International Journal of Intelligent Networks 5 (2024) 19–29

Contents lists available at ScienceDirect

International Journal of Intelligent Networks

journal homepage: www.keaipublishing.com/en/journals/international-journal-of-intelligent-networks

# A method of vehicle networking environment information sharing based on distributed fountain code

Jianhang Liu [a], Xinyao Wang [a,*], Haibin Zhai [b], Shibao Li [a], Xuerong Cui [a], Qian Zhang [a]

[a] China University of Petroleum (East China), Qingdao, 266580, China
[b] National Computer Network Emergency Reponse Technical Team/Coordination Center of China, Beijing, 100029, China

ARTICLE INFO

*Keywords:*
Cooperative perception
Mobility prediction
Autonomous vehicles
V2V
Vehicular ad-hoc network
Neural network
Wireless network

ABSTRACT

The exchange of perceptual information between autonomous vehicles could significantly improve driving safety. In general, obtaining more information means driving more safely. However, Frequent information sharing consumes a significant amount of channel bandwidth resources, which will reduce transmission efficiency and increase delay, especially in crowded cities. This paper presents a novel method of motion prediction compensation to solve this problem. Firstly, we propose a distributed fountain coding scheme to improve transmission efficiency and reduce vehicles' delay in acquiring peripheral information. Secondly, we design a mobile prediction model and information transmission control algorithm to reduce traffic while ensuring information reliability. The simulation results show that the prediction accuracy of this method is above 94 %, the information transmission is reduced by more than 50 %, and the vehicle perception rate is increased by 34 %.

## 1. Introduction

In recent years, Intelligent Transportation Systems (ITS) have emerged as a promising technology in smart cities [1]. By leveraging information processing and communication technologies, ITS can effectively enhance the quality of core transportation operations and services within a city. Autonomous vehicles (AVs) [2–4], as a significant component of ITS, have the ability to operate without human involvement by sensing their surrounding environment. Autonomous driving vehicles have been deployed and tested in certain pilot areas but still face challenges such as slow operating speeds and long response delays. The fundamental reason behind these issues is the limited perception range of individual vehicles due to obstructions from surrounding vehicles, buildings, and other obstacles. The aforementioned limitation has sparked widespread concerns regarding the reliability and safety [5] of current autonomous driving systems.

To enhance autonomous driving vehicles' operational efficiency and safety, researchers have extensively explored and studied how to expand the perception range through vehicle cooperation, as shown in Fig. 1. Researchers [6] are actively investigating methods to integrate data acquired from sensors mounted on surrounding vehicles. The objective is to create a more comprehensive perception area, leading to enhanced safety and reliability in autonomous driving. However, transmitting raw data requires a significant amount of bandwidth and computational resources, which greatly limits the persistence of data transmission and affects the real-time nature of the vehicle's perception range. Even with the fusion of roadside infrastructure and vehicle detection results [7], the amount of information transmitted remains substantial, potentially leading to vehicles receiving outdated data. These approaches aim to build a larger perception range to avoid certain dangers. Still, due to limited channel resources, the information may not be received in time or may become outdated before being processed by the receiving vehicles.

In the context of vehicular networking, vehicles exhibit fast mobility, frequent topology changes, and high packet loss rates. However, the existing TCP protocol's timeout retransmission mechanism leads to frequent retransmissions in complex vehicular network environments, resulting in many invalid data packets and reduced transmission efficiency. Additionally, there is a need to balance channel occupancy and information effectiveness in in-vehicle communication. Frequent data transmission can ensure timely information delivery but may also result in high channel occupancy, impacting other vehicles' communication. On the other hand, longer intervals between information transmissions can reduce channel occupancy but may lead to poorer timeliness of information, affecting driving safety for vehicles.

To tackle these challenges, we suggest a data-sharing solution based

* Corresponding author.
*E-mail addresses:* liujianhang@upc.edu.cn (J. Liu), s21160011@s.upc.edu.cn (X. Wang).

https://doi.org/10.1016/j.ijin.2024.01.001
Received 28 June 2023; Received in revised form 27 December 2023; Accepted 9 January 2024
Available online 18 January 2024

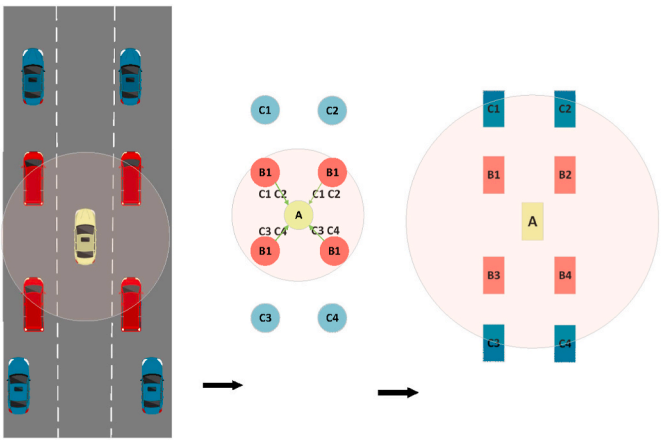



**Fig. 1.** Vehicle collaborative perception.

on distributed fountain codes called the Vehicle Information Perception Method using Distributed Fountain Codes (VIPMDFC).To reduce data transmission, we extract features from the sensed vehicles and utilize the strong error correction capability, low retransmission rate, and fast transmission speed of fountain codes to segment and encode the data, thus improving transmission reliability. Moreover, we dynamically adjust the code length based on the environment to enhance transmission efficiency. The proposed distributed fountain code sharing method allows the receiving end to accept encoded packets from different vehicles, avoiding decoding failures due to network anomalies in a specific transmitting vehicle, effectively addressing high packet loss and long delays in-vehicle communication, and improving communication reliability and efficiency. Additionally, we introduce a trajectory prediction algorithm that uses an attention-based encoder-decoder model to predict object motion states, compensating for information latency caused by network delays and reducing the frequency of information transmission. The main contributions of this paper are as follows.

1. A scheme for vehicular data sharing based on distributed fountain codes is proposed, improving the information reception rate by 34 % in complex environments.
2. A mobility prediction model is proposed for the ”human-vehicle-object” scenario, aiming to compensate for the accuracy of vehicular data transmission in the V2X communication. This model reduces the amount of data by over 50 %. The model accurately predicts future speeds by utilizing the spatiotemporal features of historical traffic, outperforming baseline models in various metrics. The prediction accuracy of this model exceeds 94 %.
3. A simulation experiment platform based on motion trajectory inference is constructed to validate the effectiveness of the vehicle cooperative-assisted driving method.

The main contents of the rest of our paper are organized as follows. In the second section, we summarize and review the related research work. In the third section, the system model and problem formulation are described, and the constraints and performance indexes of the algorithm are given. Section 4 compares VIPMDFC with other baselines and analyzes simulation results. Finally, we summarize this thesis in section 5.

## 2. Related works

Efficient collaborative perception methods should consider the size of information and network efficiency, focusing on the effectiveness of shared knowledge. Capitalizing on sensor data’s inherent characteristics makes it possible to filter or compress the data before transmission. This approach enhances the availability of shared messages, ensuring a more efficient and reliable data-sharing process in autonomous driving systems. In recent years, numerous studies have been proposed on vehicle collaborative perception and edge intelligence. Below, we will introduce relevant research from different perspectives.

### 2.1. Vehicle cooperative perception information transmission scheme

Numerous research studies have tried to mitigate the communication burden on the channel by scrutinizing the information shared among vehicles. In Ref. [8], the authors explore the realm of Vehicle-to-Vehicle (V2V) cooperative communication and employ index coding techniques to minimize the number of transmissions necessary for efficient data exchange. In Ref. [9], the authors design a new type of vehicle-to-vehicle message and define a new transmission protocol called CEM, enabling connected vehicles to share raw data from the global satellite navigation system. In Ref. [10], the authors have devised a deep reinforcement learning-based approach due to the potential redundancy in data transmission through the V2V broadcast communication mode for collaborative perception. This method is designed to facilitate centralized decision-making and optimize the collaborative effects of cooperative perception. By leveraging deep reinforcement learning techniques, the system aims to enhance the efficiency and effectiveness of data sharing among vehicles for improved perception capabilities. In Ref. [11], the authors propose using deep reinforcement learning to select the data to be transmitted, which can alleviate the network load in vehicle networks. In Ref. [12], addressing the collaborative message scheduling problem, the authors present an optimization method for a distributed set of sensor nodes to achieve real-time situational awareness and improve safety-critical decision-making. In Ref. [13], the authors design a mathematically optimized strategy for collaborative augmented reality (AVR) underlay, effectively improving system throughput and energy savings.

These methods are designed for collaborative perception scenarios in broadcast V2V communication, and they can improve collective perception efficiency based on the characteristics of in-vehicle sensors. However, in broadcast communication scenarios, wireless channel resources are limited, and multiple vehicles have to share the same wireless channel, which imposes limitations on the interaction between cars and neighboring vehicles. To address this, we have designed an information-sharing method based on distributed fountain codes at the application layer, which effectively enhances the reception rate of information.

### 2.2. Vehicle mobility prediction

With the development of vehicular networking technology, vehicle speed prediction has been widely applied in autonomous driving and vehicle cooperation. In Ref. [14], the authors present an enhanced method for predicting the short-term speed of intelligent fuel cell vehicles, incorporating spatiotemporal visual deep neural networks. This method capitalizes on image sequences captured by front cameras, which serve as environmental information, and historical speed sequences, which provide motion information. By combining these two data sources, the method achieves improved accuracy in predicting vehicle speeds. In Ref. [15], the authors proposed a learning model based on human driving preferences for vehicle speed prediction. This model addresses the shortcomings of existing research that overlook human driving preferences. They also developed an optimized speed prediction algorithm that retrieves the optimal speed trajectory by estimating driving preferences. In Ref. [16], The authors present an integrated spatiotemporal framework that utilizes historical and real-time data collected from vehicles and infrastructure. By leveraging this combined dataset, the framework enables comprehensive analysis and modeling of the spatial and temporal aspects of the transportation system, facilitating improved decision-making and optimization in various applications. The authors utilize this framework to forecast traction loads specific to particular locations and predict speed profiles

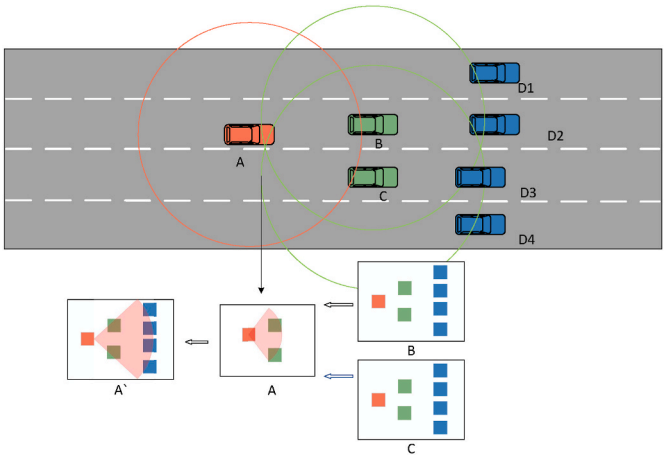



**Fig. 2.** Car A can only perceive car B and car C, and car B and car C will transmit the perceived information to car A, and car A can perceive a wider range.

specific to different time intervals. This enables the facilitation of multi-range traction power and speed prediction, enhancing the efficiency and performance of various transportation systems and applications. The primary objective is to optimize the energy efficiency of connected vehicles. The authors also incorporate Bayesian networks and shockwave models to enhance the accuracy of these predictions. In Ref. [17], the authors proposed a deep learning-based customized path speed prediction algorithm called Path Speed Prediction Neural Network (PSPNN). This framework introduces the capability to customize prediction units specifically for individualized path units instead of fixed road segments. This customization enhances the ability of the system to cater to the unique needs of traffic managers and individual travelers, providing them with more accurate and personalized predictions for their respective paths. In Ref. [18], the authors proposed a fast, interactive multi-vehicle trajectory prediction method. This method achieves high-speed and accurate long-term trajectory prediction by leveraging parallel computing scenarios. It accomplishes this by effectively learning and modeling multiple vehicles' spatial and temporal dependencies. This parallel computing approach enables efficient and simultaneous processing of multiple vehicles' trajectories, improving prediction performance. In Ref. [19], the authors presented an energy-saving driving technique for heavy-duty vehicles in mountainous areas. Through careful observation of the deceleration patterns exhibited by the preceding vehicle during uphill driving, the authors estimate its power capability and utilize this information to predict its future speed within the defined look-ahead range.

Various methods are available to predict the state of a vehicle in different scenarios. However, these methods usually require a lot of computation in common scenarios and may not be able to provide real-time computation on onboard chips. This paper proposes an information-sharing method, which features a lightweight mobility prediction model suitable for on-board edge computing while ensuring prediction accuracy.

### 2.3. Vehicle collaborative perception scheme

In [20], the authors suggest modifying the existing ETSI solution to enhance redundancy control and prevent the transmission of unnecessary CPM (Control Plane Messages) data or messages. In Ref. [21], a scalable 5G Multi-Access Edge Computing (MEC)-driven vehicle-infrastructure cooperative system is proposed. The system takes advantage of the edge offloading capability provided by 5G MEC. Its primary objective is to provide high-precision map-based environment perception specifically tailored for autonomous driving applications. By leveraging the computational resources at the network edge, the system enhances the efficiency and accuracy of environment perception, thereby contributing to the advancement of autonomous driving technology. In Ref. [22], the authors utilize an information dissemination mechanism to improve the perception capabilities of collaborating vehicles within their maximum communication range. This mechanism facilitates the efficient exchange and sharing of relevant information among vehicles, enabling them to have a more comprehensive and up-to-date understanding of their surroundings. By disseminating information among collaborating vehicles, the system enhances collective perception, improving coordination and decision-making in various cooperative driving scenarios. This mechanism facilitates the sharing of relevant information among vehicles, enabling them to comprehensively understand their surrounding environment. A probabilistic dynamic collaboration priority beacon scheme is proposed to select the most suitable collaborating vehicles for transmission based on their onboard sensor facilities, significantly reducing contention on the channel. In Ref. [23], a set of collective perception message generation rules is proposed, defining the timing and content of message generation by vehicles to reduce network traffic and congestion. In Ref. [24], a perception framework is proposed for sending and compensating for the timeliness of perception information from transmitting vehicles. In Ref. [25], the authors propose Enhanced Information Cooperation Perception (AICP), the first rapid filtering system that optimizes the information content of shared vehicle data to improve fusion rendering. A lightweight routing protocol is also proposed to facilitate data encapsulation, quick interpretation, and transmission while reducing latency.

These works optimize the information content of shared vehicle data by selecting receiving vehicles or determining which information to send. However, in complex environments, the real-time nature of information is often not guaranteed due to deteriorating channel quality, leading to delayed information reception by vehicles. This paper improves the information reception rate through distributed fountain codes and replaces continuous data transmission with mobility prediction. This approach significantly reduces the amount of data transmitted while ensuring a certain level of real-time information delivery.

## 3. System model

### 3.1. System model overview

The system model, as shown in Fig. 2, consists of a collection of $N = \{1, 2, 3, \ldots, N\}$ vehicles driving in the area, with velocities $V^t = \{v_1^t, v_2^t, \ldots, v_N^t\}$ at time t. Each participating vehicle is equipped with an Advanced Driver Assistance System (ADAS) or autonomous system comprising multiple cameras oriented in different directions to view the vehicle's surroundings comprehensively. The system incorporates GNSS (Global Navigation Satellite System) and IMU (Inertial Measurement Unit) technologies for real-time kinematics and positioning. These technologies provide accurate and precise location information, enabling the system to determine the vehicle's position and movements in real-time. Additionally, the system utilizes a wireless interface, such as DSRC (Dedicated Short-Range Communication) or C–V2X (Cellular Vehicle-to-Everything), to establish communication with other devices on the road. This wireless connectivity enables the exchange of information and coordination among vehicles, contributing to enhanced safety and efficiency in various intelligent transportation systems applications. The sensing vehicles gather information about the surrounding road environment through onboard sensors such as radar, LiDAR, GPS, and cameras and broadcast this information to nearby vehicles. Our system assumes that it possesses various capabilities to perceive object information accurately. These capabilities include precise positioning and localization, as supported by Refs. [26,27]. Additionally, the system incorporates relative velocity estimation, distance and angle estimation, and perspective transformation, as outlined in

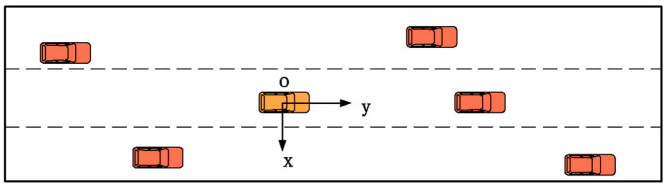



**Fig. 3.** The coordinate system used for vehicle mobility prediction displays the predicted vehicle in brownish-yellow color, while the neighboring vehicles are shown in red color.

Ref. [28]. These capabilities collectively enable the system to perceive objects in its environment accurately, forming a foundation for effective decision-making and autonomous driving functionality.

### 3.2. Communication model

The communication loss between the sending and receiving vehicles is described by equation (1) [26].

$$P_{RX}(d) = P_{TX} + G - \sum PL(d) \tag{1}$$

Where $P_{RX}$ is the received power, $P_{TX}$ is the transmitted power, $G$ is the antenna gain, $PL(d)$ is the path loss component due to various fading effects. We employ an empirical shadow fading model suitable for VANETs, which adopts a dual-slope form discussed in Refs. [27,28].

In the context of line of sight (LoS) or obstructed line of sight (OLoS) scenarios, we can express the path loss using Equation (2). In this equation, the variable $d$ denotes the distance separating the transmitter and the receiver. The term $PL_0$ represents the path loss observed at the reference distance $d_0$, while $d_b$ corresponds to the breakpoint distance. The values of $n_1$ and $n_2$ are loss exponents derived through regression analysis. Additionally, we incorporate a zero-mean Gaussian random variable, denoted as $X_\sigma$, which accounts for inherent variability with a standard deviation of $\sigma$. By utilizing this mathematical model, we can accurately estimate the path loss encountered in various scenarios, accounting for factors such as distance, reference path loss, breakpoints, loss exponents, and random fluctuations.

For non-line of sight (NLoS) scenarios, where the path between the transmitter and the receiver is obstructed by buildings, obstacles, etc., such as in urban areas or at intersections, the path loss model used is shown in equation (3) at the bottom of the page, where $d_t$ is the distance from the transmitter to the center of the intersection, $d_r$ is the distance from the receiver to the center of the intersection, $w_r$ is the width of the street, $d_w$ is the distance between the transmitter and the wall, and $\lambda$ is the wavelength.

$$PL_{LoS/OLoS}(d) = \begin{cases} PL_0 + 10n_1\log_{10}\left(\dfrac{d}{d_0}\right) + X_\sigma & \text{if } d_0 \le d \le d_b \\ PL_0 + 10n_1\log_{10}\left(\dfrac{d_b}{d_0}\right) + 10n_2\log_{10}\left(\dfrac{d}{d_b}\right) + X_\sigma & \text{if } d_b \le d \end{cases} \tag{2}$$

$$PL_{NLoS}(d_t, d_r) = \begin{cases} PL_{0_{NLoS}} + 10n_3\log_{10}\left(\dfrac{d_t^{0.957}}{(d_w w_r)^{0.81}}\dfrac{4\pi d_r}{\lambda}\right) + X_\sigma & \text{if } d_r \le d_b \\ PL_{0_{NLoS}} + 10n_3\log_{10}\left(\dfrac{d_t^{0.957}}{(d_w w_r)^{0.81}}\dfrac{4\pi d_r^2}{\lambda d_b}\right) + X_\sigma & \text{if } d_b < d_r \end{cases} \tag{3}$$

We model the channel of inter-vehicle communication as a quasi-static flat Rayleigh fading channel, and the following formula calculates the information transmission rate:

$$R_{i,j}(t) = W_{i,j}(t) log_2\left(a + \frac{S_{i,j}(t)}{N_0}\right) \tag{4}$$

Where $W_{i,j}(t)$ is the channel transmission bandwidth, $N_0$ is the Gaussian noise power, and $S_{i,j}(t)$ is the vehicle received power, calculated by the following formula:

$$S_{i,j}(t) = P(t)d^{-v}|h|^2 \tag{5}$$

Where $P(t)$ is the vehicle transmission power, $d$ represents the channel fading coefficient, and $h$ represents the distance between vehicles.

### 3.3. Vehicle mobility prediction

The prediction of future vehicle arrivals and departures relies on mobility prediction techniques. Extensive research, such as the analysis of mobility patterns from a dataset of 100,000 anonymous mobile phone users [29] and the findings presented in Ref. [30], has highlighted the predictability of vehicle mobility. Unlike previous studies that make assumptions about fixed velocity or acceleration, our approach involves predicting the speed of vehicles for specific time intervals based on their historical trajectories. By combining the predicted speed and position information, we can forecast future electric vehicles' arrival and departure times. To accomplish this, we propose a mobility prediction model incorporating the Attention mechanism, enabling enhanced accuracy in predicting vehicle movements.

#### 3.3.1. Mobility prediction problem definition

This section aims to forecast the future speed of the target vehicle T by leveraging the historical trajectories and speeds of both the target vehicle T and the surrounding vehicles within a given time interval $t_{obs}$. Our objective is to analyze the collected data and employ predictive techniques to estimate the speed the target vehicle T will likely attain in the future. By considering the past movements and speeds of the target vehicle T and its neighboring vehicles within the defined time interval $t_{obs}$, we aim to generate reliable predictions regarding the future speed of the target vehicle T.

We use the past state information of the target vehicle and n neighboring vehicles as input. The state trajectory of vehicle $i$ is defined as follows:

$$X_i = \{x_i^1, \ldots, x_i^{t_{obs}}\} \tag{6}$$

where $x_i^t$ is the state vector, given by equation (7).

$$x_i^t = (x_i^t, y_i^t, v_i^t, a_i^t, class) \tag{7}$$

The status of the target vehicle $X_T$ will be recorded.

We use a stationary reference frame where the origin O, at time t, represents the position of the target vehicle, as shown in Fig. 3. The y-axis points in the direction of motion on the highway, while the x-axis is perpendicular to the motion. This allows our model to be independent of how vehicle trajectories are obtained, especially applicable to scenarios involving onboard sensors on autonomous vehicles. It also makes the model unaffected by road curvature as long as a lane estimation algorithm is available for the onboard vehicle.

The parameter T represents the output of our model, which characterizes the probability distribution of the predicted position for the target vehicle:

$$T_{pred} = [y_{pred}^{t_{obs}+1}, \ldots, y_{pred}^{t_{obs}+t_f}] \tag{8}$$

Where $y^t$ represents the predicted velocity of the target vehicle.

The distribution of velocity at time $t \in \{t_{obs} + 1, \ldots, t_{obs} + t_f\}$ can be represented as a Gaussian distribution with parameters $\Theta = (\mu^t, \Sigma^t)$, as inferred by our model, i.e., the conditional probability distribution $P(Y|X)$:

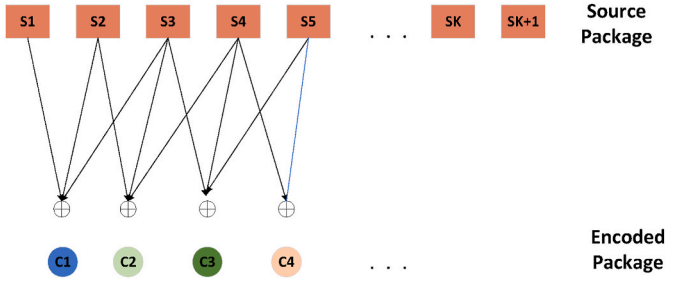



**Fig. 4.** Fountain code coding phase.

$$y^t \sim \mathbb{N}(\mu^t, \Sigma^t) \tag{9}$$

Where $\mu^t$ is the mean vector and $\Sigma^t$ is the covariance matrix:

$$\mu^t = \begin{pmatrix} \mu_x^t \\ \mu_x^t \end{pmatrix}, \Sigma^t = \begin{pmatrix} (\sigma_x^t)^2 & \sigma_x^t \sigma_y^t \rho^t \\ \sigma_x^t \sigma_y^t \rho^t & (\sigma_y^t)^2 \end{pmatrix} \tag{10}$$

We evaluate our model by using the mean vector $\mu^t$ as the predicted position $y^t$.

### 3.3.2. Network model

Our model architecture consists of three main components.

- The encoding layer of the system involves utilizing a GRU (Gated Recurrent Unit) encoder to capture the temporal evolution of vehicle trajectories and their motion characteristics. This encoding process enables the representation of the trajectories' dynamic nature and associated motion features.
- The attention module connects the encoder and decoder's hidden states. It is based on spatiotemporal encoding of the output features from the encoding layer and forms a vector representing the contextual influence.
- In the decoding layer, a contextual vector is received, containing information about neighboring vehicles and the encoded motion of the target vehicle. This vector is used to generate the predicted future velocity of the target vehicle. By analyzing combined information from neighboring vehicles and the encoded motion of the target vehicle, our decoding layer estimates the velocity of the target vehicle in the upcoming time frame.

The encoding layer encodes the trajectories of vehicles belonging to the neighborhood of the target vehicle at time $t = t_{obs}$. We use a GRU layer to map the state vectors $x_i^t$ of each vehicle i to the vector $e_i^t$.

$$e_i^t = \psi(x_i^t; W_{embedding}) \tag{11}$$

Where $\psi$ is the fully connected function with the activation function LeakyReLU, and $W_{embedding}$ represents the weight parameters of the GRU layer.

The GRU encoder is provided with the embedding vectors of each vehicle $i$, with a time step of $t = \{1, \ldots, t_{obs}\}$.

$$h_i^t = GRU(h_i^{t-1}, e_i^t; W_{enc}) \tag{12}$$

Where $h_i^t$ is the hidden state of the $i$ neighboring vehicle at time $t$, and $W_{enc}$ represents the weights of the GRU. The weights are shared across all encoders. We use the hidden states of the GRU encoder, computed over the observation time, as the input to the attention module instead of computing vehicle relationships at each time step.

We utilize vehicle interactions in the following way.

- We map the encoding outputs of the vehicles to query.

$$Q_l = \theta_l(h_T^{t_{obs}}, W_{\theta_l}) \tag{13}$$

- Map the target vehicle code to the keys.

$$K_l = \varphi_l(H, W_{\varphi_l}) \tag{14}$$

- The encoding of the target vehicle is mapped to keys.

$$K_l = \rho_l(H, W_{\rho_l}) \tag{15}$$

Where $W_{\theta_l}$, $W_{\varphi_l}$ and $W_{\rho_l}$ are weight matrices learned for each attention head $l$.

Here we use dot-product attention, where the weights represent the influence between the historical states of the vehicles. They are the product of the query $Q$ and the key $K$.

$$\alpha_l = softmax\left(\frac{Q_l K_l^T}{\sqrt{d}}\right) V \tag{16}$$

$Q_l K_l^T$ is the matrix used for calculating the dot-product similarity through matrix multiplication. d is a scaling factor equal to the dimension of the projection space.

### 3.3.3. Mobility prediction

We use GRU as the decoder for mobility prediction. The GRU decoder takes the context vector $z$, which includes selected information about vehicle interactions and the motion encoding of the target vehicle, as input. It generates the predicted parameters of the future velocity distribution for the target vehicle, with a time step of $t = t_{obs} + 1, \ldots, t_{obs} + t_f$.

$$\Theta^t = \Lambda(GRU(h_{dec}^{h-1}; W_{dec})) \tag{17}$$

The parameter $\Theta^t$ denotes the predicted parameters of the velocity distribution at time $t$. It is computed by applying a fully connected function $\Lambda$, with the weight parameter $W_{dec}$ of the GRU decoder, to the hidden state $h_{dec}^{t-1}$. This transformation enables the extraction and representation of the relevant features and dependencies in the hidden state, leading to the estimation of the velocity distribution parameters for the given time step.

Our model uses the mean squared error (MSE) function as the loss function during training.

$$L_{MSE}(T_{pred}) = \frac{1}{n}\Sigma_{i=1}^{n}(y_{pred} - y^t) \tag{18}$$

## 3.4. Vehicle networking data sharing method

### 3.4.1. Information sharing scheme based on distributed fountain code

In the context of vehicular networks, different vehicles can detect pedestrians, other vehicles, and various objects on the road using sensors and cameras. Whenever a vehicle detects an element (such as a pedestrian or a vehicle), it encodes it using fountain codes to create an information packet. This packet contains the location information of the element (e.g., longitude and latitude), current timestamp, and the color of the vehicle, uniquely identifying the encoded element on the road. Once the phase of gathering information is completed, the process of encoding using fountain codes is initiated. Fountain codes possess a

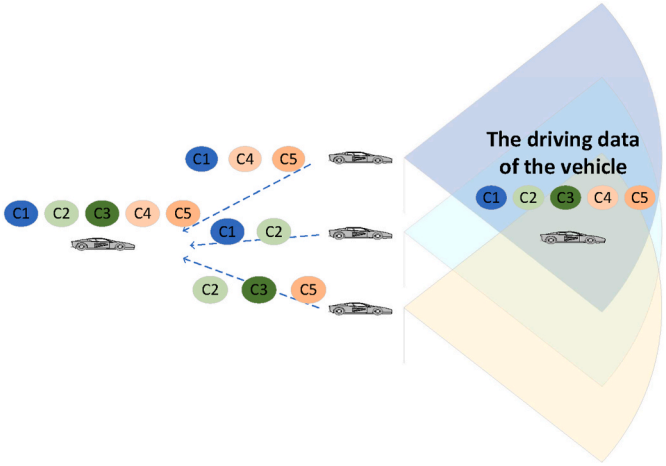
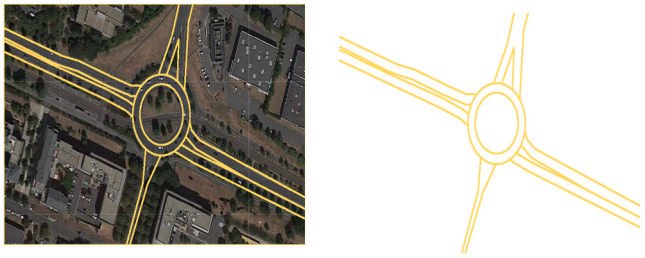



**Fig. 5.** Multiple vehicles have detected the same target.



(a) The yellow lines are the roads segments for simulations (b) Roads abstracted from digital maps

**Fig. 6.** The simulation map.

distinctive characteristic whereby receiving a satisfactory number of encoded data packets enables the retrieval of the original data. If the source file comprises $K$ source symbols, the recipient only requires obtaining $N = (1 + \psi)K$ bits to decode the source data, where $\varphi$ represents a minimal coding overhead. This prevents scenarios where the absence of specific data packets hinders the upload of the source data, making fountain codes especially well-suited for facilitating data sharing in vehicular networks.

The encoding process using LT codes is illustrated in Fig. 4. Firstly, the original task data is divided into several equal parts. Each part is then further divided into $K$ subparts, which are called source data packets. To generate encoded output data packets, an integer $d$ is selected from a given probability mass function. The integer $d$ represents the degree of the output data packet and is referred to as the degree distribution. In a specific system, vehicles negotiate and agree on the value of $d$. In our work, we employ a robust solution distribution:

$$\rho(d) = \begin{cases} \frac{1}{k} & d = 1 \\ \frac{1}{d(d-1)} & d = 2, 3, \ldots, k \end{cases} \tag{19}$$

Throughout the encoding procedure, the size of the source data undergoes expansion, transitioning from bits to $N$ bytes. The source data is partitioned into $K$ equitably-sized sets of data packets, denoted as $S = [S_1, S_2, \ldots, S_k]$, where each packet comprises $L$ bits of data. The encoder

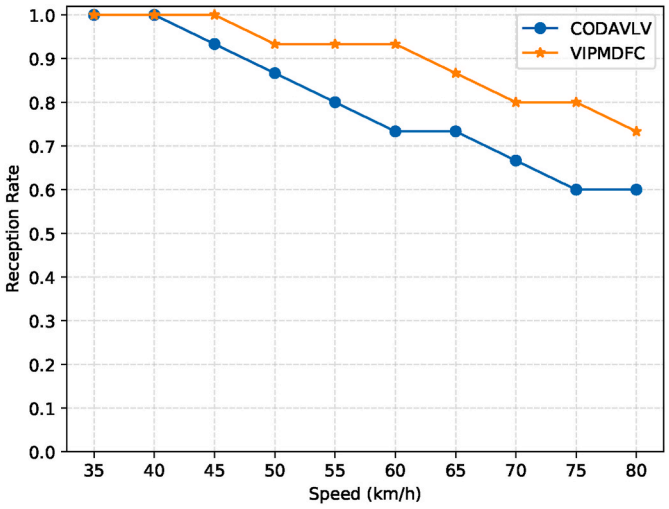



**Fig. 7.** Effect of vehicle speed on the reception rate.

generates a random degree $\rho(d)$, ranging from 1 to $K$, guided by the degree distribution function $\rho(d)$. This degree signifies the quantity of source data packets involved in the encoding process. Subsequently, the encoder randomly selects $\rho(d)$ source data packets and employs XOR operations to produce an encoded data packet $E_i$. This process is reiterated until a satisfactory number of encoded data packets are generated.

In the decoding phase, the decoder implements the backpropagation (BP) decoding algorithm. The decoder first looks for an encoded packet $E_j$ with a degree of 1, which means that the packet $E_j$ corresponds to a single source packet. The source packet $S_j$ can be directly recovered from this encoded packet. Then, the decoder decreases the degree of all encoded packets related to $S_j$ by 1. This process is repeated until the degree of all encoded packets becomes 1, indicating that all the source packets have been recovered.

As shown in Fig. 5, when other vehicles also detect the same element, they will generate encoded packets for that element as well. When the target vehicle receives a sufficient number of these encoded packets from different vehicles, it can use the decoding method of fountain codes to decode all these packets and ultimately identify the targeted element on the road. Through this distributed encoding and decoding mechanism, different vehicles can share their detected environmental information and road objects, leading to more accurate perception. For example, a vehicle may be limited in fully detecting and recognizing a pedestrian due to angle, distance, or occlusion by obstacles. However, by receiving encoded information provided by other vehicles, it can obtain a more comprehensive and accurate judgment.

### 3.4.2. *Packet variance correction*

In practical applications, different vehicles may have variations in sensor configurations and performance, leading to certain errors and biases in the detected position of the same object. This means that although the encoded packets generated by these vehicles correspond to the same object, the included position information (longitude and latitude) may slightly differ. When decoding and identifying objects, the target vehicle needs to consider this reality. It needs to associate these slightly different position information within a certain tolerance range to the same object without mistakenly identifying them as different objects. This requires some noise reduction and correction processing to eliminate the noise introduced by sensor errors. Therefore, after

receiving a certain number of encoded packets, the target vehicle cannot directly use the position information contained in the packets. Instead, it clusters the position information and groups the encoded packets that describe the same element together, as shown in Algorithm 1.

**Algorithm 1.** Data Packet Difference Correction

**Input :** fountain code package *data*, channel congestion ratio *SBR*
**Output:** a collection containing decoded vehicles

```
Initialize feature_data_map and full_message_map as empty maps;
while new CBR value is available do
    for all_info in data do
        package_head ← header information of encode_package;
        if package_head exists in feature_set_map then
            add package_content to feature_set_map [package_head] ;
            elements ← the number of encoded packages;
            if elements is sufficient then
                add elements to full_message_map;
                remove corresponding element from feature_data_map;
            end
        else
            add the new feature to feature_set_map;
        end
    end
    decode elements in full_message_map;
    return full_message_map;
end
```

This helps to effectively mitigate the deviations introduced by individual sensor errors and obtain a more accurate object position. The target vehicle then combines this optimized position information with other details, such as timestamps, to determine whether all these encoded packets belong to the same object. In addition to calculating the optimized position, the target vehicle can also assess whether it belongs to the same encoded element based on the number of packets and the degree of variation in the position information within the packets. If the number of packets is sufficiently large and the positional differences are within a reasonable range, it can be inferred that they correspond to the same object, with the data exhibiting some noise due to sensor errors.

### 3.5. Vehicle networking information transmission optimization

#### 3.5.1. Send message encoding selection

This proposed encoding packet selection method, as described in Algorithm 2, addresses the issue of excessive duplicate information in vehicle perception. Broadcasting the perceived information as is would result in a high volume of redundant information in the vehicle-to-vehicle network, significantly reducing the efficiency of useful information transmission. In order to mitigate this, we employ a coding packet selection approach. Since we broadcast data to surrounding vehicles, each vehicle also receives perception information from other vehicles. If a vehicle receives coding packets that contain information about its own currently perceived target, the transmission frequency is reduced. As we encode the information using distributed fountain codes before transmission, reducing the transmission frequency does not diminish the perception level for the corresponding target.

**Algorithm 2.** Select Send Coded Package

**Output:** coded packets to be sent

```
if perceived_info is a new target then
    add perceived_info to perception_targets;
    encoded_package ← encoder perceived_info;
    return encoded_package;
else
    received_package ← receive package;
    if received_package contains perceived_info then
        send_frequency ← calculate send frequency by perceived_info;
        if send_frequency should send then
            encoded_package ← encoder perceived_info;
            return encoded_package;
        end
    else
        perception_targets remove perceived_info;
    end
end
```

#### 3.5.2. Delivery Timing Based on Mobility Prediction

The continuous broadcasting of perceived information by vehicles would result in a large amount of data in the network, affecting transmission efficiency. To address this issue, we propose the use of mobility prediction instead of uninterrupted data broadcasting, as outlined in Algorithm 3. We also specify that vehicles should send information in whole seconds because allowing information to be sent at any arbitrary time could result in too few coding packets for the receiver to decode using distributed fountain codes.

Vehicles construct a virtual map based on the perceived data and include the perceived vehicle information within it. They then use this information to predict their future states. If the perceived and predicted information align closely, the vehicle does not send this portion of the content until there is a significant deviation between the predicted information and the actual vehicle state. As a safety measure, vehicles are mandated to broadcast information forcefully every 2 s. This time interval may be adjusted when the prediction accuracy decreases. Additionally, when a vehicle enters the communication range, a broadcast of information is sent to accommodate the newly joined vehicle.

**Algorithm 3.** Delivery Timing Based on Mobility Prediction

**Output:** coded packets to be sent
**if** *two second passed* **then**
  **return** *perceived_info*;
**end**
**if** *perceived_info from a new target* **then**
  add *perceived_info* to *perception_targets*;
  update $virtual_map$ by *perception_targets*;
  *spredict_info* ← predict by *perception_targets*;
  **if** *abs(predict_info - perceived_info)* ≥ *threshold* **then**
    *perception_targets* remove *perceived_info*;
  **end**
  *virtual_map* update *perception_targets*;
**end**
**if** *new car detected* **then**
  **return** *perceived_info*;
**end**
*error* ← abs(*predict_info - perceived_info*);
**if** *error* ≥ *threshold* **then**
  **return** *perceived_info*;
**end**

## 4. Experiments

### 4.1. Experimental simulation of data sharing based on distributed fountain code

#### 4.1.1. Simulation setup

We downloaded a road map from OpenStreetMap to simulate a roundabout, as shown in Fig. 6 (a). We extracted the information of the main road, as shown in Fig. 6 (b).

In this section, we simulate the use of C–V2X technology in vehicle-to-vehicle communication. We compare the performance under different vehicle speeds, network packet loss rates, and the number of sensing vehicles and sensed vehicles, using both conventional methods and distributed fountain coding. We assess the effectiveness of our proposed method by evaluating its performance based on the reception rate. The reception rate is defined as follows:

$$r_{recv} = \frac{n_{received}}{n_{all}} \tag{20}$$

Where $r_{recv}$ represents the reception rate, $n_{received}$ is the number of complete vehicle information received by vehicle T, where only the vehicles for which T successfully decodes the entire information are counted, and $n_{all}$ is the total number of vehicles that can be sensed. We compare our proposed method, VIPMDFC, with the communication method used in CODAVLV [31], which serves as the baseline. The comparison is based on the reception rate, which measures the effectiveness of the two methods in transmitting and decoding vehicle information in the C–V2X scenario.

#### 4.1.2. Effect of vehicle speed on the reception rate

In the simulation, we investigated the variation of communication quality over time for different vehicle speeds, as shown in Fig. 7. The experimental results indicate that the communication quality between vehicles with higher speeds and the target vehicle decreases over time. This is because higher vehicle speeds result in surrounding vehicles quickly moving out of the communication range of the target vehicle, affecting the communication effectiveness. It is worth noting that at the beginning of the experiment, some sending vehicles in CODAVLV were far from the target vehicle and immediately left the communication area after sending the information. As a result, subsequent information could not be received in a timely manner, leading to a rapid decrease in the reception rate.

#### 4.1.3. Impact of Different Packet Loss Rates

The experimental results indicate that in the simulation experiments based on vehicular communication networks, VIPMDFC outperforms CODAVLV, as shown in Fig. 8. We considered two different packet loss rates, 10 %, and 20 % [32], and the results show that both methods exhibit the ability to tolerate noise interference. However, during the process of receiving data from the target vehicle, VIPMDFC demonstrates a better improvement in communication robustness. In contrast, distributed fountain code (DFC) technology enhances data reliability by adding redundancy and effectively addresses the issue of neighboring vehicles quickly leaving the communication range of the target vehicle, resulting in a higher reception rate. Therefore, in vehicular communication network applications, VIPMDFC shows significant superiority over CODAVLV and holds promising prospects for wide-ranging applications.

#### 4.1.4. The effect of different vehicle numbers

We conducted simulations with different numbers of sending vehicles: 5 (Fig. 9(a)), 6 (Figs. 9 b), and 7 (Fig. 9 c), with a packet loss rate of 0.2, to analyze the reception rates at different speeds. The results are shown in the figures. It can be observed that as the number of vehicles increases, the reception rate of information significantly improves. However, as the vehicle speed increases and issues such as packet loss during the communication process occur, some vehicles quickly move out of the communication range, resulting in a decrease in reception rate. From the figures, it can be seen that our proposed VIPMDFC approach performs better than CODAVLV. The VIPMDFC method achieves higher reception rates by effectively addressing the challenges associated with increased vehicle speed and packet loss during communication.

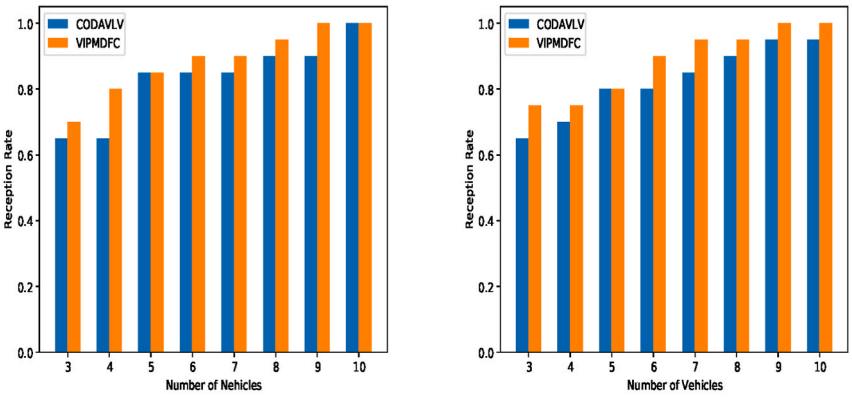
#### 4.1.5. The Effect of Different Perceived Vehicle Numbers

We conducted simulations with a sending vehicle count of 5 and speeds of 70 km/h (Fig. 10 a) and 90 km/h (Fig. 10 b), with a channel packet loss rate of 0.2. The reception rates under different numbers of perceived vehicles were analyzed and are shown in the figures. As the number of perceived vehicles increases, the reception rates of both methods decrease due to the increased amount of information being transmitted by the vehicles. However, in the VIPMDFC approach, the target vehicle receives a portion of the encoded packets from each sending vehicle, resulting in a higher utilization of resources and the ability to decode more information. As the speed increases, the reception rate of CODAVLV decreases, while VIPMDFC shows no significant decline. This is because the distributed fountain code-based approach has higher interference resistance and reception efficiency within a certain speed range.

### 4.2. Information transfer based on mobility prediction

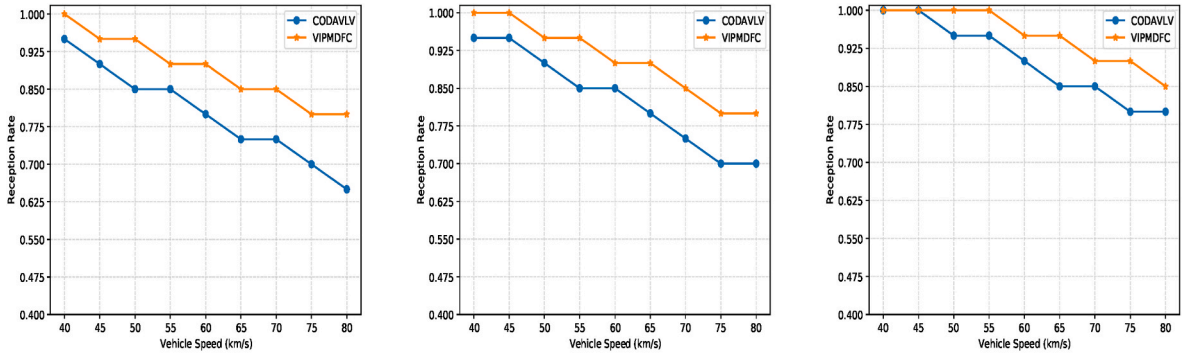
#### 4.2.1. Training setup

In this section, we compare the performance of our proposed model based on the Attention mechanism with linear models and Long Short



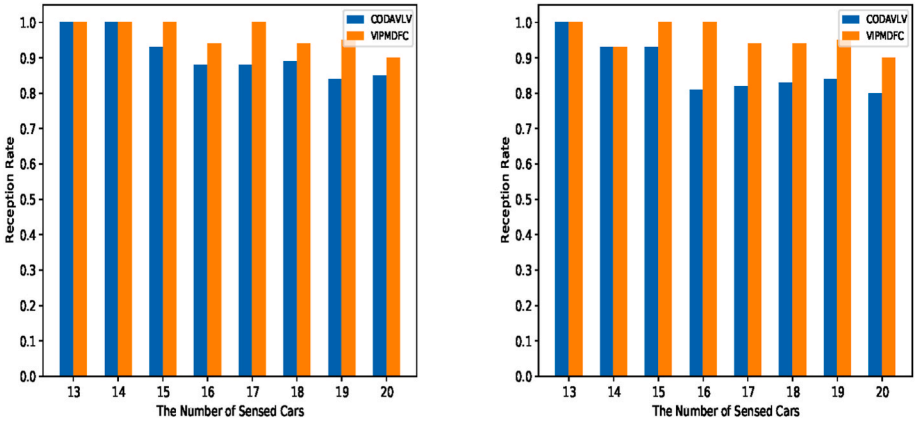

(a) Receiving rate at 10% packet loss rate (b) Receiving rate at 20% packet loss rate

**Fig. 8.** Impact of different packet loss rates.




(a) Five delivery vehicles (b) Six delivery vehicles (c) Seven delivery vehicles

**Fig. 9.** The effect of different vehicle numbers.




(a) Five delivery vehicles (b) Six delivery vehicles

**Fig. 10.** The effect of different perceived vehicle numbers.

**Table 1**
Comparison of prediction accuracy of three algorithms.

| Model | R2 | MSE | RMSE | MAE |
|---|---|---|---|---|
| Linear | 0.813 | 7.139 | 2.672 | 2.133 |
| LSTM | 0.917 | 3.131 | 1.770 | 1.361 |
| Attention + GRU | 0.943 | 2.170 | 1.473 | 0.947 |

Term Memory (LSTM) models. We also analyze the amount of transmitted information between the approach using mobility prediction and the conventional approach.

For the purpose of training our mobility prediction model, we employ a dataset of vehicle mobility that has been gathered by the Departmental Committee of Val de Marne (94) in France [33]. This dataset encompasses various details, including time intervals, vehicle

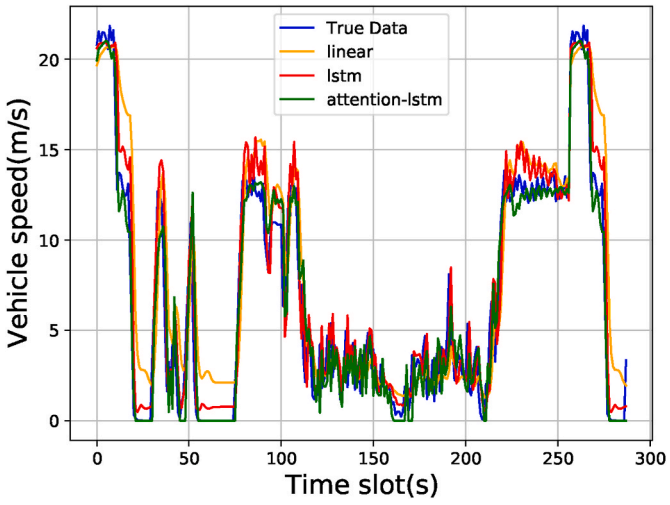



**Fig. 11.** The effect of different perceived vehicle numbers.

IDs, positions, heading angles, vehicle coordinates in a 2D plane (x and y coordinates measured in meters), and vehicle velocities (expressed in meters per second). To establish our training and testing sets, we randomly select 45,000 trajectories for training and 5,000 trajectories for testing. These trajectories are transformed into samples, adhering to a multi-input-multi-output structure. Each sample incorporates the trajectories of ten consecutive time slots, with the objective of predicting the velocities for the subsequent five-time slots.

#### 4.2.2. Indicators and comparison models

To compare with the current representative models, we begin by comparing the predicted velocities with the ground truth velocities. Then, we evaluate the models using the following metrics: coefficient of determination (R2), mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE). We consider the comparison between the following two approaches.

- The fully connected model consists of two fully connected layers. Between these layers, a dropout layer with a dropout rate of 0.2 is included to prevent overfitting. The activation function used in both layers is the sigmoid function.
- The LSTM model consists of two LSTM layers followed by a fully connected output layer. A dropout layer with a dropout rate of 0.2 is added between the second LSTM layer and the output layer to prevent overfitting. The activation function used in the output layer is the sigmoid function.

#### 4.2.3. Prediction results

Fig. 11 displays the real speed and predicted speed for the three algorithms. To ensure clarity, we sampled only 275 data points from the test dataset. It can be observed that the performance of the fully connected model is the poorest, especially at lower speeds. Our proposed model's curve fits the real data better than the LSTM model. In Table 1, we compare the metrics of the three models. It can be seen that our proposed model outperforms the other algorithms in all metrics. This is because the spatiotemporal features of human activities do not improve the judgment of the driver's driving intentions, and the combination of the Attention mechanism and GRU is more conducive to learning on large and complex datasets compared to simply stacking LSTM layers. Our proposed model achieves a prediction accuracy of 94.3 %, which is sufficient to meet the prediction requirements.

#### 4.2.4. Information transfer based on mobility prediction

In this experiment, we conducted simulations with two vehicles (Fig. 12 (a)) and three vehicles (Fig. 12 (b)) respectively. The number of sensed vehicles was set to 5, 6, 7, and 8. We compared the performance of regular periodic broadcasting and information transmission using mobility prediction. The results are shown in the graph. It can be observed that due to the adoption of mobility prediction, the amount of data transmitted when there is little change in the sensed vehicle information can be significantly reduced. Therefore, using mobility prediction instead of continuous data transmission improves the efficiency of information sharing.

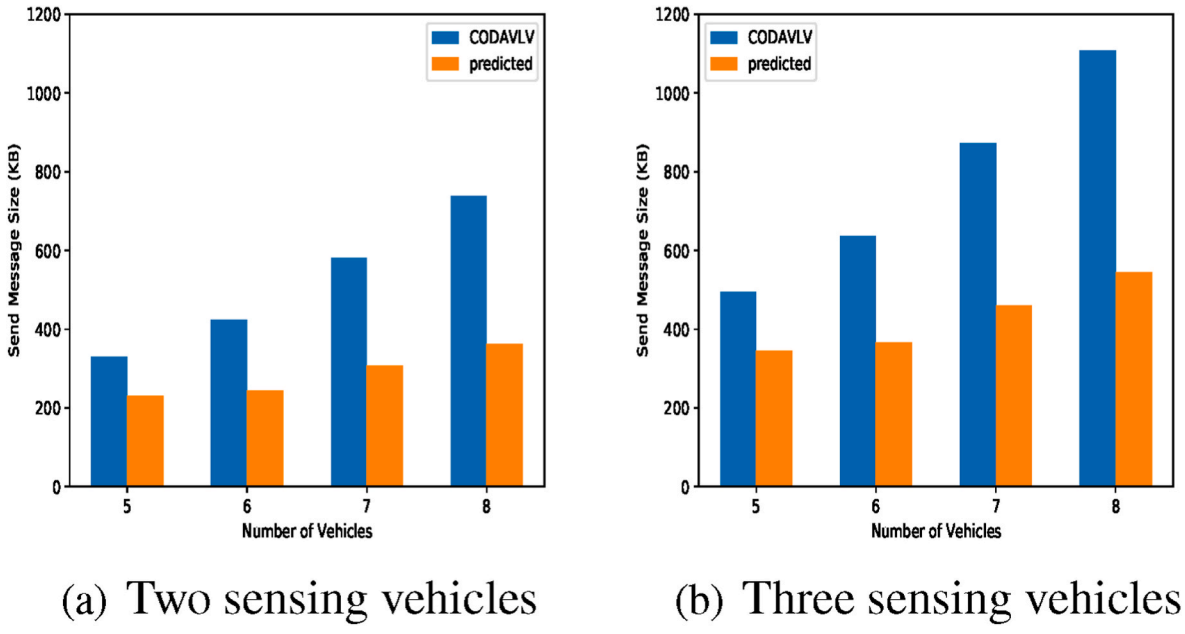



**Fig. 12.** Information transfer based on mobility prediction.

## 5. Conclusions

In this paper, the information-sharing method in collaborative vehicle perception is studied, and an information-sharing method is proposed. We extract the features of vehicle information in the sensing range and propose a distributed fountain code encoding method to reduce the delay and improve the stability of information transmission through multi-vehicle transmission. A vehicle movement prediction network based on an attention mechanism is also designed, and a movement prediction model and information transmission control algorithm are proposed to accurately predict the missing current state information of the vehicle and reduce the amount of data communicated while ensuring the reliability of the data. A large number of simulations were performed to demonstrate that our method is effective compared to other baseline methods. This method significantly reduces the amount of perceptual data transmission and improves the efficiency of information sharing.

## CRediT authorship contribution statement

**Jianhang Liu:** Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – review & editing. **Xinyao Wang:** Conceptualization, Investigation, Methodology, Software, Validation, Writing – original draft. **Haibin Zhai:** Conceptualization, Formal analysis. **Shibao Li:** Resources, Software. **Xuerong Cui:** Methodology, Software. **Qian Zhang:** Resources, Software.

## Declaration of competing interest

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

## References


[1] P. Ghorai, A. Eskandarian, Y.-K. Kim, G. Mehr, State estimation and motion prediction of vehicles and vulnerable road users for cooperative autonomous driving: a survey, IEEE Trans. Intell. Transport. Syst. 23 (2022) 16983–17002.

[2] S. Feng, X. Yan, H. Sun, Y. Feng, H.X. Liu, Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment, Nat. Commun. 12 (2021) 748.

[3] D. Wang, W. Fu, Q. Song, J. Zhou, Potential risk assessment for safe driving of autonomous vehicles under occluded vision, Sci. Rep. 12 (2022) 4981.

[4] V.V. Dixit, S. Chand, D.J. Nair, Autonomous vehicles: disengagements, accidents and reaction times, PLoS One 11 (2016) e0168054.

[5] E. Arnold, O.Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, A. Mouzakitis, A survey on 3d object detection methods for autonomous driving applications, IEEE Trans. Intell. Transport. Syst. 20 (2019) 3782–3795.

[6] D. LaChapelle, T. Humphreys, L. Narula, P. Iannucci, E. Moradi-Pari, Automotive collision risk estimation under cooperative sensing, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 9200–9204.

[7] S. Shi, J. Cui, Z. Jiang, Z. Yan, G. Xing, J. Niu, Z. Ouyang, Vips: real-time perception fusion for infrastructure-assisted autonomous driving, in: Proceedings of the 28th Annual International Conference on Mobile Computing and Networking, 2022, pp. 133–146.

[8] J. Pachat, N.S. Karat, P. Deepthi, B.S. Rajan, Index coding in vehicle to vehicle communication, IEEE Trans. Veh. Technol. 69 (2020) 11926–11936.

[9] F. Raviglione, S. Zocca, A. Minetto, M. Malinverno, C. Casetti, C.F. Chiasserini, F. Dovis, From collaborative awareness to collaborative information enhancement in vehicular networks, Vehicular Communications 36 (2022) 100497.

[10] T. Higuchi, M. Giordani, A. Zanella, M. Zorzi, O. Altintas, Value-anticipating v2v communications for cooperative perception, in: 2019 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2019, pp. 1947–1952.

[11] H. Xu, X. Liu, Perception synergy optimization with deep reinforcement learning for cooperative perception in c-v2v scenarios, Vehicular Communications 38 (2022) 100536.

[12] S. Aoki, T. Higuchi, O. Altintas, Cooperative perception with deep reinforcement learning for connected vehicles, in: 2020 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2020, pp. 328–334.

[13] B. Dai, F. Xu, Y. Cao, Y. Xu, Hybrid sensing data fusion of cooperative perception for autonomous driving with augmented vehicular reality, IEEE Syst. J. 15 (2020) 1413–1422.

[14] Y. Zhang, Z. Huang, C. Zhang, C. Lv, C. Deng, D. Hao, J. Chen, H. Ran, Improved short-term speed prediction using spatiotemporal-vision-based deep neural network for intelligent fuel cell vehicles, IEEE Trans. Ind. Inf. 17 (2020) 6004–6013.

[15] S. Yang, W. Wang, J. Xi, Leveraging human driving preferences to predict vehicle speed, IEEE Trans. Intell. Transport. Syst. 23 (2021) 11137–11147.

[16] M.R. Amini, Q. Hu, A. Wiese, I. Kolmanovsky, J.B. Seeds, J. Sun, A Data-Driven Spatio-Temporal Speed Prediction Framework for Energy Management of Connected Vehicles, IEEE Transactions on Intelligent Transportation Systems, 2022.

[17] H. Yang, C. Liu, M. Zhu, X. Ban, Y. Wang, How fast you will drive? predicting speed of customized paths by deep neural network, IEEE Trans. Intell. Transport. Syst. 23 (2021) 2045–2055.

[18] L. Hou, S.E. Li, B. Yang, Z. Wang, K. Nakano, Structural transformer improves speed-accuracy trade-off in interactive trajectory prediction of multiple surrounding vehicles, IEEE Trans. Intell. Transport. Syst. 23 (2022) 24778–24790.

[19] N.K. Sharma, A. Hamednia, N. Murgovski, E.R. Gelso, J. Sjöberg, Optimal eco-driving of a heavy-duty vehicle behind a leading heavy-duty vehicle, IEEE Trans. Intell. Transport. Syst. 22 (2020) 7792–7803.

[20] G. Thandavarayan, M. Sepulcre, J. Gozalvez, Redundancy mitigation in cooperative perception for connected and automated vehicles, in: 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), IEEE, 2020, pp. 1–5.

[21] Y. Lian, L. Qian, L. Ding, F. Yang, Y. Guan, Semantic fusion infrastructure for unmanned vehicle system based on cooperative 5g mec, in: 2020 IEEE/CIC International Conference on Communications in China (ICCC), IEEE, 2020, pp. 202–207.

[22] S. Faiz, N. Achir, K. Boussetta, Increasing vehicles perception using cooperative relaying and priority-based beaconing, in: 2021 IEEE 22nd International Conference on High Performance Switching and Routing (HPSR), IEEE, 2021, pp. 1–6.

[23] G. Thandavarayan, M. Sepulcre, J. Gozalvez, Generation of cooperative perception messages for connected and automated vehicles, IEEE Trans. Veh. Technol. 69 (2020) 16336–16341.

[24] Y. Zhu, C. Zhao, Y. Du, Latency impact analysis of point cloud fusion modes for cooperative perception in autonomous driving, in: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2022, pp. 1024–1029.

[25] P. Zhou, P. Kortoçi, Y.-P. Yau, B. Finley, X. Wang, T. Braud, L.-H. Lee, S. Tarkoma, J. Kangasharju, P. Hui, Aicp: augmented informative cooperative perception, IEEE Trans. Intell. Transport. Syst. 23 (2022) 22505–22518.

[26] L. Cheng, B.E. Henty, D.D. Stancil, F. Bai, P. Mudalige, Mobile vehicle-to-vehicle narrow-band channel measurement and characterization of the 5.9 ghz dedicated short range communication (dsrc) frequency band, IEEE J. Sel. Area. Commun. 25 (2007) 1501–1516.

[27] T. Abbas, K. Sjöberg, J. Karedal, F. Tufvesson, et al., A measurement based shadow fading model for vehicle-to-vehicle network simulations, Int. J. Antenn. Propag. (2015) 2015.

[28] M. Noor-A-Rahim, G.M.N. Ali, H. Nguyen, Y.L. Guan, Performance analysis of ieee 802.11 p safety message broadcast with and without relaying at road intersection, IEEE Access 6 (2018) 23786–23799.

[29] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 327 (2010) 1018–1021.

[30] W. Liu, Y. Shoji, Deepvm: rnn-based vehicle mobility prediction to support intelligent vehicle applications, IEEE Trans. Ind. Inf. 16 (2019) 3997–4006.

[31] J. Wang, Y. Zeng, Y. Gong, Collaborative 3d Object Detection for Autonomous Vehicles via Learnable Communications, IEEE Transactions on Intelligent Transportation Systems, 2023.

[32] A. Rayamajhi, A. Yoseph, A. Balse, Z. Huang, E.M. Leslie, V. Fessmann, Preliminary performance baseline testing for dedicated short-range communication (dsrc) and cellular vehicle-to-everything (c-v2x), in: 2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall), IEEE, 2020, pp. 1–5.

[33] M.-A. Lèbre, F. Le Mouël, E. Ménard, On the importance of real data for microscopic urban vehicular mobility trace, in: 2015 14th International Conference on ITS Telecommunications (ITST), IEEE, 2015, pp. 22–26.