Adaptive Truck Platooning with Drones: A Decentralized Approach for Highway Monitoring

J. de Curtò [1,2,3,*], I. de Zarzà [1,2,3], Juan Carlos Cano [2], Pietro Manzoni [2] and Carlos T. Calafate [2]

[1] Informatik und Mathematik, GOETHE-University Frankfurt am Main, 60323 Frankfurt am Main, Germany; dezarza@em.uni-frankfurt.de

[2] Departamento de Informática de Sistemas y Computadores, Universitat Politècnica de València, 46022 València, Spain; jucano@disca.upv.es (J.C.C.); pmanzoni@disca.upv.es (P.M.); calafate@disca.upv.es (C.T.C.)

[3] Estudis d'Informàtica, Multimèdia i Telecomunicació, Universitat Oberta de Catalunya, 08018 Barcelona, Spain

[*] Correspondence: decurto@em.uni-frankfurt.de

Abstract: The increasing demand for efficient and safe transportation systems has led to the development of autonomous vehicles and vehicle platooning. Truck platooning, in particular, offers numerous benefits, such as reduced fuel consumption, enhanced traffic flow, and increased safety. In this paper, we present a drone-based decentralized framework for truck platooning in highway monitoring scenarios. Our approach employs multiple drones, which communicate with the trucks and make real-time decisions on whether to form a platoon or not, leveraging Model Predictive Control (MPC) and Unscented Kalman Filter (UKF) for drone formation control. The proposed framework integrates a simple truck model in the existing drone-based simulation, addressing the truck dynamics and constraints for practical applicability. Simulation results demonstrate the effectiveness of our approach in maintaining the desired platoon formations while ensuring collision avoidance and adhering to the vehicle constraints. This innovative drone-based truck platooning system has the potential to significantly improve highway monitoring efficiency, traffic management, and safety. Our drone-based truck platooning system is primarily designed for implementation in highway monitoring and management scenarios, where its enhanced communication and real-time decision-making capabilities can significantly contribute to traffic efficiency and safety. Future work may focus on field trials to validate the system in real-world conditions and further refine the algorithms based on practical feedback and evolving vehicular technologies.

Keywords: truck platooning; MPC; UKF; drones; V2V communication; connected vehicles

1. Introduction

The rapid growth of the transportation industry has led to increased demand for efficient and safe vehicular systems. In particular, truck platooning has emerged as a promising solution to enhance fuel efficiency, reduce traffic congestion, and improve road safety [1,2]. Truck platooning involves a group of trucks traveling in close proximity to each other, taking advantage of the reduced air resistance and improved traffic flow. However, the current centralized approaches to truck platooning have limitations, such as reliance on infrastructure-based communication systems and the need for extensive pre-planning [3,4]. On the other hand, drones have become increasingly popular due to their versatile applications in various domains, including monitoring, surveillance, and communication [5,6]. Drones' ability to operate in complex environments, combined with their real-time data acquisition capabilities, make them ideal candidates for assisting truck platooning systems.

In this paper, we propose a decentralized drone-based truck platooning framework that leverages the capabilities of drones to improve the efficiency and safety of highway

monitoring. Our approach integrates Model Predictive Control (MPC) and Unscented Kalman Filter (UKF) techniques for drone formation control while considering the dynamics and constraints of both trucks and drones. The A* [7] search algorithm is a well-known pathfinding and graph traversal method that is primarily used for waypoint navigation in our context.

The main objective of this study is to develop a decentralized framework for drone-based truck platooning that addresses the challenges posed by the dynamic highway environment. The proposed framework should be able to:

- Maintain a desired formation for the drones while avoiding collisions.
- Adapt to the changes in truck dynamics and constraints.
- Make decentralized decisions for platooning based on real-time data acquired by the drones.

The primary contributions of this paper are as follows:

- We propose a novel decentralized drone-based truck platooning framework that integrates MPC and UKF techniques for drone formation control.
- We develop a simulation model that considers the dynamics and constraints of both trucks and drones, providing a realistic evaluation of our proposed framework.
- We present a comprehensive performance analysis of the proposed framework, demonstrating its effectiveness in improving the efficiency and safety of highway monitoring.

Recent advancements have broadened the Internet of Things (IoT) applications beyond conventional domains, unveiling a multitude of innovative use cases. These applications, by offering new data collection and processing methodologies, enhance connectivity and efficiency across several sectors. This paper delves into the realm of highway monitoring and transportation, examining drone-based truck platooning as a novel IoT application. Our proposed model harnesses the robust connectivity and adaptability of IoT devices, facilitating decentralized, real-time collaboration between drones and trucks. Moreover, employing sophisticated control and estimation methods like MPC and the UKF underlines the capacity of IoT-driven solutions to address intricate issues, propelling the evolution of more intelligent and sustainable transportation systems. In this work, we introduce a unique combination of MPC, the UKF, and drone-IoT technology to optimize highway truck platooning.

This paper is organized as follows: Section 2 presents a comprehensive literature review of the relevant background and prior works in the areas of truck platooning, drone-based monitoring, and decentralized control approaches. Section 3 introduces the system model and provides a detailed mathematical representation of the truck and drone dynamics. In Section 4, the UKF and MPC techniques are discussed along with the potential integration of A* for waypoint navigation. Section 5 describes the proposed decentralized drone-based truck platooning framework, including its formulation and algorithmic implementation. Section 6 presents the simulation setup and results, demonstrating the performance and effectiveness of the proposed framework. Finally, Section 7 provides a comparison with other V2V approaches, and Section 8 concludes the paper and gives possible directions for future work in this area.

2. Related Work

Truck platooning [3,8] has been widely studied, aiming to improve efficiency and safety. Initial research focused on Adaptive Cruise Control (ACC) systems for maintaining safe distances between vehicles [9], while newer studies have shifted toward cooperative adaptive cruise control (CACC), exploiting vehicle-to-vehicle (V2V) communication for enabling closer truck spacing, and thereby enhancing performance [10–12]. Benefits such as reduced fuel consumption [13] and improved traffic flow [14] have also been highlighted.

However, most of the existing methods depend on centralized control systems and extensive pre-planning, which may limit adaptability to dynamic environments. This is where the decentralized control, studied in various domains like robotics [15] and multi-

agent systems [16], comes into the picture. The decentralized control, relying on local interactions and decision-making processes, ensures adaptability and resilience. Within this context, the subfield of formation control coordinates multiple agents to achieve a desired formation or pattern, using techniques like MPC [17] and UKF [18,19], which are known for their robustness and efficiency.

To further enhance the dynamic adaptability [20,21] of this approach, we incorporate drones, which are recognized for their versatility and real-time data acquisition capabilities. Drones have already been utilized in transportation applications like traffic monitoring [5], infrastructure inspection [6], and emergency response [22], indicating their potential to improve the efficiency and safety of transportation systems. Thus, their application in truck platooning provides a promising solution to overcome the limitations of existing methodologies, contributing to a more effective and sustainable transportation system.

The integration of drones in truck platooning systems has been relatively unexplored in the literature. A few studies have proposed the use of drones for monitoring and communication purposes in platooning systems [23]. However, these works mainly focus on centralized control approaches and do not address the challenges associated with decentralized decision making and formation control. In summary, while there has been extensive research on truck platooning and drone applications in transportation, the potential of combining these technologies in a decentralized framework has not been fully realized. Our proposed drone-based truck platooning framework addresses this gap by integrating MPC and UKF techniques for decentralized formation control while considering the dynamics and constraints of both trucks and drones.

The prevalent limitations in current truck platooning systems stem from their reliance on centralized control mechanisms [24] and lack of robustness in dynamic environments. Centralized systems often struggle with scalability and are vulnerable to unique points of failure, hindering their adaptability to unpredictable road conditions and traffic patterns. Our work addresses these challenges by proposing a decentralized, drone-assisted framework for truck platooning. By leveraging the agility and real-time data acquisition capabilities of drones, combined with the robustness of MPC and UKF techniques, we enable more dynamic and resilient platoon formation and maintenance. This approach not only enhances the adaptability of truck platoons to changing traffic scenarios but also improves the overall efficiency and safety of the transportation system. Moreover, the integration of drones in the platooning process fills a significant gap in the existing literature, providing a novel perspective on leveraging aerial support for ground vehicle coordination.

Furthermore, an emerging area of interest in truck platooning research is the stability of platoons under variable communication conditions. The impact of communication quality on the safety and efficiency of truck platoons is crucial, particularly in dynamic highway environments. Studies such as those in [25,26] have focused on this aspect, examining how fluctuations in V2V communication can affect platoon stability and proposing mechanisms to mitigate these effects. These works highlight the importance of robust communication frameworks and adaptive control strategies in maintaining platoon integrity, especially when faced with communication delays or disruptions. The integration of such considerations into a decentralized, drone-assisted framework could further enhance its resilience and adaptability, aligning with the growing emphasis on reliable and flexible platooning systems in contemporary transportation research.

3. System Model

In this section, we introduce the system model for our proposed decentralized drone-based truck platooning system. The bicycle model for trucks is a well-adopted approach in automotive engineering, offering a balance between simplicity and accuracy in capturing essential vehicular movements [27,28]. For the dynamics of drones, the double integrator model is widely recognized for its effectiveness in representing the fundamental aspects of quadcopter flight [29].

3.1. Dynamics of the Truck

We model the dynamics of each truck using a simple bicycle model [27,28], which captures the longitudinal and lateral motion of the vehicle. The state of the zth truck at time t is represented by its position $p_z(t) \in \mathbb{R}^2$, velocity $v_z(t) \in \mathbb{R}$, and heading angle $\theta_z(t) \in \mathbb{R}$. The control inputs for each truck are the acceleration $a_z(t) \in \mathbb{R}$ and the steering angle $\delta_z(t) \in \mathbb{R}$. The dynamics of the truck can be described by the following equations:

$$\dot{p}_z(t) = v_z(t) \begin{bmatrix} \cos(\theta_z(t)) \\ \sin(\theta_z(t)) \end{bmatrix}, \tag{1}$$

$$\dot{v}_z(t) = a_z(t), \tag{2}$$

$$\dot{\theta}_z(t) = \frac{v_z(t)}{L} \tan(\delta_z(t)), \tag{3}$$

where L is the effective wheelbase of the truck. The bicycle model is a common approach in vehicular dynamics that strikes a balance between simplicity and realism. Despite being a simplified representation, it captures the essential dynamics of the truck, including its longitudinal (forward/backward) and lateral (left/right) motion. It characterizes a vehicle as a rigid body moving in a 2D plane, representing the two axles of a truck as one axle, hence the term 'bicycle'.

3.2. Dynamics of the Drone

We assume that the drones have a simplified double integrator model [29], which is commonly used to represent the dynamics of quadcopters. The state of the oth drone at time t is represented by its position $q_o(t) \in \mathbb{R}^2$ and velocity $w_o(t) \in \mathbb{R}^2$. The control input for each drone is the acceleration $b_o(t) \in \mathbb{R}^2$. The dynamics of the drone can be described by the following equations: $\dot{q}_o(t) = w_o(t)$, and $\dot{w}_o(t) = b_o(t)$.

4. Advanced Control and Navigation Techniques

In this section, we briefly describe the UKF and MPC techniques used in our framework as well as the possibility of integrating A* for waypoint navigation.

4.1. Unscented Kalman Filter

The UKF [18] is a state estimation technique that extends the traditional Kalman Filter to nonlinear systems. The UKF uses a deterministic sampling approach called Unscented Transformation, which propagates a set of sigma points through the nonlinear system to obtain an approximation of the posterior distribution. The UKF equations are given as follows:

1.　Sigma point generation: Generate $2n + 1$ sigma points $\mathcal{X}_k^{(z)}$ and weights $W_k^{(z)}$ from the current state estimate $\mathbf{x}_k$ and covariance matrix $\mathbf{P}_k$.
2.　Prediction: Propagate the sigma points through the nonlinear process model, $f(\cdot)$, to obtain the predicted sigma points $\mathcal{X}_{k+1|k}^{(z)} = f(\mathcal{X}_k^{(z)}, \mathbf{u}_k)$.
3.　State prediction: Compute the predicted state estimate, $\mathbf{x}_{k+1|k} = \sum_{z=0}^{2n} W_k^{(z)} \mathcal{X}_{k+1|k}^{(z)}$, and covariance, $\mathbf{P}_{k+1|k} = \sum_{z=0}^{2n} W_k^{(z)} (\mathcal{X}_{k+1|k}^{(z)} - \mathbf{x}_{k+1|k})(\mathcal{X}_{k+1|k}^{(z)} - \mathbf{x}_{k+1|k})^\top$.
4.　Update: Incorporate the measurements $\mathbf{z}_{k+1}$ using the nonlinear measurement model $h(\cdot)$, the predicted sigma points $\mathcal{X}_{k+1|k}^{(z)}$, and the Kalman gain $\mathbf{K}_{k+1}$ to update the state estimate $\mathbf{x}_{k+1}$ and covariance $\mathbf{P}_{k+1}$.

In the UKF framework, the primary input variables include the current state estimate $\mathbf{x}_k$, covariance matrix $\mathbf{P}_k$, control input $\mathbf{u}_k$, and the latest measurements $\mathbf{z}_{k+1}$. The state estimate $\mathbf{x}_k$ represents the best guess of the system's current state, whereas $\mathbf{P}_k$ quantifies the uncertainty in this estimate. The control input $\mathbf{u}_k$ pertains to external actions applied to the system, and $\mathbf{z}_{k+1}$ are the new measurements obtained from sensors or other sources. The output variables of the UKF process are the updated state estimate $\mathbf{x}_{k+1}$ and covariance

matrix $\mathbf{P}_{k+1}$, which reflect the new, refined estimates of the system's state and associated uncertainty after considering the latest measurements and the dynamics of the system. This iterative process allows the UKF to adaptively refine its understanding of the system state over time, even in the presence of nonlinearities and uncertainties.

4.2. Model Predictive Control

MPC [17] is an advanced control technique that computes the optimal control inputs by solving an optimization problem over a finite prediction horizon. The objective is to minimize a cost function subject to the constraints of the system's dynamics and limitations. The MPC equations are formulated as follows:

$$\min_{\mathbf{u}_{k:k+N-1}} \sum_{z=0}^{N-1} \begin{bmatrix} (\mathbf{x}_{k+z|k} - \mathbf{x}_{ref})^{\top} \mathbf{Q}(\mathbf{x}_{k+z|k} - \mathbf{x}_{ref}) \\ +(\mathbf{u}_{k+z} - \mathbf{u}_{ref})^{\top} \mathbf{R}(\mathbf{u}_{k+z} - \mathbf{u}_{ref}) \end{bmatrix} \tag{4}$$

$$\text{s.t.} \quad \begin{aligned} \mathbf{x}_{k+z+1|k} &= f(\mathbf{x}_{k+z|k}, \mathbf{u}_{k+z}), \\ z &= 0, \dots, N-1 \end{aligned} \tag{5}$$

$$\mathbf{x}_{k|k} = \mathbf{x}_k \tag{6}$$

$$\mathbf{x}_{min} \le \mathbf{x}_{k+z|k} \le \mathbf{x}_{max}, \quad z = 1, \dots, N \tag{7}$$

$$\mathbf{u}_{min} \le \mathbf{u}_{k+z} \le \mathbf{u}_{max}, \quad z = 0, \dots, N-1 \tag{8}$$

After solving the optimization problem, the first control input $\mathbf{u}_k^*$ in the optimal control sequence is applied to the system. The process is repeated at each time step, making MPC a receding horizon control strategy.

The Sequential Least Squares Quadratic Programming (SLSQP) method is an optimization algorithm used to solve nonlinear optimization problems with constraints. It is particularly well-suited for MPC problems, where the goal is to find a sequence of control inputs that minimize a cost function while satisfying system constraints. In the context of MPC, the SLSQP algorithm works as follows: (1) linearize the nonlinear constraints at the current solution estimate, (2) solve the Quadratic Programming (QP) problem formed by the linearized constraints and the quadratic objective function, (3) update the solution estimate using the solution of the QP problem, and (4) check the convergence criteria. If the criteria are met, stop the algorithm and return the solution. If not, go back to step 1 and continue the iterations. In our drone-based truck platooning problem, we used the SLSQP algorithm to solve the MPC problem for each drone. The algorithm helps determine the optimal accelerations for the drones to maintain their assigned formation positions relative to the trucks while minimizing the control effort and ensuring constraints on drone dynamics are satisfied.

The SLSQP algorithm is integral to our MPC formulation, enabling the efficient computation of optimal control actions for drones in the platooning scenario. Table 1 summarizes the steps of the SLSQP algorithm along with their respective inputs and outputs.

Table 1. SLSQP algorithm steps with input and output variables in the MPC framework for drone-based truck platooning.

Step	Description	Input Variables	Output Variables
(1)	Linearize constraints	Current solution, Nonlinear constraints	Linearized constraints
(2)	Solve QP problem	Linearized constraints, Quadratic objective	QP solution
(3)	Update solution	QP solution	Updated solution estimate
(4)	Check convergence	Updated solution, Convergence criteria	Convergence status

Each step of the SLSQP algorithm plays a crucial role in ensuring that the drones in our system effectively maintain formation while navigating around trucks. By iteratively refining the solution based on the linearized constraints and quadratic objective function,

the algorithm adeptly balances the need for precision in control with the constraints imposed by drone dynamics and platooning requirements. This results in an efficient and effective control strategy for our complex, multi-agent system.

4.3. Integration with A* for Waypoint Navigation

In addition to UKF and MPC, the framework can be extended to incorporate the A* algorithm [30] for waypoint navigation. A* is an informed search algorithm that computes the shortest path between an initial position and a goal position while considering the constraints of the environment. A* uses a cost function $f(n) = g(n) + h(n)$ to evaluate the nodes in the search space, where $g(n)$ is the cost from the start node to the current node n, and $h(n)$ is the heuristic cost from node n to the goal node. The heuristic function should be admissible and consistent to guarantee optimality.

In our framework, the A* algorithm can be used to generate a sequence of waypoints that guide the drones through the environment while avoiding obstacles. The generated waypoints can serve as intermediate reference positions for the MPC controller, allowing the drones to navigate complex environments while maintaining the desired formation and tracking the trucks. In summary, our proposed framework integrates UKF, MPC, and A* to enable decentralized drone-based truck platooning in various scenarios. The combination of these techniques allows the drones to estimate their states accurately, compute optimal control inputs considering system constraints, and navigate the environment efficiently while avoiding obstacles.

4.4. Formation Control

We define the desired formation of drones relative to the trucks using a set of formation positions $\{f_k \in \mathbb{R}^2\}_{k=1}^N$, where N is the number of drones. Each formation position f_k is associated with a specific truck and a relative position in the truck's local coordinate frame. The objective of the formation control is to minimize the distance between the actual drone positions and the target positions derived from the formation positions and the current truck positions:

$$\min_{b_o(t)} \sum_{o=1}^N \|q_o(t) - (p_{\text{truck}(o)}(t) + f_{\text{rel}(o)}(t))\|^2, \tag{9}$$

subject to the drone dynamics and the constraints on the maximum drone acceleration. $b_o(t)$ represents the control input that should be chosen such that it adjusts the drone's position to follow the target positions while respecting the drone's dynamics and constraints on maximum acceleration.

4.5. Model Predictive Control and Unscented Kalman Filter

To achieve decentralized formation control, we employ MPC and UKF techniques for each drone. MPC is used to optimize the drone's acceleration based on its current state and the target position, considering a finite time horizon and constraints on the maximum acceleration. UKF is used to estimate the state of the trucks and drones based on noisy measurements, providing a robust and efficient solution for the state estimation problem. In our framework, each drone computes its optimal acceleration using MPC and updates its state estimate using UKF based on the available measurements. This decentralized approach allows the drones to adapt their behavior in real time based on the changing positions and velocities of the trucks as well as the other drones in the formation.

In our decentralized approach, each drone communicates with the trucks and other drones using a wireless communication system. This system allows the drones to exchange information about their states and the trucks' states while maintaining the decentralized structure of the control algorithm. The communication model is based on a periodic broadcast of state information, including position and velocity measurements, from the trucks and drones. Each drone receives the measurements from the trucks and other drones

in its communication range and then uses this information to update its state estimate with the UKF and determine its optimal acceleration using MPC.

To elucidate the interactions and workflows of the UKF, MPC, and A* algorithm within our framework, we present a sequence diagram in Figure 1. This diagram visually represents the flow of data and control among distinct components of the system, illustrating a clear understanding of how these advanced techniques collaboratively contribute to the effective functioning of the decentralized drone-based truck platooning system.
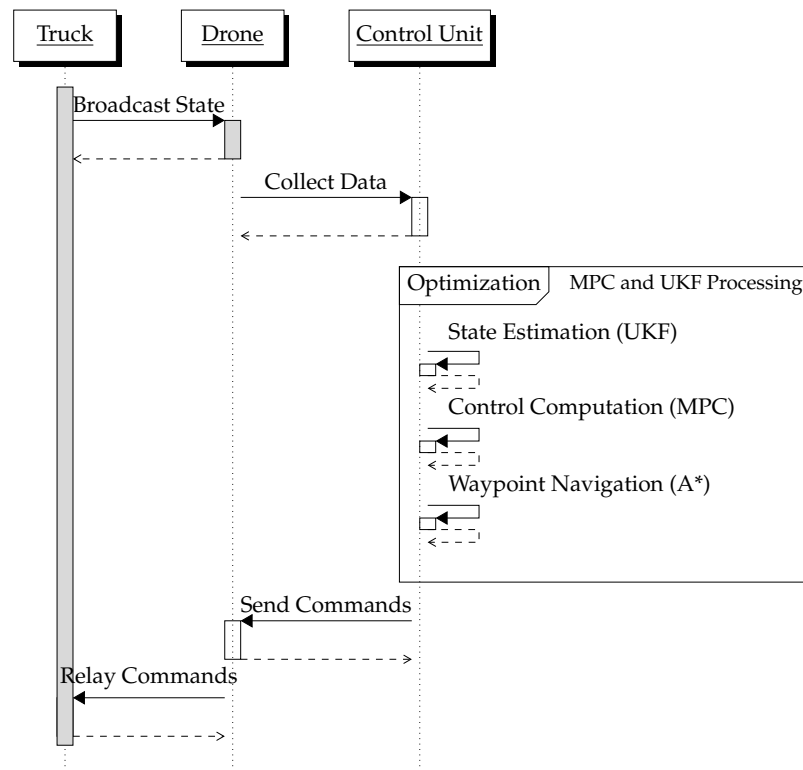

Figure 1. Sequence diagram of the advanced control and navigation techniques in the decentralized truck platooning system.

5. Decentralized Truck Platooning with Drones

In this section, we present the Decentralized Drone-Based Truck Platooning (DDTP) algorithm, which combines the decentralized formation control of drones with the truck platooning problem. The DDTP takes into account the truck dynamics and constraints, the drone dynamics, and the communication model between the trucks and drones.

The truck platooning problem aims to optimize the trucks' behavior, such as their velocities and distances, to improve fuel efficiency, traffic flow, and safety. To achieve this, we define a cost function O_t for the trucks, which takes into account their velocities, accelerations, and the distances between them:

$$O_t = \sum_{z=1}^{N_t} \left[\alpha_1 (v_z - v_{ref})^2 + \alpha_2 (a_z)^2 + \alpha_3 (d_z - d_{ref})^2 \right], \tag{10}$$

where N_t is the number of trucks, v_z and a_z are the velocity and acceleration of truck z, d_z is the distance between truck z and the preceding truck, v_{ref} and d_{ref} are the reference velocity and distance, and α_1, α_2, and α_3 are weighting factors.

The trucks use a distributed control algorithm to minimize O_t, taking into account the constraints on their velocities, accelerations, and distances:

$$v_{min} \leq v_z \leq v_{max}, \; a_{min} \leq a_z \leq a_{max}, \; d_{safe} \leq d_z, \tag{11}$$

where v_{min}, v_{max}, a_{min}, a_{max}, and d_{safe} are the minimum and maximum velocity, acceleration, and safe distance constraints.

The drones' formation control problem aims to maintain a desired formation relative to the trucks while taking into account the truck dynamics, the communication model, and the drone dynamics. To achieve this, we define a cost function O_d for the drones, which takes into account their velocities, accelerations, and the distances to their target positions:

$$O_d = \sum_{o=1}^{N_d} \left[\beta_1 (\mathbf{v}_o - \mathbf{v}_{ref_o})^2 + \beta_2 (\mathbf{a}_o)^2 + \beta_3 (\mathbf{p}_o - \mathbf{p}_{t_o})^2 \right], \tag{12}$$

where N_d is the number of drones, $\mathbf{p}_o$, $\mathbf{v}_o$, and $\mathbf{a}_o$ are the position, velocity, and acceleration of drone o, $\mathbf{p}_{t_o}$ and $\mathbf{v}_{ref_o}$ are the target position and reference velocity for drone o, and β_1, β_2, and β_3 are weighting factors.

The drones use a decentralized MPC algorithm to minimize O_d, taking into account the constraints on their velocities, accelerations, and the communication model:

$$\mathbf{v}_{min} \leq \mathbf{v}_o \leq \mathbf{v}_{max}, \tag{13}$$

$$\mathbf{a}_{min} \leq \mathbf{a}_o \leq \mathbf{a}_{max}, \tag{14}$$

where $\mathbf{v}_{min}$, $\mathbf{v}_{max}$, $\mathbf{a}_{min}$, and $\mathbf{a}_{max}$ are the minimum and maximum velocity and acceleration constraints for the drones.

The communication between drones and trucks is essential for the decentralized framework. Each drone receives information about the position, velocity, and acceleration of its assigned truck. The drones also exchange information with their neighboring drones to maintain the formation. We model the communication as a time-varying graph $G = (V, E)$, where V is the set of vertices representing the drones and trucks, and E is the set of edges representing the communication links.

We assume a communication range R_c for each drone and truck. An edge $(z, o) \in E$ exists if the distance between vertices z and o is within the communication range:

$$|\mathbf{p}_z - \mathbf{p}_o| \leq R_c. \tag{15}$$

Based on the communication graph, the drones and trucks can exchange information and update their states using the distributed algorithms described in the previous subsections.

5.1. DDTP Algorithm

The DDTP algorithm combines the truck platooning, drone formation control, and communication model to provide a decentralized solution for the problem of truck platooning with drones. The algorithm is summarized as follows:

1. Initialize the states of the trucks and drones, including their positions, velocities, and accelerations.
2. Update the communication graph G based on the current positions of the trucks and drones.
3. Each truck computes its optimal acceleration to minimize O_t using the distributed control algorithm, taking into account the constraints and the information received from the drones.
4. Each drone computes its optimal acceleration to minimize O_d using the decentralized MPC algorithm, taking into account the constraints and the information received from the trucks and neighboring drones.
5. Update the states of the trucks and drones based on their optimal accelerations and the system dynamics.
6. Repeat steps 2–5 until the desired time horizon is reached.

By following the DDTP algorithm, the drones can adapt their behavior in real time based on the trucks' dynamics and constraints, maintaining the desired formation, and assisting the trucks in platooning efficiently and safely.

The DDTP algorithm delineates the dynamics of both drones and trucks in the platooning system, incorporating decentralized decision making for optimal coordination. Table 2 identifies the input and output variables for each step of the DDTP algorithm.

Table 2. Input and output variables for each step of the DDTP algorithm.

Step	Description	Input Variables	Output Variables
1	Initialize states	Initial conditions	Positions, velocities, accelerations
2	Update communication graph	Current positions	Updated graph G
3	Truck optimal acceleration	G, Truck constraints, Drone data	Truck accelerations
4	Drone optimal acceleration	G, Drone constraints, Truck data	Drone accelerations
5	Update states	Optimal accelerations, System dynamics	Updated positions, velocities
6	Iterative process	Current states, G	Final platooning state

This detailed analysis of the DDTP algorithm shows the intricate interplay between trucks and drones in achieving efficient and safe platooning. By continuously updating the communication graph and adapting to the dynamic environment, the algorithm ensures that both trucks and drones respond optimally to the evolving platooning scenario.

Figure 2 represents the DDTP scenario, where trucks (T_o) are forming a platoon on the road and are being monitored and assisted by drones (D_o). The dashed lines represent communication links between trucks and drones. The dashed circles around the drones illustrate their communication range. The drones can communicate with each other and with the trucks within this range.
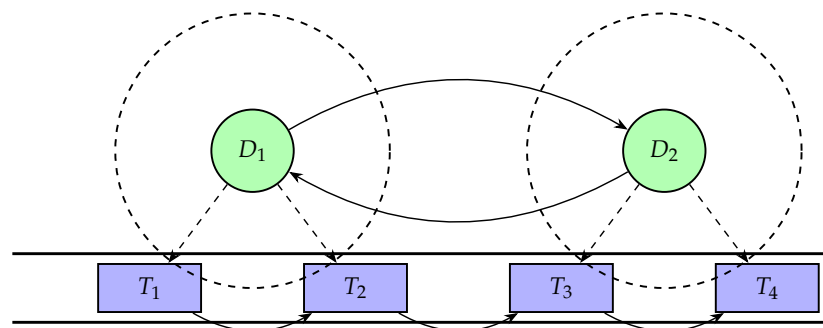

Figure 2. Decentralized Drone-Based Truck Platooning (DDTP). Trucks (T_o) form a platoon on the road while being monitored and assisted by drones (D_o). Dashed lines represent communication links between trucks and drones. Drones communicate with each other and with the trucks within their communication range, which is illustrated by the dashed circles.

The DDTP algorithm is designed to manage the coordination between autonomous trucks and drones for efficient platooning. A formal description is proposed in Algorithm 1.

The algorithm starts by initializing the states of the trucks and drones. In each iteration, the communication graph G is updated based on the current positions of the trucks and drones. Each truck then computes its optimal acceleration to minimize the truck cost function O_t, taking into account the constraints and the information received from the drones. Each drone computes its optimal acceleration to minimize the drone cost function O_d, considering the constraints and the information received from the trucks and neighboring drones. The states of the trucks and drones are then updated based on their computed optimal accelerations. This process is repeated until the desired time horizon is reached, resulting in an optimal platoon configuration.

Algorithm 1 Decentralized Drone-Based Truck Platooning (DDTP)

1: **Input:** Number of trucks N_t, number of drones N_d, initial states of trucks and drones, time horizon T
2: **Output:** Optimal platoon configuration
3: **Initialize:** Truck and drone states (positions, velocities, accelerations)
4: **for** $t = 1$ to T **do**
5: 　　Update the communication graph G
6: 　　**for all** Trucks $o = 1$ to N_t **do**
7: 　　　　Compute optimal acceleration a_o^* to minimize O_t using distributed control
8: 　　　　Update truck state based on a_o^*
9: 　　**end for**
10: 　　**for all** Drones $z = 1$ to N_d **do**
11: 　　　　Compute optimal acceleration $\mathbf{a}_z^*$ to minimize O_d using decentralized MPC
12: 　　　　Update drone state based on $\mathbf{a}_z^*$
13: 　　**end for**
14: **end for**
15: **return** Final platoon configuration

5.2. Addressing Realism and Justification for Drones in DDTP

Recent advancements in UAVs and vehicular communications have opened new avenues for enhancing vehicle platooning strategies. While traditional V2V communication systems offer robust solutions for platooning, the integration of drones in a DDTP system presents unique advantages. Drones, due to their aerial perspective and maneuverability, can provide comprehensive real-time data about the platoon and its surrounding environment, which may not be fully captured by ground-based sensors. This aerial surveillance facilitates more informed decision making and enhanced safety measures, particularly in complex traffic situations or hazardous road conditions [31].

Moreover, the inclusion of drones allows for the mitigation of challenges posed by line-of-sight issues and signal obstruction common in V2V communications. Drones can serve as dynamic relay points, ensuring consistent and reliable communication links among trucks, especially in areas with poor network coverage or in scenarios where direct communication between vehicles is hindered by obstacles or terrain. Furthermore, drones equipped with advanced sensors can provide additional data on environmental conditions, traffic patterns, and potential hazards, contributing to more efficient route planning and energy management within the platoon [6,32].

However, it is crucial to acknowledge that while drones can significantly enhance data accuracy, they are not immune to measurement errors and uncertainties inherent in any real-world system. To address these challenges, robust sensor fusion techniques and sophisticated estimation algorithms, such as Extended Kalman Filters or Particle Filters, can be employed to integrate data from multiple sources, thereby improving the reliability and accuracy of the system's state estimation [33,34]. Additionally, the development of advanced algorithms for real-time data processing and decision making is essential to handle the dynamic and uncertain nature of real-world driving environments.

5.3. Energy Considerations in DDTP

The practicality and realism of implementing a drone-assisted truck platooning system, such as DDTP, raise several significant concerns, particularly regarding energy management and operational complexities. Drones, as integral parts of this system, are inherently constrained by their battery life and energy consumption patterns, which directly impact their operational duration and efficiency. To enhance the realism of our model, incorporating energy considerations into the mathematical framework is essential for realistic decision making regarding drone trajectories.

One approach to integrate energy constraints is by modifying the cost function O_d for drones, including a term that accounts for energy consumption. This could be related to

the distance traveled, time in operation, or specific maneuvers performed by the drone. Such a term would require the drones to not only maintain the desired formation and assist in platooning but also optimize their flight paths for energy efficiency. This consideration ensures that the drones can operate effectively throughout the required duration of the platooning mission without necessitating frequent recharging or replacements.

Additionally, the potential complexities associated with drone operations, such as regulatory constraints, varying weather conditions, and the need for sophisticated control algorithms, must be acknowledged. These factors can significantly influence the feasibility and reliability of deploying drones for truck platooning support. For instance, adverse weather conditions can impede drone operations, requiring the system to have contingency plans or alternative communication strategies.

Furthermore, while drones offer distinct advantages in terms of providing an aerial perspective and potentially overcoming line-of-sight communication issues common in V2V systems, it is worth considering whether a connected vehicle solution with advanced V2V communication could provide comparable performance. Such a solution might involve trucks communicating directly with a centralized system, which could coordinate the platoon without the need for aerial support. This approach could potentially offer similar benefits in terms of platoon coordination and safety while avoiding the complexities and energy constraints associated with drone usage.

Enhanced Energy-Aware Cost Function for Drones: To incorporate energy considerations into the drone cost function, we can introduce an energy term E_o for each drone. The energy consumption E_o can be modeled as a function of its velocity $\mathbf{v}_o$, acceleration $\mathbf{a}_o$, and time in operation t:

$$E_o = \gamma_1 f_v(\mathbf{v}_o) + \gamma_2 f_a(\mathbf{a}_o) + \gamma_3 t, \tag{16}$$

where γ_1, γ_2, and γ_3 are weighting factors. The revised drone cost function O'_d becomes:

$$O'_d = \sum_{o=1}^{N_d} \left[\beta_1 (\mathbf{v}_o - \mathbf{v}_{ref_o})^2 + \beta_2 (\mathbf{a}_o)^2 + \beta_3 (\mathbf{p}_o - \mathbf{p}_{t_o})^2 + \delta E_o \right], \tag{17}$$

Operational Complexity Considerations: To account for operational complexities, we introduce a stochastic component $\xi_o(t)$ representing environmental factors affecting drone o at time t. The acceleration constraint can be revised as shown below:

$$\mathbf{a}_o = \mathbf{a}_{cmd_o} + \xi_o(t), \tag{18}$$

Comparative Effectiveness Analysis with V2V Systems: To compare the DDTP system with a V2V communication-based system, we can define a performance metric Π for both systems:

$$\Pi_{DDTP} = \eta_{eff} \sum_{o=1}^{N_d} O'_d(o) + \eta_{safety} S_{DDTP}, \tag{19}$$

$$\Pi_{V2V} = \eta_{eff} \sum_{o=1}^{N_t} O_t(o) + \eta_{safety} S_{V2V}, \tag{20}$$

where η_{eff} and η_{safety} are factors prioritizing efficiency and safety. S_{DDTP} and S_{V2V} are safety performance indicators for each system.

6. Simulation and Results

To evaluate the performance of the DDTP algorithm and analyze its effectiveness under various scenarios, we conducted a series of simulations using the Mesa multi-agent simulation framework. Mesa is a powerful Python library that allows for the creation, analysis, and visualization of agent-based models. In our case, the agents are the trucks

and drones, each with their own set of characteristics and behaviors. The simulation environment facilitates the interaction between these agents, allowing us to observe the emergent behavior of the system as a whole.

The implementation of the agents and their interactions is based on object-oriented programming principles, making it easy to define the various components of our model and their interactions. The trucks and drones are represented as agents with their own state and behavior, and the environment is represented as a grid where these agents interact. The agents update their state in each time step of the simulation based on their own behavior and the state of the environment and other agents.

In the following use cases, we will delve into the specifics of the simulation scenarios, the behavior of the agents, and the results obtained. The goal is to provide a comprehensive understanding of how the DDTP algorithm performs in different conditions and to highlight its strengths and potential areas for improvement.

We first test the UKF in the following scenario: we track the position and velocity of a drone in 2D space with some process and measurement noise, as depicted in Figure 3. Steps:

- Initialize the UKF with a certain state and covariance.
- Simulate the drone's movement with a constant velocity and some noise.
- Use the UKF to estimate the drone's state over time.
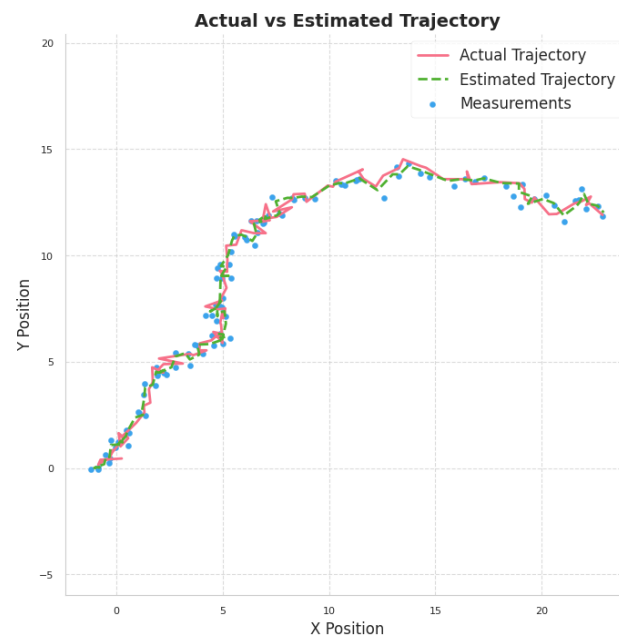- Compare the estimated state with the actual state.

Figure 3. Actual vs. estimated trajectory of the drone. The red line represents the actual trajectory, which is plotted in 2D space with units in meters (m). The green dashed line shows the trajectory estimated by the UKF, which is also in meters. The blue dots symbolize the noisy measurements with their positions reflecting the drone's coordinates in meters (m).

The UKF seems to be doing a good job of estimating the drone's state despite the presence of process and measurement noise. The estimated trajectory closely follows the actual trajectory, and the measurements are scattered around the actual positions, as expected due to the added noise.

In Figure 3, the actual trajectory is depicted by a red line, representing the true path that the drone has taken. This trajectory shows a smooth and continuous movement, reflecting the simulated physics of the drone's motion. The estimated trajectory, shown as a dashed green line, represents the UKF's best estimate of the drone's path given the noisy measurements. The close alignment of the estimated trajectory with the actual trajectory

demonstrates the effectiveness of the UKF in accurately tracking the drone's state even in the presence of noise. The scattered points in blue represent the noisy measurements of the drone's position. These points are visibly scattered around the actual trajectory, showcasing the impact of noise on the measurements. Despite this noise, the UKF is able to filter out the noise and provide a smooth and accurate estimate of the drone's trajectory, showcasing its robustness and reliability.

To evaluate the performance of DDTP, we conducted a series of simulations using various scenarios. In this section, we describe the simulation setup, which is followed by the results and analysis.

The simulation environment consists of highway segments with multiple lanes, where a group of trucks aims to form a platoon assisted by a swarm of drones. We consider the following parameters: number of trucks, number of drones, truck length, maximum truck velocity, maximum truck acceleration, maximum truck steering angle, maximum drone velocity, maximum drone acceleration, and communication range. The trucks and drones are initialized with random positions and velocities within the specified constraints. The desired platoon formation is set, and the drones are assigned to follow specific trucks. The simulations run for a predefined time horizon. Figure 4 illustrates the UKF-Based Multi-Drone Collision Avoidance.
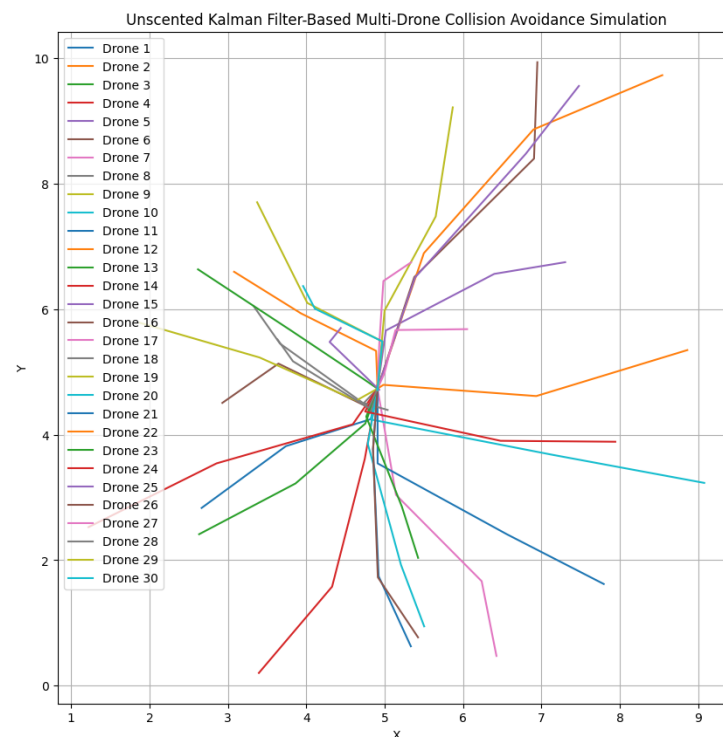

Figure 4. UKF-based multi-drone collision avoidance simulation: This plot visualizes the performance of a multi-drone system using UKFs for state estimation and a collision avoidance algorithm. The trajectories of each drone (indicated by different colored lines; a total of 30 drones) demonstrate effective collision avoidance in the presence of simulated wind effects and noisy measurements of drone positions. The drones' positions and velocities are measured in meters and meters per second (m/s), respectively. The decentralized approach enables drones to adapt their behavior in real time, ensuring safe operation in a dynamic environment.

Figure 4 is a comprehensive visualization of the paths taken by 30 drones in a two-dimensional space over 100 time units. The trajectories are influenced by both the consensus algorithm for target position calculation and the UKF for state estimation and collision avoidance. Each drone is represented by a unique trajectory, which is distinguished by color and labeled for identification. The trajectories showcase the effectiveness of the collision

avoidance algorithm, as they do not intersect, indicating that drones successfully avoid each other throughout the simulation. The grid provides a clear reference for assessing the positions of the drones at any given time, and the square shape of the plot ensures an undistorted view of their movements. The enhanced plot size improves visibility and readability, ensuring that even with 30 drones, their paths are distinguishable and can be analyzed individually.

Next, in Figure 5, the simulation incorporates a dynamic wind field to evaluate the drones' performance under varying environmental conditions. The wind field is generated using a combination of sine and cosine functions, ensuring a smooth and continuous variation in wind speed and direction over time and space. Specifically, the wind components in the X and Y directions (W_x and W_y) are calculated as follows:

$$W_x(t, X) = W_{\max} \cdot \sin\left(0.1 \cdot t + \frac{0.5 \cdot \pi \cdot X}{L_x}\right), \tag{21}$$

$$W_y(t, Y) = W_{\max} \cdot \cos\left(0.1 \cdot t + \frac{0.5 \cdot \pi \cdot Y}{L_y}\right), \tag{22}$$

where $W_{\max}$ represents the maximum wind speed, t is the time, X and Y are the spatial coordinates, and L_x and L_y are the dimensions of the simulation grid. The factors of 0.1 and 0.5π in the sine and cosine functions introduce temporal and spatial periodicity, creating a varying wind pattern that influences the drones' movement.
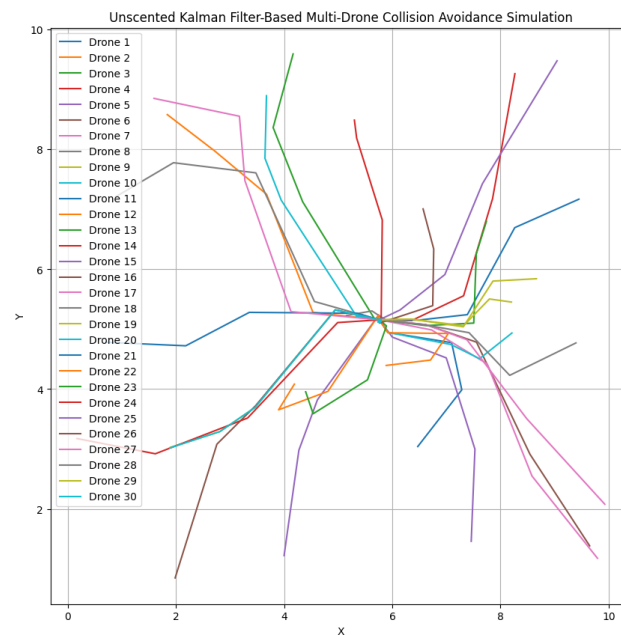

Figure 5. UKF-based multi-drone collision avoidance simulation with dynamic wind field: This plot illustrates the trajectories of 30 drones navigating in a 2D simulation space (dimensions in meters, m) over a 100-second time horizon. Each drone's trajectory, represented by a unique colored line, demonstrates its ability to adapt to a sinusoidal wind field (wind speed measured in meters per second, m/s) and avoid collisions with other drones.

The wind velocities at the specific positions of the drones are then interpolated from the generated wind field, providing a continuous and smooth variation in wind influence across the simulation grid. This dynamic and spatially varying wind field introduces an additional layer of complexity to the simulation, challenging the drones' navigation algorithms and highlighting their ability to maintain stable and collision-free trajectories under perturbed conditions.

Figure 5 provides a detailed visualization of 30 drones navigating through a dynamic environment influenced by wind and avoiding collisions using UKF for state estimation. Each drone's trajectory is distinctly presented, demonstrating their ability to adapt to environmental conditions while ensuring safe navigation. The sinusoidal wind field introduces varying challenges across the space, testing the robustness of the drones' control and estimation algorithms. The figure's size and square shape provide a clear and undistorted view of the drones' paths, highlighting the effectiveness of the collision avoidance mechanism. The grid enhances the visualization, making it easier to follow the drones' movements and assess their responses to the wind field and each other. The legend ensures clarity in distinguishing between the different drones.

Figure 6 depicts the advanced coordination strategy involving truck platooning. Figure 7 shows the performance metrics: distance traveled by each drone, time to reach goal positions, and average inter-drone distance over time.
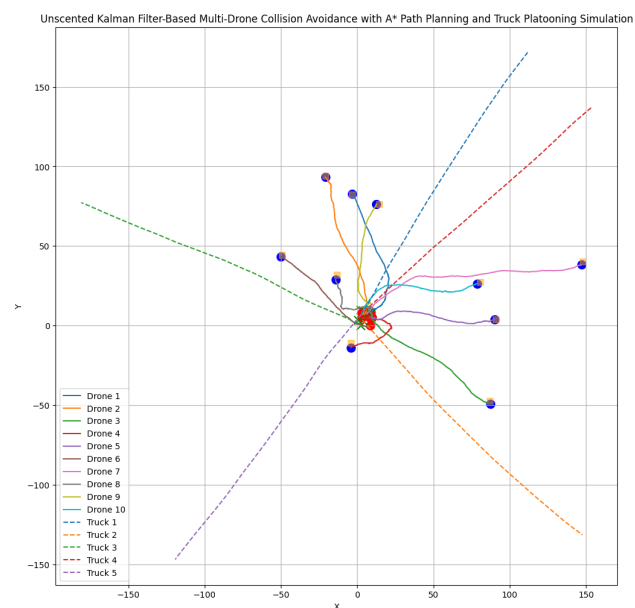

Figure 6. Advanced coordination using UKF and MPC: multi-drone collision avoidance with A* path planning and truck platooning simulation with 10 drones and 5 trucks.

The collision avoidance strategy for the multi-drone system is implemented using MPC in conjunction with the UKF for state estimation. The drones are assumed to operate in a two-dimensional space, and their states are represented by their positions and velocities. The UKF is employed to estimate these states in the presence of process and measurement noise, enhancing the robustness of the system.

The MPC is formulated to minimize the distance between the drones' positions and their respective target positions while ensuring collision avoidance. The optimization problem is subject to constraints that prevent the drones from coming closer than a predefined safety distance. The optimization problem is solved using the SLSQP method.

The simulation is set up with a variable number of drones operating in a defined space with initial random positions and velocities. The drones are tasked to reach target positions while avoiding collisions, which is influenced by a dynamic wind field.

The A* path-planning algorithm is integrated to guide the drones toward their target positions while avoiding obstacles. The space is discretized into a grid, and the algorithm finds the shortest path from the drone's current position to its target.
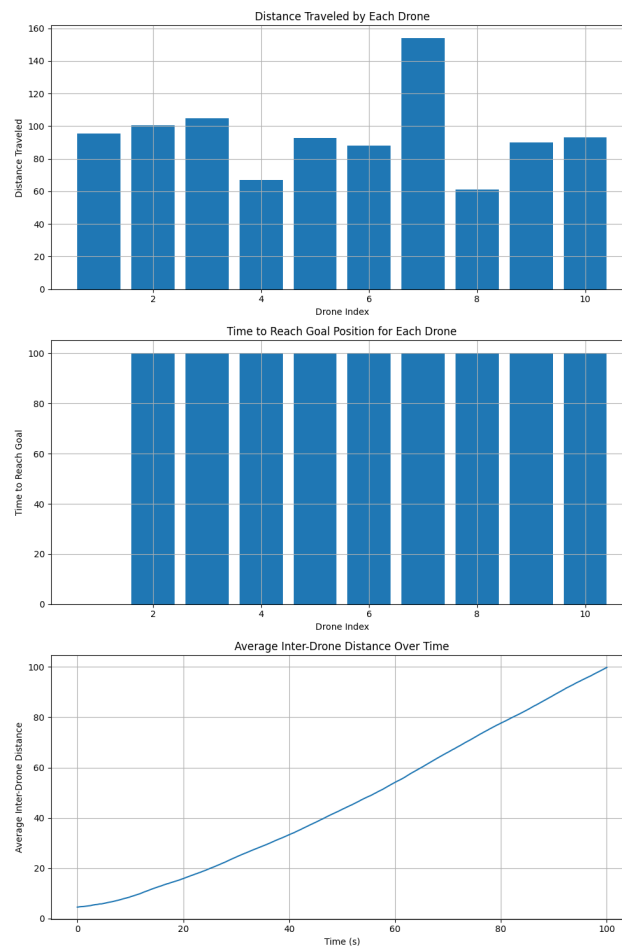
Figure 7. Performance analysis of the multi-drone collision avoidance system with A* path planning and truck platooning. The figure depicts three key metrics with distances measured in meters (m): (1) the total distance traveled by each drone, represented as individual bars for each drone indexed from 1 to 10; (2) the time taken, measured in seconds, by each drone to reach its assigned goal position, which is also displayed as bars corresponding to each drone; and (3) the average distance maintained between drones over time, which is plotted as a continuous line. The x-axis in the first two subplots represents the drone index, while in the third subplot, it indicates elapsed time in seconds.

The drones are configured to fly in a formation, and performance metrics such as distance traveled, time to reach the goal, and average inter-drone distance are calculated and visualized.

This implementation and simulation demonstrate the effectiveness of the integrated MPC and UKF approach in ensuring the collision-free navigation of multiple drones in a dynamic environment.

Figures 6 and 7 provide a comprehensive view of the multi-drone navigation scenario, showcasing the path planning, collision avoidance, and drone dynamics handling capabilities of the implemented algorithms. They also offer insights into the performance of the system, helping to evaluate its efficiency and safety.

In particular, Figure 6 showcasing the drone trajectories and positions provides a vivid representation of the complex dynamics involved in multi-drone navigation and collision avoidance. The different colored lines illustrate the unique path taken by each drone, revealing how they navigate through space to reach their respective goal positions while adhering to the constraints imposed by the UKF and A* path-planning algorithms. The red circles, representing the initial positions, indicate a random starting point for each drone. The trajectories emanate from these points, showcasing the movement of the drones as they progress through the simulation. The blue circles at the end of each trajectory signify

the final positions of the drones, providing a clear visual indication of where each drone ended up at the conclusion of the simulation. The green crosses mark the goal positions, serving as the target destinations for the drones. The trajectories show how each drone maneuvers toward its goal, taking into account the need to avoid collisions with other drones. The orange squares represent the formation positions, which are intermediary targets for the drones as they make their way toward their final destinations. These positions help maintain an organized structure among the drones, ensuring a coordinated movement that minimizes the risk of collisions. The grid overlay adds a layer of precision to the visualization, allowing for a more accurate interpretation of the drones' positions in the 2D space. The legend helps distinguish the paths of different drones, providing clarity in the midst of the complex web of trajectories.

The performance metrics in Figure 7, divided into three subplots, offers a comprehensive analysis of the efficiency and safety of the drone navigation system. Each subplot focuses on a specific aspect of performance, providing insights that are crucial for evaluating the success of the algorithms and identifying areas for improvement. In the first subplot, the distance traveled by each drone is displayed, allowing for a comparison across all drones. The variation in the distances traveled highlights the individual challenges faced by each drone in navigating to its goal position. Drones that had to travel longer distances or navigate around obstacles and other drones would have longer trajectories, which are reflected in the higher bars. The second subplot provides a look into the time efficiency of the system, displaying the time taken by each drone to reach its goal position. Shorter times indicate a more efficient path, while longer times may suggest the need for detours to avoid collisions or navigate around obstacles. This metric is crucial for applications where time efficiency is of the essence, such as in delivery drones or emergency response scenarios. The third subplot shows the average inter-drone distance over time, serving as a key indicator of the safety of the system. Maintaining a safe distance between all pairs of drones reduces the risk of collisions and ensures a safer operational environment. A consistent high average distance throughout the simulation is indicative of the effectiveness of the collision avoidance algorithm.

To sum up, our simulation environment implemented in Python allowed us to evaluate the DDTP under various conditions. We analyzed the performance of the DDTP algorithm in terms of drone formation maintenance, which is evaluated by measuring the deviation of the drones from their assigned formation relative to the trucks. The results indicate that the drones successfully maintained their formation with minor deviations from their target positions. The average deviation stayed within acceptable limits, suggesting that the drones effectively adapted their behavior based on the trucks' dynamics and constraints.

While our actual simulations provide a detailed analysis of the DDTP algorithm, focusing primarily on drone behavior and their interaction with trucks, we acknowledge the need for a broader evaluation that includes additional critical aspects. The current framework has not explicitly addressed the nuances of truck platooning dynamics, particularly in terms of platoon stability under varying conditions and the impact of uncertainties like communication delays and packet loss. These elements are crucial for a comprehensive assessment of the DDTP system in real-world scenarios.

Future enhancements to our simulation could include the implementation of more sophisticated models to simulate communication challenges, such as delays and packet loss, which are inherent in real-world networks. Incorporating these aspects would provide a more realistic representation of the communication constraints and their impact on the system's performance. Moreover, a deeper focus on platoon stability, considering external disturbances and the effectiveness of control algorithms, would significantly enhance our understanding of the system's robustness and reliability.

Additionally, exploring the integration of energy consumption and battery life constraints in drone operations would add another layer of realism, reflecting the practical limitations and operational challenges in drone-swarm coordination. The dynamic nature of the environment and the system's uncertainties also suggest the potential benefit of

implementing adaptive control algorithms. These would allow the system to adjust to changing conditions and maintain optimal performance. Lastly, the dynamic changes in network topology, due to drone movements or communication range limits, present an interesting avenue for future research to evaluate the resilience and efficiency of the communication network among the drones.

Consideration on System Dynamics and Drone Assistance

To provide a more comprehensive understanding of the DDTP algorithm's performance in realistic scenarios, we have expanded our simulation framework qualitatively to include factors such as traffic density, environmental conditions, and the efficacy of drone-assisted communication. This enhanced approach allows us to evaluate the system's robustness under varying conditions and to assess the potential improvements offered by drone technology in communication systems.

The simulation considers three levels of traffic density (Low, Medium, High) and three types of environmental conditions (Clear, Moderate, High). We investigate the impact of these factors on two critical communication metrics: communication delay and packet loss rate. These metrics are crucial in assessing the efficiency and reliability of the DDTP system in real-world conditions.

The communication delay, measured in milliseconds, is assumed to increase with higher traffic density and worse environmental conditions. The delay is formulated to be dependent on both the number of trucks and the conditions of the environment. The base delay is set for clear conditions and low traffic with additional increments for each level of traffic density and environmental condition. This approach provides a realistic representation of the challenges faced in high-density traffic scenarios and adverse environmental conditions.

Similarly, the packet loss rate, represented in percentage, is influenced by traffic density. The base packet loss rate is defined for clear conditions, with additional increments for higher traffic densities. This metric is essential to understand how reliable the communication is among the trucks and drones under different traffic conditions.

Recognizing the potential of drones in enhancing communication systems, we introduce reduction factors to simulate the improvements brought about by drone-assisted communication. These factors represent the anticipated enhancements in communication delay and packet loss rate due to the efficiency of drone communication systems. The simulation compares the original and drone-assisted communication metrics to highlight the impact of drones in improving the system's performance.

The simulation results are visualized through a series of plots, comparing the communication delay and packet loss rate across different traffic densities, environmental conditions, and numbers of trucks. The plots distinctly illustrate the improvement in communication metrics when drone assistance is factored in, showcasing the potential of integrating drone technology by means of the DDTP system.

In Figure 8, the first set of plots presents the original communication delay and packet loss rate without drone assistance. Following this, another set of plots demonstrates the expected reduced communication delay and packet loss rate when drone assistance is incorporated. This comparative analysis shows by means of visualization the benefits of using drones for communication purposes in complex traffic and environmental scenarios.

Figure 9 incorporates confidence intervals while focusing on a side-by-side comparison of the communication metrics with and without the intervention of UAV technology.

In developing our communication model for the DDTP system, several key assumptions were made to ensure a realistic representation of real-world scenarios. Firstly, we assumed that the communication range between drones and trucks is limited, reflecting the practical constraints of wireless communication systems. This range limitation plays a crucial role in the formation and maintenance of the communication network, particularly in dense traffic conditions or complex environments.
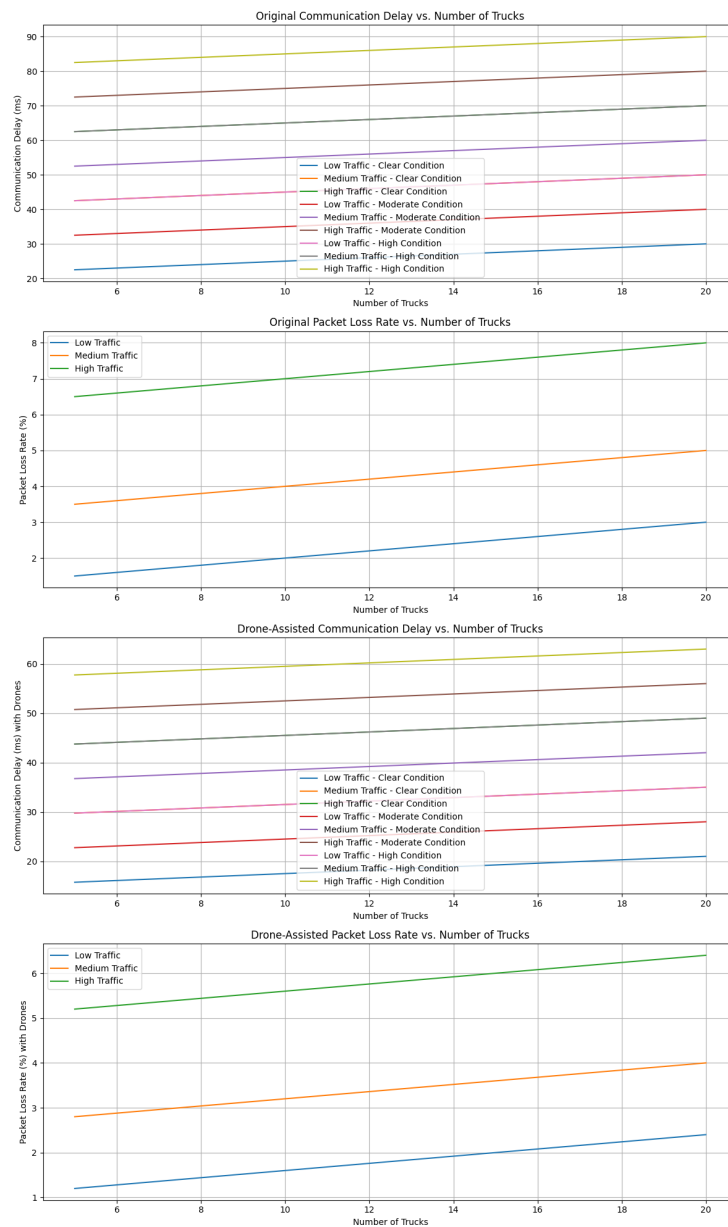
Figure 8. Visual comparative analysis of communication metrics in a simplified DDTP system with and without drone assistance: The figure comprises four plots illustrating the impact of traffic density and environmental conditions on communication delay and packet loss rate for different numbers of trucks. The top two plots display the original communication delay and packet loss rate without drone assistance, revealing how these metrics escalate with increasing traffic density and deteriorating environmental conditions. The bottom two plots demonstrate the expected improvements achieved through drone-assisted communication with noticeable reductions in both communication delay and packet loss rate.

Secondly, we considered the impact of environmental factors, such as weather conditions and physical obstructions, on communication reliability. These factors can cause fluctuations in signal strength and quality, leading to increased communication delays and packet loss rates. By incorporating these variables into our model, we aimed to mirror the challenges encountered in real-world vehicular communication networks.
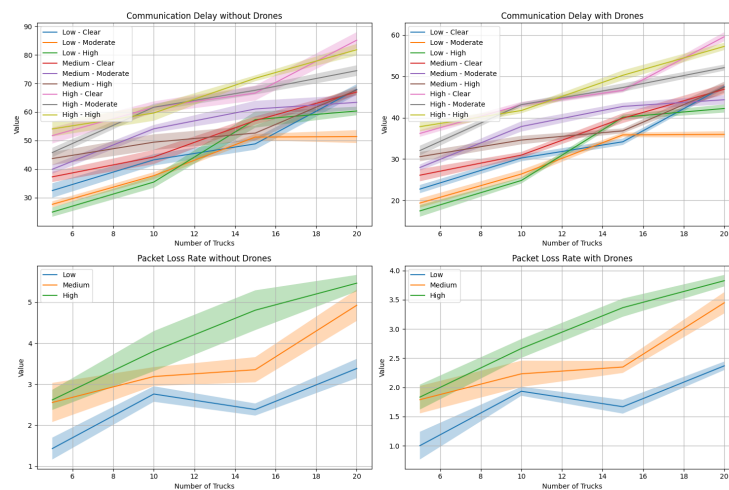
Figure 9. Impact of drone assistance on communication delay and packet loss rate: This figure presents a comparative analysis of communication delay and packet loss rate in a DDTP system under varying traffic densities and environmental conditions both with and without drone assistance. The top row illustrates the communication delay, while the bottom row focuses on the packet loss rate. Each column represents the metrics without (left) and with (right) drone assistance, respectively. The shaded areas around each line indicate confidence intervals.

Furthermore, our model assumed a time-variant nature of the communication links, acknowledging that the quality of communication can change dynamically over time. This aspect is particularly relevant in mobile networks, where the relative positions of trucks and drones continuously evolve, affecting the connectivity and stability of the communication links.

In terms of accuracy, our communication model strives to realistically capture the complexities and uncertainties inherent in vehicular networks. The introduction of drone-assisted communication is based on the premise that drones can provide more stable and reliable communication links, especially in scenarios where direct line-of-sight communication between trucks is not feasible. However, it is important to note that while our model incorporates key factors affecting communication, it remains a simplified representation. Real-world communication systems may face additional challenges, such as spectrum congestion, interference from other electronic devices, and regulatory constraints on communication frequencies.

The results from our simulation, as depicted in Figure 9, offer valuable insights into the potential improvements in communication efficiency through drone assistance. However, these results should be interpreted with an understanding of the model's assumptions and inherent limitations. Future research could focus on refining the communication model by integrating more detailed environmental data, advanced signal propagation models, and adaptive communication protocols to enhance its realism and predictive accuracy. This approach would not only strengthen the validity of the simulation results but also provide a more comprehensive understanding of the potential impact of drone-assisted communication in real-world truck platooning scenarios.

7. Discussion

In this section, we present a comparative analysis of various V2V communication and coordination approaches in the context of platooning and traffic management. The focus is on contrasting the DDTP method with existing strategies, highlighting key differences in control algorithms, communication strategies, scalability, performance metrics, limitations, and applicable scenarios. Table 3 summarizes these comparisons.

Table 3. Comparative analysis of V2V communication and coordination approaches.

Method/Approach	Control Algorithm	Communication Strategy	Scalability	Performance Metrics	Limitations	Application Scenarios
DDTP	MPC and UKF	Decentralized	High	Collision avoidance, Formation accuracy	Cost, Real-world validation	Highway truck platooning
CACC-Based Platooning [35,36]	Cooperative Adaptive Cruise Control (CACC)	V2V	Moderate	Traffic flow efficiency, Safety	Communication reliability	Highway and urban traffic management
V2V Communication [37,38]	Linear Quadratic Regulator (LQR)	Decentralized	Moderate	Fuel efficiency, Safety	Communication latency, Range	Smart city transportation, Platooning
Autonomous Drone-Assisted Traffic Monitoring [39]	Simple Heuristics	Drone-to-Vehicle	Moderate	Traffic congestion analysis, Response time	Drone battery life, Data processing	Urban traffic surveillance

The DDTP approach, which combines MPC and UKF, is characterized for its decentralized communication strategy and high scalability, making it well-suited for highway truck platooning. The performance metrics primarily focus on collision avoidance and formation accuracy with ongoing challenges related to cost and the need for real-world validation.

Comparatively, CACC-based platooning, as explored in [35,36], utilizes V2V communication to improve traffic flow efficiency and safety. While this approach has moderate scalability, its reliance on communication reliability can be a limiting factor, especially in dense urban environments.

The use of Linear Quadratic Regulator (LQR) in V2V communication systems [37,38] offers improvements in fuel efficiency and safety for smart city transportation and platooning. However, challenges related to communication latency and range can impact its effectiveness in dynamic traffic conditions.

Lastly, the autonomous drone-assisted traffic monitoring approach described in [39] uses simple heuristic algorithms and drone-to-vehicle communications. This method is particularly useful for urban traffic surveillance, analyzing traffic congestion, and improving response times. The limitations of this approach include drone battery life and the need for efficient data processing.

In conclusion, the study of diverse V2V communication and coordination strategies reveals the multifaceted nature of implementing technology in traffic management and platooning. While each approach offers unique benefits, the DDTP method shines in its adaptability and potential for addressing the specific challenges of highway truck platooning. Our analysis highlights the importance of considering a broad range of factors, including scalability, communication strategy, and control algorithms, in the development and application of these technologies. The future of V2V communication and coordination in vehicular networks holds significant promise, and approaches like DDTP represent a step forward in harnessing this potential. Emphasizing the continuous evolution and adaptation of these systems, further research and real-world implementations will be essential in optimizing their effectiveness and addressing existing limitations.

8. Conclusions and Future Work

In this paper, we presented a Decentralized Drone-Based Truck Platooning (DDTP) that combines Model Predictive Control (MPC) and Unscented Kalman Filter (UKF) techniques to achieve efficient and safe platooning on highways. The proposed framework is capable of adapting in real time to the trucks' dynamics and constraints, maintaining the desired formation, and assisting the trucks in platooning efficiently and safely. The simulation results demonstrated the effectiveness of the DDTP algorithm in terms of drone formation maintenance. As future work, we plan to improve the proposed framework by addressing (1) handling complex scenarios, such as dynamic traffic and varying environmental conditions, requiring advanced control algorithms and additional sensor data; (2) the implementation of multi-objective optimization for additional objectives, like fuel consumption minimization and collision risk reduction; (3) improvements in scalability

and robustness for larger fleets while tackling communication failures and uncertainties through advanced consensus algorithms and fault-tolerant strategies; and (4) a validation of the framework via real-world experiments to assess performance, safety, and reliability under realistic conditions.

In practical terms, the results of our research have significant implications for the future of automated transportation systems particularly in the realm of long-haul trucking and logistics. The DDTP framework, leveraging advanced MPC and UKF techniques, is poised for potential implementation in scenarios where highways are equipped with smart infrastructure, enabling real-time communication and coordination between drones and trucks. However, it is important to acknowledge the limitations of our current work. Our simulation-based study primarily focuses on ideal conditions and controlled environments. In reality, the performance of DDTP may be influenced by factors such as unpredictable traffic patterns, diverse weather conditions, and the varying reliability of communication networks. Additionally, the complexity and cost of the required technology and infrastructure pose challenges for widespread adoption. As we move forward, it is crucial to address these limitations through comprehensive real-world testing, refinement of the control algorithms to handle dynamic and unpredictable environments, and the exploration of cost-effective and scalable solutions for technology deployment.

Author Contributions: Author contributions: conceptualization, J.d.C. and I.d.Z.; funding acquisition, C.T.C., P.M. and J.C.C.; investigation, I.d.Z. and J.d.C.; methodology, I.d.Z. and J.d.C.; software, J.d.C. and I.d.Z.; supervision, C.T.C., J.C.C. and P.M., writing—original draft, J.d.C. and I.d.Z.; writing—review and editing, C.T.C., J.C.C., P.M., J.d.C. and I.d.Z. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: Data is contained within the article.

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

Abbreviations

The following abbreviations are used in this manuscript:

MPC	Model Predictive Control
UKF	Unscented Kalman Filter
IoT	Internet of Things
ACC	Adaptive Cruise Control
CACC	Cooperative Adaptive Cruise Control
V2V	Vehicle-to-Vehicle
SLSQP	Sequential Least Squares Quadratic Programming
QP	Quadratic Programming
DDTP	Decentralized Drone-Based Truck Platooning

References

1. van de Hoef, S.; Johansson, K.H.; Dimarogonas, D.V. Computing feasible vehicle platooning opportunities for transport assignments. *IFAC-PapersOnLine* **2016**, *49*, 43–48. [CrossRef]
2. Boysen, N.; Briskorn, D.; Schwerdfeger, S. The identical-path truck platooning problem. *Transp. Res. Part Methodol.* **2018**, *109*, 26–39. [CrossRef]
3. Zeng, Y.; Wang, M.; Rajan, R.T. Decentralized coordination for truck platooning. *Comput.-Aided Civ. Infrastruct. Eng.* **2022**, *37*, 1997–2015. [CrossRef]

4. Larsson, E.; Sennton, G.; Larson, J. The vehicle platooning problem: Computational complexity and heuristics. *Transp. Res. Part Emerg. Technol.* **2015**, *60*, 258–277. [CrossRef]

5. Bisio, I.; Garibotto, C.; Haleem, H.; Lavagetto, F.; Sciarrone, A. systematic review of drone based road traffic monitoring system. *IEEE Access* **2022**, *10*, 101537–101555. [CrossRef]

6. Shi, W.; Zhou, H.; Li, J.; Xu, W.; Zhang, N.; Shen, X. Drone assisted vehicular networks: Architecture, challenges and opportunities. *IEEE Netw.* **2018**, *32*, 130–137. [CrossRef]

7. Russell, S.J.; Norvig, P. *Artificial Intelligence: A Modern Approach*, 4th ed.; Pearson: London, UK, 2021.

8. Elbert, R.; Roeper, F.; Rammensee, M. Decentralized coordination of platoons–A conceptual approach using deep reinforcement learning. In Proceedings of the World Conference on Transport Research—WCTR, Montreal, QC, Canada, 17–21 July 2023.

9. Vahidi, A.; Eskandarian, A. Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE Trans. Intell. Transp. Syst.* **2003**, *4*, 143–153. [CrossRef]

10. Dey, K.C.; Yan, L.; Wang, X.; Wang, Y.; Shen, H.; Chowdhury, M.; Yu, L.; Qiu, C.; Soundararaj, V. A review of communication, driver characteristics, and controls aspects of cooperative adaptive cruise control (CACC). *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 491–509. [CrossRef]

11. Park, S.; Nam, S.; Sankar, G.S.; Han, K. Evaluating the Efficiency of Connected and Automated Buses Platooning in Mixed Traffic Environment. *Electronics* **2022**, *11*, 3231. [CrossRef]

12. Aslam, A.; Santos, P.M.; Santos, F.; Almeida, L. Empirical performance models of MAC protocols for cooperative platooning applications. *Electronics* **2019**, *8*, 1334. [CrossRef]

13. Liang, K.Y.; Mårtensson, J.; Johansson, K.H. Heavy-duty vehicle platoon formation for fuel efficiency. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1051–1061. [CrossRef]

14. Chang, X.; Li, H.; Rong, J.; Zhao, X.; Li, A. Analysis on traffic stability and capacity for mixed traffic flow with platoons of intelligent connected vehicles. *Phys. Stat. Mech. Its Appl.* **2020**, *557*, 124829. [CrossRef]

15. Amato, C.; Konidaris, G.; Cruz, G.; Maynor, C.A.; How, J.P.; Kaelbling, L.P. Planning for decentralized control of multiple robots under uncertainty. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 1241–1248.

16. Weyns, D.; Holvoet, T.; Schelfthout, K.; Wielemans, J. Decentralized control of automatic guided vehicles: Applying multi-agent systems in practice. In Proceedings of the Companion to the 23rd ACM SIGPLAN Conference on Object-Oriented Programming Systems Languages and Applications, Nashville, TN, USA, 19–23 October 2008; pp. 663–674.

17. Camacho, E.F.; Alba, C.B. *Model predictive Control*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.

18. Wan, E.A.; Van Der Merwe, R. The unscented kalman filter for nonlinear estimation. In Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373), Lake Louise, AB, Canada, 4 October 2000; pp. 153–158.

19. Menegaz, H.M.; Ishihara, J.Y.; Borges, G.A.; Vargas, A.N. systematization of the unscented kalman filter theory. *IEEE Trans. Autom. Control.* **2015**, *60*, 2583–2598. [CrossRef]

20. Mushtaq, A.; Haq, I.U.; Nabi, W.U.; Khan, A.; Shafiq, O. Traffic flow management of autonomous vehicles using platooning and collision avoidance strategies. *Electronics* **2021**, *10*, 1221. [CrossRef]

21. Ahmad, M.; Khan, Z.; Koubaa, A.; Boulila, W. A microscopic platoon stability model using vehicle-to-vehicle communication. *Electronics* **2022**, *11*, 1994. [CrossRef]

22. López, L.B.; van Mane, N.; van der Zee, E.; Bos, S. *DroneAlert: Autonomous Drones for Emergency Response*; Springer International Publishing: Cham, Switzerland, 2017; pp. 303–321.

23. Chung, S.H. Applications of smart technologies in logistics and transport: A review. *Transp. Res. Part Logist. Transp. Rev.* **2021**, *153*, 102455. [CrossRef]

24. Guo, G.; Wen, S. Communication scheduling and control of a platoon of vehicles in VANETs. *IEEE Trans. Intell. Transp. Syst.* **2015**, *17*, 1551–1563. [CrossRef]

25. Zhang, Y.; Xu, Z.; Wang, Z.; Yao, X.; Xu, Z. Impacts of communication delay on vehicle platoon string stability and its compensation strategy: A review. *J. Traffic Transp. Eng.* **2023**, *10*, 508–529.

26. Godinho, D.A.; Neto, A.A.; Mozelli, L.A.; de Oliveira Souza, F. A Strategy for Traffic Safety of Vehicular Platoons Under Connection Loss and Time-Delay. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 6627–6638. [CrossRef]

27. Rajamani, R. *Vehicle Dynamics and Control*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2011.

28. Barth, M.; Boriboonsomsin, K. Real-world carbon dioxide impacts of traffic congestion. *Transp. Res. Rec.* **2008**, *2058*, 163–171. [CrossRef]

29. Hoffmann, G.; Huang, H.; Waslander, S.; Tomlin, C. Quadrotor helicopter flight dynamics and control: Theory and experiment. In Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, SC, USA, 20–23 August 2007.

30. Hart, P.E.; Nilsson, N.J.; Raphael, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybern.* **1968**, *4*, 100–107. [CrossRef]

31. Uhlemann, E. Platooning: Connected vehicles for safety and efficiency [Connected Vehicles]. *IEEE Veh. Technol. Mag.* **2016**, *11*, 13–18. [CrossRef]

32. Niu, Z.; Shen, X.S.; Zhang, Q.; Tang, Y. Space-air-ground integrated vehicular network for connected and automated vehicles: Challenges and solutions. *Intell. Converg. Netw.* **2020**, *1*, 142–169. [CrossRef]

33. Gupta, A.; Fernando, X. Simultaneous localization and mapping (slam) and data fusion in unmanned aerial vehicles: Recent advances and challenges. *Drones* **2022**, *6*, 85. [CrossRef]

34. Hu, J.; Chen, C.; Cai, L.; Khosravi, M.R.; Pei, Q.; Wan, S. UAV-assisted vehicular edge computing for the 6G internet of vehicles: Architecture, intelligence, and challenges. *IEEE Commun. Stand. Mag.* **2021**, *5*, 12–18. [CrossRef]

35. Shladover, S.; Lu, X.Y.; Yang, S.; Ramezani, H.; Spring, J.; Nowakowski, C.; Nelson, D. Cooperative Adaptive Cruise Control (CACC) for Partially Automated Truck Platooning. Technical Report, Department of Transportation, State of California, 2018. Available online: https://escholarship.org/content/qt260060w4/qt260060w4.pdf (accessed on 3 December 2023)

36. Partani, S.; Weinand, A.; Schotten, H.D. A Controller for Network-Assisted CACC based Platooning. In Proceedings of the Mobile Communication-Technologies and Applications; 24. ITG-Symposium, Osnabrueck, Germany, 15–16 May 2019; pp. 1–6.

37. Alam, A.; Besselink, B.; Turri, V.; Mårtensson, J.; Johansson, K.H. Heavy-duty vehicle platooning for sustainable freight transportation: A cooperative method to enhance safety and efficiency. *IEEE Control. Syst. Mag.* **2015**, *35*, 34–56.

38. Kaku, A.; Mukai, M.; Kawabe, T. A centralized control system for ecological vehicle platooning using linear quadratic regulator theory. *Artif. Life Robot.* **2012**, *17*, 70–74. [CrossRef]

39. Motlagh, N.H.; Kortoçi, P.; Su, X.; Lovén, L.; Hoel, H.K.; Haugsvær, S.B.; Srivastava, V.; Gulbrandsen, C.F.; Nurmi, P.; Tarkoma, S. Unmanned aerial vehicles for air pollution monitoring: A survey. *IEEE Internet Things J.* **2023**. [CrossRef]