standard adaptive cruise modes. A related study [163] proved stability for multiple time-varying delays and external disturbance. It employed a squared error-based adaptation law for control gains. Lyapunov–Krasovskii theory was employed to derive delay-dependent LMI conditions ensuring stability with disturbance attenuation. However, the approach relied on bounded-delay assumptions. To address the limitations of fixed feedback gains, a fully adaptive synchronisation approach was designed that adapted both feedback and coupling gains online in [50]. This approach was based on formulating the MRAC problem. Each agent tracked a virtual reference model with parameters tuned by learning laws based on the system's design parameters. A scalable DAC was proposed in [178]. The controller gains were constant, and a time-varying coupling gain was adjusted online using an adaptation law proportional to the squared tracking error, with a leakage term to keep the gain bounded. Its principal advantage was its role in a parallel computing architecture, which made the simulations scalable. Within the DAC framework, a tracking adaptive control strategy was introduced in [206], which guaranteed both finite-time and asymptotic convergence. The adaptation mechanism utilised gradient-based laws, with controller parameters updated proportional to the tracking errors. This design achieved convergence without relying on the initial conditions. In another work [60], an MRAC framework was extended to address the nonlinear input uncertainties and imperfect communication. It included a neural network to reduce uncertainty and an input estimator to handle communication failures. An error-driven gradient-based adaptation mechanism adjusted the controller parameters. The MRAC framework was further applied in [174] to platoons with uncertainties, consisting of a reference model and an adaptive control system. Subsequently, this framework was improved in [173] to handle non-zero leader inputs using a two-layer architecture and introduced a cooperative disagreement error as a synchronisation mechanism. This design exhibited strong synchronisation, with fixed gains calculated from an algebraic solution to a Riccati equation, without the need for online parameter adaptation. In another work, a distributed adaptive fuzzy controller was designed for controlling platoons with state constraints and an unknown dead-zone input nonlinearity [208]. It applied a combination of backstepping, barrier Lyapunov functions, and fuzzy logic systems to approximate the unknown dynamics. An adaptation law proportional to the squared fuzzy basis function with sigma-modification terms ensured safety bounds. It also compensated for the actuator's dead zone. Its limitations were the need to know the signs of dead-zone parameters. The work in [37] proposed an event-triggered consensus protocol to address random communication noises. It featured a decaying, time-varying gain to mitigate noise. The work in [205] developed a backstepping-based adaptive scheme using radial basis function neural networks for approximating nonlinearities in heterogeneous platoons. It incorporated two event-triggering conditions. This design helped improve the communication efficiency. In a further application of the backstepping framework [212], an adaptive function was proposed in order to cope with unknown lag time in nonlinear platoons. It featured a novel event-triggered mechanism. To address input saturation, a

fully distributed event-triggered protocol was developed in [223] for a linear heterogeneous platoon. It adjusted the adaptive coupling gains according to a law proportional to the squared error. Finally, a random-switching topology-distributed scheme was presented in [180], in which communication changes were modelled as a Markov process. The controller incorporated time-varying adaptive gains, modulated by an adaptation law proportional to the squared neighbour-tracking error. Table 13 provides a classification of the literature on adaptive mechanisms used in vehicle platooning. It summarises the reviewed articles in five separate categories. The table shows that the squared-error adaptation mechanism is the most commonly implemented technique. There are also a good number of publications on adaptation mechanisms that are directly proportional to the error in the states. The model reference adaptive control (MRAC) and the gradient-based adaptation law are also used frequently as adaptation mechanisms. Finally, the "Others" category describes alternative, lesser-known adaptive mechanisms.

Table 13. Adaptive control methodologies for vehicle platooning.

Methodology	References
Squared error of the states	[41,59,160,163,178,180,205,220,223]
Directly proportional to error of the states	[141,143,174,206]
Gradient based	[212]
Model Reference Adaptive Control	[50,60,173]
Others	[37,144,208]

4.7. Distributed Model Predictive Control (DMPC)

Model predictive control (MPC) is an optimisation-based approach that uses a system's model to predict its behaviour over a finite time period. It calculates the optimal control inputs by minimising a cost function subject to state, input, and output constraints. The output states are controlled to track desired set points (reference), while control inputs are sent to actuators to implement the control actions [258]. For vehicle platooning, DMPC is an extension of this strategy in which each vehicle solves a local optimisation problem at every time step. The DMPC for vehicle platooning depends on the vehicle's dynamic model, physical constraints, and predicted state trajectories based on neighbouring vehicles. The objective is to minimise a local cost function penalising deviations from desired spacing, velocity, in addition to errors between a vehicle's own predicted trajectory and those assumed for its neighbours. The whole process is repeated in the next time step [106,134]. Figure 10 summarises the neighbour-coupled DMPC loop. At each sampling instant t, follower i receives from each connected neighbour $j \in N_i$ a presumed (communicated) predicted state trajectory $\bar{x}_j(k|t)$ over the prediction horizon N_p. Using its own model and these neighbour predictions, vehicle i solves a local constrained optimisation to obtain an optimal input sequence and the corresponding predicted trajectory. Only the first control move $u_i^*(0|t)$ is applied, and the updated predicted trajectory $\bar{x}_i(k|t+1)$ is then broadcast to the neighbours for the next receding-horizon step. A typical DMPC cost function can be represented as

$$J_i = \sum_{k=0}^{N_p-1} \left(\left\| x_i(k|t) - r_i(k|t) \right\|_{Q_i}^2 + \left\| u_i(k|t) \right\|_{R_i}^2 + \left\| x_i(k|t) - \bar{x}_i(k|t) \right\|_{F_i}^2 + \sum_{j \in N_i} \left\| C_s x_i(k|t) - C_s \bar{x}_j(k|t) - d_{ij} \right\|_{G_i}^2 \right), \tag{37}$$

where $x_i(k|t)$ and $u_i(k|t)$ are the predicted state and control input of vehicle i at prediction step k and time t, respectively. The term $r_i(k|t)$ is the local reference (state) trajectory. The

signals $\bar{x}_i(k|t)$ and $\bar{x}_j(k|t)$ are the presumed (communicated/assumed) self and neighbour predicted trajectories available at time t. The matrix C_s selects the spacing-related component(s) from the state (e.g., if $x_i = [s_i, v_i, a_i]^\top$, then $C_s = [1 \;\; 0 \;\; 0]$ so that $C_s x_i = s_i$). The desired inter-vehicular spacing is d_{ij}, and N_i denotes the neighbour set of vehicle i. The weighting matrices satisfy $Q_i \succeq 0$, $F_i \succeq 0$, $G_i \succeq 0$, and typically $R_i \succ 0$, penalising the tracking error, control effort, deviation from the assumed self-trajectory, and spacing disagreement with neighbours, respectively. The implementation of DMPC to vehicle platoons in the literature can be broadly categorised into three categories: (i) ideal node with ideal communication; (ii) ideal node with communication imperfection; (iii) external disturbance on a node with communication imperfection.

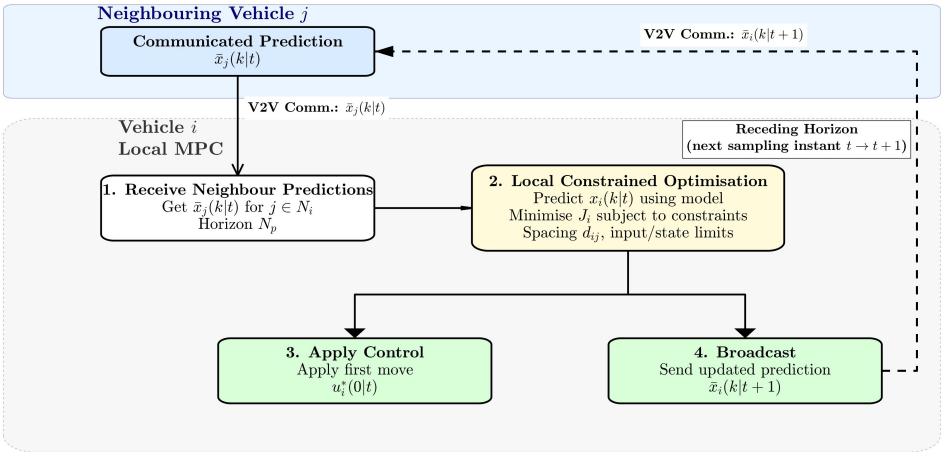


Figure 10. Neighbour-coupled DMPC workflow for vehicle platooning. Follower i receives presumed neighbour trajectories $\bar{x}_j(k|t)$ via V2V communication, solves a local constrained finite-horizon optimisation, applies the first control input $u_i^*(0|t)$, and broadcasts its shifted plan $\bar{x}_i(k|t+1)$ for the next receding-horizon step.

(i). Ideal node with ideal communication

This paragraph surveys the paper, focusing on DMPC when there is no disturbance at the node, and the network has no imperfections. The structure shown in Equation (37) is a popularly adopted structure given in the DMPC literature for platooning [84,134]. Building on this base DMPC framework, collision avoidance was further incorporated in [65] using the alternating direction method of multipliers (ADMM), rather than conventional quadratic programming solvers. A low-complexity variant was proposed that performed only one ADMM iteration per time step, thereby reducing computational load at the expense of lower tracking performance. An energy-oriented DMPC approach was presented in [45], in which the vehicle dynamics were integrated with a distance-dependent aerodynamic drag model to minimise energy consumption. This approach calculated variable spacing policies online to balance energy savings and safety. To address parameter uncertainty, a DMPC scheme was proposed in [177] to ensure stability under varying engine inertia parameters. This method, however, was limited to homogeneous vehicles. A different approach was followed in [80] with a hierarchical framework first optimising platoon formation to achieve better cohesion and fuel economy, followed by the application of a centralised robust MPC. However, the controller's cost function did not consider the neighbour coupling terms of Equation (37).

(ii). Ideal node with Communication Imperfections

To address the computational challenge of DMPC within an imperfect communication environment, the following works have been published. In one study, the control trajectory was parametrised using discrete orthonormal Laguerre functions [66]. This, in turn, reduced the number of optimisation variables and addressed input delays, resulting in rapid, smoother convergence. To overcome the issue of communication intermittency, another framework combined a V2V communication model grounded on stochastic geometry to calculate the probability of receiving information [213]. An Extended Kalman Filter (EKF) was implemented for state estimation of disconnected vehicles, thereby increasing robustness against random network interruptions. However, the effect of topology switching was not accounted for. The problem of uncertain changes in the network environment was also addressed in [91] by representing the communication topology switching as a Markov chain. This probabilistic design increased stability without requiring an average dwell time. The work also introduced constraints to prevent disturbance amplification. Continuing research on the DMPC framework led to work on a heterogeneous platoon in [198]. The cost function was extended to include a penalty term for deviations from the trajectory of the succeeding vehicle, thereby improving safety. This approach handled uniform delays by modifying the desired trajectory based on local predictions and managed packet dropouts by switching to a no-leader configuration.

(iii). External Disturbance on nodes with Communication Imperfections

This paragraph increases the challenges of vehicle platooning by incorporating a DMPC framework that addresses external disturbances at the node and network imperfections. Typical DMPC applications for platooning, which have constraints on vehicle states such as velocity and acceleration [18,51], control inputs [198] and inter-vehicle distances [45]. An advanced DMPC framework was developed in [185] with a dynamic event-triggering mechanism that utilises dynamic adaptation for predicted states and imperfect communication, including delays and packet losses. This adaptive triggering helped better manage computational load and ensured input-to-state stability. Similarly, an intermittent communication algorithm was proposed in [84] that estimated a consensus-based reference velocity to address time-varying delays and packet losses, but with the limitation that it could only be applied to homogeneous vehicles. Further progressing the field of communication-aware design, an adaptive event-triggered robust MPC was designed for linear multiagent systems with time-varying delays and arbitrary network topologies [221]. Its cost function did not include neighbour coupling terms. LMI-based optimisation was used to ensure the robustness and to limit the growth of computational complexity. A nonlinear DMPC was coupled with a discrete-time control barrier function (DCBF) to enforce hard safety constraints, thereby guaranteeing a minimum battery level [90]. This work also mentioned the use of a Markov Decision Process for strategic selection of charging stations. To manage totally unknown system dynamics, a data-based DMPC framework was proposed in [222], which learnt the system model with input–output data before controller design. However, its performance relied on data quality, and it failed to cope with communication uncertainties. Lastly, a tube-based DMPC was designed in [18] to simultaneously combat communication delays and external disturbances, utilising MPC, a consensus protocol, a delay compensation scheme, data buffers, and state predictors. Table 14 summarises the various cost–function structures utilised in the literature. As seen in the table, quadratic tracking with control efforts remains the dominant approach, reducing errors while minimising the cost function. Moreover, another class of cost functions relates to energy-consciousness. It is formulated to promote energy efficiency, a key factor in achieving sustainable transportation. Table 15 gives an overview of the most common constraints considered. The table shows the three constraints, which are actuator/control

input bounds, inter-vehicular spacing, and state/output bounds, as being important for physical feasibility and safety. Conversely, constraints on jerk, which are related to passenger comfort as well as smooth operation, and bounds on model-parameter uncertainties, which are of great importance for robustness, are less frequently treated as constraints in the existing literature on DMPC for vehicle platoons.

Table 14. Cost functions.

Type	References
Quadratic tracking only	[65]
Quadratic tracking and control effort	[18,51,66,80,84,91,106,134,148,177,185,198,213,221,222]
Energy-aware augmentation	[45,90]

Table 15. Constraints.

Type	Description	References
Actuator/Control input bounds	Hard/soft constraint on inputs or increments of input	[18,45,51,66,80,84,90,91,106,134,148,177,185,198,213,221,222]
Inter-vehicular spacing	Hard/soft constraint on distance/headway	[45,51,65,80,90,91,134,148,185,198,213]
State/Output bounds	Hard/soft constraint on states/outputs (e.g., position/velocity/acceleration)	[18,45,66,80,84,90,91,106,148,185,198,213,222]
Jerk bounds	Hard/soft constraint on the rate of change of acceleration (jerk)	[106,148]
Uncertainty bounds	Bounds on model parameter uncertainties	[90,177,222]

4.8. Distributed Sliding-Mode Control (DSMC)

DSMC is a type of nonlinear control that adapts sliding-mode control (SMC) for multi-agent systems. An SMC is a control strategy known for mitigating the effects of nonlinearities, model uncertainties, and exogenous disturbances [259]. Such factors are typically due to the inherent nonlinear dynamics, actuator saturation, or external environmental effects [260]. The primary strategy of SMC is to move the state trajectory to a predefined sliding surface and then force the system to remain on it. When the state is on the surface, the plant's effective dynamics collapse to a reduced-order form, the tracking error tends to zero, and the output follows a prescribed reference signal. The control objectives are to take the plant to a stable state [261] and to reject unmatched uncertainties via disturbance observers [262]. A vehicle platoon is an example of a system that is loaded with intrinsic nonlinearities and uncertainties that require reliable control strategies. Hence, SMC control is extended to distributed sliding-mode control (DSMC) [200,207,263]. In this environment, each vehicle runs a local SMC controller whose control law is computed solely from information received from its neighbours and the underlying communication topology. The nonlinearities occur due to the aerodynamic drag, interaction between tyres and the road, rolling resistance, limitation of the actuator, and effect of grade, which are more pronounced under variable roadway conditions [207]. Parametric uncertainties include unknown vehicle mass, aerodynamic coefficients, wheel radius and efficiency factors due to loading, manufacturing tolerances, wear [263], and model uncertainty due to unmodelled dynamics, approximations, and omitted high frequency modes [196]. The design of the DSMC law consists of three systematic steps. First, the design of the sliding surface is based on a mathematical equation that describes the desired dynamic behaviour. In the context of platoon control, each vehicle i is assigned a local sliding surface comprising local state errors and interactions among vehicles. An example of the sliding surface is:

$$\varpi_i = k_1 e_{s_{i0}} + k_2 e_{v_{i0}} + \sum_{j \in \mathcal{N}_i} \tilde{a}_{ij} \left(e_{s_{ij}} + k_3 e_{v_{ij}} \right),$$

where $k_1, k_2, k_3 > 0$ are design gains, $\tilde{a}_{ij}$ are entries of the adjacency matrix, $\mathcal{N}_i$ is the neighbour set of the vehicle i [95]. The second step is the reaching law, which specifies the rate at which the state trajectories approach the sliding surface. Classic examples are the constant rate law $\dot{\varpi}_i = -\varkappa \operatorname{sgn}(\varpi_i)$, $\varkappa > 0$ is the gain constant [207]; the exponential reaching law $\dot{\varpi}_i = -\varkappa \operatorname{sgn}(\varpi_i) - k\varpi_i$ [263], and the power-rate law $\dot{\varpi}_i = -k|\varpi_i|^{\alpha} \operatorname{sgn}(\varpi_i)$ for finite-time convergence [86]. The third step is to design the switching law, which introduces robustness by adding a discontinuous control action to the main control law. It forces the system to remain at the sliding surface. Standard choices include the sign function, the saturation function, etc. The latter is used to mitigate the effect of chattering [95]. The literature about DSMC applied to vehicle platoon control can be divided into three categories: (i) ideal node with ideal communication; (ii) ideal node with communication imperfection; (iii) external disturbance on a node with communication imperfection.

(i). Ideal Node with Ideal Communication

This paragraph accounts for the platoon having no disturbances on nodes and for imperfections in communication. In the work [207], the authors employed a linear sliding surface in combination with an exponential reaching law. Instead of using a traditional sign-based switching rule, they proposed an adaptive update scheme to replace discontinuous switching and avoid chattering. A follow-up study [95] introduced a DSMC architecture for heterogeneous platoons that utilised a linear sliding surface and sign-based switching, incorporating a boundary layer to mitigate chattering. To address these model nonlinearities, ref. [97] employed an adaptive neural network, a nonlinear sliding surface, and a power-rate law, incorporating a sign function adaptation. Likewise, ref. [100] implemented a radial basis function neural network to directly approximate nonlinearities, thereby avoiding the need for a switching law and reducing chattering. Notably, the authors in [47] proposed a distributed fixed-time approach with an integral sliding surface using a sign-function law to address uncertainties, although they did not discuss chattering. To compensate for unmeasured states, a prescribed-time scheme was proposed in [103], which constructed a nonlinear sliding surface, thereby making the control signal smoother. Other works have adapted integral sliding surfaces in combination with sign-function switching to address external disturbances [189], which was later extended in [190] by a variable-time headway policy. The study in [196] also employed an integral sliding surface and found that chattering could be reduced by using a saturation function rather than a sign function. The authors in [98] deployed a disturbance observer to estimate disturbance and neighbour acceleration. They employed an integral sliding surface and an adaptive reaching law, which effectively reduced chattering. A hierarchical approach in [192] operated by an adaptive integral sliding-mode design with an adaptive switching law to directly estimate the physical parameters. The authors of [99] combined a disturbance observer and a neural network based on an integral sliding surface and a sign function. Taking advantage of the integral sliding-mode principle, ref. [102] proposed an optimal nominal controller by Hamiltonian theory to suppress the occurrence of chattering. In [146], a PID-type sliding surface was coupled with a continuous double high-power reaching law to improve the anti-chattering effect. Finally, an energy-aware DSMC with a linear sliding surface and a saturation-shaped reaching law was proposed in [200]. To address the communication load issue, ref. [188] proposed a DSMC framework for the Round-Robin scheduling protocol, with a scheduling-dependent integral sliding surface and a discrete-time exponential-like reaching law. The novelty lay in explicitly incorporating the communication protocol into the controller and the surface design, thereby reducing the communication burden.

(ii). Ideal Node with Communication Imperfection

This paragraph addresses the case where the communication network is not perfect, even though disturbances acting on the node remain zero. The authors of [199] proposed a distributed adaptive sliding-mode controller that explicitly addressed the problem of unknown topologies for vehicle platoons. Their technique involved a sliding surface that accounted for local errors and neighbour interactions, utilising an exponential reaching law to ensure convergence. A major contribution was to manage unknown communication topologies with only eigenvalue bounds (not the full matrix). To counteract communication delays, ref. [151] designed a DSMC with a linear sliding surface and an exponential reaching law, rather than a switching law, to avoid chattering. The authors demonstrated their tolerance to communication delays through Lyapunov analysis.

(iii). External disturbance on the node with communication imperfection

The most challenging cases are for controllers that must deal with uncertainty and imperfect communication networks. Moving towards learning-based solutions, ref. [204] formulated a framework combining DSMC and RL to achieve the performance objectives. Their design was based on an integral sliding surface, but the key innovation was an actor–critic RL design that learned the optimal control policy online. The advantage of this type of architecture was that it eliminated the need to model the node or have complete knowledge of the communication topology. Chattering was reduced by using an adaptive gain in the sign-based switching term, which modulated the switching effort based on the system state. Table 16 categorises the literature depending on the type of DSMC for vehicle platooning. The table shows an increasing trend in the publications, with the integral sliding surface being the most popular. The traditional linear sliding surface also encompasses a significant number of studies, whereas research on the PID-type and nonlinear sliding surfaces is comparatively small. Table 17 summarises the reaching laws adopted in the different studies. The exponential reaching law is one of the most common reaching laws owing to its ease of implementation and predicted rate of convergence. Another common law, which is the power law. It is beneficial for addressing chattering. The adaptive reaching law is less common in the literature, suggesting scope for further research in this area. Table 18 categorises the literature on the basis of the switching law. It is apparent that the conventional sign function is the most popular; however, it also remains a cause of chattering. In its place, a more advanced switching law, known as adaptive switching, is often used. Table 19 lists the other approaches to the DSMC design that are also used to support its development. Some common approaches in the literature include adaptive parameter estimation and the observer-based framework in DSMC.

Table 16. Sliding surface.

Surface Type	References
Linear Sliding Surface	[95,100,151,199,200,207]
Integral Sliding Surface	[47,98,99,102,188–190,192,196,204]
PID-type sliding surface	[146]
Nonlinear Sliding Surface	[97,103]

Table 17. Reaching law.

Type	References
Constant	[200]
Exponential	[95,99,151,199,207]
Power	[47,97,102,146]
Adaptive	[98]
Other	[103]
Not Mentioned	[189,190,204]

Table 18. Switching law.

Type	References
Sign Function	[47,95,99,102,103,188–190], [196] *
Saturation Function	[196] *, [200]
Adaptive Switching Law	[97,192]
Not Mentioned	[151,204]

* This reference utilises both sign and saturation functions.

Table 19. Other approaches of DSMC.

Methodology	References
Adaptive parameter estimation	[100,199,207]
Observer	[98,200]

4.9. Distributed Nonlinear Control (DNC)

In the previous Section 4.8, a type of nonlinear control, i.e., SMC, extended to DSMC was reviewed. This subsection reviews a general distributed nonlinear controller (DNC) applied to the vehicle platoons. The fundamentals of Nonlinear systems theory are related to systems that do not follow the superposition principle, i.e., for which the relationship between input and output is described by nonlinear differential equations. These systems can be characterised by phenomena such as multiple equilibria, limit cycles, and finite escape times, which are not present in linear systems. The controllers that govern such systems are known as nonlinear controllers [264]. The field has been extended to distributed nonlinear control (DNC), which involves designing individual nonlinear controllers for each vehicle to address the complexities of real-time vehicle platooning. The literature on DNC applied to vehicle platoons can be sorted into three categories: (i) ideal node with ideal communication; (ii) ideal node with communication imperfection; (iii) external disturbance on a node with communication imperfection.

(i). Ideal Node with ideal communication

This paragraph discusses the application of DNC to vehicle platoons characterised by no disturbance at the node and a perfect communication network. In [73], the distributed control protocol took into consideration vehicle dynamic information, i.e., the electronic throttle (ET) opening angle, to achieve consensus on position and velocity. The authors in [104] adopted a third-order vehicle dynamics model to characterise heterogeneous platoons. Their controller embedded the acceleration difference between consecutive vehicles, a factor of passenger comfort and car-following interactions through a nonlinear tanh-based function. In the study performed in [74], the electronic throttle aperture was incorporated into DNC. This integration improved passenger comfort by attenuating the acceleration amplitude. However, the scheme was restricted to platoons composed of uniform vehicles. To cope with the diversity of vehicles, ref. [145] proposed a third-order model to describe the difference of powertrain characteristics and designed a controller, which also realised the car-following algorithm to avoid the rear-end problem. An important part of this work focused on preventing the disturbances from increasing as they travelled through the platoon. To reduce communication load, the work in [72] designed a DNC system based on event-triggered control. The controller was designed to combat Zeno behaviour.

(ii). Ideal Node with Communication Imperfection

This subsection involved the DNC implemented on a vehicle platoon in the presence of network imperfection but with no disturbances on the node. A practical validation was presented in [85], where a nonlinear controlled platoon model was tested against experimental data from a robotic testbed. Its performance was analysed against time-varying communication delays. In another study, the DNC was developed to address

the issue of topology switching. Here, inter-vehicular distances were adjusted to enhance safety during stop-and-go manoeuvres. Moreover, some works on DNC development incorporated practical constraints on platoons. For example, the authors in [83] designed the DNC to account for actuator input saturation, thereby incorporating the physical constraints of acceleration and braking. In another work [75], the authors developed the DNC based on a spring-damper energy system. The novelty of this work was that it maintained platoon performance, i.e., stability and safety, even in complete loss of communication.

(iii). External disturbance on the node with communication imperfection

This paragraph reviews the platoon's challenges, including disturbances on the nodes and communication defects. The work in [167] extended the third-order model to explicitly treat time-varying communication delays and limited communication ranges by deriving the allowed upper bound of physical delays. The method produced a non-zero steady-state spacing error. In a similar approach, the authors in [147] also addressed communication delays within a third-order heterogeneous framework. Their controller incorporated the acceleration difference between vehicles to enhance comfort, employing a hierarchical approach to improve the fidelity between the desired and actual acceleration in the simulation. A similar event-triggered scheme was adopted in [210], where a nonlinear function was introduced to generate an accurate model of the dynamic interactions between neighbouring vehicles. This work included the contribution of an event-triggered controller that explicitly accounted for input delays and external disturbances. The upper limit of the communication delay was determined by applying Lyapunov stability theory to ensure consensus. Table 20 classifies the literature of DNC-controlled platoon according to the nonlinear elements used. The table shows a common trend in the implementation of tanh-based optimal velocity functions or similar saturating functions to model the DNC. These functions capture more of the vehicle's physical limitations on speed. The other methodologies for DNC development are less common and include potential-energy-based functions, nonlinear event-triggered mechanisms, and input-saturation considerations.

Table 20. Nonlinear elements.

Methodology	Nonlinear Element	References
Tan-h-based function of optimal-velocity	Nonlinear velocity function or a similar saturating function.	[73,74,85,104,145,147,149,167,210]
Others	Potential Energy based nonlinear function	[75]
	Nonlinear event triggered function	[72]
	Input saturated	[83]

4.10. Distributed Artificial Intelligence (AI) Based Control

Artificial intelligence (AI)-based control methodologies offer a spectrum of techniques for controlling dynamical systems. Amongst the main methods are reinforcement learning (RL), where agents successively discover optimal policies by interacting with their environment; deep reinforcement learning (DRL), where deep neural networks are used to address high-dimensional state spaces; adaptive optimal control, which usually uses actor–critic architectures to approximate policies for real-time decision making; and data-driven methods using a radial basis function neural network (RBFNN) [133,193,209,215]. These techniques learn directly from data; for example, Q-learning uses the values of actions estimated by policy iteration, and actor–critic DRL separates the action selection and the value estimation into two different components [133,193,209,215]. The literature on AI-based control applied to vehicle platoons can be categorised into three main areas: (i) ideal node with ideal communication; (ii) ideal node with communication imperfection; (iii) external disturbance on a node with communication imperfection.

(i). Ideal Node with Ideal Communication

This paragraph reviews all the literature related to the ideal case, i.e, no disturbances on the node and perfect communication. Both the RL and DRL control methods implement distributed learning, where the vehicles use the local observations to adaptively update their tracking policies [133,193]. The authors in [133] developed a DRL controller for the platoon formation that addressed the system uncertainty and dimensionality issues. They introduced a method based on conditional Kullback–Leibler (KL) divergence to quantify the value of V2X information. This, in turn, led to the identification of the data required to construct a low-dimensional state space. A distributed training framework based on the Deep Q-Network was presented in [54], decomposing the task into single-agent problems of maintaining equidistant spacing. This featured a two-phase learning process to protect privacy and reduce communication load, where vehicles first train their Q-network locally before a consensus algorithm averages the learned parameters. In a similar work [201], the authors developed an RL framework on the heterogeneous platoons with unknown dynamics. They used an identifier-critic-actor architecture, in which three neural networks were employed for the following purposed: (i) to estimate the system dynamics; (ii) to synthesise the policies; (iii) to evaluate the controlled system performance. The update of the learning algorithm was based on the negative gradient of the function, which was equivalent to the Hamilton–Jacobi–Bellman (HJB) equation. This led to the removal of the stringent persistence-of-excitation (POE) condition, thereby increasing computational efficiency. A further contribution to the literature, the work presented in [153] was a data-driven hybrid iteration algorithm. This algorithm combined adaptive dynamic programming with a distributed internal model designed to reduce the effects of external disturbances. The learning was first performed using value iteration to find a preliminary stabilising policy, and then switched to policy iteration to accelerate convergence. This hybrid learning strategy demonstrated robustness in traffic-congestion scenarios. In [215], a distributed data-driven formation-control scheme for heterogeneous platoons was proposed. It combined radial basis function neural networks (RBFNNs) with a model-free adaptive control framework and an event-triggered mechanism to significantly reduce computational burden. Building on previous work, the authors in [216] developed a data-driven iterative learning control that established a linear data model for the adjacent vehicles in the platoon. The algorithm required only control information from earlier iterations and information from neighbouring vehicles, thus eliminating the need for prior system modes. This, in turn, improved accuracy. In another work, the authors in [55] employed the distributed learning and real-time adaptation (for the model-free behaviour). In another work, a data-driven iterative learning control method utilising adjacent dynamic linearization was proposed. A linear data model was established to characterise the dynamic relationship between communicating adjacent vehicles. By leveraging control knowledge accumulated from previous iterations and information from neighbouring vehicles, the need for precise system models was circumvented, thereby enhancing accuracy and robustness against uncertainties.

(ii). Ideal Node with Communication Imperfection

This paragraph surveys papers that consider a non-ideal communication network with ideal node dynamics in the platoon. To address the challenges of switching topologies, a distributed leader-state observer was developed in [55]. This observer provided an estimate of the leader's state despite changes in communication links. This estimation was then used by a model-free Q-learning algorithm that learnt the optimal control policy from system data to minimise tracking errors and energy consumption. Another contribution was a distributed control strategy based on distributed proximal policy optimisation [53]. The response to communication failure was to explicitly account for it by embedding a realistic communication model in the training environment and designing an active information-

fusion mechanism to smoothly accelerate vehicles under intermittent connectivity. The reward design aimed to balance minimising deviations from platoon-wide consensus with penalties for harsh acceleration. Using actual trajectory data for training contributed to the robustness.

(iii). External Disturbance on the node with Communication Imperfection

This paragraph presents work on AI-based platoon control that addresses realistic situations involving disturbances at the node and imperfect communication. An RL-based adaptive optimal control scheme was constructed in [209] to accommodate platoons characterised by unmodelled dynamics, external disturbances and uncertain communication topologies. The architecture combined an actor–critic neural network with adaptive backstepping to enable online self-learning from input–output data without explicit vehicle models. The optimisation objective penalised both the tracking errors and the control effort. The advantage of the latter approach was that it could be scaled to uncertain topologies. In the work [193], the authors developed the DRL-based platoon control. They implemented neural networks augmented with integral action to generate incremental acceleration commands and counter unwanted disturbances. The DRL-based controller operated on the deterministic policy gradient approach. To make the controller robust across any communication topologies, they modelled the neighbouring teammates as a single 'virtual' average agent during training. This made the overall controlled system robust to any topology changes. Table 21 summarises literature employing AI-based control approaches for vehicle platoon formation. The range of methods discussed covers a large portion of current research: from reinforcement learning (RL) methods, inclusive of value-based methods such as Q-learning, to more advanced deep reinforcement learning (DRL) methods; from adaptive optimal control methodologies, which dynamically approximated the Hamilton–Jacobi–Bellman equation, to other data-driven methods, which uses radial basis function neural networks (RBFNNs) to directly design the controllers based on the system input/output data. The relatively even distribution of publications across these AI categories indicates an actively evolving area of study.

Table 21. AI-based control.

Methodology	Approach	References
Reinforcement learning (RL)—value-based (Q-learning/Deep Q-Network)	Agents iteratively learn policies through environmental interactions to maximise rewards; value-based RL, like Q-learning, estimates action values via policy iteration on system trajectories.	[54,55]
Deep reinforcement learning (DRL)—actor–critic	Integrates deep neural networks to handle complex, high-dimensional state spaces; actor–critic DRL uses an actor for action selection and a critic for value estimation; enables decentralised learning using local observations to adapt policies for tracking.	[53,133,193]
Adaptive optimal control—actor–critic (Hamilton–Jacobi–Bellman)	Employs actor–critic architectures to approximate optimal policies online using input–output data; approximates solutions to the Hamilton–Jacobi–Bellman equation; removes reliance on exact vehicle models or topological matrices; promotes model-free, robust platoon behaviour via learning and real-time adaptation.	[153,201,209]
Data-driven methods—radial basis function neural networks (RBFNNs); data-driven iterative learning	Directly uses system input/output data for controller design	[215,216]

4.11. Distributed Control (Other Approaches)

Besides the commonly used distributed control approaches discussed in the previous subsections, there are other, less popular yet effective approaches that facilitate state synchronisation in the platoon. These are discussed in this subsection.

4.11.1. Data-Driven Control

Most of the time, it is not possible to completely accurately model the vehicles due to unobserved dynamics and changing operational conditions. The distributed data-driven control (DDC) approach is suitable for this type of problem. In this type of control, there is no requirement for a physical (first-principles) model of the vehicle, which is quite the opposite of conventional control strategies, which require accurate modelling for precise platoon control. In the DDC, control laws are developed directly from input–output data. Some of the works on DDC for platooning include the following: A DDC was developed in [219], which linearly approximated the nonlinear vehicle dynamics of the heterogeneous platoon using an incremental data-based design. This approach achieved consensus among states of vehicles without the need for pre-training, thereby offering less computational demand. The work in [217] addressed communication imperfections within the platoon. They designed the DDC strategy that accounted for the effects of quantisation and switching topologies. Moreover, an improved uniform quantiser was introduced, embedded with an encoding–decoding mechanism, to minimise data loss.

4.11.2. Optimal Control

Optimal control of vehicle platoons has become a research area aimed at enhancing the overall system performance. Some examples in of platoon controlled via optimal control are as follows: In [105], a cooperative optimal control framework was presented, which was based on a cost function balancing the performance of the platoon with the amount of energy consumed. The optimal control gains were determined by solving a linear matrix inequality (LMI) optimisation problem, which ensured stability by limiting the amplification of disturbances. The work in [168] designed the distributed linear quadratic regulator (DLQR) controller with the DSMC controller. The DLQR handled the plan parameters by computing optimal values to minimise the cost function. The cost function consisted of the state errors and the controller efforts. In addition, the DSMC part tackled the exogenous perturbations. The sliding surface designed was an integral sliding surface with a switching law, thereby ensuring finite convergence of the platoon-controlled system (with optimal parameters) to the sliding surface and thereby rejecting disturbances.

4.11.3. Miscellaneous Distributed Controllers

These are the distributed control algorithms which are rarely implemented. Thus, there are high chances of expanding them into numerous distributed controllers for the vehicle platoon in the future. For instance, the authors in [43] used game theory to solve the platoon problem within a differential game setting. It yielded analytical solutions for the trajectories of the Nash equilibrium. This, in turn, enabled the pre-computation of trajectories, thereby reducing the computational load. Another unique example is the use of fractional calculus, which, in turn, leads to the development of fractional-order controllers. This approach has also been rarely synthesised for the formation control of vehicles in a platoon, especially for general topologies (although there are a few works on distributed fractional-order controllers for specific topologies). Recently, three novel distributed fractional-order controllers were presented in [81], with their stability regions characterised using the Root Boundary Locus (RBL) method. The work showed that the fractional-order controllers outperformed integer-order ones. In a separate study, a switched pinning control method

was proposed in [67], in which control inputs were applied only to selected pinning agents, while the other vehicles adjusted their velocities autonomously to reach consensus. The optimal switching of these agents was formulated as a Mixed Logical Dynamical (MLD) system that integrates continuous dynamics with logical constraints. Combined with model predictive control (MPC), this framework minimised velocity errors. Another uniquely developed controller is shown in [94]. To reduce communication overhead, a distributed event-triggered impulsive control (ETIC) scheme was developed. In this scheme, control actions were sent as discrete impulses that instantaneously changed a vehicle's velocity. The design also ensured Zeno behaviour was addressed by having a time-dependent term within the event triggering function

4.12. Distributed Observer-Based Control

In control systems, an observer is a dynamical system designed to estimate the internal states of a plant whose states are not directly measurable due to some technical or economic limitations. An observer utilises the system's inputs and outputs to estimate the states required for implementing state-feedback control. In multiagent systems, such as vehicle platoons, the observer concept is extended to a distributed framework because actual information is either unavailable or unmeasurable. Each vehicle has a local observer. The vehicle uses its own measurements and the data received from its neighbours to estimate the necessary states. This estimated information is used to compute the control action [42,64]. Moreover, DO not only estimate the states of followers or the lead vehicle, but also estimate the consensus state error [61,64]. A type of DO for the vehicle platoon is shown:

$$\dot{\hat{x}}_i(t) = A\hat{x}_i(t) + B\,u_i(t) + L\Big(\sum_{j\in N_i} \tilde{a}_{ij}\big(\hat{x}_j(t) - \hat{x}_i(t)\big) + b_i\big(\hat{x}_0(t) - \hat{x}_i(t)\big)\Big), \tag{38}$$

where $\hat{x}_i(t)$ is the estimated state of the ith follower vehicle, $\hat{x}_j(t)$ and $\hat{x}_0(t)$ are, respectively, the estimates of a neighbouring jth vehicle and the leader vehicle. The matrices A , B represent the system and input matrix, respectively, and L is the observer gain matrix. After estimation, the controller uses the estimated state $\hat{x}_i(t)$ to compute a suitable control action $u_i(t)$, leading to platoon stability and consensus. The observer shown in (38) is a Luenberger-type DO. the literature on DO are also catergorised into three groups: (i) ideal node with ideal communication; (ii) ideal node with communication imperfection; (iii) external disturbance on the node with communication imperfection.

(i). Ideal Node with Ideal Communication

This paragraph surveys the research using the DO framework as an auxiliary unit to the primary controller when the platoon operates in ideal conditions, i.e., with no disturbances at the node and ideal communication between the vehicles. Within the framework of DSMC, an adaptive Luenberger-type velocity DO was designed for each follower [200]. The novelty of this distributed observer-based platoon control lies in the use of acceleration-dependent adaptive gain to control the difference between the predicted and measured speeds, thereby improving fuel economy.

(ii). Ideal Node with Communication Imperfection

In this paragraph, a review of studies on platoons with no disturbances at the nodes, yet with imperfect communication, was conducted. Work in [38] proposed a DO to address the problem where the velocity of the leader was unmeasurable and hence was estimated. However, the leader acceleration was assumed to be uniform. The control law incorporated the estimated leader velocity while also addressing the constant time delay. The stability condition was analysed using frequency-domain methods. In [61], the observer estimated consensus tracking errors while the platoon was heterogeneous and subject to commu-

nication time delays. The stability of the platoon could be proven using the generalised Nyquist criteria. In the study [42], stochastic uncertainties were compensated with the development of an observer based on the Kalman filter. The work was performed on homogeneous platoons. The proposed design enabled reconstruction of relative states from available input and output measurements, while mitigating the effects of packet dropouts and bounded delays. More recently, the authors in [265] used a Kalman filter as DO to estimate and counteract longitudinal external disturbances in the presence of sensor measurement noise. They developed a distributed linear controller with an auxiliary DO architecture for addressing vehicle platoons communicating in a PFL topology. Although the methodology is restricted to a particular network topology, the DO responsible for disturbance estimation and compensation is relevant to observer-based platoon control. Notably, it can be generalised to generic communication topology graphs using adjacency and pinning matrices. Further extending the Luenberger-type DO frameworks, the work in [117] addressed the challenge of fading communication channels. The design employed relative output feedback to estimate the vehicle states, with observer gains determined via modified Riccati inequalities. The authors in [135] addressed communication imperfections and parametric uncertainty by developing a unified two-layer architecture comprising an upper-level observer and a controller. This DO relied on sign-based terms to estimate the leader's state. Meanwhile, the tracking controller tracked the lead vehicle, whose states were already estimated by the observer. A comparative study in [116] used a Luenberger-type DO to analyse the performance of different directed communication topologies with constant delays. A similar DO for periodically intermittent information was developed in [125]; the novelty was to derive an information-rate condition that ensured stability over on and off communication periods. An adaptive DO framework was proposed in [64] to estimate the leader's state and the leader's unknown system matrix, thereby eliminating the need for a known leader model. To address cyber attacks, a conventional Luenberger-type DO was used in [122] to ensure resilience against Denial-of-Service (DoS) attacks and maintain state estimation during communication blackouts.

(iii). External Disturbance on the Node with Communication Imperfection

This paragraph reviews the platoon's challenges, including disturbances on the nodes and communication defects. A distributed proportional–integral DO, which extended conventional Luenberger designs by including integral terms, was developed in [166] in order to cope with mixed disturbances. It estimated states of velocity and acceleration and increased robustness in the presence of measurement noise. In [211], the DO estimated the leader's unknown jerk dynamics. The approach used sign–error integration for robustness to unknown reaction-time delays in nonlinear car-following models. To overcome communication limitations, a neural network-based DO for limited bit-rate operating environments was introduced in [203]. The design was based on a coding–decoding scheme to reduce the communication load, and the NNs approximated the unknown nonlinearities. The DO estimated the vehicles' states from coded data.

Table 22 categorises different types of DO. Since observers are auxiliary units that provide state estimates to the controllers, the table classifies the primary controllers by the type of observers they use. From the table, it is evident that the Luenberger observer is a popular choice for augmentation with a wide variety of controllers. Thus, highlighting its versatility and effectiveness in state estimation for a vehicle platoon. One key observation from the table is that observers and controllers of the same nature are often developed, for example, adaptive controllers with adaptive observers, and AI-based controllers with NN-based observers. In contrast, some other well-established methods, such as the Kalman filter, are less frequently used in recent literature on platoon control.

Table 22. Observer types used with different main controllers.

Main Controller	Observer Type	References
Linear	Luenberger	[38,61,116,117,122,125]
	Kalman filter	[42]
	PI/PID-inspired	[166]
	Other	[135]
Adaptive	Adaptive	[64,79]
	Luenberger	[154,178]
	Other	[101,155]
SMC	Luenberger	[95,98,196]
	Adaptive	[200]
	Other	[103]
AI control	NN-based	[203]
Other control	Luenberger	[176,197]
	NN-based	[183]
	Nonlinear-based	[211]

4.13. Observed Trends Across the Reviewed Literature

Table 23 provides a detailed comparison of the distributed control algorithms discussed in Section 4 for longitudinal vehicle platooning, including their attributes, core mechanisms, robustness, model suitability, computational complexity, and key limitations. A definitive ranking of distributed controller families is rarely reported in the literature. Most studies focus on a specific controller class, a specific vehicle model, and a specific set of assumptions (e.g., topology, delays, packet drops, or disturbances), rather than benchmarking multiple controller families under the same conditions. For this reason, the Table 23 should be read as a structured synthesis of reported capabilities, typical validation settings, and commonly stated limitations across the reviewed works, rather than as a claim that one controller universally outperforms the others.

After a comprehensive review of the literature, across challenging network and operating conditions, a certain set of usage patterns is observed. For network-induced delays and packet losses, the most preferred controller is H_∞ frameworks. This is because the reduction H_∞ norm carried by Lyapunov–Krasovskii and LMI-based formulations allows for providing systematic robustness conditions. Adaptive controllers are typically used when parametric uncertainty, platoon heterogeneity, or unknown dynamics are present. This is because of the online parameter tuning capability built into the design. These studies are also often paired with an event-triggered approach to limit transmissions and safe communication resources. Model-predictive control (MPC) methods are primarily used when hard constraints (e.g., spacing safety, actuator constraints, comfort) are to be enforced alongside multi-objective performance (e.g., energy and tracking). Although this generally increases the computational load. Sliding-mode and other nonlinear controllers are often introduced when strong robustness to uncertainties and/or disturbances is desired, with a growing preference for integral surfaces and continuous/adaptive laws to minimise chattering. AI-based controllers are the most recent distributed controllers added to the class of consensus controller families. They are being implemented in complex models, i.e., data-driven environments, or uncertain topologies. Their stability and safety assurances are typically strengthened by combining learning with model-based elements such as observers, backstepping or safety constraints. Finally, observer-based augmentation is used across a variety of controller families, typically to reconstruct leader states or estimate disturbances in the presence of communication imperfections or other uncertainties.

Table 23. Comparison of distributed control strategies for longitudinal vehicle platooning.

Attribute	DLC	DHC	DAC	DSMC	DMPC	AI Control
Core Mechanism	Fixed-gain state feedback [12]. Consensus-based constant-gain neighbour–leader coupling [88].	LMI-based gain synthesis [16]. $\mathcal{H}_\infty$ designs target disturbance-to-error attenuation [17]. Some formulations assume ideal communication (delay/drop neglected) [164].	Online gain adjustment via adaptation laws [41,220]. Gradient/MRAC-type updates [141]. Neuro-adaptive variants with NN/RBF estimators [97,205].	Sliding surfaces built from spacing/velocity errors [95,207]. Reaching/switching laws provide robustness [95]. Integral SMC variants may yield explicit delay bounds [190].	Receding-horizon optimisation with prediction [134]. Neighbour-coupled MPC under constraints [106]. Distributed trajectory optimisation also reported [146].	Model-free RL/DRL or NN policies [54,133]. Learning-based control under uncertain topology/dynamics [193,209,215].
Computational load and Implementation	Low computational load [107,139]. Mostly offline tuning [39].	Moderate load; offline LMI synthesis [164,169]. Lightweight online execution once gains are fixed [16].	Moderate–high load due to continuous adaptation [141,205]. Mainly online due to parameter updates [64].	Low–moderate load; online switching terms [95]. Higher complexity with observers/NN smoothing [100].	High online cost (optimisation each step) [65,66]. Offline tuning of horizons/weights; online solve [134].	High training cost [54]. Low-cost online inference after training [215].
Performance: nonlinearity and disturbance	Typically linear/linearised; gain-limited rejection [39]. Noise-handling via relative-state/relative I/O designs [42].	Explicit $\mathcal{L}_2/\mathcal{H}_\infty$ attenuation [17]. Robustness margins can be conservative [15]. Some works focus on disturbance rejection under ideal comm [164].	Bounded disturbances handled via adaptation (UUB/ISS-type) [174]. NN/RBF estimators approximate unknown nonlinearities [97,205].	Strong robustness to disturbances/nonlinearities [95]. Disturbance-observer/energy-aware SMC variants reported [200].	Nonlinear/constraint-aware; tube/robust variants mitigate disturbances [18]. Performance depends on prediction/estimation quality [134].	Robustness possible if trained across disturbances [193]. Generalisation remains data-dependent [53].
Performance: mismatch and uncertainty	Sensitive to mismatch; robustness often implicit [113]. Works best under mild heterogeneity [39].	Structured uncertainty via robust LMI synthesis [16]. Can be conservative for broad uncertainty sets [17].	Main strength: online compensation of unknown parameters [174]. Often requires boundedness/PE-type assumptions [163].	Robust to bounded uncertainties; adaptive SMC relaxes bounds [199]. Heterogeneous platoons under disturbances reported [190,200].	Uncertainty handled via robust/stochastic constraints [18]. Prediction mismatch directly affects performance [65].	Robust training can cover uncertainty [193]. Formal guarantees uncommon without safety layers [133].
Performance: delays and packet drops	Sufficient conditions under bounded delays/drops [42]. Bounds may be conservative at scale [71].	Admissible delay/drop parameters enter LMIs as feasibility regions [17,156]. Some $\mathcal{H}_\infty$ works neglect delay/drop effects [164].	Event-triggered/adaptive schemes tolerate variable delays/drops [144,160]. Explicit margins not always stated [163].	Explicit delay margins reported in policy-specific designs (e.g., VTH) [190]. Packet-delivery probability modelled; quantitative errors reported [207].	Predictive designs accommodate delays; tube MPC adds robustness [18]. Losses handled via estimation; computation heavy [66].	Communication constraints handled in training; tolerance often empirical unless analysed [53,203]. Designs for uncertain topology without explicit bounds also reported [209].
Performance: switching topology	Needs connectivity and dwell-time tools [93]. Often conservative under frequent switching [137].	Markov/switching modelling supported via LMI synthesis [67]. Conservatism rises with many modes [157].	Clear handling under Markov-switching formulations [180]. Requires transition/dwell-time constraints [209].	Robust to uncertain interaction topology [199]. Random topology and delivery-probability coupling reported [207].	Switching handled via dwell-time inside optimisation [119]. Complexity rises with mode changes [134].	Topology adaptation via Q-learning/DRL [55,209]. Retraining/graph generalisation affects scalability [54].
Model suitability and limitations	Best for linear/LTI or linearised dynamics [88]. Linearisation reduces realism [113].	Best for linear/linearised models in LMI-based $\mathcal{H}_\infty$ [17]. Some robust designs neglect comm imperfections [164].	Linear/nonlinear models possible [143,208]. NN/fuzzy adds tuning/PE needs [60,97].	Suited to nonlinear dynamics and matched disturbances [95]. Chattering/gain tuning central [151].	Linear and nonlinear MPC reported [134]. Heavy computation; depends on prediction accuracy [65].	Model-free/data-driven; suited to unknown dynamics [55,215]. Guarantees typically need safety layers [53,133].

5. Countering Faults and Attacks

This section reviews the literature on operational faults in vehicle platoons. The two main faults can be broadly categorised into the following types: (i) cyber faults—Denial-of-Service (DoS), replay attacks, spoofing, etc.; (ii) mechanical faults—actuator faults, sensor faults, etc. The remaining section briefly focuses on these faults, their modelling, and the mitigation strategies to address them.

5.1. Denial-of-Service (DoS) Attacks

Denial-of-Service (DoS) attacks are one of the most common types of cyberattacks. A distinct characteristic of these attacks is that they jam the wireless channel. As a result, the platoon's vehicles do not receive the packets for some time intervals. Hence, DoS attacks are modelled by incorporating communication delays into the platoon network. The authors in [122] developed a DO-based control method to address aperiodic DoS attacks. The observer ensured that tracking was maintained, thus optimising the tolerable duration of the attack. In another work [121], the authors developed distributed predictor-based control to address the communication delay and network latency caused by DoS. The authors in [76] modelled DoS as a constant communication delay. To mitigate these attacks, an algorithm was developed to detect them. The affected vehicle is either retrieved or isolated from the platoon. Additional works on DoS aimed to prove the stability conditions in terms of upper limits on DoS frequency and duration. The works in [52,63,182,191] showed similar strategies of designing even-triggered control schemes that handled both DoS attacks and parameter uncertainties. These works assumed a bound on the frequency and duration of DoS. In some cases, the developed gains were also topology-dependent owing to network latencies. In one of the works by [127], the authors synthesised a DMPC framework; established a quantitative relationship between the DoS attack parameters and the exponential decay rate of the system [127]. The authors in [162] modelled DoS as switching topology and proposed a similar strategy by developing H_∞ control approach to suppress the external disturbances and under intermittent attacks.

5.2. Data Integrity Attacks (Spoofing, Falsification and Byzantine Attacks)

Preserving the integrity and quality of data is a major concern that is often compromised in the event of cyberattacks. The authors in [176] developed a distributed H_∞ control approach to prevent the falsified information. To protect against spoofing and data falsification, distributed, adaptive, event-triggered approaches were developed in [172]. The data-driven adaptive controllers were also designed in [214] to handle such types of attacks. In the work of [82], the authors designed a neural network-based adaptive strategy to compensate for the falsified driving commands. They also embedded a mechanism that avoided collisions. A distributed model predictive control framework was designed in [123], which used a detector to distinguish unreliable data in the presence of Byzantine attacks.

5.3. Physical Faults and Mixed Cyber Physical Threats

In addition to cyber defaults, there is significant literature on mechanical malfunctions in vehicle platooning, such as actuator and sensor faults. A Fault-Tolerant Control (FTC) is implemented to mitigate such type of faults. The authors in [62,195] implemented learning-based state estimation for such faults. They devised a prescribed-time FTC approach. Building on this study, the works in [175,179] addressed actuator faults in the presence of cyber threats. More information about their modelling and mitigation with respect to the different controllers applied is given in Table 24.

Table 24. Controller-based classification of fault modelling and mitigation.

Type	Fault type	How Is Modelled	Addressed	References
Linear	Cyber fault	Additive faulty signal injected into the control law	Estimate the bounds of the fault term	[186]
			Control law with random noise compensation	[152]
		Communication delay	Event-triggered control	[52]
			Switching-based controller	[76]
		Communication delay and additive signals injected to states	Event-triggered control	[63]
	Mechanical Faults	Additive faulty signal injected into the control law	Event-triggered control	[161]
	Both Cyber and mechanical faults	Additive faulty signal injected into the control law	Analysis-only; mitigation not addressed.	[181]
Adaptive	Cyber fault	Additive faulty signal injected into the control law	Neural Network-based adaptive gains	[82]
			Adaptive event-triggered communication	[172]
		Communication time delay	Control law having time-varying delay	[124]
	Mechanical Faults	Additive faulty signal injected into the control law	Estimate the bounds of the fault term	[86,96,195]
			Event-triggered control	[171]
MPC	Cyber fault	Switching Signal	Embedding timing bounds in control law	[127]
		Additive faulty signal injected into the control law	Detect and isolate attack-corrupted data from prior broadcasted data	[123]
			Event-triggered control	[191]
Observer Control	Cyber fault	Switching Signal	Estimate the states of its neighbours based on its onboard sensor	[122]
		Additive faulty signal injected into the control law	Estimate the states of the vehicle based on its onboard sensor	[176,187]
	Mechanical Faults	Additive faulty signal injected into the control law	Estimate fault detection and isolate the fault or compensate it	[175,202,218]
	Both Cyber and mechanical faults	Cyber fault—Time delay; Mechanical Fault—Additive faulty signal injected into the control law	Event-triggered control and estimate the states of the vehicle based on its onboard sensors	[150,179]
H_∞ Control	Cyber fault	Switching Signal	Control to find bounds on the allowable frequency and duration of attack	[162]
		Intermittent V2V communication	Event-triggered control	[182]
AI Control	Cyber fault	Additive faulty signal injected into the control law	Learning-based estimation to actively compensate for the fault	[62]
Nonlinear Control	Cyber fault	Additive faulty signal injected into the control law	Compensates for the faulty signal	[194]
Others approaches	Cyber fault	Communication time delay	Estimate the delay created by the attack	[121]
		Additive faulty signal injected into the control law	Estimate from past valid data; when the error exceeds a fixed threshold, activate the compensation algorithm.	[214]
			Game-theoretic defence design	[128]

5.4. Strategic Defence and Vulnerability Analysis

Shifting the focus from controller design to strategic defence, a game-theoretic approach was proposed to determine the optimal placement of defensive actuators [128]. This work formulated the problem as a Stackelberg game in which the defender would place actuators to maximise the energy the attacker would need to inject to disrupt the platoon; this was quantified using controllability metrics. A systematic vulnerability analysis in [181] also showed that middle platoon members are often the most impactful targets in highly connected topologies; hence, more resilient techniques are needed to make them more robust.

Tables 24 and 25 provide complementary perspectives on the fault-related literature in vehicle platooning. Table 25 is organised primarily by fault type and its modelling, and then maps the supporting literature across controller families. Additionally, Table 24 is organised first by controller family (e.g., linear, adaptive, MPC, $\mathcal{H}_\infty$, AI, nonlinear, observer, and others) and for each family, the table separates the cyber, mechanical, and combined faults, and also reports how the fault is modelled and is mitigated. When investigated together, the tables reveal that cyber faults dominate the research landscape and are addressed in the majority of controller families, whereas mechanical faults are comparatively less studied, and compounded faults, i.e., cyber-mechanical faults, are poorly studied. Event-triggered mechanisms and observer-based designs are the most common reported mitigation strategies. It is also supported by Figure 11. The pie chart shows that cyber faults account for about 65% of research on all faults in vehicle platooning. Moreover, the second pie chart shows that among the cyber faults, DoS attacks are the most common. The last pie chart shows that among the distributed control strategies, distributed linear and observer-based are by far the most commonly implemented, accounting for 25.0% and 20.8%, respectively.

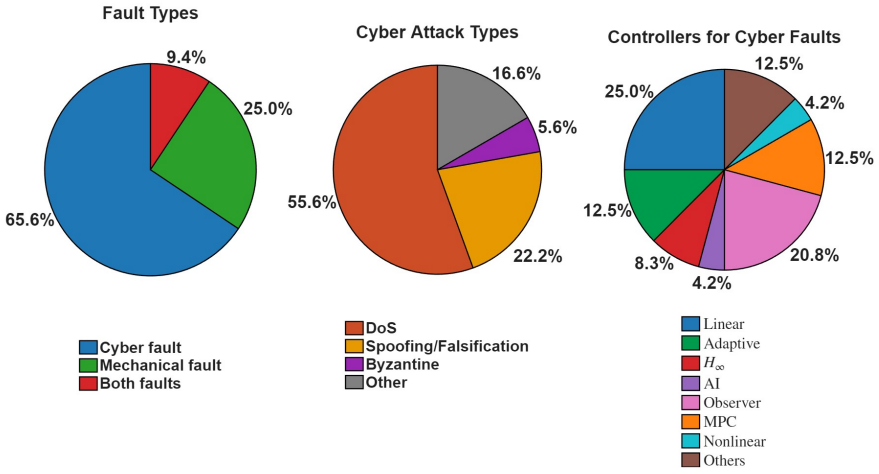


Figure 11. Fault/attack overview: (**left**)—fault types; (**centre**)—cyber attack types; (**right**)—controllers for cyber faults.

Table 25. Fault-type vs. controller family mapping for fault mitigation.

Fault Type	Fault Modelled as	Controller							
		Linear	Adaptive	MPC	H_∞	AI	Nonlinear	Observer	Others
Cyber	Additive faulty signal in control law	[152,186]	[82,172]	[123,191]	—	[62]	[194]	[176,187]	[128,214]
	Comm. delay in control law	[52,76]	[124]	—	—	—	—	—	[121]
	Comm. delay and Additive signal in control law	[63]	—	—	—	—	—	—	—
	Switching topology or intermittent comm.	—	—	[127]	[162,182]	—	—	[122]	—
Mechanical	Additive faulty signal in control law	[86,96,161, 171,195]	—	—	—	—	—	[175,202,218]	—
Cyber and Mechanical	Additive faulty signal in control law	[181]	—	—	—	—	—	—	—
	Comm. delay and Additive signal in control law	—	—	—	—	—	—	[150,179]	—

6. Platoon Performance Analysis and Stability Metrics

6.1. Internal Stability

Internal stability of a vehicle platoon is a property that ensures that the state trajectories of all the vehicles asymptotically converge with time. This attribute is fundamental to the platoon system's stable operation. It makes sure that the platoon objectives, namely spacing error, velocity error and acceleration error, converge to a steady state (or zero) as time goes to infinity [8]. The internal stability of the platoon is analysed in four main ways: (i) Routh–Hurwitz stability criterion; (ii) basic Lyapunov stability; (iii) Lyapunov–Krasovskii stability; (iv) Lyapunov–Razumikhin stability [8,70].

(i). *Routh–Hurwitz Stability Criterion*
The Routh–Hurwitz criterion is one of the most commonly used criteria for finding stability. It determines the platoon's internal stability by calculating the characteristic matrix (or the characteristic equation in the case of the platoon's decomposition into its individual subsystems) and examining its eigenvalues. If the real part of all the eigenvalues is negative, then the platoon is considered internally stable [40,71,88,137]. The closed-loop controlled dynamics of the full platoon system can be expressed in matrix form as:

$$\dot{X} = \big(I_N \otimes A - \mathcal{H} \otimes BK\big)X, \tag{39}$$

where $X = \mathrm{col}(x_1, \ldots, x_N)$ is a vector of follower states (or error states), and $\mathcal{H}$ is the Information matrix [71] and K is the controller gain. The decomposition of the closed-loop platoon system having N vehicles depends on the property of $\mathcal{H}$. If $\mathcal{H}$ has real, distinct eigenvalues (in case of undirected topologies [71]), then $\mathcal{H}$ decomposes into N independent modes of eigenvalue λ_i.

$$\dot{e}_i = \big(A - \lambda_i BK\big)e_i, \qquad i = 1, \ldots, N, \tag{40}$$

where e_i is the state error. Each mode has its own characteristic polynomial $p_i(s) = \det\big(sI - (A - \lambda_i BK)\big)$; the Routh–Hurwitz test is applied to each $p_i(s)$ to ensure $\Re(s) < 0$ [40,71]. If $\mathcal{H}$ has repeated real eigenvalues or complex eigenvalues (directed topologies [12]), the decomposition is not directly applicable, and the stability test is carried out using an equivalent real-domain representation (or by forming the characteristic matrix of the closed loop platoon), as detailed in [12].

(ii). *Basic Lyapunov Stability*
This approach to stability analysis constructs a function similar to an energy, called a Lyapunov function, to evaluate the stability of the platoon in the vicinity of an equilibrium point. If the time derivative of this Lyapunov function is negative in the neighbourhood of the equilibrium, then the platoon is said to be asymptotically stable [37,88,136]. It is better if the platoon system does not experience any time delay; otherwise, modified forms of Lyapunov stability are used to address time delays. Consider the closed-loop platoon (error) model $\dot{X}(t) = A_{\mathrm{cl}}X(t)$, where $X(t) \in \mathbb{R}^n$ denotes the stacked platoon state-error vector, $A_{\mathrm{cl}} \in \mathbb{R}^{n\times n}$ is the closed-loop system matrix. A standard Lyapunov candidate is

$$V(X) = X^\top PX, \qquad P = P^\top \succ 0, \tag{41}$$

where $V(X) \in \mathbb{R}$ is the Lyapunov function, $P \in \mathbb{R}^{n\times n}$ is a symmetric positive definite matrix (i.e., $\succ 0$ denotes positive definiteness). This gives $\dot{V}(X) = X^\top(A_{\mathrm{cl}}^\top P + PA_{\mathrm{cl}})X$, where $\dot{V}(X)$ is the time derivative of $V(X)$. Hence, internal stability follows if there exists $P \succ 0$, such that

$$A_{\mathrm{cl}}^\top P + PA_{\mathrm{cl}} \prec 0, \tag{42}$$

where $\prec 0$ denotes negative definiteness, implying that $X(t) \to 0$ as $t \to \infty$ and thus the platoon state errors converge to zero [37,88].

(iii). *Lyapunov–Krasovskii Stability*

If a vehicle platoon experiences communication time delays (constant or varying), then one of the powerful methods of determining internal stability is by the Lyapunov–Krasovskii stability method. In this method of stability, Lyapunov–Krasovskii (LK) functionals are employed. These functionals include delay-dependent terms in addition to the basic Lyapunov function. The addition of extra terms captures the effect of past vehicle states on current stability [37,70]. A delayed closed loop platoon model can be represented as $\dot{X}(t) = A_0X(t) + A_dX(t-\phi)$, where $X(t) \in \mathbb{R}^n$ denotes the stacked platoon state-error vector, $A_0, A_d \in \mathbb{R}^{n\times n}$ are constant system matrices, and $\phi > 0$ is the communication delay, a common LK functional is

$$V(t) = X^\top(t)PX(t) + \int_{t-\phi}^{t} X^\top(\theta)QX(\theta)\,d\theta, \qquad P \succ 0,\ Q \succeq 0, \tag{43}$$

where $P \in \mathbb{R}^{n\times n}$ is symmetric positive definite, $Q \in \mathbb{R}^{n\times n}$ is symmetric positive semidefinite, and θ is the integration variable. Sufficient internal-stability conditions are obtained by enforcing $\dot{V}(t) < 0$ through matrix inequalities [37,70].

(iv). *Lyapunov–Razumikhin Stability*

Lyapunov–Razumikhin-based stability method is another alternative approach to analyse the stability of a platoon in times of communication time delays. In this method of stability, a basic Lyapunov function is constructed along a condition that relates the delayed platoon state to the current platoon state [37,70]. A typical Lyapunov–Razumikhin starts with $V(X) = X^\top PX$, $P \succ 0$, and thereby the Razumikhin condition is imposed. It is shown as

$$V\big(X(t-\theta)\big) \le \rho\, V\big(X(t)\big), \qquad \forall\, \theta \in [0,\phi],\ \rho > 1, \tag{44}$$

where $\phi \in \mathbb{R}_+$ denotes the communication time delay, and $\rho \in \mathbb{R}$ is a chosen scalar value. In the Lyapunov–Razumikhin stability approach $\dot{V}(X(t))$ is upper-bounded by a negative expression, yielding internal stability under delay [37,70]. Compared with the Lyapunov–Krasovskii method for analysing stability, the Lyapunov–Razumikhin method is easier to implement and computationally lighter; however, the latter yields more conservative results [266].

Note: When communication or actuation delays are present, internal stability is analysed using either time-domain or frequency-domain methodologies. The common time-domain methodologies include Lyapunov–Krasovskii functionals and Lyapunov–Razumikhin methods (discussed before), while popular frequency-domain methodologies include the Routh-Hurwitz Stability criterion (also discussed before). Other frequency-domain methods, such as Cluster Treatment of Characteristic Roots (CTCR) [40], the general Nyquist criterion [267] (these are not discussed), along with other less prominent time domain techniques, are placed in the "Others" category in Table 26.

From Table 26, it can be seen that the Routh–Hurwitz criterion is less in use, whereas the frequently employed method for stability analysis is the Lyapunov-based family of methods. This preference stems from the fact that the Routh–Hurwitz criterion is primarily used for linear time-invariant (LTI) models and is not applicable to platoons with nonlinear dynamics at each node. In communication-time-delay vehicle platoons, the Lyapunov–Krasovskii functional is often preferred over the Lyapunov–Razumikhin approach due to the latter's conservative results.

Table 26. Internal stability methodologies. Publications with more than two stability analysis methods are denoted by (*).

Methodologies	References
Routh–Hurwitz	[36,40,43,71,73,74,79,85,88,104,115,137,139,142,145,149,159,165,186, 192,196,219]
Basic Lyapunov	[12,16,39,47,51,55,59,60,62,63,70,75,86,91,92,94–100,102,103,106,107,116,117,119,120,122,125,127,132,134,143,146,153, 154,157,160,162,168,171,173,174,176,177,179,180,182,183,186–188,191,194,195,197,199,203,204,206,208,209,212,213,220,223]
Lyapunov–Krasovskii	[83] *, [17,41,42,49,52,76,78,87,89,105,108,111,121,124,131,135,136,144, 147,150,156,161,163,169,170,172,184,190,210,221]
Lyapunov–Razumikhin	[83] *, [123,138,167]
Others	[15,61,66,72,81,85,112,118,126,130,140,152,193,214,217,218]

6.2. String Stability

String stability is a performance characteristic of vehicle platoons that ensures that state errors, i.e., spacing, velocity or acceleration, do not grow as they travel along the vehicle string. Depending on the method of analysing string stability, it is generally categorised into: (i) Lyapunov-based, (ii) input–output-based, and (iii) input–state-based [20].

(i). *Lyapunov-based*

The string stability, when analysed using a Lyapunov-based method, treats the platoon error as an unforced system and focuses on the effect of initial perturbations [20]. Let $\boldsymbol{e}(t) = \mathrm{col}(e_1(t), \ldots, e_N(t))$ be a collection of individual errors (e.g., vector of spacing errors) of all follower vehicles, where $\mathrm{col}(\cdot)$ is column-stacking, N is the number of follower vehicles and $e_i(t) \in \mathbb{R}^m$ is the individual error for vehicle i. Distinct from the method of analysis for internal stability using Lyapunov, which uses one Lyapunov matrix for the entire platoon state, Lyapunov-based string stability uses a composite Lyapunov function, which is a sum of individual vehicle contributions:

$$V(\boldsymbol{e}) = \sum_{i=1}^{N} e_i^\top P e_i, \qquad P \succ 0, \tag{45}$$

where $P \in \mathbb{R}^{m \times m}$ is constant. Then the Lyapunov function for the closed-loop platoon derivative should satisfy a uniform decay bound as

$$\dot{V}(\boldsymbol{e}(t)) \leq -\alpha \sum_{i=1}^{N} \|e_i(t)\|^2, \qquad \alpha > 0, \tag{46}$$

where $\alpha \in \mathbb{R}$ is constant and $\|\cdot\|$ denotes the Euclidean norm. where the constants P and α do not depend on the platoon length N. The inequalities above are obtained by substituting the closed-loop error model into $\dot{V}$ and enforcing negativity via matrix inequalities during controller tuning [20].

(ii). *Input-to-output (IO)*

The input-to-output method treats string stability as a disturbance-to-output attenuation property, neglecting initial conditions of the platoon system [20]. It is commonly used in the frequency domain. It is often analysed assuming that the vehicles are communicating in a PF topology (even if the entire platoon is not). Defining the error propagation transfer function for the vehicle i

$$H(s) = \frac{E_i(s)}{E_{i-1}(s)}, \tag{47}$$

where s is the Laplace variable and $E_i(s)$ is the Laplace transform of the error of vehicle i with respect to its predecessor. The controller is IO string-stable if

$$\|H(j\omega)\|_\infty = \sup_{\omega \in \mathbb{R}} |H(j\omega)| \leq 1, \tag{48}$$

where $\omega \in \mathbb{R}$ is the angular frequency, and $\|\cdot\|_\infty$ denotes the $\mathcal{H}_\infty$ norm. The key step is to derive $H(s)$ from the closed-loop linear model (vehicle dynamics, controller, and the selected error signal), and then verify or enforce the $\mathcal{H}_\infty$ bound to be string stable according to the condition in (48) [20,268].

(iii). *Input-to-state (IS)*

Input-to-state string stability accounts for both initial and persistent external perturbations. A platoon is ISS string-stable if there are a class $\mathcal{KL}$ function β and a class $\mathcal{K}$ function γ for which

$$\|e_s(t)\|_p \leq \beta\big(\|e_s(0)\|_p, t\big) + \gamma\big(\|\epsilon_s\|_{\mathcal{L}_\infty[0,t]}\big), \qquad \forall t \geq 0, \tag{49}$$

where $e_s(t)$ is the vector of string errors and ϵ_s are the disturbances, $\|\cdot\|_p$ is the ℓ_p-norm, $\|\cdot\|_{\mathcal{L}_\infty[0,t]}$ is the supremum norm over the interval $[0, t]$, and $\|\cdot\|$ denotes the Euclidean norm; $\mathcal{K}$ and $\mathcal{KL}$ are standard classes of comparison functions, and $\mathcal{K}_\infty$ denotes the class of $\mathcal{K}$ functions that are unbounded. A typical route is to build an ISS–Lyapunov function $V(e_s)$ satisfying for some class $\mathcal{K}_\infty$ functions $\underline{\alpha}, \overline{\alpha}, \alpha, \sigma$ (i.e., comparison functions),

$$\underline{\alpha}(\|e_s\|) \leq V(e_s) \leq \overline{\alpha}(\|e_s\|), \qquad \dot{V}(e_s) \leq -\alpha(\|e_s\|) + \sigma(\|\epsilon_s\|). \tag{50}$$

Using comparison arguments, this differential inequality directly gives an ISS bound of the form above, which ensures that the disturbances will not cause unbounded growth of string errors, and that the effect of the initial conditions will decay with time in the form of [20].

A recent trend is to include string stability constraints in the controller synthesis process, rather than performing the check $\|H(j\omega)\|_\infty \leq 1$ after the controller is developed. In particular, work in [269] accounted for heterogeneous platoons with network disconnections and switching information patterns. Their approach imposes an explicit non-amplification constraint on successive spacing errors and incorporates it into an LMI-based robust adaptive output–feedback design, thus enforcing stability, robustness, and string non-amplification simultaneously at design time.

Table 27 classifies the literature according to the above three definitions of string stability. The table indicates that frequency-domain IOSS is the most widely used method for evaluating string stability. This means that most works assume a PF topology in proving the platoon string stability of the vehicles. Other time-domain methods, i.e., LSS and ISSS, are still limited.

Table 27. String stability methodologies.

Methodologies	References
Lyapunov-based string stability	[82,143,183,192,205]
Input-to-output string stability	[36,38,40,51,63,80,83,85,99,100,102,103,105,111,112,135,138, 140,142,144,146,147,151,158,160,164,167,176,180,181,184,185, 190,193,214]
Input-to-State String Stability	[55,98,173,204,209]

6.3. Robust Stability

Robust stability for the vehicle platoon is the ability of the controlled platoon to maintain the desired performance even in the presence of uncertainties and external disturbances [15]. Some factors that challenge the robust stability of the platoon are:

(i). *Communication imperfections:* This category is discussed in detail in Sections 3.3 and 5. Examples of this category include communication delays [163], packet drops [15], and malicious cyberattacks [162].
(ii). *External disturbances:* This category is discussed in detail in Section 3. Examples of this category include wind gusts, road friction, surface gradient [12,92].
(iii). *Model uncertainties:* This type of imperfection arises from model mismatches, i.e., while carrying out linearisation or if there is some unmodelled dynamics (both types are explained briefly in Section 3). Some examples of work are [92,95].

Below are the two common methods to assess robust stability in the platoon:

(i). Matrix Inequality-Based Analysis (LMI and Riccati Methods)

This is the most common approach for assessing the robustness of the platoon system. The robust stability problem is transformed into a set of solvable matrix inequalities to meet the H_∞ performance criterion [15]. Hence, if the energy of the tracking errors remains proportionally smaller than that of the disturbances up to a certain attenuation level, the platoon is termed as robustly stable [12]. A solution of these inequalities gives the robust gains [15].

(ii). Input-to-State Stability (ISS) Analysis

The input-to-state stability (ISS) approach is based on the concept that bounded disturbance inputs lead to the bounded state errors [170]. Hence, this method for assessing the robustness of the system constructs a Lyapunov function and computes its time derivative. The Lyapunov function is chosen such that this derivative is negative whenever the magnitude of the system's state is sufficiently larger than the disturbance magnitude. From this type of approach, it is proven that errors would not grow but will be bounded provided the disturbances are bounded [170].

Table 28 outlines the methodologies used to evaluate the robust stability of platoon systems. The use of Matrix Inequalities, including linear matrix inequalities (LMIs) and Riccati methods, is a popular approach to testing robust stability. The ISS analysis, although less comprehensive than the former approach, also appears quite frequently in the literature.

Table 28. Robust stability methodologies.

Methodologies	References
Matrix Inequality-Based Analysis (LMI and Riccati Methods)	[12,15–17,80,92,112,121,127,161–163,182,184,221]
Input-to-State Stability (ISS) Analysis	[95,96,159,170,185,186,200,222]

7. Simulation and Experimental Validation

The transition from theoretical control design to real-world deployment requires rigorous validation. However, before deployment of the algorithms on the real-time hardware, the software platforms are used to test and validate their effectiveness; by emulating similar situations of real world. The primary platforms used in the literature are critically compared below in terms of realism, scalability, and implementation constraints.

(i) MATLAB/Simulink

MATLAB/Simulink is currently the most popular for validating platoon algorithms because it provides a mature environment for modelling dynamics, synthesising controllers, and conducting closed-loop validation.

- Realism: It shows a high realism for the control loop fidelity if the modules of the vehicle, the actuator and the sensors are accurately modelled. It contains vehicle-modelling tool chains for making it realistic; however, the final level of realism is determined by model detail and parameter calibration selection [270].
- Scalability: It supports simulation of small to medium-length platoons. Extension to larger platoons is possible with model simplification; however, fine nonlinear dynamics, stiff solvers and small time steps add considerable computational cost and time.
- Constraints: Licensing restrictions and the availability of tools can restrict reproducibility. Real-time and hardware-oriented testing usually needs more real-time execution support (e.g., Simulink Real-Timing) and careful timing/IO configuration [271].

(ii) Simulation of Urban Mobility (SUMO)

SUMO is a microscopic open-source traffic simulator for large road networks. It simulates realistic traffic interactions.

- Realism: It shows strong realism at the level of traffic-interaction (network flows, merging, intersections, and mixed traffic). Nevertheless, vehicle dynamics are typically simplified, so that low-level control validation is typically performed using co-simulation.
- Scalability: SUMO is well-suited to network-scale simulation of large numbers of vehicles, making it attractive for analysing platoons embedded in larger traffic networks.
- Constraints: Platooning control laws are not commonly designed within SUMO, but external controllers are coupled via interfaces such as TraCI; this introduces integration effort and requires the user to carefully consider what is being validated (controller validation, traffic-level interactions) [272,273].

(iii) Plexe

Plexe is specifically oriented to the platooning simulation framework. It is employed when the communication effects are an integral part of the validation. Thus, it not only validates controllers but also network performance. It is typically run in combination with mobility and network simulation environments.

- Realism: Realism is high for cooperative platooning functions if communication and mobility are represented together (e.g., communication performance like delays, latency, as well as vehicle motion), thereby enabling more implementation-relevant assessment than control-only simulations.
- Scalability: It supports moderate platoon sizes. However, larger scenarios are possible but are limited by the overhead of discrete-event network simulation and the complexity of multi-tool co-simulation.
- Constraints: Practical barriers include the engineering overhead of co-simulation software, cross-software parameter consistency (mobility, networking, and controller settings), and reproducibility and dependencies across software. The basis for such network mobility studies using the OMNeT++-SUMO co-simulation is summarised in the Veins documentation [274,275].

(iv) CARLA (Car Learning to Act)

CARLA is an open-source autonomous driving simulator that aims at providing high-fidelity 3D environments and sensor simulation.

- Realism: It provides the highest realism for scenario and perception layers, including controllable environments and sensor simulation, which is valuable when platooning algorithms are working with perception/localisation stacks or when they demand scenarios beyond abstract traffic models [276,277]
- Scalability: It supports limited traffic scaled when compared to traffic-only simulators due to the computational demands of rendering and sensor pipelines
- Constraints: Some of the common constraints are computation constraints, workload of the scenario authoring system, integration effort required for co-simulation with external control toolchains (if controllers are not implemented on CARLA) [276].

(v) PreScan

PreScan is a commercial platform for scenario-based testing and autonomous driving validation, including research on sensors.

- Realism: Realism is good for sensor-based validation workflows that can improve application relevance in comparison with only control algorithm validations.
- Scalability: It supports moderate fidelity, as when the scenario is complex and high-fidelity sensors are simulated, the number of vehicles is reduced.
- Constraints: Commercial licensing and proprietary components reduce accessibility. scenario and sensing constraints, as illustrated by works that adopt it for platooning-related validation [190].

In addition to the platforms mentioned above, a few lesser-known simulation tools are also used. In contrast to fully simulated validation, experimental configurations provide more direct evidence of feasibility in real time by subjecting the sensing and actuation limits, timing constraints, and communication artefacts to scrutiny that are difficult to capture perfectly in software. Table 29 classifies the reviewed studies based on their validation (simulation-based or hardware-based) and then sub-classifies based on the particular platform used for the validation. Simulation-based validation easily outweighs hardware-based validation; the most common approach is MATLAB/Simulink, due to its low barrier to control development and its ability to support repeatable testing. Traffic and networking-oriented tools (e.g., SUMO and Plexe) are usually chosen when realistic network-scale dynamics or communication effects are of primary importance, rather than for controller prototyping. Although MATLAB/Simulink is still the most commonly used platform for validating platooning control laws, less commonly used are traffic-simulation environments such as SUMO and Plexe, which primarily model network-level traffic interactions and are not designed for detailed controller-oriented vehicle-dynamics validation. However, SUMO and Plexe offer greater realism in inter-vehicle communication within a fleet, and their wider use is recommended for future research studies. Among platoon sizes used for validation, platoons between five and ten vehicles are most frequently tested in simulation, whereas hardware validations are commonly limited to fewer than five vehicles due to cost, safety, and infrastructure constraints.

Table 29. Numerical analysis. Publications with more than one simulation or co-combined with experimental setup are marked with (*).

Cases	Simulation Platform	Platoon Length (Incl. Leader)	References
Only simulations (platform used)	MATLAB/Simulink	<5	[18,65,86,188]
		5–10	[92,109,147,161,192] *, [43,45,47,51,75,77,84,87,96,108,111,121,122,131,136,144, 160,163,167,176,181,185,191,197,198,200,201,203,205]
		>10	[15,17,80,112,113,151,157], [109] *
	PLEXE	<5	—
		5–10	[41,83,105,110,124,129]
		>10	—
	Other simulators	<5	[85,220] *, [202]
		5–10	[92,147,192] *, [89,91,132,148,165,187,193]
		>10	[178]
	Not mentioned simulators	<5	[97] *, [38,62,76,78,90,128,164,168,171,174,175,195,208,216]
		5–10	[117,159,179,180] *, [14,36,37,42,52,54,55,59–61,63,64,66,67,71–74,82,88,93–95,98–100,102–104,115,116,118,119,123,125,130,133,134,138,140,142,145,149, 150,154,158,162,170,172,173,182–184,186,189,194,196,206,209,210,212,214,215, 217–219,221–223]
		>10	[117,180,220] *, [12,39,70,81,126,137,141,146,152,156,169,207]
		Number not mentioned	[16,49,53,79,139]
Experimental	Hardware setup	<5	[106,114,204], [85,97,159,161] *
		5–10	[179] *, [120,135,143,190]
		>10	[213]

8. Technology Readiness and Barrier Deployment

While much of the contemporary research on platoon control remains theoretical, there are hardware setups too for the control algorithms, as shown in Table 29.

8.1. Technology Readiness

The experimental evidence from the literature reviewed indicates that the readiness of different families of controllers is uneven, and is largely determined by the computational load, the sensing/communication requirements, and their tolerance to non-ideal networked operation.

(i). Distributed Linear Control:

Distributed linear controllers are the best candidates for immediate deployment due to their compatibility with the constraints imposed by conventional automotive electronic control units (ECUs). They are also easily applied and require very little computation as compared to other controllers. Large-scale experimental trials conducted on a platoon of three passenger vehicles (Toyota Prii) with V2V communications, coupled with radar and camera sensors, provide evidence of the efficacy of these controllers with realistic actuation dynamics and delay-aware modelling [114]. In low-speed service scenarios, linear distributed control has also been validated with unmanned ground vehicle (UGV) platoons; the experimental results have been obtained using traditional autonomous system elements, such as LiDAR and cameras, differential GPS/INS, and wireless communication links [120].

(ii). Adaptive and Learning-Based Control:

The next to be deployed after linear controllers are distributed adaptive controllers. More precisely, adaptive with learning-based methodologies. They have been validated on small-to-medium testbeds. For example, experimental studies based on fleets of several robots with embedded controllers and short-range wireless communication have supported the feasibility of these techniques under tight onboard constraints, but such studies have been at a much smaller size than that used in heavy-duty vehicle platooning (as compared to the linear controllers) [143].

(iii). Model Predictive Control (MPC):

Once practised as theoretical controllers, MPC has advanced. MPC-based platoon control has been demonstrated on early hardware, combining it with state estimation and communications-aware design. Real-vehicle investigations using an onboard 5G-V2V computing node have shown real-time optimisation with latencies down to 15 ms, and performance under artificially induced V2V disconnection conditions [213]. Other DMPC formulations in this collection are still limited to simulation, and thus more in the proof-of-concept stage [134].

8.2. Barriers to Deployment

Across the hardware studies, three barriers to deployment stand out, which are: (i) network imperfections, which comprise delay, loss, and cyber attack, (ii) heterogeneity paired with actuator and safety constraints, and (iii) limitations in sensing and estimation, which adds extra latency.

(i). Network Constraints and Stability Margins:

A significant amount of theoretical literature assumes that V2X channels are constant-delay links; however, hardware measurements contradict this assumption. Experimental studies on autonomous-robot platoons show that apart from the magnitude of the delay, the frequency characteristics of the delay can also affect the stability [85]. Reliability and security also matter in practice. Fault-tolerant consensus has been tested on a five-vehicle unmanned platform under DoS attacks, and it has shown that the attack models, characteristics of packet loss and recovery mechanisms need to be co-designed with the control architecture [179]. Concurrently, periodical event-triggered fault detection has been tested on a two-vehicle WiFi-based communication link, thus highlighting the need for bandwidth-conscious safety layers [161].

(ii). Heterogeneity and Saturation:

Real-world platoon configurations exhibit heterogeneity due to powertrain constraints and safety regulations, leading to actuator saturation. Experiments carried out with full-scale passenger cars show that velocity restrictions reduce platoon cohesion, thus making topological designs and controllers that can account for constrained followers rather than uniform feasible speed profiles necessary [114]. Heterogeneity is also treated experimentally at smaller scales: variable time-headway policy control is validated on intelligent micro-vehicle platforms, supporting the view that flexible spacing policies are more realistic for heterogeneous connected vehicles under disturbances than rigid constant-spacing assumptions [190].

(iii). Sensor Accuracy and Estimation:

Empirical studies of hardware show that the limits of sensing and estimation are key factors in determining system stability. Experiments with mini-platoon using ultra-wideband (UWB) positioning have revealed that jitter and positioning errors are reflected in variations in tracking performance and have therefore led to the use of filtering techniques, estimation procedures, or robustness measures [97,204]. In larger-scale tests, network-aware control integrates state estimation to sustain performance during V2V degradation, implying an estimation layer is necessary for deployment [213].

9. Conclusions and Future Research Directions

9.1. Conclusions

This review provides a full overview of distributed longitudinal control methods for autonomous vehicle platoons, from 2013 to 2025. The surveyed literature is organised systematically into a four-component framework: node dynamics, formation geometry, in-

formation flow topology, and distributed controllers. The overall synthesis suggests that the suitability of any given controller is inherently application-dependent and is essentially determined by modelling uncertainty, spatial arrangements, and communication conditions.

(i). Topology and Performance Domains

Information flow topology is a major factor affecting platoon behaviour, as it determines the distribution of closed-loop eigenvalues, which in turn affects stability margins, scalability, convergence rates, and robustness. It is worth noting that leader-pinned communication topologies, such as BDL and PFL, exhibit more favourable eigenvalue properties than unpinned topologies, such as BD and PF, thereby enhancing stability and alleviating performance degradation as the platoon length grows.

(ii). Robustness and Latency Management

Under delays, packet losses, and bandwidth limitations, the most reliable approaches are delay-aware robust designs (commonly Lyapunov–Krasovskii/LMI-based and H_∞-type methods), often supported by observers and event-triggered communication when continuous data exchange is not feasible. Conservative spacing policies remain a practical stabilising layer when communication quality degrades. In the context of delays, packet losses, bandwidth limitations, and switching topologies, the most reliable approaches are delay-aware robust designs (mostly of Lyapunov–Krasovskii/LMI and H_∞-type), often complemented by observers and event-triggered communication when continuous data exchange is infeasible. Conservative spacing policies, such as CD or CTH, remain a practical approach in vehicle platoon operations.

(iii). Heterogeneity, Disturbances, and Model Uncertainty.

For heterogeneous platoons and uncertain dynamics, adaptive and sliding-mode controllers provide robust resilience against disturbances and parameter variations. On the other hand, MPC-based designs are best suited when explicit constraint handling, e.g., safety distances and actuator limits, is of prime importance, despite their increased computational effort.

(iv). Validation and Technology Readiness.

Validation efforts tend to be dominated by simulation-based research over hardware setups. There is a prevalence of mathematical software, primarily Matlab/Simulink, being the main platform for controller development. SUMO/Plexe-based environments are preferred when traffic scale, interactions, or communication effects are central. Hardware experiments, however, are still relatively limited due to cost, safety, and infrastructure requirements; therefore, there is a need for more robust translation from theory to hardware.

Finally, the review paper supports a consistent practical hierarchy: topology choices that improve eigenvalue properties (especially leader pinning), combined with low-complexity robust controllers and realistic validation settings, provide the most credible route towards scalable and deployable platooning under non-ideal communication and uncertain modelling conditions.

9.2. Future Research Directions

(i). Unified Control Frameworks for Realistic Multi-Factor Scenarios

A prominent avenue for subsequent research is the development of unified control frameworks that can simultaneously address a range of operational challenges rather than addressing each challenge individually. Current investigations into distributed controllers tend to focus on external disturbances, uncertainty, nonlinearities, heterogeneous platoons, imperfect communication, and cyberattacks, either in isolation or in simplified combinations. The next generation of platoon controllers should have a holistic view and obtain

control laws that simultaneously account for the main contingencies faced in realistic operating scenarios. Such an integrative approach would significantly improve the resilience of platoon systems to the full range of deployment time uncertainties. Moreover, while heterogeneity in vehicle dynamics has been well studied (summarised in Table 2), the heterogeneity of connectivity, i.e., differences in communication capabilities, reception ranges, or access to leader information, is largely neglected. Future investigations should systematically test the effect of asymmetric information availability on string stability and develop adaptive reconfiguration strategies tailored to variable network topologies.

(ii). Formal Verification and Safety Guarantees for Learning-Based Methods

The development of data-driven and artificial intelligence-based control methods is a promising direction in the design of distributed controllers. As shown in Figure 9, the usage of AI-based approaches has shown a constant growth in the previous five years (from 5% to 12% of recent publications). Reinforcement-learning- and neural-network-based controllers provide excellent adaptation capabilities to complex, multi-dimensional environments. Nevertheless, they poorly meet formal guarantees of stability and safety at both the training and the deployed stages. To overcome this limitation, future work should incorporate the control theoretic guarantees, such as control barrier functions (CBFs) that provide forward-invariance certificates for safety-critical constraints, and reachability-analysis tools aimed at verifying bounded behaviour under worst-case disturbances. A particularly promising approach involves providing reinforcement learning policies with safety shields based on CBF that only intervene when unsafe actions are proposed, thereby preserving the flexibility of learning while maintaining collision avoidance and separation constraints. Beyond classical Lyapunov and LMI techniques, emerging verification frameworks, such as sum-of-squares programming, contraction theory, and neural Lyapunov methods, should be systematically explored to certify AI-based platoon controllers.

(iii). Cyber-Physical Resilience

In parallel, cybersecurity research must advance beyond isolated attack cases (e.g., Denial-of-Service only) towards resilient frameworks that can address combinations of cyber threats, such as Denial-of-Service, Byzantine faults, and spoofing attacks. Such frameworks should integrate attack-resilient observers with attack-detection, isolation, and reconfiguration protocols so that, under adversarial conditions, systems degrade slowly rather than fail catastrophically. A largely unexplored domain is the potential of fractional-order control, which provides long-memory properties that may offer greater flexibility for compensating for communication delays and external disturbances than integer-order counterparts.

Author Contributions: Conceptualisation, O.H. and U.M.; Methodology, O.H. and U.M.; Investigation, O.H.; Data curation, O.H.; Writing—original draft preparation, O.H.; Writing—review and editing, O.H., U.M., P.G. and A.S.; Supervision, U.M., P.G. and A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Engineering and Physical Sciences Research Council (EPSRC), UK, Project Reference Number 2757376.

Data Availability Statement: No new data were created or analysed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study, in the collection, analysis, or interpretation of data, in the writing of the manuscript, or in the decision to publish the results.

Appendix A. Network Synchronisation Framework

This appendix summarises the theoretical background behind network synchronisation for multi-agent systems (MAS). Consider a leader agent with state $x_0(t) \in \mathbb{R}^n$ and N follower agents with states $x_i(t) \in \mathbb{R}^n$, $i \in \{1, \ldots, N\}$. Each follower has dynamics $\dot{x}_i = f(x_i)$, where $f : \mathbb{R}^n \to \mathbb{R}^n$ is a vector field. Communication between agents is represented by a directed graph, where an edge (j, i) means that agent i receives information from agent j.

Appendix A.1. Modelling Network via Graph Theory

Let $\tilde{a}_{ij} \geq 0$ be the (possibly weighted) adjacency weight from agent j to agent i for $i \neq j$, and define the neighbour set

$$\mathcal{N}_i = \{\, j \in \{1, \ldots, N\} \mid \tilde{a}_{ij} > 0 \,\}. \tag{A1}$$

The (in-degree) Laplacian $\mathcal{L} = [\mathcal{L}_{ij}] \in \mathbb{R}^{N \times N}$ is

$$\mathcal{L}_{ij} = \begin{cases} -\tilde{a}_{ij}, & i \neq j, \\ \sum\limits_{k \in \mathcal{N}_i} \tilde{a}_{ik}, & i = j, \end{cases} \qquad i, j \in \{1, \ldots, N\}. \tag{A2}$$

Pinning to the leader is described by $\mathcal{P} = \mathrm{diag}(p_1, \ldots, p_N)$ with $p_i \in \{0, 1\}$, where $p_i = 1$ indicates that follower i can access $x_0(t)$ directly. The corresponding Information matrix is $\mathcal{H} = \mathcal{L} + \mathcal{P}$.

Appendix A.2. Diffusive Coupling and Pinning Control

A commonly used nonlinear network model with *diffusive* coupling can be written in the compact "coupling-matrix" form (as in the pinning-control literature) by using a coupling configuration matrix whose rows sum to zero [278,279]:

$$\dot{x}_i(t) = f\big(x_i(t)\big) + c \sum_{j=1}^{N} \tilde{a}_{ij}\, \Gamma x_j(t) + u_i(t), \qquad i = 1, \ldots, N, \tag{A3}$$

where $c > 0$ is the coupling strength and $\Gamma \in \mathbb{R}^{n \times n}$ is an inner-coupling (selection) matrix that specifies which state components are coupled. Here, the diagonal entries are chosen as $\tilde{a}_{ii} = -\sum_{j \in \mathcal{N}_i} \tilde{a}_{ij}$ so that $\sum_{j=1}^{N} \tilde{a}_{ij} = 0$, making (A3) equivalent to a standard diffusive term $c \sum_{j \in \mathcal{N}_i} \tilde{a}_{ij} \Gamma\big(x_j - x_i\big)$ [278,279].

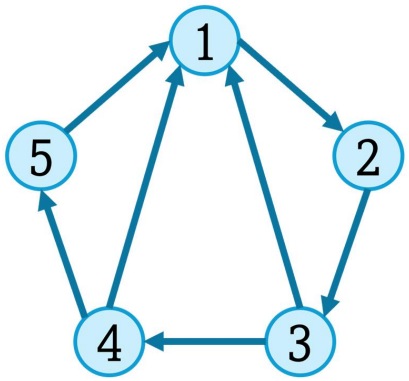


Figure A1. Example follower communication digraph used for pinned leader–follower synchronisation (pinning is encoded by $p_i \in \{0, 1\}$).

For leader–follower synchronisation (pinning control), a typical local pinning law is

$$u_i(t) = -c\, k_p\, p_i\, \Gamma\big(x_i(t) - x_0(t)\big), \qquad i = 1, \ldots, N, \tag{A4}$$

where $k_p > 0$ is a pinning gain. The objective is leader–follower consensus, i.e., $x_i(t) \to x_0(t)$ as $t \to \infty$ for all followers using only neighbour information (via $\tilde{a}_{ij}$) plus leader information available to pinned agents (via p_i) [278–285].

References

1. Wang, Z.; Wu, G.; Barth, M.J. A Review on Cooperative Adaptive Cruise Control (CACC) Systems: Architectures, Controls, and Applications. In Proceedings of the 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 2884–2891.
2. United States Department of Transportation. NHTSA Estimates for 2022 Show Roadway Fatalities Remain Flat after Two Years of Dramatic Increases. Washington, DC, USA, April 2023. Available online: https://www.nhtsa.gov/press-releases/traffic-crash-death-estimates-2022 (accessed on 3 December 2025).
3. Brake, the Road Safety Charity. UK Collision and Casualty Statistics. United Kingdom, October 2023. Available online: https://www.brake.org.uk/get-involved/take-action/mybrake/knowledge-centre/uk-road-safety (accessed on 3 December 2025).
4. United States Environmental Protection Agency. Carbon Pollution From Transportation. May 2023. Available online: https://www.epa.gov/transportation-air-pollution-and-climate-change/carbon-pollution-transportation (accessed on 3 December 2025).
5. Department for Transport. Transport and Environment Statistics: 2023. United Kingdom, October 2023. Available online: https://www.gov.uk/government/statistics/transport-and-environment-statistics-2023/transport-and-environment-statistics-2023 (accessed on 3 December 2025).
6. Green, M. How Long Does It Take to Stop? Methodological Analysis of Driver Perception–Brake Times. *Transp. Hum. Factors* **2000**, *2*, 195–216. [CrossRef]
7. Rebelo, M.; Rafael, S.; Bandeira, J.M. Vehicle Platooning: A Detailed Literature Review on Environmental Impacts and Future Research Directions. *Future Transp.* **2024**, *4*, 591–607. [CrossRef]
8. Li, S.E.; Zheng, Y.; Li, K.; Wang, J. An Overview of Vehicular Platoon Control under the Four-Component Framework. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Seoul, Republic of Korea, 28 June–1 July 2015; pp. 286–291.
9. Fakhfakh, F.; Tounsi, M.; Mosbah, M. Vehicle Platooning Systems: Review, Classification and Validation Strategies. *Int. J. Netw. Distrib. Comput.* **2020**, *8*, 203–213. [CrossRef]
10. Nguyen, T.; Xie, M.; Liu, X.; Arunachalam, N.R.; Rau, A.; Lechner, B.; Busch, F.; Wong, Y.D. Platooning of Autonomous Public Transport Vehicles: The Influence of Ride Comfort on Travel Delay. *Sustainability* **2019**, *11*, 5237. [CrossRef]
11. Badnava, S.; Meskin, N.; Gastli, A.; Al-Hitmi, M.A.; Ghommam, J.; Mesbah, M.; Mnif, F. Platoon Transitional Maneuver Control System: A Review. *IEEE Access* **2021**, *9*, 88327–88347. [CrossRef]
12. Li, S.E.; Qin, X.; Zheng, Y.; Wang, J.; Li, K.; Zhang, H. Distributed Platoon Control under Topologies with Complex Eigenvalues: Stability Analysis and Controller Synthesis. *IEEE Trans. Control Syst. Technol.* **2019**, *27*, 206–220. [CrossRef]
13. Elahi, A.; Alfi, A.; Chadli, M. Fixed-Time H∞ Consensus Control of Uncertain Vehicular Platooning Systems with Homogeneous Time-Varying Actuator Delay. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 11716–11728. [CrossRef]
14. Xiao, S.; Ge, X.; Han, Q.-L.; Zhang, Y. Dynamic Event-Triggered Platooning Control of Automated Vehicles under Random Communication Topologies and Various Spacing Policies. *IEEE Trans. Cybern.* **2022**, *52*, 11477–11490. [CrossRef]
15. Halder, K.; Montanaro, U.; Dixit, S.; Dianati, M.; Mouzakitis, A.; Fallah, S. Distributed H∞ Controller Design and Robustness Analysis for Vehicle Platooning under Random Packet Drop. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 4373–4386. [CrossRef]
16. Zheng, Y.; Li, S.E.; Li, K.; Ren, W. Platooning of Connected Vehicles with Undirected Topologies: Robustness Analysis and Distributed H-Infinity Controller Synthesis. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 1353–1364. [CrossRef]
17. Halder, K.; Gillam, L.; Dixit, S.; Mouzakitis, A.; Fallah, S. Stability Analysis with LMI-Based Distributed H-Infinity Controller for Vehicle Platooning under Random Multiple Packet Drops. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 23517–23532. [CrossRef]
18. Yi, P.; Liao, J.; Zhao, D. Tube-Based Distributed Model Predictive Control for Connected Vehicle Platoons with Communication Delay. In Proceedings of the 43rd Chinese Control Conference, Kunming, China, 28–31 July 2024.
19. Li, S.E.; Zheng, Y.; Li, K.; Gao, F.; Wu, Y.; Zhang, H.; Hedrick, J.K. Dynamical Modeling and Distributed Control of Connected and Automated Vehicles: Challenges and Opportunities. *IEEE Intell. Transp. Syst. Mag.* **2017**, *9*, 46–58. [CrossRef]
20. Feng, S.; Zhang, Y.; Li, S.E.; Cao, Z.; Liu, H.X. String Stability for Vehicular Platoon Control: Definitions and Analysis Methods. *Annu. Rev. Control* **2019**, *47*, 81–97. [CrossRef]
21. Li, Q.; Chen, Z.; Li, X. A Review of Connected and Automated Vehicle Platoon Merging and Splitting Operations. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 22790–22806. [CrossRef]

22. Salek, M.S.; Basu Thakur, M.; Ala, P.S.K.; Chowdhury, M.; Schmid, M.; Murray-Tuite, P.; Khan, S.M.; Krovi, V. An Overview of Automated Vehicle Longitudinal Platoon Formation Strategies. *ACM J. Auton. Transp. Syst.* **2025**, *3*, 8. [CrossRef]
23. Nandhini, M.; Mohamed Rabik, M. A Comprehensive Overview of the Alignment between Platoon Control Approaches and Clustering Strategies. *Green Energy Intell. Transp.* **2024**, *3*, 100223. [CrossRef]
24. Viadero-Monasterio, F.; Meléndez-Useros, M.; Jiménez-Salas, M.; López Boada, B.; López Boada, M.J. Key Influencing Factors in Vehicle Platoons: A Systematic Study and Review. *Evol. Syst.* **2025**, *16*, 116. [CrossRef]
25. Chu, D.; Zhao, C.; Wang, R.; Xiao, Q.; Wang, W.; Cao, D. A Survey of Multi-Vehicle Consensus in Uncertain Networks for Autonomous Driving. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 19319–19341. [CrossRef]
26. Guanetti, J.; Kim, Y.; Borrelli, F. Control of Connected and Automated Vehicles: State of the Art and Future Challenges. *Annu. Rev. Control* **2018**, *45*, 18–40. [CrossRef]
27. Xie, S.; Li, Z.; Arvin, F.; Ding, Z. A Review of Multi-Vehicle Cooperative Control System in Intelligent Transportation. *Int. J. Automot. Manuf. Mater.* **2023**, *2*, 5. [CrossRef]
28. Soni, A.S.; Hu, H. Formation Control for a Fleet of Autonomous Ground Vehicles: A Survey. *Robotics* **2018**, *7*, 67. [CrossRef]
29. Montanaro, U.; Dixit, S.; Fallah, S.; Dianati, M.; Stevens, A.; Oxtoby, D.; Mouzakitis, A. Towards Connected Autonomous Driving: Review of Use-Cases. *Veh. Syst. Dyn.* **2019**, *57*, 779–814. [CrossRef]
30. Martínez-Díaza, M.; Al-Haddad, C.; Soriguera, F.; Antoniou, C. Platooning of Connected Automated Vehicles on Freeways: A Bird's Eye View. *Transp. Res. Procedia* **2021**, *58*, 479–486. [CrossRef]
31. Malik, S.; Khan, M.A.; El-Sayed, H. Collaborative Autonomous Driving—A Survey of Solution Approaches and Future Challenges. *Sensors* **2021**, *21*, 3783. [CrossRef] [PubMed]
32. Jia, D.; Lu, K.; Wang, J.; Zhang, X.; Shen, X. A Survey on Platoon-Based Vehicular Cyber-Physical Systems. *IEEE Commun. Surv. Tutor.* **2015**, *18*, 263–284. [CrossRef]
33. Lesch, V.; Breitbach, M.; Segata, M.; Becker, C.; Kounev, S.; Krupitzer, C. An Overview on Approaches for Coordination of Platoons. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 10049–10065. [CrossRef]
34. Schwarting, W.; Alonso-Mora, J.; Rus, D. Planning and Decision-Making for Autonomous Vehicles. *Annu. Rev. Control. Robot. Auton. Syst.* **2018**, *1*, 187–210. [CrossRef]
35. Wu, C.; Cai, Z.; He, Y.; Lu, X. A Review of Vehicle Group Intelligence in a Connected Environment. *IEEE Trans. Intell. Veh.* **2024**, *9*, 1865–1890. [CrossRef]
36. Li, H.; Chen, Z.; Fu, B.; Wu, Z.; Ji, X. Event-Triggered Vehicle Platoon Control under Random Communication Noises. *IEEE Access* **2021**, *9*, 51722–51735. [CrossRef]
37. Li, H.; Chen, Z.; Fu, B.; Wu, Z.; Ji, X. Consensus-Based Vehicle Platoon Control with Communication Noises. In Proceedings of the 40th Chinese Control Conference (CCC), Shanghai, China, 26–28 July 2021; pp. 7483–7488.
38. Guo, H.; Liu, C.-L. Observer-Based Consensus Protocol of Vehicles Platooning with Time Delay. In Proceedings of the 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 22–24 May 2021; pp. 4949–4954.
39. Ma, F.; Wang, J.; Yang, Y.; Wu, L.; Yang, L. *Control Synthesis for Distributed Vehicle Platoon Under Different Topological Communication Structures*; SAE Technical Paper 2019-01-0494; SAE: Washington, DC, USA, 2019.
40. Zhang, Z.; Zhu, X. Internal Stability and String Stability Analysis for Vehicle Platoon with Time Delays via the Distributed PID Controller. In Proceedings of the 41st Chinese Control Conference (CCC), Hefei, China, 25–27 July 2022; pp. 49–54.
41. Petrillo, A.; Salvi, A.; Santini, S.; Valente, A.S. Adaptive Multi-Agents Synchronization for Collaborative Driving of Autonomous Vehicles with Multiple Communication Delays. *Transp. Res. Part Emerg. Technol.* **2018**, *86*, 372–392. [CrossRef]
42. Elahi, A.; Alfi, A.; Modares, H. Distributed Consensus Control of Vehicular Platooning under Delay, Packet Dropout and Noise: Relative State and Relative Input–Output Control Strategies. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 20123–20133. [CrossRef]
43. Jond, H.B.; Yildiz, A. Connected and Automated Vehicle Platoon Formation Control via Differential Games. *IET Intell. Transp. Syst.* **2023**, *17*, 312–326. [CrossRef]
44. Viadero-Monasterio, F.; Meléndez-Useros, M.; Jiménez-Salas, M.; López Boada, M.J. Fault-Tolerant Robust Output-Feedback Control of a Vehicle Platoon Considering Measurement Noise and Road Disturbances. *IET Intell. Transp. Syst.* **2025**, *19*, e70007. [CrossRef]
45. Caiazzo, B.; Coppola, A.; Petrillo, A.; Santini, S. Distributed Nonlinear Model Predictive Control for Connected Autonomous Electric Vehicles Platoon with Distance-Dependent Air Drag Formulation. *Energies* **2021**, *14*, 5122. [CrossRef]
46. Feng, J.; Gao, Z.; Guo, B. State-Feedback and Nonsmooth Controller Design for Truck Platoon Subject to Uncertainties and Disturbances. *World Electr. Veh. J.* **2024**, *15*, 251. [CrossRef]
47. Coppola, A.; Lui, D.G.; Petrillo, A.; Santini, S. Distributed Fixed-Time Leader-Tracking Control for Heterogeneous Uncertain Autonomous Connected Vehicles Platoons. In Proceedings of the 29th Mediterranean Conference on Control and Automation (MED), Puglia, Italy, 22–25 June 2021; pp. 554–559.
48. Ioannou, P.; Xu, Z. Throttle and Brake Control Systems for Automatic Vehicle Following. *IVHS J.* **1994**, *1*, 345–377. [CrossRef]

49. Wen, S.; Guo, G. Sampled-Data Control for Connected Vehicles with Markovian Switching Topologies and Communication Delay. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 2930–2942. [CrossRef]
50. Baldi, S.; Frasca, P. Adaptive Synchronization of Unknown Heterogeneous Agents: An Adaptive Virtual Model Reference Approach. *J. Frankl. Inst.* **2019**, *356*, 935–955. [CrossRef]
51. Liangye, L.; Defeng, H.; Shiming, Y.; Shijian, D. Distributed Predictive Control for String Stability of Heterogeneous Vehicle Platoons with Multiple Communication Topologies. In Proceedings of the Chinese Control Conference (CCC), Guangzhou, China, 27–30 July 2019; pp. 2930–2935.
52. Xiao, S.; Ge, X.; Han, Q.-L.; Cao, Z.; Zhang, Y.; Wang, H. Resilient Distributed Event-Triggered Control of Vehicle Platooning under DoS Attacks. *IFAC-PapersOnLine* **2020**, *53*, 1807–1812. [CrossRef]
53. Shi, H.; Zhou, Y.; Wang, X.; Fu, S.; Gong, S.; Ran, B. A Deep Reinforcement Learning-Based Distributed Connected Automated Vehicle Control under Communication Failure. *Comput.-Aided Civ. Infrastruct. Eng.* **2022**, *37*, 2033–2051. [CrossRef]
54. Liu, B.; Ding, Z.; Lv, C. Platoon Control of Connected Autonomous Vehicles: A Distributed Reinforcement Learning Method by Consensus. In Proceedings of the 21st IFAC World Congress, Berlin, Germany, 11–17 July 2020; pp. 15241–15246.
55. Jun, M.; Zuo, L.; Yan, M. Q-Learning Based Model Unknown Vehicle Platoon Tracking Control with Switching Topologies. In Proceedings of the 42nd Chinese Control Conference (CCC), Tianjin, China, 24–26 July 2023; pp. 6446–6452.
56. Chow, Y.; Nachum, O.; Duenez-Guzman, E.; Ghavamzadeh, M. A Lyapunov-based Approach to Safe Reinforcement Learning. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32, pp. 8092–8101.
57. Berkenkamp, F.; Turchetta, M.; Schoellig, A.P.; Krause, A. Safe Model-based Reinforcement Learning with Stability Guarantees. In *Advances in Neural Information Processing Systems*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30, pp. 908–918.
58. Yang, J.; Chu, D.; Lu, L.; Meng, Z.; Deng, K. Model-data-driven Control for Human-leading Vehicle Platoon. *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.* **2024**. [CrossRef]
59. Lui, D.G.; Petrillo, A.; Santini, S. Leader Tracking Control for Heterogeneous Uncertain Nonlinear Multi-Agent Systems via a Distributed Robust Adaptive PID Strategy. *Nonlinear Dyn.* **2022**, *108*, 363–378. [CrossRef]
60. Arevalo-Castiblanco, M.F.; Tellez-Castro, D.; Sofrony, J.; Mojica-Nava, E. Adaptive Control for Unknown Heterogeneous Vehicles Synchronization with Unstructured Uncertainty. In Proceedings of the IEEE 4th Colombian Conference on Automatic Control (CCAC), Medellín, Colombia, 15–18 October 2019; pp. 1–6.
61. Jiang, W.; Chen, Y.; Charalambous, T. Consensus of General Linear Multi-Agent Systems with Heterogeneous Input and Communication Delays. *IEEE Control Syst. Lett.* **2021**, *5*, 851–856. [CrossRef]
62. Liu, J.; Dong, J. Fixed-Time Cooperative Fault-Tolerant Control of Heterogeneous Multi-Agent Systems via Learning-Based Uncertainty Estimation. *IEEE Internet Things J.* **2025**, *12*, 42803–42812. [CrossRef]
63. Xu, Y.; Guo, G. Event-Triggered Control of Connected Vehicles under Multiple Cyber Attacks. *Inf. Sci.* **2022**, *582*, 778–796. [CrossRef]
64. Zhang, H.; Liu, J.; Wang, Z.; Yan, H.; Zhang, C. Distributed Adaptive Event-Triggered Control and Stability Analysis for Vehicular Platoon. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 1627–1638. [CrossRef]
65. Vlachos, E.; Lalos, A.S. ADMM-Based Cooperative Control for Platooning of Connected and Autonomous Vehicles. In Proceedings of the IEEE International Conference on Communications (ICC), Seoul, Republic of Korea, 16–20 May 2022; pp. 4242–4247.
66. Resmi, R.; Mija, S.J.; Jacob, J. Discrete Laguerre-Based Model Predictive Control for Dynamic Consensus of a Vehicle Platoon with Time Delay. *Int. J. Syst. Sci.* **2022**, *53*, 2566–2583. [CrossRef]
67. Wakasa, T.; Nagatani, Y.; Sawada, K.; Shin, S. Switched Pinning Control for Vehicle Platoons via Mixed Logical Dynamical Modeling. In Proceedings of the IEEE/SICE International Symposium on System Integration (SII), Honolulu, HI, USA, 12–15 January 2020; pp. 449–454.
68. Liu, Y.; Li, Z.; Guo, G. Interaction Network-Based Resilient Consensus of Connected Vehicles against Cyber-Attacks. *IET Control Theory Appl.* **2023**, *17*, 744–756. [CrossRef]
69. Guo, H.; Liu, J.; Dai, Q.; Chen, H.; Wang, Y.; Zhao, W. A Distributed Adaptive Triple-Step Nonlinear Control for a Connected Automated Vehicle Platoon with Dynamic Uncertainty. *IEEE Internet Things J.* **2020**, *7*, 3861–3871. [CrossRef]
70. Qin, X.; Bian, Y.; Hu, Z.; Sun, N.; Hu, M. Distributed Vehicular Platoon Control Considering Communication Topology Disturbances. In Proceedings of the IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–6.
71. Shen, Z.; Liu, Y.; Li, Z.; Nabin, M.H. Cooperative Spacing Sampled Control of Vehicle Platoon Considering Undirected Topology and Analog Fading Networks. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 18478–18491. [CrossRef]
72. Linsenmayer, S.; Dimarogonas, D.V.; Allgöwer, F. Event-Based Vehicle Coordination Using Nonlinear Unidirectional Controllers. *IEEE Trans. Control Netw. Syst.* **2018**, *5*, 1575–1584. [CrossRef]

73. Li, Y.; He, C. Connected Autonomous Vehicle Platoon Control Considering Vehicle Dynamic Information. In Proceedings of the 37th Chinese Control Conference (CCC), Wuhan, China, 25–27 July 2018; pp. 7834–7839.
74. Zhu, Y.; Li, Y.; Yu, S. Nonlinear Longitudinal Control for Vehicle Platoon Considering the Effect of Electronic Throttle Opening Angle. In Proceedings of the 39th Chinese Control Conference (CCC), Shenyang, China, 27–29 July 2020; pp. 3196–3201.
75. Xie, S.; Hu, J.; Ding, Z.; Arvin, F. Distributed Cooperative Autonomous Driving of Intelligent Vehicles Based on Spring-Damper Energy System. In *Proceedings of the 2023 ICM—IEEE International Conference on Mechatronics*; IEEE: Loughborough, UK, 2023.
76. Mokari, H.; Firouzmand, E.; Sharifi, I.; Doustmohammadi, A. DoS Attack Detection and Resilient Control in Platoon of Smart Vehicles. In Proceedings of the 9th RSI International Conference on Robotics and Mechatronics (ICRoM), Tehran, Iran, 19 November 2021.
77. Gao, T.; Tian, Y.; Tao, H.; Ni, Y. Vehicle Platoon Control Based on Information Exchanges in Internet of Vehicles. In *Lecture Notes in Electrical Engineering*; Ren, Z., Ed.; Springer: Singapore, 2023; Volume 934, pp. 985–995.
78. Su, Y.; Yang, X.; Shi, P.; Wen, G.; Xu, Z. Consensus-Based Vehicle Platoon Control under Periodic Event-Triggered Strategy. *IEEE Trans. Syst. Man Cybern. Syst.* **2024**, *54*, 533–542. [CrossRef]
79. Paré, P.E.; Hashemi, E.; Stern, R.; Sandberg, H.; Johansson, K.H. Networked Model for Cooperative Adaptive Cruise Control. *IFAC-PapersOnLine* **2019**, *52*, 151–156. [CrossRef]
80. Zhang, P.; Tian, D.; Zhou, J.; Duan, X.; Sheng, Z.; Zhao, D.; Cao, D.; Bi, L. Efficient Robust Model Predictive Control for Behaviorally Stable Vehicle Platoons. *IEEE Trans. Intell. Transp. Syst.* **2024**, *26*, 1671–1688. [CrossRef]
81. Hanif, O.; Gruber, P.; Sorniotti, A.; Montanaro, U. Network-Based Fractional-Order Control Algorithms for Vehicle Platooning. *IEEE Access* **2025**, *13*, 139992–140019. [CrossRef]
82. Xiao, S.; Ge, X.; Han, Q.-L.; Zhang, Y. Secure and Collision-Free Multi-Platoon Control of Automated Vehicles under Data Falsification Attacks. *Automatica* **2022**, *145*, 110531. [CrossRef]
83. Chen, J.; Liang, H.; Li, J.; Lv, Z. Connected Automated Vehicle Platoon Control with Input Saturation and Variable Time Headway Strategy. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 4929–4940. [CrossRef]
84. Zhao, X.; Guo, S.; Mu, J.; He, D. Distributed Model Predictive Control for Vehicle Platooning with Intermittent Communication. In Proceedings of the 36th Chinese Control and Decision Conference (CCDC), Shenyang, China, 25–27 May 2024; pp. 869–874.
85. Al-Darabsah, I.; Hsueh, K.-F.; Khalil, A.; Al Janaideh, M.; Campbell, S.A.; Kundur, D. Validation of an Autonomous Vehicle Platoons Model with Time-Varying Communication Delays. *Chaos Solitons Fractals* **2024**, *184*, 114983. [CrossRef]
86. Han, J.; Zhang, J.; He, C.; Lv, C.; Li, C.; Hou, X.; Ji, Y. Adaptive Distributed Finite-Time Fault-Tolerant Controller for Cooperative Braking of the Vehicle Platoon. *IET Intell. Transp. Syst.* **2021**, *15*, 1562–1581. [CrossRef]
87. 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*, 15050–15064. [CrossRef]
88. Manfredi, S.; Petrillo, A.; Santini, S. Distributed PI Control for Heterogeneous Nonlinear Platoon of Autonomous Connected Vehicles. *IFAC-PapersOnLine* **2020**, *53*, 15229–15234. [CrossRef]
89. Bifulco, G.N.; Coppola, A.; Mungiello, A.; Petrillo, A.; Santini, S. CAVs Platoons under Nonlinear Spacing Policy and Heterogeneous Communication Delays as a Formation Control Problem. In Proceedings of the European Control Conference (ECC), Stockholm, Sweden, 25–28 June 2024; pp. 1393–1398.
90. Hao, Q.; Dong, J.; Zhao, J. Adaptive Cruise Control Algorithm for Persistence and Security of Distributed Heterogeneous Vehicle Platoon Based on Discrete-Time Control Barrier Function. *IEEE Trans. Veh. Technol.* **2025**, *74*, 15320–15330. [CrossRef]
91. He, D.; Yang, Z.; Mu, J.; Li, Y. Distributed Model Predictive Platooning Control of Heterogenous CAVs under Markovian Switching Communication Topologies. *IEEE Trans. Veh. Technol.* **2026**, *75*, 145–155. [CrossRef]
92. Coppola, A.; Lui, D.G.; Petrillo, A.; Santini, S. Cooperative Driving of Heterogeneous Uncertain Nonlinear Connected and Autonomous Vehicles via Distributed Switching Robust PID-Like Control. *Inf. Sci.* **2023**, *625*, 277–298. [CrossRef]
93. Wang, W.; Wang, C.; Guo, Y.; Luo, X.; Gao, Y. Self-Triggered Consensus of Vehicle Platoon System with Time-Varying Topology. *Front. Neurorobot.* **2020**, *14*, 53. [CrossRef]
94. Li, X.; Pham, D.T.; Zhang, X.; Zhao, Q. Impulsive Control of Second-Order Intelligent Vehicle Longitudinal Platooning under the Event-Triggered Mechanism. *Int. J. Robust Nonlinear Control* **2025**, *35*, 6475–6484. [CrossRef]
95. Wu, Y.; Li, S.E.; Cortés, J.; Poolla, K. Distributed Sliding Mode Control for Nonlinear Heterogeneous Platoon Systems with Positive Definite Topologies. *IEEE Trans. Control Syst. Technol.* **2020**, *28*, 1272–1283. [CrossRef]
96. Caiazzo, B.; Lui, D.G.; Petrillo, A.; Santini, S. Resilient Adaptive Finite-Time Fault-Tolerant Control for Heterogeneous Uncertain and Nonlinear Autonomous Connected Vehicles Platoons. *IEEE Open J. Intell. Transp. Syst.* **2023**, *4*, 481–492. [CrossRef]
97. Huang, J.; Chen, J.; Yang, H.; Li, D. Vehicle Platoon Tracking Control Based on Adaptive Neural Network Algorithm. *Int. J. Control Autom. Syst.* **2023**, *21*, 3405–3418. [CrossRef]
98. Zhu, Y.; Li, Y.; Zhu, H.; Yu, S.; Yuan, J.; Wen, G. Joint Sliding-Mode Controller and Observer for Vehicle Platoon Subject to Disturbance and Acceleration Failure of Neighboring Vehicles. *IEEE Trans. Intell. Veh.* **2023**, *8*, 2345–2357. [CrossRef]

99. Wang, J.; Luo, X.; Zhang, Y.; Guan, X. Distributed Integrated Sliding Mode Control via Neural Network and Disturbance Observer for Heterogeneous Vehicle Systems with Uncertainties. *Trans. Inst. Meas. Control* **2023**, *45*, 2204–2215. [CrossRef]
100. Wang, Y.; Dong, H.; Li, X. Distributed Cooperative Neural Control for Nonlinear Heterogeneous Platoon Systems with Unknown Uncertainties. *Arab. J. Sci. Eng.* **2024**, *49*, 15957–15970. [CrossRef]
101. Zhang, L.; Jin, S.; Bu, X.; Yin, C. Disturbance Observer Based Model-Free Adaptive Longitudinal Control of Vehicle Platoon. In Proceedings of the IEEE 14th Data Driven Control and Learning Systems Conference (DDCLS), Wuxi, China, 9–11 May 2025; pp. 1–6.
102. Zuo, L.; Peng, X.; Cheng, M.; Zhang, Z. Optimal Integral Sliding-Mode Based Nonlinear Vehicle Platoon Tracking Control with Directed Communication Topology and External Disturbance. *Nonlinear Dyn.* **2025**, *113*, 18107–18119. [CrossRef]
103. Yu, Y.; Song, Z.; Meng, H. Practical Prescribed-Time Sliding-Mode Output Feedback Tracking Control for Heterogeneous Vehicle Platoon. *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.* **2025**, *239*, 1066–1083. [CrossRef]
104. Jiao, A.; Li, Y.; Yu, S.; Cui, Z. Nonlinear Longitudinal Control for Connected Vehicle Platoon Considering the Acceleration Difference. In Proceedings of the 39th Chinese Control Conference (CCC), Shenyang, China, 27–29 July 2020; pp. 5467–5472.
105. Chen, J.; Li, J.; Xu, Z.; Wu, X. Cooperative Optimal Control for Connected and Automated Vehicles Platoon. *Phys. A Stat. Mech. Its Appl.* **2022**, *607*, 128197. [CrossRef]
106. Yan, M.; Ma, W.; Zuo, L.; Yang, P. Distributed Model Predictive Control for Platooning of Heterogeneous Vehicles with Multiple Constraints and Communication Delays. *J. Adv. Transp.* **2020**, *2020*, 4657584. [CrossRef]
107. Prayitno, A.; Nilkhamhang, I. V2V Network Topologies for Vehicle Platoons with Cooperative State Variable Feedback Control. In Proceedings of the Second International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP), Bangkok, Thailand, 20–22 January 2021; pp. 1–4.
108. Caiazzo, B.; Di Rosa, F.; Lui, D.G.; Petrillo, A.; Santini, S. Distributed Resilient Control for Autonomous Connected Vehicles Platoon over Uncertain V2X Communication Links. In Proceedings of the IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), Bilbao, Spain, 24–28 September 2023; pp. 5241–5246.
109. Zhao, C.; Duan, X.; Cai, L.; Cheng, P. Vehicle Platooning with Non-Ideal Communication Networks. *IEEE Trans. Veh. Technol.* **2021**, *70*, 18–32. [CrossRef]
110. Oliveira, R.; Montez, C.; Boukerche, A.; Wangham, M.S. Co-Design of Consensus-Based Approach and Reliable Communication Protocol for Vehicular Platoon Control. *IEEE Trans. Veh. Technol.* **2021**, *70*, 9510–9524. [CrossRef]
111. Gao, Y.; Wen, S.; Zhao, Y. Control of Connected Vehicles with Multi-Type Communication Topology and Delay. In Proceedings of the 35th Chinese Control and Decision Conference (CCDC), Yichang, China, 20–22 May 2023; pp. 798–803.
112. Zhao, C.; Cai, L.; Cheng, P. Stability Analysis of Vehicle Platooning with Limited Communication Range and Random Packet Losses. *IEEE Internet Things J.* **2021**, *8*, 262–277. [CrossRef]
113. Qin, X.; Bian, Y.; Yang, Z.; Hu, M. Distributed Vehicular Platoon Control with Heterogeneous Communication Delays. In Proceedings of the 23rd IEEE International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 3219–3224.
114. Zegers, J.C.; Semsar-Kazerooni, E.; Ploeg, J.; van de Wouw, N.; Nijmeijer, H. Consensus Control for Vehicular Platooning with Velocity Constraints. *IEEE Trans. Control Syst. Technol.* **2018**, *26*, 1592–1605. [CrossRef]
115. Wei, J.; Zhu, X.; Shen, Y. Assignment of Rightmost Pole in Vehicle Platoon Based on Proportional-Retarded Controller. In Proceedings of the 43rd Chinese Control Conference, Kunming, China, 28–31 July 2024; pp. 6445–6450.
116. Prayitno, A.; Indrawati, V.; Nilkhamhang, I. A Comparative Study of Vehicle Platoon with Limited Output Information in Directed Topologies. In Proceedings of the Third International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP), Bangkok, Thailand, 18–20 January 2023; pp. 110–113.
117. Shen, Z.; Guo, H.; Liu, Y.; Li, Z.; Wu, Y. Cooperative Spacing Control of Vehicle Platoon via Relative Output Feedback Considering Analog Fading Networks. *Int. J. Robust Nonlinear Control* **2025**, *35*, 909–923. [CrossRef]
118. Shen, Z.; Liu, Y.; Li, Z.; Wu, Y. Distributed Vehicular Platoon Control Considering Communication Delays and Packet Dropouts. *J. Frankl. Inst.* **2024**, *361*, 106703. [CrossRef]
119. Jun, M.; Zuo, L.; Yan, M.; Zhang, J. Distributed Vehicle Platoon Tracking Control for Optimal Energy Consumption with Switching Topologies. In Proceedings of the 35th Chinese Control and Decision Conference (CCDC), Yichang, China, 22 May 2023; pp. 161–167.
120. Zhao, Y.; Ni, J.; Hu, J. Dynamics Control of Topology Switching for Multi-Functional Unmanned Ground Vehicle Platoon with VCBDL Topology. *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.* **2023**, *237*, 2708–2720. [CrossRef]
121. Caiazzo, B.; Lui, D.G.; Mungiello, A.; Petrillo, A.; Santini, S. On the Resilience of Autonomous Connected Vehicles Platoon under DoS Attacks: A Predictor-Based Sampled Data Control. In Proceedings of the IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), 24–28 September 2023; pp. 4907–4912.
122. Khodadadi, S.; Tasooji, T.K.; Marquez, H.J. Observer-Based Secure Control for Vehicular Platooning under DoS Attacks. *IEEE Access* **2023**, *11*, 20542–20552. [CrossRef]

123. Wei, H.; Zhang, H.; Al-Haddad, K.; Shi, Y. Ensuring Secure Platooning of Constrained Intelligent and Connected Vehicles against Byzantine Attacks: A Distributed MPC Framework. *Engineering* **2024**, *33*, 35–46. [CrossRef]
124. Petrillo, A.; Pescapé, A.; Santini, S. A Secure Adaptive Control for Cooperative Driving of Autonomous Connected Vehicles in the Presence of Heterogeneous Communication Delays and Cyberattacks. *IEEE Trans. Cybern.* **2021**, *51*, 1134–1149. [CrossRef]
125. Prayitno, A.; Indrawati, V.; Nilkhamhang, I. Distributed Model Reference Control for Synchronization of a Vehicle Platoon with Limited Output Information and Subject to Periodical Intermittent Information. *Int. J. Appl. Math. Comput. Sci.* **2023**, *33*, 537–551. [CrossRef]
126. Shen, Y.; Zhu, X.; Yan, M.; Zuo, L. Most Exigent Eigenvalue Problem for the Stability Analysis of the Vehicle Platoon with Delay. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 12267–12286. [CrossRef]
127. Zeng, H.; Ye, Z.; Zhang, D.; Lu, Q. Robust Distributed Model Predictive Control of Connected Vehicle Platoon against DoS Attacks. *Intell. Robot.* **2023**, *3*, 288–305. [CrossRef]
128. Basiri, M.H.; Pirani, M.; Azad, N.L.; Fischmeister, S. Security-Aware Optimal Actuator Placement in Vehicle Platooning. *Asian J. Control* **2022**, *24*, 1557–1574. [CrossRef]
129. Liu, F.; Long, Y.; Zhou, H. Analysis of Autonomous Vehicle Platoon with Disturbance Based on PLEXE. In Proceedings of the Chinese Control and Decision Conference (CCDC), Nanchang, China, 3–5 June 2019; pp. 134–139.
130. Wu, Z.; Sun, J.; Xu, R. Consensus-Based Connected Vehicles Platoon Control via Impulsive Control Method. *Phys. A Stat. Mech. Its Appl.* **2021**, *580*, 126190. [CrossRef]
131. Wang, W.; Wang, C.; Wang, Z.; Han, B.; He, C.; Cheng, J.; Luo, X.; Yuan, M.; Kurths, J. Nonlinear Consensus-Based Autonomous Vehicle Platoon Control under Event-Triggered Strategy in the Presence of Time Delays. *Appl. Math. Comput.* **2021**, *404*, 126246. [CrossRef]
132. Zhang, L.; Sun, J.; Yang, Q. Distributed Model-Based Event-Triggered Leader–Follower Consensus Control for Linear Continuous-Time Multiagent Systems. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 6457–6465. [CrossRef]
133. Lei, L.; Liu, T.; Zheng, K.; Hanzo, L. Deep Reinforcement Learning Aided Platoon Control Relying on V2X Information. *IEEE Trans. Veh. Technol.* **2022**, *71*, 5811–5826. [CrossRef]
134. Li, K.; Bian, Y.; Li, S.E.; Xu, B.; Wang, J. Distributed Model Predictive Control of Multi-Vehicle Systems with Switching Communication Topologies. *Transp. Res. Part C Emerg. Technol.* **2020**, *118*, 102717. [CrossRef]
135. Hu, M.; Bu, L.; Bian, Y.; Qin, H.; Sun, N.; Cao, D.P.; Zhong, Z. Hierarchical Cooperative Control of Connected Vehicles: From Heterogeneous Parameters to Heterogeneous Structures. *IEEE/CAA J. Autom. Sin.* **2022**, *9*, 1590–1602. [CrossRef]
136. Yan, M.; Tang, Y.; Yang, P. Consensus Based Control Algorithm for Nonlinear Vehicle Platoons in the Presence of Time Delays and Packet Losses. In Proceedings of the IEEE Conference on Decision and Control (CDC), Miami Beach, FL, USA, 17–19 December 2018; pp. 3339–3344.
137. Zhang, L.; Sun, J.; Cai, T. Distributed Topology Switching Strategy Designing for Heterogeneous Vehicle Platoons. In Proceedings of the IEEE International Conference on Control and Automation (ICCA), Anchorage, AK, USA, 12–15 June 2018; pp. 734–738.
138. Yang, L.; Liu, Y.; Sun, Z.; Yan, Q.; Qin, Z.; Wang, X. Stability Analysis of Connected Automated Vehicle Platoon Controller under Unreliable Communication Conditions. In Proceedings of the IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), Bilbao, Spain, 24–28 September 2023; pp. 3647–3652.
139. Zheng, Y.; Bian, Y.; Li, S.E.; Li, S. Cooperative Control of Heterogeneous Connected Vehicles with Directed Acyclic Interactions. *IEEE Intell. Transp. Syst. Mag.* **2019**, *11*, 108–124. [CrossRef]
140. Zhan, J.; Zhang, L.; Chen, Y. Asynchronous Platoon Control for Connected Vehicles with Intermittent and Delayed Information Transmission. *IEEE Trans. Autom. Control* **2023**, *68*, 6875–6882. [CrossRef]
141. Montanaro, U.; Fallah, S.; Dianati, M.; Oxtoby, D.; Mizutani, T.; Mouzakitis, A. On a Fully Self-Organizing Vehicle Platooning Supported by Cloud Computing. In Proceedings of the Fifth International Conference on Internet of Things: Systems, Management and Security (IoTSMS), Valencia, Spain,15–18 October 2018; pp. 295–302.
142. Chen, Y.; Yan, B. Vehicle Platoon Control Based on Third-Order Heterogeneous Model and Predictive Spacing Strategy. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 4458–4469. [CrossRef]
143. Hu, J.; Bhowmick, P.; Arvin, F.; Lanzon, A.; Lennox, B. Cooperative Control of Heterogeneous Connected Vehicle Platoons: An Adaptive Leader-Following Approach. *IEEE Robot. Autom. Lett.* **2020**, *5*, 977–984. [CrossRef]
144. Yu, G.; Wong, P.K.; Huang, W.; Zhao, J.; Wang, X.-B.; Yang, Z.-X. Distributed Adaptive Consensus Protocol for Connected Vehicle Platoon with Heterogeneous Time-Varying Delays and Switching Topologies. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 17620–17631. [CrossRef]
145. He, C.; Li, Y.; Xu, J.; Hao, W. Third-Order Model Based Nonlinear Longitudinal Control for Heterogeneous Connected Vehicle Platoon. In Proceedings of the IEEE International Conference on Real-time Computing and Robotics (RCAR), Irkutsk, Russia, 4–9 August 2019; pp. 106–111.
146. Guo, G.; Yang, D.; Zhang, R. Distributed Trajectory Optimization and Platooning of Vehicles to Guarantee Smooth Traffic Flow. *IEEE Trans. Intell. Veh.* **2023**, *8*, 684–694. [CrossRef]

147. Li, Y.; Zhang, S.; Huang, L.; Huang, G. Nonlinear Longitudinal Cooperative Control of Heterogeneous Connected Vehicle Platoon Considering Car-Following Interactions and Communication Delay. *Transp. B Transp. Dyn.* **2023**, *11*, 1687–1706. [CrossRef]
148. Razzaghpour, M.; Valiente, R.; Zaman, M.; Fallah, Y.P. Predictive Model-Based and Control-Aware Communication Strategies for Cooperative Adaptive Cruise Control. *IEEE Open J. Intell. Transp. Syst.* **2023**, *4*, 232–243. [CrossRef]
149. Zhang, S.; Li, Y.; Huang, L.; Huang, X. Longitudinal Control of Heterogeneous Connected Vehicle Platoon Considering the Motion Coupling Interactions with Switching Topology. In Proceedings of the International Conference on Frontiers of Artificial Intelligence and Machine Learning (FAIML), Beijing, China, 14–16 April 2023; pp. 210–213.
150. Chehardoli, H.; Shafie, E. Adaptive Resilient Observer-Based Control of Leader-Following Smart Vehicle Platoons Subject to Compound Cyber Attacks. *Int. J. Dyn. Control* **2025**, *13*, 32. [CrossRef]
151. Li, H.; Wen, S.; Guo, G.; Liang, H.; He, J. An Improved Sliding Mode Control (SMC) Approach for Enhancement of Communication Delay in Vehicle Platoon System. *IET Intell. Transp. Syst.* **2022**, *16*, 958–970. [CrossRef]
152. Chen, G.; Zhao, W.; Xia, J.; Wang, Z.; Park, J.H. Differentially Private Aperiodic Sampled-Data Consensus for Intelligent Interconnected Heterogeneous Vehicular Platoons. *IEEE Syst. J.* **2025**, *19*, 612–623. [CrossRef]
153. Liang, X.; Gao, W.; Hu, C.; Chai, T. Cooperative Adaptive Cruise Control of Connected and Autonomous Vehicles via Hybrid Iteration. *IEEE Trans. Veh. Technol.* **2025**, 1–12, *Early Access*.
154. Prayitno, A.; Indrawati, V.; Yusuf, Y.G. Neighborhood Controller-Local Observer in Adaptive Control Approach for Heterogeneous Vehicle Platoon Formation. In Proceedings of the International Seminar on Intelligent Technology and Its Applications (ISITIA), Surabaya, Indonesia, 26–27 July 2023; pp. 484–489.
155. Guo, Z.; Fan, S.; Wang, X.; Wang, B.; Yan, J.-J.; Deng, C. Platoon Control for Cyber-Physical Vehicle Systems with Intermittent Communication. *IEEE Internet Things J.* **2025**, *12*, 35773–35783. [CrossRef]
156. Elahi, A.; Alfi, A.; Modares, H. H∞ Consensus of Homogeneous Vehicular Platooning Systems with Packet Dropout and Communication Delay. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 3680–3691. [CrossRef]
157. Halder, K.; Montanaro, U.; Dixit, S.; Dianati, M.; Mouzakitis, A.; Fallah, S. Distributed Controller Design for Vehicle Platooning under Packet Drop Scenario. In Proceedings of the IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–6.
158. Ouyang, Z.; Li, Z.; Chen, S.; Yang, Z. Distributed Vehicle Platoon Control with Leader Selection Strategy. In Proceedings of the 40th Chinese Control Conference (CCC), Shanghai, China, 26–28 July 2021; pp. 7536–7541.
159. Zhao, Y.; Ni, J.; Hu, J. A Novel Platoon Topology and Performance Analysis with Cloud Brain Control Center for Multi-Functional Unmanned Ground Vehicle. *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.* **2021**, *237*, 2458–2473. [CrossRef]
160. Zhao, R.; Liu, G.; Wang, G.; Zhao, J. Distributed Event-Triggered Adaptive Robust Platoon Control of Connected Vehicles with Uncertainties Subject to Invalid Communication. *Nonlinear Dyn.* **2023**, *111*, 19187–19207. [CrossRef]
161. Wang, L.; Hu, M.; Bian, Y.; Guo, G.; Li, S.E.; Chen, B.; Zhong, Z. Periodic Event-Triggered Fault Detection for Safe Platooning Control of Intelligent and Connected Vehicles. *IEEE Trans. Veh. Technol.* **2024**, *73*, 5064–5077. [CrossRef]
162. Zhao, Y.; Liu, Z.; Wong, W.S. Resilient Platoon Control of Vehicular Cyber Physical Systems under DoS Attacks and Multiple Disturbances. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 10945–10956. [CrossRef]
163. di Vaio, M.; Petrillo, A.; Santini, S. On the Robustness of a Distributed Adaptive Synchronization Protocol for Connected Autonomous Vehicles with Multiple Disturbances and Communication Delays. In Proceedings of the IEEE Conference on Decision and Control (CDC), Miami Beach, FL, USA, 17–19 December 2018; pp. 3622–3627.
164. Li, S.E.; Gao, F.; Li, K.; Wang, L.-Y.; You, K.; Cao, D. Robust Longitudinal Control of Multi-Vehicle Systems—A Distributed H-Infinity Method. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 2779–2790. [CrossRef]
165. Godinho, D.A.; Neto, A.A.; Souza, F.O. Distributed Control and Reorganization of Heterogeneous Vehicle Platoons Subject to Time Delay and Limited Communication Range. *arXiv* 2023, arXiv:2307.03714.
166. Li, Y.; Chen, Y.; Li, S. Proportional-Integral Observer-Based Consensus Control for Heterogeneous Vehicular Platoon Systems with Mixed Disturbances and Unavailable Acceleration Information. *Trans. Inst. Meas. Control* **2025**, *47*, 1–14. [CrossRef]
167. Li, H.; Chen, Z.; Fu, B.; Sun, M. Nonlinear Control of Heterogeneous Vehicle Platoon with Time-Varying Delays and Limited Communication Range. *Int. J. Control Autom. Syst.* **2023**, *21*, 1727–1738. [CrossRef]
168. Zhu, Y.; Li, Y.; Hu, S.; Yu, S. Optimal Control for Vehicle Platoon Considering External Disturbances. In Proceedings of the IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–10 October 2022; pp. 453–458.
169. Ge, X.; Xiao, S.; Han, Q.-L.; Zhang, X.-M.; Ding, D. Dynamic Event-Triggered Scheduling and Platooning Control Co-Design for Automated Vehicles over Vehicular Ad-Hoc Networks. *IEEE/CAA J. Autom. Sin.* **2022**, *9*, 31–46. [CrossRef]
170. Zhang, J.; Peng, C.; Xie, X. Platooning Control of Vehicular Systems by Using Sampled Positions. *IEEE Trans. Circuits Syst. II Express Briefs* **2023**, *70*, 2435–2439. [CrossRef]
171. Wu, G.; Chen, G.; Zhang, H.; Huang, C. Fully Distributed Event-Triggered Vehicular Platooning with Actuator Uncertainties. *IEEE Trans. Veh. Technol.* **2021**, *70*, 6601–6612. [CrossRef]

172. Wang, J.; Guo, J.; Li, K.; Zheng, H. Distributed Adaptive Event-Triggered Control of Connected Automated Vehicle Platoon Systems with Spoofing Cyber Attacks. *IEEE Trans. Veh. Technol.* **2024**, *73*, 18186–18198. [CrossRef]
173. Prayitno, A.; Nilkhamhang, I. Distributed Model Reference Control for Cooperative Tracking of Vehicle Platoons Subjected to External Disturbances and Bounded Leader Input. *Int. J. Control Autom. Syst.* **2022**, *20*, 2067–2080. [CrossRef]
174. Prayitno, A.; Nilkhamhang, I. Distributed Model Reference Adaptive Control for Vehicle Platoons with Uncertain Dynamics. *Eng. J.* **2021**, *25*, 173–185. [CrossRef]
175. Liu, C.; Zhao, J.; Patton, R.J. Distributed Antiattack Fault-Tolerant Tracking Control for Vehicle Platoon Systems under Cyber-Physical Threats. *IEEE Trans. Ind. Inform.* **2023**, *19*, 7825–7834. [CrossRef]
176. Zhou, A.; Wang, J.; Peeta, S. Robust Control Strategy for Platoon of Connected and Autonomous Vehicles Considering Falsified Information Injected through Communication Links. *J. Intell. Transp. Syst.* **2023**, *27*, 735–751. [CrossRef]
177. Zeng, H.; Ye, Z.; Zhang, D. Robust Model Predictive Control Based Cooperative Control of Uncertain Connected Vehicle Platoon Systems. In Proceedings of the International Conference on Control, Automation and Robotics (ICCAR), Beijing, China, 21–23 April 2023; pp. 256–262.
178. Wu, Q.; Ge, X.; Han, Q.-L.; Cole, C.; Spiryagin, M. A Scalable Parallel Computing Method for Autonomous Platoons. *Veh. Syst. Dyn.* **2024**, *62*, 2283–2303. [CrossRef]
179. Liu, C.; Xia, Z.; Patton, R.J. Distributed Fault-Tolerant Consensus Control of Vehicle Platoon Systems with DoS Attacks. *IEEE Trans. Veh. Technol.* **2024**, *73*, 14438–14449. [CrossRef]
180. Ding, S.; Ai, H.; Xie, X.; Jing, Y. Distributed Adaptive Platooning Control of Connected Vehicles with Markov Switching Topologies. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 18421–18432. [CrossRef]
181. Hendrix, C.; Bloom, G. Platoon Vulnerability Due to Network Topology and Targeted Vehicle. In Proceedings of the IEEE Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 6–9 January 2024; pp. 771–776.
182. Han, Q.; Ma, J.; Zuo, Z.; Wang, X.; Yang, B.; Guan, X. Resilient Event-Triggered Control of Vehicle Platoon under DoS Attacks and Parameter Uncertainty. *IEEE Trans. Intell. Veh.* **2025**, *10*, 169–178. [CrossRef]
183. Ning, B.; Han, Q.-L.; Ge, X.; Sanjayan, J. A Zeroing Neural Network-Based Approach to Parameter-Varying Platooning Control of Connected Automated Vehicles. *IEEE Trans. Intell. Veh.* **2024**, *9*, 1977–1988. [CrossRef]
184. Wang, J.; Zheng, H.; Guo, J.; Fan, J.; Li, K. Distributed Adaptive Robust H∞ Control of Intelligent-Connected Electric Vehicles Platooning Subject to Communication Delay. *IET Intell. Transp. Syst.* **2021**, *15*, 699–711. [CrossRef]
185. Han, Q.; Xu, C.; Wang, X.; Yang, H.; Zuo, Z.; Yang, B.; Guan, X. Dynamic Event-Triggered Model Predictive Control for Multiple Platoons under Nonideal Communications. *IEEE Internet Things J.* **2025**, *12*, 29411–29426. [CrossRef]
186. Li, Z.; Li, Z.; Liu, Y. Resilient Control Design of the Third-Order Discrete-Time Connected Vehicle Systems against Cyber-Attacks. *IEEE Access* **2020**, *8*, 157470–157482. [CrossRef]
187. Wang, J.; Chen, X.; Guo, J.; Li, K.; Deng, X. Observer-Based Event-Triggered Platoon Control for Connected Automated Vehicles under FDI Attacks. *IEEE Internet Things J.* **2025**, *12*, 38783–38794. [CrossRef]
188. Liu, P.; Ju, Y.; Ding, D.; Sun, Y.; Zhang, M. Sliding Mode Platooning Control of Automated Vehicles under Round-Robin Protocols. *Int. J. Robust Nonlinear Control* **2025**, *35*, 7736–7747. [CrossRef]
189. Lv, Q.; Li, Y.; Yu, S.; Li, Y. Platoon Control for Connected Trucks Considering External Disturbances. In Proceedings of the CAA International Conference on Vehicular Control and Intelligence (CVCI), Tianjin, China, 29–31 October 2021; pp. 1–6.
190. Li, Y.; Qin, Z.; Zhu, Y.; Yu, S.; Gao, X. Variable Time Headway Policy Based Platoon Control for Heterogeneous Connected Vehicles with External Disturbances. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 21190–21201. [CrossRef]
191. Zeng, H.; Ye, Z.; Zhang, D. Dynamic Event-Triggering-Based Distributed Model Predictive Control of Heterogeneous Connected Vehicle Platoon under DoS Attacks. *ISA Trans.* **2024**, *153*, 1–12. [CrossRef]
192. Zhu, Y.; Li, Y.; Zhu, H.; Yu, S.; Yuan, J.; Wen, G. Hierarchical Control of Connected Vehicle Platoon by Simultaneously Considering the Vehicle Kinematics and Dynamics. *IEEE Trans. Intell. Veh.* **2024**, *9*, 1333–1344. [CrossRef]
193. Neto, A.A.; Mozelli, L.A. Robust Longitudinal Control for Vehicular Platoons Using Deep Reinforcement Learning. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 14401–14411. [CrossRef]
194. Alsinai, A.; Piqueira, J.R.C.; Hassan, W.U.; Niazi, A.U.K.; Safder, F. Dynamic Nonlinear Control Strategies for Resilient Heterogeneous Vehicle Platooning and Handling Byzantine Attacks in Communication Networks. *Heliyon* **2025**, *11*, e41574. [CrossRef]
195. Liu, J.; Dong, J. Hierarchical Prescribed-Time Platoon Control for Heterogeneous Vehicles with Actuator Faults and System Uncertainties. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 6965–6976. [CrossRef]
196. Li, Y.; Qin, Z.; Zhu, H.; Peeta, S.; Gao, X. Platoon Control of Connected Vehicles with Heterogeneous Model Structures Considering External Disturbances. *Green Energy Intell. Transp.* **2022**, *1*, 100038. [CrossRef]
197. Wang, B.; Su, R. A Distributed Platoon Control Framework for Connected Automated Vehicles in an Urban Traffic Network. *IEEE Trans. Control Netw. Syst.* **2022**, *9*, 1717–1730. [CrossRef]

198. Tapli, T.; Akar, M. Cooperative Adaptive Cruise Control Algorithms for Vehicular Platoons Based on Distributed Model Predictive Control. In Proceedings of the Mediterranean Conference on Control and Automation (MED), Saint-Raphael, France, 15–18 September 2020; pp. 305–310.
199. Gao, F.; Hu, X.; Li, S.E.; Li, K.; Sun, Q. Distributed Adaptive Sliding Mode Control of Vehicular Platoon with Uncertain Interaction Topology. *IEEE Trans. Ind. Electron.* **2018**, *65*, 6352–6361. [CrossRef]
200. Sun, C.; Zhang, Y.; Xiahedan, H. Energy-Aware Distributed Sliding Mode Control for Heterogeneous Vehicle Platoons. In Proceedings of the International Conference on Robotics, Intelligent Control and Artificial Intelligence (RICAI), Nanjing, China, 6–8 December 2024; pp. 328–334.
201. Bejarbaneh, E.Y.; Du, H. Cooperative Control of Vehicle Platoons with Unknown Dynamics Using Neural Network-Enhanced Reinforcement Learning. In Proceedings of the IEEE International Conference on Mechatronics (ICM), Wollongong, NSW, Australia, 28 February–2 March 2025; pp. 1–6.
202. Guo, S.; Yang, J.; Zhang, K.; Zhang, S.; Chen, D.; Tang, C. Consensus Control and Sensor Fault Diagnosis of Heterogeneous Intelligent Connected Vehicles with Time-Varying Parameters and Random Noises under Zero-Trust Architecture. *IEEE Trans. Veh. Technol.* **2025**, *74*, 13713–13727. [CrossRef]
203. Xie, M.; Ding, D.; Shen, B.; Song, Y. Learning-Based Platooning Control of Automated Vehicles under Constrained Bit Rate. *IEEE Trans. Intell. Veh.* **2024**, *9*, 1104–1117. [CrossRef]
204. Wen, S.; Guo, G.; Zhao, Y. Practical Prescribed Performance Tracking Control and Optimization for Nonlinear Vehicular Platoon. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 9625–9639. [CrossRef]
205. Wu, Z.; Sun, J.; Hong, S. RBFNN-Based Adaptive Event-Triggered Control for Heterogeneous Vehicle Platoon Consensus. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 18761–18772. [CrossRef]
206. Liu, Y.; Yao, D.; Li, H.; Lu, R. Distributed Cooperative Compound Tracking Control for a Platoon of Vehicles with Adaptive NN. *IEEE Trans. Cybern.* **2022**, *52*, 7039–7049. [CrossRef]
207. Gao, F.; Liu, B.; Qi, J.; Wang, C. Distributed Sliding Mode Control for Formation of Multiple Nonlinear AVs Coupled by Uncertain Topology. *SN Appl. Sci.* **2019**, *1*, 374. [CrossRef]
208. Wei, J.; Liu, Y.-J.; Chen, H.; Liu, L. Fuzzy Adaptive Control for Vehicular Platoons with Constraints and Unknown Dead-Zone Input. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 4403–4412. [CrossRef]
209. Wen, S.; Guo, G.; Zhao, Y.; Gao, Z. Adaptive Optimal Control for Vehicle Platoons with Uncertain Communication Topology. *IEEE Trans. Veh. Technol.* **2024**, *73*, 12500–12514. [CrossRef]
210. Alsinai, A.; Piqueira, J.R.C.; Niazi, A.U.K.; Hassan, W.U.; Mustafa, M.W.; Zaidi, S.B.R. Optimising Consensus Efficiency in Autonomous Vehicle Platoons through Nonlinear Distributed Control. *J. Control Decis.* **2025**, 1–12, *Early Access*. [CrossRef]
211. Wang, B.; Shen, C.; Lin, C.; Deng, C.; Shi, Y. Distributed Cooperative Control and Robust Optimization for Nonlinear Connected Automated Vehicles with Unknown Reaction Time Delays and Jerk Dynamics. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 6715–6733. [CrossRef]
212. Li, Z.; An, T.; Dong, B.; Yuan, X. Event-Triggered V2V Communication-Based Cooperative Adaptive Tracking Control for Nonlinear Vehicle Platoon Systems with Unknown Lag Time. *Nonlinear Dyn.* **2025**, *113*, 519–532. [CrossRef]
213. Gao, J.; Peng, Z.; Jang, S.; Piao, C. Platoon Control Leveraging Network Performance and State Estimation under Dynamic V2V Network. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 12093–12105. [CrossRef]
214. Zhu, P.; Jin, S.; Bu, X.; Hou, Z. Distributed Data-Driven Control for a Connected Autonomous Vehicle Platoon Subjected to False Data Injection Attacks. *IEEE Trans. Autom. Sci. Eng.* **2024**, *21*, 7527–7538. [CrossRef]
215. Zhu, P.; Jin, S.; Yin, C. RBFNN-Based Event-Triggered Data-Driven Formation Control for a Connected Heterogeneous Vehicle Platoon. In Proceedings of the IEEE Data Driven Control and Learning Systems Conference (DDCLS), Kaifeng, China, 17–19 May 2024; pp. 122–127.
216. Zhang, Y.; Hui, Y.; Chi, R. Adjacent Dynamic Linearization Based Data-Driven Iterative Learning Control for Vehicle Formation. In Proceedings of the IEEE Data Driven Control and Learning Systems Conference (DDCLS), Wuxi, China, 9–11 May 2025.
217. Zhu, P.; Jin, S.; Bu, X.; Hou, Z. Distributed Data-Driven Control for a Connected Heterogeneous Vehicle Platoon under Quantized and Switching Topologies Communication. *IEEE Trans. Veh. Technol.* **2023**, *72*, 9796–9807. [CrossRef]
218. Zhu, P.; Jin, S.; Bu, X.; Hou, Z. Distributed Data-Driven Event-Triggered Fault-Tolerant Control for a Connected Heterogeneous Vehicle Platoon with Sensor Faults. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 5498–5509. [CrossRef]
219. Ma, Y.-S.; Che, W.-W.; Deng, C.; Wu, Z.-G. Data-Driven Distributed Vehicle Platoon Control for Heterogeneous Nonlinear Vehicle Systems. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 2373–2382. [CrossRef]
220. Zhang, L.; Sun, J.; Lai, G. Event-Based Leader-Follower Consensus for Linear Multi-Agent Systems with Adaptive Weighting under Directed Communication Topologies. In Proceedings of the Asian Control Conference (ASCC), Kitakyushu, Japan, 9–12 June 2019; pp. 925–930.
221. Li, J.; Wang, L.; Xu, M.; Fang, Y. Adaptive Event-Triggered Robust Model Predictive Control for Multiagent Systems with Time-Varying Delay under Arbitrary Network Topology. *IEEE Syst. J.* **2023**, *17*, 5381–5392. [CrossRef]

222. Li, Y.; Zhang, H.; Wang, Z.; Huang, C.; Yan, H. Data-Based Distributed Model Predictive Control for Large-Scale Systems. *Nonlinear Dyn.* **2025**, *113*, 3965–3980. [CrossRef]
223. Yan, C.; Yan, L.; Lv, Y.; Xia, Y. Fully Distributed Event-Triggered Anti-Windup Consensus of Heterogeneous Systems with Input Saturation and an Active Leader. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 12993–13004. [CrossRef]
224. Wijnbergen, P.; Jeeninga, M.; Besselink, B. Nonlinear Spacing Policies for Vehicle Platoons: A Geometric Approach to Decentralized Control. *Syst. Control Lett.* **2021**, *153*, 104954. [CrossRef]
225. Yanakiev, D.; Kanellakopoulos, I. Nonlinear Spacing Policies for Automated Heavy-Duty Vehicles. *IEEE Trans. Veh. Technol.* **1998**, *47*, 1365–1377. [CrossRef]
226. Liu, H.; Wang, Z.; Cui, J.; Zhang, Y. Close-Range Coordination to Enhance Constant Distance Spacing Policies for Platoons of Automated Vehicles. *Sensors* **2024**, *24*, 4753.
227. Tsugawa, S.; Jeschke, S.; Shladover, S.E. A Review of Truck Platooning Projects for Energy Savings. *IEEE Trans. Intell. Veh.* **2016**, *1*, 68–77. [CrossRef]
228. Shladover, S.E.; Su, D.; Lu, X.-Y. Impacts of Cooperative Adaptive Cruise Control on Freeway Traffic Flow. *Transp. Res. Rec.* **2012**, *2324*, 63–70. [CrossRef]
229. Nowakowski, C.; O'Connell, J.; Shladover, S.E.; Cody, D. *Cooperative Adaptive Cruise Control: Testing Drivers' Choices of Following Distances*; California PATH Research Report UCB-ITS-PRR-2010-01; California PATH Program, University of California at Berkeley: Berkeley, CA, USA, 2010.
230. Milanes, V.; Shladover, S.E.; Spring, J.; Nowakowski, C.; Kawazoe, H.; Nakamura, M. Cooperative Adaptive Cruise Control in Real Traffic Situations. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 296–305. [CrossRef]
231. Zhang, H.; Chen, J. Bipartite Consensus of Multi-Agent Systems over Signed Graphs: State Feedback and Output Feedback Control Approaches. *Int. J. Robust Nonlinear Control* **2017**, *27*, 3–14. [CrossRef]
232. Fax, J.A.; Murray, R.M. Graph Laplacians and Stabilisation of Vehicle Formations. In Proceedings of the IFAC World Congress, Barcelona, Spain, 21–26 July 2002; pp. 55–60.
233. Zheng, Y.; Li, S.E.; Wang, J.; Cao, D.; Li, K. Stability and Scalability of Homogeneous Vehicular Platoon: Study on the Influence of Information Flow Topologies. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 14–26. [CrossRef]
234. Martín-Hernández, J.; Wang, H.; Van Mieghem, P.; D'Agostino, G. Algebraic Connectivity of Interdependent Networks. *Phys. A Stat. Mech. Its Appl.* **2014**, *404*, 92–105. [CrossRef]
235. Xu, Q.; Chang, X.; Wang, J.; Chen, C.; Cai, M.; Wang, J.; Li, K.; Cao, D. Cloud-Based Connected Vehicle Control Under Time-Varying Delay: Stability Analysis and Controller Synthesis. *IEEE Trans. Veh. Technol.* **2023**, *72*, 14074–14086. [CrossRef]
236. Zhang, L.; Orosz, G.; Huang, Y. Design and Analysis of Longitudinal Controller for the Platoon with Time-Varying Delay. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 21407–21418. [CrossRef]
237. Ru, S.; Jiao, X. Robust H_∞ Control of Heterogeneous Connected Vehicles Platoon with Random Time-Varying Communication Delay. *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.* **2024**, *238*, 1437–1449. [CrossRef]
238. Zheng, Y.; Li, S.E.; Wang, J.; Cao, D.; Li, K. Distributed Consensus Strategy for Platooning of Vehicles in the Presence of Time-Varying Heterogeneous Communication Delays. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 205–219.
239. Khalifa, A.; Kermorgant, O.; Dominguez, S.; Martinet, P. Platooning of Car-like Vehicles in Urban Environments: Longitudinal Control Considering Actuator Dynamics, Time Delays, and Limited Communication Capabilities. *IEEE Trans. Control Syst. Technol.* **2020**, *29*, 2670–2677.
240. Wang, L.; Duan, Y.; Lai, Y.; Mu, S.; Li, X. V2I-Based Platooning Design with Delay Awareness. *arXiv* **2020**, arXiv:2012.03243.
241. Wang, M.; Hoogendoorn, S.P.; Daamen, W.; van Arem, B.; Happee, R. Situation-Aware Switching of the Communication Topology in Vehicle Platooning. *IFAC-PapersOnLine* **2021**, *54*, 78–84.
242. Xiao, S.; Ge, X.; Han, Q.-L.; Zhang, Y. Adaptive Switched Control for Connected Vehicle Platoon with Unknown Actuator Faults Based on Filter-Connected Multiple Models Strategy. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 7752–7765.
243. Zhang, L.; Boukas, E.-K. Stability and Stabilization of Markovian Jump Linear Systems with Partly Unknown Transition Probabilities. *Automatica* **2009**, *45*, 463–468. [CrossRef]
244. Liberzon, D. *Switching in Systems and Control*; Birkhäuser: Boston, MA, USA, 2003. [CrossRef]
245. Kannan, R.S.; Cheng, L. Analysis of Packet Drops and Channel Crowding in Vehicle Platooning Using V2X Communication. In Proceedings of the IEEE Vehicular Technology Conference (VTC-Fall), Chicago, IL, USA, 27–30 October 2018; pp. 1–7.
246. Wang, T.; Zhao, Y.; Liang, H.; Lam, H.K. Stability Analysis of Connected Vehicle Platoons with Markovian Packet Losses. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 6535–6548.
247. Bae, J.; Choi, J.; Yi, K. Reducing Time Headway in Homogeneous CACC Vehicle Platoons in the Presence of Packet Drops. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; pp. 1726–1731.
248. Lin, X.; Shi, Y.; Johansson, J. Distributed Model Predictive Control of Vehicle Platoons with Random Packet Loss. In Proceedings of the IEEE Conference on Decision and Control (CDC), Austin, TX, USA, 14–17 December 2021; pp. 1836–1841.

249. Ahmed, A.; Guo, G. Event-Triggered-Based Robust Control of Heterogeneous Connected Vehicles Platoon with Communication Delay and Random Packet Loss. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 13808–13822.
250. Halder, S.; Ahlawat, A. Stability Analysis with LMI-Based H_∞ Approach for Synchronisation of Multi-Agent Systems. *IEEE Access* **2020**, *8*, 128152–128162.
251. Tabuada, P. Event-Triggered Real-Time Scheduling of Stabilizing Control Tasks. *IEEE Trans. Autom. Control* **2007**, *52*, 1680–1685. [CrossRef]
252. Dimarogonas, D.V.; Frazzoli, E.; Johansson, K.H. Distributed Event-Triggered Control for Multi-Agent Systems. *IEEE Trans. Autom. Control* **2012**, *57*, 1291–1297. [CrossRef]
253. Heemels, W.P.M.H.; Johansson, K.H.; Tabuada, P. An Introduction to Event-Triggered and Self-Triggered Control. In Proceedings of the IEEE Conference on Decision and Control (CDC), Maui, HI, USA, 10–13 December 2012; pp. 3270–3285.
254. da Silva, E.A.; Mozelli, L.A.; Neto, A.A.; Souza, F.O. Improving the Design of Linear Controllers for Platooning under Disturbances. *IFAC-PapersOnLine* **2025**, *59*, 109–114. [CrossRef]
255. Heeger, D.J. *Linear Systems*; Center for Neural Science, New York University: New York, NY, USA, 2025. Available online: https://www.cns.nyu.edu/david/handouts/linear-systems/linear-systems.html (accessed on 6 June 2025).
256. Apkarian, P.; Noll, D. The $\mathcal{H}_\infty$ Control Problem Is Solved. *AerospaceLab* **2017**, *13*, 1–11.
257. Tzafestas, S.G. *Introduction to Mobile Robot Control*; Elsevier: Amsterdam, The Netherlands, 2014.
258. Rawlings, J.B.; Mayne, D.Q.; Diehl, M. *Model Predictive Control: Theory, Computation, and Design*, 2nd ed.; Nob Hill Publishing: Madison, WI, USA, 2017.
259. Sliding Mode Controller. ScienceDirect Topics. Available online: https://www.sciencedirect.com/topics/engineering/sliding-mode-controller (accessed on 2 November 2025).
260. Utkin, V.I. Sliding Mode Control Design Principles and Applications to Electric Drives. *IEEE Trans. Ind. Electron.* **1993**, *40*, 23–36. [CrossRef]
261. Xu, G.; Wang, Z.; Li, C. Synchronization of Lorenz System Based on Fast Stabilization Sliding Mode Control. *Math. Probl. Eng.* **2015**, *2015*, 282919. [CrossRef]
262. Yang, J.; Li, S.; Yu, X. Sliding-Mode Control for Systems with Mismatched Uncertainties via a Disturbance Observer. *IEEE Trans. Ind. Electron.* **2013**, *60*, 160–169. [CrossRef]
263. Zhu, Y.; Zhu, F.; Lu, T. Distributed Integrated Sliding Mode Control for Vehicle Platoons Based on Disturbance Observer and Multi-Power Reaching Law. *IEEE Access* **2020**, *8*, 208459–208469.
264. Khalil, H.K. *Nonlinear Systems*, 3rd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2002.
265. Viadero-Monasterio, F.; Meléndez-Useros, M.; Jiménez-Salas, M.; López Boada, M.J. Robust Adaptive Heterogeneous Vehicle Platoon Control Based on Disturbances Estimation and Compensation. *IEEE Access* **2024**, *12*, 96924–96935. [CrossRef]
266. Wang, Y.-E.; Sun, X.-M.; Wu, B. Lyapunov–Krasovskii Functionals for Input-to-State Stability of Switched Non-Linear Systems with Time-Varying Input Delay. *IET Control Theory Appl.* **2015**, *9*, 1717–1722. [CrossRef]
267. Wang, X.; Lv, S.; Guo, X.; Li, X. Computation of the Exact Time Delay Margin for Vehicle Platoon Control Systems Using Algebraic Geometry Methods. *J. Syst. Sci. Complex.* **2024**, *37*, 1–20.
268. Swaroop, D.; Hedrick, J.K. String Stability of Interconnected Systems. *IEEE Trans. Autom. Control* **1996**, *41*, 349–357. [CrossRef]
269. Viadero-Monasterio, F.; Meléndez-Useros, M.; Jiménez-Salas, M.; Boada, B.L. Robust Adaptive Control of Heterogeneous Vehicle Platoons in the Presence of Network Disconnections with a Novel String Stability Guarantee. *IEEE Trans. Intell. Veh.* **2026**, *11*, 63–75. [CrossRef]
270. MathWorks. Vehicle Dynamics Blockset. Available online: https://www.mathworks.com/products/vehicle-dynamics.html (accessed on 15 November 2025).
271. MathWorks. Simulink Real-Time Documentation. Available online: https://www.mathworks.com/help/slrt/index.html (accessed on 15 November 2025).
272. DLR Institute of Transportation Systems. SUMO Documentation: TraCI Protocol. Available online: https://sumo.dlr.de/docs/TraCI/Protocol.html (accessed on 15 November 2025).
273. Acosta, A.F.; Espinosa, J.E.; Espinosa, J. TraCI4Matlab: Enabling the Integration of the SUMO Road Traffic Simulator and Matlab® through a Software Re-Engineering Process. In *Modeling Mobility with Open Data*; Behrisch, M., Weber, M., Eds.; Springer: Cham, Switzerland, 2015; pp. 155–170.
274. Veins Project. Frequently Asked Questions (FAQ)—Veins: The Open Source Vehicular Network Simulation Framework. Available online: https://veins.car2x.org/documentation/faq/ (accessed on 15 November 2025).
275. Segata, M.; Joerer, S.; Bloessl, B.; Sommer, C.; Dressler, F.; Lo Cigno, R. PLEXE: A Platooning Extension for Veins. In Proceedings of the IEEE Vehicular Networking Conference (VNC), Paderborn, Germany, 3–5 December 2014; pp. 53–60.
276. Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; Koltun, V. CARLA: An Open Urban Driving Simulator. *arXiv* **2017**, arXiv:1711.03938. [CrossRef]

277. Catozzi , S. Design of a NMPC System for Automated Driving and Integration into the CARLA Simulation Environment. Master's Thesis, Politecnico di Torino, Turin, Italy, 2024.
278. Chen, X.; Zhang, L.; Li, W. An Overview about Pinning Control of Complex Networks. In Proceedings of the International Conference on Modelling, Identification and Control (ICMIC), Wuhan, China, 24–26 June 2012.
279. Wang, X.; Li, X.; Lu, J. Control and Flocking of Networked Systems via Pinning. *IEEE Circuits Syst. Mag.* **2010**, *10*, 83–91. [CrossRef]
280. Olfati-Saber, R.; Fax, J.A.; Murray, R.M. Consensus and Cooperation in Networked Multi-Agent Systems. *Proc. IEEE* **2007**, *95*, 215–233. [CrossRef]
281. Olfati-Saber, R.; Murray, R.M. Consensus Problems in Networks of Agents with Switching Topology and Time-Delays. *IEEE Trans. Autom. Control* **2004**, *49*, 1520–1533. [CrossRef]
282. Ren, W.; Beard, R.W. Consensus Seeking in Multiagent Systems under Dynamically Changing Interaction Topologies. *IEEE Trans. Autom. Control* **2005**, *50*, 655–661. [CrossRef]
283. Moreau, L. Stability of Multiagent Systems with Time-Dependent Communication Links. *IEEE Trans. Autom. Control* **2005**, *50*, 169–182. [CrossRef]
284. Jadbabaie, A.; Lin, J.; Morse, A.S. Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbour Rules. *IEEE Trans. Autom. Control* **2003**, *48*, 988–1001. [CrossRef]
285. Cao, Y.; Yu, W.; Ren, W.; Chen, G. An Overview of Recent Progress in the Study of Distributed Multi-Agent Coordination. *IEEE Trans. Ind. Inform.* **2013**, *9*, 427–438. [CrossRef]

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.