A Mobility Prediction-Based Relay Cluster Strategy for Content Delivery in Urban Vehicular Networks

Shaoqi Yue 1,2 and Qi Zhu 1,2,*

1 Jiangsu Key Laboratory of Wireless Communications, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; 1019010230@njupt.edu.cn
2 Engineering Research Center of Health Service System Based on Ubiquitous Wireless Networks, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
* Correspondence: zhuqi@njupt.edu.cn; Tel.: +86-136-0518-9346

Abstract: In recent years, cache-enabled vehicles have been introduced to improve the efficiency of content delivery in vehicular networks. However, because of the high dynamic of network topology, it is a big challenge to increase the success probability of content delivery. In this paper, we propose a relay strategy based on cluster’s prediction trajectory for the situation of no cache near the request vehicles. In our strategy, the roadside unit (RSU) divides vehicles into clusters by their prediction trajectory, and then proactively caches contents at a cluster that will be about to meet the request vehicle. In order to decrease the probability of unsuccessful content delivery caused by communication duration that is too short between the request vehicle and content source vehicle, RSU caches content chunks at multiple vehicles in a cluster. By letting the request vehicle communicate with vehicle-caching content chunks one by one, our strategy enlarges the communication duration and increases the success probability. Our strategy also maximizes the success probability by optimizing the number of vehicles selected to cache content chunks. Besides, based on statistical characteristics of vehicles’ speed, we derive the formula of success probability of content delivery. The simulation results show that our strategy can increase the success probability of content delivery, as well as decrease time delay, for example. For example, we increase the success probability by about 20%. Since the trajectory prediction-based cluster-dividing mechanism can improve clusters’ stability at intersections, this method is well suited for urban road scenarios.

Keywords: vehicular networks; relay; content chunks; content delivery; prediction trajectory; cluster

1. Introduction

With the dramatic development of intelligent transportation systems (ITS), various vehicular applications, including road safety, intelligent transportation, in-vehicle entertainment, and self-driving [1], occur in our daily lives. The smart road can supply vehicles with autonomous drive, but it needs a large quantity of data transmitted between road facilities and vehicles [2,3], which will absolutely increase the traffic of vehicular network. The increasing number of vehicles on the road sharply increases the traffic burden of vehicular networks, and the highly dynamic vehicular network makes vehicular communication links unstable. Consequently, the quality-of-experience of vehicle users is poor. In order to solve these problems, researchers and scholars have introduced in-network caching into the vehicular network.

In-network caching is a mechanism to cache contents in every node of the vehicular network, and vehicle members can obtain desired contents from neighbor nodes. However, caching increases the redundancy of contents cached in networks, which can shorten the communication links between content source nodes and request vehicles and decrease traffic burden of content servers and network. The authors of [4] introduced the Leave Copy Everywhere (LCE) mechanism, which lets nodes cache every passing content by itself. The authors of [5] introduce the probability mechanism, which lets nodes cache contents...
passing by itself with fixed value of probability. Non-Cooperative caching mechanisms, characterized by their relatively high redundancy of contents, waste the caching space of the network. Meanwhile, for cooperative caching mechanisms, nodes in the network can cooperate with each other, which can effectively reduce content redundancy. The Leave Copy Down (LCD) mechanism, described by the authors of [6], caches content in the next node after the source node along the transmission path. The authors of [7] demonstrated the Probcache mechanism, which caches content with probability values derived from some characteristics of content and the caching node. However, the mechanisms from researches discussed above fail to consider the factor of vehicular trajectory and ignore the frequent communication interruption problem caused by rapid vehicle movement.

Aiming at weakening the influence of vehicle movement characteristics, some researchers and scholars have chosen to predict future trajectory of vehicles and cache contents at suitable nodes to improve the system’s performance. The strategy described by the authors of [8] comprehensively considers the characteristics of the request vehicle, request content, and roadside units (RSUs), and caches the specific content chunks at suitable RSUs. The authors of [9] analyzed the request information and movement characteristics, constructed an optimization to minimize the time delay as well as the caching cost, and finally determined which contents the RSU should cache. The strategy described by the authors of [10] makes use of the neural network to predict the vehicles’ future trajectory, and caches the request content at the RSUs on the future trajectory of the vehicle. By analyzing vehicle’s historical trajectory, the authors of [11] used a prediction based on the Partial Matching (PPM) method to predict the vehicle’s probability of reaching different hotspot regions and to obtain the future trajectory of vehicles. Then, the prediction was used to determine whether to cache content at the vehicle. Using a dynamic Markov chain model to predict vehicle’s mobility, the authors of [12] cached corresponding contents before the vehicle generated its request and optimized the content retrieval latency. By modeling the movement of mobility clients as a second-order Markov chain, the authors of [13] established a network transition probability table to predict mobility clients’ trajectories and caches desired content at an access point (AP) on the prediction trajectories to increase the caching hit ratio of the network. Using deep learning to learn vehicles’ behavior, the authors of [14] introduced an effective proactive caching strategy to minimize the time delay. However, scholars from abovementioned research have only considered the movement characteristics of request nodes and have failed to take into account the relationship between request nodes and caching nodes, such as relative motion.

Vehicular network is characterized by high dynamic topology, which can be effectively reduced by dividing vehicles into clusters because vehicles staying at the same cluster have similar attributes, such as speed, position, and so on, and communication links among vehicles are more stable. The Lowest-ID (Identity) mechanism, depicted by the authors of [15], allocate sole IDs to every vehicle in the network and select the vehicle with the minimum ID value as the cluster head (CH). All vehicles in the coverage area of the CH are then considered cluster members (CM) of the CH. The authors of [16] demonstrated the Mobility Based on Clustering (MOBIC) mechanism, which selects the vehicle with minimum value of motion discrepancy between vehicles in the coverage area of itself as the CH, and all vehicles in the coverage area of the CH are CMs of the CH. The authors of [17] introduced the weighted clustering algorithm (WCA), which analyzes influences from motion, density, and energy, and calculates a weighted result as the metric to select the CH. Besides, cooperation between clusters and RSUs can lower the amount of redundant content, as well as increase the utilization ratio of caching space. The mechanisms described by the authors of [18,19] let RSUs cache contents with high popularity at CHs to fit vehicles from the present cluster or surrounding clusters. With the mechanism described by the authors of [20], vehicles determine whether to cache some specific contents by considering contents cached in themselves and clusters in which they stayed, which decreases the amount of redundant content. The authors of [21] let leaving vehicles transfer contents to vehicles entering the fixed region, whereby contents with high popularity can be cached at
the area. Although the mechanisms from the abovementioned research can weaken the influence of high dynamic topology, the researchers have only focused on past or current positions and have failed to take into account the vehicles’ future trajectory, so the clusters have poor stability.

To sum up, the existing cluster-dividing mechanisms fail to well accommodate the high dynamic topology, especially in situations of urban areas, and vehicle clusters can easily disintegrate at intersections. So, we take into account vehicles’ future trajectories to enlarge the stability of clusters. In order to increase the success probability of content delivery, this paper proactively caches request contents at clusters which will meet the request vehicle on its future trajectory. This paper also adopts the multiuser multichannel transmission mode in the communication process.

The main contributions of our paper are summarized as follows:

• We propose a proactive caching strategy based on the cluster’s prediction trajectory, which divides vehicles into clusters according to the vehicles’ prediction trajectory. RSUs proactively cache content chunks at a cluster that will meet the request vehicle on the prediction trajectory. By letting vehicles receive content chunks from vehicles on the opposite road one by one, we enlarge the communication duration between the request vehicle and content source vehicle and increase the success probability of content delivery. Besides, in order to increase the success probability of content delivery between the request cluster and content source cluster, we introduce the multiuser multichannel transmission mode into the communication process between clusters.

• Aiming at increasing stability of clusters, we treat vehicles’ prediction trajectory as one of considerations of cluster division. By giving CMs the same prediction trajectory, we can obtain the cluster’s prediction trajectory, which is the same as that of the vehicles’ in the cluster.

• Based on the prediction trajectory of the request vehicle and clusters, as well as the vehicle’s speed, the RSU proactively caches content chunks at multiple vehicles in the corresponding cluster. During the process, the RSU formulates the optimal number of content chunks to maximize the success probability of content delivery.

• Based on the statistical characteristics of vehicle’s speed, vehicle flow, and the number of request arrivals in RSU, we theoretically derive the success probability of content delivery. The simulation results verify the validity of the process of derivation and demonstrate that this paper improve the system’s performance in terms of time delay, as well as the success probability of content delivery. In comparison with the results achieved by the authors of [19], we increase success probability by about 20%.

The rest of this paper is organized as follows. The system model is presented in Section 2. In Section 3, we depict the content request and delivery process, introduce the cluster dividing algorithm, optimize the caching strategy, and derive a formula for the success probability of content delivery. Section 4 shows the simulation results and illustrates the performance of our proposed strategy. Section 5 summarizes the paper.

2. System Model

In this section, as shown in Figure 1, we consider a bidirectional two-lane urban area which has 2I lanes that have the same size with length E and width D. I and i + 1, i ∈ [I, I] denote the two lanes of the dual carriageway, respectively. The traffic flow is different in different lanes, and we assume the vehicle number of an area of lane i follows a Poisson Point Process, with a rate of λ_i per unit area [22]. For example, K denotes the vehicle number of lane i and the probability distribution function of K:

\[ P_i(K = k) = \frac{(\lambda_iDE)^k}{k!}e^{-\lambda_iDE} \]  (1)
Especially, the two lanes of a dual carriageway have the same distribution parameter, and there holds \( \lambda_i = \lambda_{i+1}, i \in [1, I] \).

Assuming that the speed of vehicles on different streets is different, and the speed of vehicles in the same lanes follows the same Truncated Normal Distribution [23], the probability density function of speed, defined by \( s_i \) of vehicles on lane \( i \), is given as follows:

\[
f(s_i) = \begin{cases} 
\frac{e^{-\frac{(s_i - \mu_i)^2}{2\sigma_i^2}}}{\sqrt{2\pi\sigma_i}}, & s_{\min} \leq s_i \leq s_{\max} \\
0, & \text{else}
\end{cases}
\]

where \( \mu_i \) and \( \sigma_i \) denote the mean and standard deviation of \( s_i \), and parameter \( c_i \) ensures the accumulated probability of \( f(s_i) \) equals 1 in the range \([s_{\min}, s_{\max}]\), which can be used to determine the value of \( c_i \), especially, \( s_i \) and \( s_{i+1} \) have the same probability density function.

Because the speed of any two vehicles is independent of each other, the joint probability density function of speed \( s_i^x \) and speed \( s_i^y \) of two vehicles in lane \( i \) is derived as follows:

\[
f(s_i^x, s_i^y) = f(s_i^x) \cdot f(s_i^y) \\
= \begin{cases} 
\frac{e^{-\frac{(s_i^x - \mu_i)^2}{2\sigma_i^2}}}{\sqrt{2\pi\sigma_i}}, & s_{\min} \leq s_i^x \leq s_{\max}, s_{\min} \leq s_i^y \leq s_{\max} \\
0, & \text{else}
\end{cases}
\]

The distribution function of speed \( s_i \) can be derived from Formula (2) as follows:

\[
F(s_i) = \begin{cases} 
1 & s_i > s_{\max} \\
\int_{s_{\min}}^{s_i} f(t)dt & s_{\min} \leq s_i \leq s_{\max} \\
0 & s_i < s_{\min}
\end{cases}
\]

Each intersection in our system has an RSU. RSUs connect with the internet, and RSUs are linked to their neighboring counterparts, whereby vehicles can obtain all desired content. Assuming that \( M \) RSUs exist in this area, RSU is denoted by \( R_m \), and the subscript \( m \) denotes the sequence number of \( R_m \), all RSUs have the same communication range with \( R^{rsu} \), as well as the same maximum transmission rate with \( R^{max} \), and the number of...
request arrival at $R_m$, $m \in [1, M]$ obeys the distribution of Poisson Process with parameter $\lambda_{m,rsu}$. Vehicles in our system are denoted by $V_x$, where the subscript $x$ denotes the sequence number of $V_x$, and all vehicles have the same transmission range with $r^v$.

The authors of [24] indicated that the request for content approximately follows a Zipf distribution. Accordingly, we assume that the amount of content in the content library is $L$, and every content has the same size of $W$. Vehicles’ requests for content in the library obeys the Zipf distribution, and content with higher popularity has a higher probability of being requested. The probability of requesting the content with popularity ranked at $\tau$ is given as follows:

$$f_{\tau} = \frac{\tau^{-\gamma}}{L \sum_{j=1}^{L} j^{-\gamma}}, \tau \in [1, L]$$

where $\gamma$ means the parameter of Zipf distribution. As the value of $\gamma$ increases, the vehicles will more frequently request content with high popularity. $\tau$ denotes the popularity rank of content in the library.

Vehicles in our system have limited caching capacity, which is divided into two parts: Caching space and relay space. The former is used to cache the content which is of high popularity or is attractive to the vehicle user, and it can accommodate content with the maximum number of $N (N < L)$. The latter is used to temporarily cache content chunks that will be delivered to the request vehicle when the vehicle meets the request vehicle. The vehicles selected to temporarily cache content are called as the relay vehicles. Because the relay space only caches part of the content, the amount of content that the relay space can cache was set to 2. The common symbols are showed in Table 1.

| Symbol | Description |
|--------|-------------|
| $I$ | The number of street |
| $D$ | The width of lane |
| $E$ | The length of lane |
| $\lambda_i$ | The parameter of Poisson Point Process for the number of vehicles on lane $i$ |
| $M$ | The number of the roadside units |
| $R_m$ | The roadside unit with sequence number of $m$ |
| $R_{su}^m$ | The coverage range of roadside unit |
| $R_{max}$ | The maximum transmission rate of roadside unit |
| $R^a$ | The average travel distance of vehicle under the coverage range of roadside unit |
| $\lambda_{m,rsu}$ | The parameter of Poisson Process for the request number arriving at $R_m$ |
| $L$ | The number of content in content library |
| $\gamma$ | The parameter of Zipf distribution |
| $W$ | The size of content |
| $V_x$ | The vehicle with sequence number of $x$ |
| $H_x$ | The number of vehicles in the same lane as $V_x$ under the coverage range of $V_x$ |
| $N$ | The maximum number of content that vehicle’s caching capacity can accommodate |
| $r^v$ | The coverage range of vehicle |
| $C_{m,i}$ | The cluster with vehicles have next prediction lane $i$ under the coverage range of $R_m$ |
| $n_{m,i}$ | The number of vehicle in $C_{m,i}$ |

### 3. Content Acquisition Process and Caching Algorithm Optimization

#### 3.1. Content Acquisition Process

In vehicular networks, vehicles can obtain desired content from neighboring vehicles as well as RSUs, and the content acquisition process is illustrated in Figure 2. In our strategy, as shown in Figure 3, there are five cases of content acquisition. In case 3 and case 4, the request vehicles obtain desired content from other vehicles on opposite lanes. Request vehicles and content source vehicles drive in opposite directions, which can easily lead to unsuccessful content transmission due to communication duration times being too short. For example, assuming that the speed of vehicles is 50 km/h and the coverage range of
vehicle is 50 m, the communication duration time will be around 7.2 s. If the transmission rate is 1 Mbit/s, then the transmission volume of content is no more than 7.2 Mbit. However, the volume of desired content of vehicle users always exceeds 7.2 Mbit, which means that the situation of unsuccessful transmission occurs easily. So, our strategy caches content chunks using multiple vehicles, which shortens the volume of content transferred from the content source vehicle to the request vehicle and increases the probability of successful content transmission. The process of content acquisition for vehicle users is as follows.

Figure 2. The flowchart of content acquisition.
The request vehicle will first searches for desired content among vehicles that are a one-hop distance from itself in the same lane. If the desired content exists, then the content source vehicle will directly transfer content to the request vehicle. As shown in Figure 3, \( V_2 \) caches the content that \( V_1 \) requests, and \( V_2 \) directly transfers the content as it receives the request from \( V_1 \).

If the request vehicle fails to find desired content among vehicles that are a one-hop distance from itself in the same lane, then the request will be delivered to vehicles that are a two-hop distance from the request vehicle in the same lane. If the desired content exists, then the content source vehicle will transfer the content to the request vehicle that is a two-hop distance away, meaning that the content source vehicle will first transfer the content to a relay vehicle, and then the relay vehicle will transfer the content to the request vehicle. As shown in Figure 3, \( V_5 \), which is a two-hop distance from \( V_3 \), caches the content \( V_3 \) desires. \( V_5 \) first transfers the content to the relay vehicle \( V_4 \), and then \( V_4 \) transfers the content to \( V_3 \).

If the request vehicle fails to find desired content among vehicles that are a one-hop or two-hop distance from itself in the same lane, then the request will be forward-delivered to vehicles that are a one-hop or two-hop distance from the request vehicle in the opposite lane. If the desired content exists, then the request vehicle and content source vehicle will separately act as CHs and make vehicles that are a one-hop distance from themselves in the same lane become CMs to construct the request cluster and the content source cluster. On this basis, according to caching optimization algorithm depicted in Section 3.2, the content source vehicle will averagely divide content into multiple chunks and cache them at multiple relay vehicles in content source cluster, and those vehicles will transfer content chunks to vehicles in request cluster using the multiuser multichannel transmission mode. As shown in Figure 3, \( V_{11} \) caches the content \( V_6 \) requests. Then, \( V_{10}, V_{11}, V_{12}, \) and \( V_{13} \) constitute the content source cluster, and \( V_6, V_7, V_8, \) and \( V_9 \) constitute the request cluster after the request arrives at \( V_{11} \). According to caching optimization algorithm, \( V_{11} \) divides the content into three chunks and separately caches them at \( V_{11}, V_{12}, \) and \( V_{13} \). During the transmission process, \( V_{11}, V_{12}, \) and \( V_{13} \) separately transfer content chunks to \( V_6, V_7, \) and \( V_8, \) and then \( V_6 \) receives the remaining content chunks from \( V_7 \) and \( V_8 \).

If the request vehicle fails to find the desired content among vehicles that are a one-hop or two-hop distance away from the request vehicle in the same lane or in the front opposite lane, then the request will be delivered to an RSU. When the request vehicle is situated in the coverage area of the RSU, the request vehicle directly obtains desired content.
content from the RSU. As shown in Figure 3, the request of $V_{14}$ is delivered to $R_4$, and then $R_4$ directly transfers content to $V_{14}$.

5. When the request vehicle is situated out of coverage area of the RSU, the RSU ahead of the request vehicle divides vehicles in its coverage area into clusters by their prediction trajectory. According to caching optimization algorithm depicted in Section 3.2, the RSU divides the content into multiple chunks averagely and caches them at multiple relay vehicles in a cluster that are about to meet request vehicle. As shown in Figure 3, $R_2$ delivers the request of $V_{15}$ to $R_3$. Vehicles going to drive in the opposite lane of $V_{15}$ constitute the caching cluster, and $R_3$ chooses $V_{16}$, $V_{17}$, and $V_{18}$ from the cluster to cache content chunks. During communication process, $V_{15}$ receives content chunks from $V_{16}$, $V_{17}$, and $V_{18}$ one by one.

3.2. Caching Optimization Algorithm

3.2.1. Cluster-Dividing Mechanism

RSUs not only provide services for vehicles in the coverage area of themselves, but also proactively cache content at clusters going to meet request vehicles based on the requests forwarded by neighboring RSUs. Because the content transmission process between the clusters and request vehicles happens on the prediction trajectory of clusters, our mechanism makes use of neural network to predict the vehicles’ behavior among turning left, going straight, turning right, and turning around at intersections. According to the prediction results, vehicles in the coverage area of RSUs were divided into different clusters.

Our mechanism makes use of a three-layer neural network to predict the vehicles’ behavior, and the structure of the used neural network is shown in Figure 4. The neural network has a SoftMax layer behind the output layer, as well as a sigmoid activation function in the hidden layer. Some attribute information of the vehicle can affect the driver’s driving behavior, for example, departure places and destinations can be used by drivers to select the most convenient path, and the drivers will take different choices at different times and different positions. Therefore, the input parameters of the neural network are defined as $x_1$, denoting departure place; $x_2$, denoting the destination; $x_3$, denoting the current position; and $x_4$, denoting the current time. A four-length vector can be input into the input layer of the used neural network. The hidden layer has two implicit layers with node numbers of 17 and 21. The output layer also a four-length vector, in which $y_1$, $y_2$, $y_3$, and $y_4$ denote the probability of turning left, going straight, turning right, and turning around. In the paper, 9149 sets of driving data were obtained from the urban road environment simulated by SUMO (Simulation of Urban Mobility) [25], and the proportion of training data and test data was 8:2. In order to depict prediction results in terms of probability, we added a SoftMax layer behind the hidden layer to normalize the output of hidden layer. The SoftMax function [26] is expressed as follows:

$$
softmax(x_d) = \frac{e^{x_d}}{\sum_{j} e^{x_j}}
$$

Figure 4. The structure of the neural network.
The SoftMax function can map the output of hidden layer to the range of \((0, 1)\), by which the neural network can predict the probability of turning left, going straight, turning right, and turning around. The neural network will then treat the result with the maximum probability as the behavior of vehicles.

The neural network considers the cross-entropy between the behavior distribution of the training output and the historical actual data as the objective function. By minimizing the value of objective function, our mechanism obtains network parameters and completes the training process. The objective function is formulated as follows:

\[
J(\theta) = -\frac{1}{N_s} \left[ \sum_{g=1}^{N_s} \sum_{j=1}^{4} Y_{g,j} \log(\text{softmax}(y_{g,j}(\theta))) \right]
\]  

(7)

where \(N_s\) denotes the sample number of training, \([Y_{g,1}, Y_{g,2}, Y_{g,3}, Y_{g,4}]\) denotes the actual result of the g-th sample, and \([y_{g,1}(\theta), y_{g,2}(\theta), y_{g,3}(\theta), y_{g,4}(\theta)]\) denotes the training output of the g-th sample.

The training process of the neural network uses the gradient descent to find the set \(\theta\) to minimize \(J(\theta)\), and the performance of the neural network can be illustrated by the prediction accuracy, which can be derived by dividing the number of correct predictions to the volume of test data. Our paper compares the prediction accuracy when the activation function is the sigmoid function and tanh function. The relationship between the prediction accuracy and training times is shown in Figure 5, which shows that prediction accuracy increased with the increase of training times, and that the prediction accuracy reached nearly 1 after 30 rounds training. Figure 5 also shows that the effect of the sigmoid function was better than that of the tanh function, so we chose the sigmoid function as the activation function. The neural network can predict the behavior of vehicles with high accuracy after finishing training. The cluster-dividing algorithm based on the vehicles’ prediction trajectory is discussed next.

![Figure 5. Prediction accuracy of the neural network.](image-url)

Based on prediction trajectory of vehicles in the coverage area of the RSUs, the RSUs divide vehicles into different clusters, and the CMs in a cluster must have the same next prediction lanes. For example, as shown in Figure 6, the next lanes of vehicles in the coverage area of \(R_m\) may be lane 1, lane 2, lane 3, and lane 4, and the clusters can be denoted by \(C_{m,i}(i = 1, 2, 3, 4)\). \(C_{m,i}^{\text{num}}\) denotes the number of CMs in cluster \(C_{m,i}\), and \(C_{m,1}^{\text{num}}\) denotes the number of CMs in cluster \(C_{m,1}\). If \(C_{m,1}^{\text{num}} = 4\), the cluster dividing algorithm is shown as Algorithm 1.
Algorithm 1 Cluster Dividing Algorithm Based on Mobility Prediction

1. Initialization: all vehicles in coverage area of R̅m constitute the set SETm, all vehicles with next prediction lane i constitute the cluster Cm,i = ∅
2. For any Vx ∈ SETm
3. input departure place x1, destination x2, current lane x3 and current time x4 of Vx into neural network, and obtain output [softmax(y1), softmax(y2), softmax(y3), softmax(y4)]
4. and the next prediction lane of Vx
5. if the next prediction lane of Vx is lane i
6. make Vx become a CM of Cm,i
7. End if
8. End For

3.2.2. Caching optimization for number k of content chunks

In our strategy, case ③ and case ④ both cache content at relay vehicles. The probability that the relay vehicles can meet the request vehicle is 1 in case ③, which can be regarded as the particular case of case ④ with a prediction probability of 1. So, our paper only discusses the caching optimization of case ④, whereby RSUs averagely divide content into k chunks and cache them at k vehicles. If the value of k increases, then the size of content chunks cached at a vehicle decreases, and the success probability of content chunks transmission increases. However, the probability of meeting between the request vehicle and all of those k vehicles decreases. On the contrary, if the value of k decreases, the probability of meeting between the request vehicle and all of those k vehicles increases, but the size of content chunks cached at a vehicle increases and the success probability of content chunks transmission decreases. So, our strategy optimizes the number k of content chunks to maximum the success probability of content transmission from k vehicles in a cluster to the request vehicle.

Assuming that Rm needs to cache content chunks at multiple vehicles in cluster Cm,i+k for vehicle Vx on lane i, Rm chooses vehicles with the top k prediction probability to cache k content chunks. Every content chunk has the same size of Wk, and the prediction probability of those k vehicles constitutes the set {p̅pre,i, p̅pre,i+1, ..., p̅pre,i+k}. Assuming that the speed of two vehicles on the opposite lanes are sx,i and sx,i+1 and the average transmission rate is Vd during the process of the two vehicles meeting, the condition of success content chunk transmission is 2sx,iVd ≥ Wk and the feasible region is G0 = \{(sx,i, sx,i+1)|sx,i + sx,i+1 ≤ 2sx,iVdWk\}. 

Figure 6. Intersection model.
During the meeting between $C_{m,j+1}$ and the request vehicle, the probability that all of those $k$ vehicles successfully transfer content chunks to the request vehicle is as follows:

$$p_{i+1}^{\text{trans}}(k) = \left[ \int_{G_0} \int f(s_i^x, s_{i+1}^y) \, ds \right]^k \quad (8)$$

The success probability of content transmission from cluster $C_{m,j+1}$ to request the vehicle is as follows:

$$p_{i+1}^{\text{succ}}(k) = p_{i+1}^{\text{trans}}(k) \cdot \prod_{j=1}^{k} p_{i+1,j}^{\text{pre}} \quad (9)$$

Our strategy maximizes the success probability of content transmission by optimizing the number of content chunks, and the optimization is formulated as follows:

$$\max_k p_{i+1}^{\text{succ}}(k) = p_{i+1}^{\text{trans}}(k) \cdot \prod_{j=1}^{k} p_{i+1,j}^{\text{pre}} \quad \text{s.t.} \quad 1 \leq k \leq \min \{ NV, C_{i+1}^{\text{num}} \} \quad (10)$$

where, in case $\bigcirc$ $NV = H_x$, $H_x$ denotes the number of vehicles that are a one-hop distance away from $V_x$, and in case $\bigcirc$, $NV = C_{i+1}^{\text{num}}$, $C_{i+1}^{\text{num}}$ denotes the number of vehicles in $C_{m,j+1}$.

By optimizing the number of content chunks, $R_m$ maximizes the success probability of content transmission from the cluster to the request vehicle. The process of $R_m$ proactively caches content chunks at multiple vehicles in a cluster, as shown in Algorithm 2.

### Algorithm 2 Caching Algorithm for Optimizing the Number of Content Chunks

**Initialization:** All vehicles in coverage area of $R_m$ constitute the set $\text{SET}_m$, $\text{Max} = \phi$, $\text{Pre} = \phi$, $NV$

1. predict behavior of all vehicles in set $\text{SET}_m$, make vehicles with prediction lane $i + 1$ become CMs of cluster $C_{m,j+1}$, and put prediction probability into set $\text{Pre}$
2. for $k = 1$ to $\min \{ C_{i+1}^{\text{num}}, NV \}$ do
3. use $k$ to calculate $p_{i+1}^{\text{trans}}(k)$ in (8)
4. choose elements with top $k$ values from $\text{Pre}$ to constitute $\{ p_{i+1,1}^{\text{pre}}, p_{i+1,2}^{\text{pre}}, \ldots, p_{i+1,k}^{\text{pre}} \}$
5. use $\{ p_{i+1,1}^{\text{pre}}, p_{i+1,2}^{\text{pre}}, \ldots, p_{i+1,k}^{\text{pre}} \}$ and $p_{i+1}^{\text{trans}}(k)$ to calculate $p_{i+1}^{\text{pre}}(k)$ in (9), and put $p_{i+1}^{\text{pre}}(k)$ into set $\text{Max}$
6. end For
7. make $k$ corresponding to maximum value in $\text{Max}$ become number of content chunks

### 3.3. Analysis of Success Probability of Content Acquisition

In our strategy, vehicles cache content in their caching space based on content’s popularity, the caching space can accommodate content with maximum number of $N_i$, and the probability that a vehicle caches the content ranked $\tau - th$ is $f^\tau(N)$. Assume that a vehicle $V_x$ on lane $i$ requests the content with a rank of $\tau$ in popularity, and $V_x$ is within or is going to enter the coverage area of $R_m$. As described in Section 3.1, there are five cases for the request vehicle to obtain desired content: Obtaining content from vehicles that are a one-hop distance away from the request vehicle in the same lane, obtaining content from vehicles that are a two-hop distance away from request vehicle in the same lane, obtaining content from vehicles that are a one-hop or two-hop distance away from the request vehicle on the front opposite lane, obtaining content directly from an RSU, and obtaining content from the cluster with content cached by the RSU. The analysis of those five cases is shown as follows.

- Obtaining content from vehicles that are a one-hop distance away from the request vehicle

Assuming that request vehicle $V_x$ drives after content source vehicle $V_y$, the speed of each vehicle is, respectively, $s_i^x$ and $s_i^y$. The average transmission rate between vehicles is
When \( s_i^x > s_i^y \), the probability that the desired content manages to be cached by vehicles that are a two-hop distance away from the request vehicle is as follows:

\[
p_{i}^{\text{one}} = p\{s_i^x > s_i^y\} \cdot \int_{1}^{\infty} f(s_i^x, s_i^y) \, ds + p\{s_i^x < s_i^y\} \cdot \int_{1}^{s_i^x} f(s_i^x, s_i^y) \, ds
\]

Let \( K_1 \) denotes the number of vehicles that are a one-hop distance away from the request vehicle in the same lane. The probability that a request vehicle on lane \( i \) can obtain desired content from a vehicle that is a one-hop distance away from itself is derived as follows:

\[
P_{i}^{\text{self}} = \sum_{k=1}^{\infty} \sum_{i=1}^{N} P_i \{ K_1 = k \} \cdot \{ f^x(N) \cdot 1 + (1 - f^x(N)) \cdot [1 - (1 - f^x(N))^{k-1}] \} \cdot f_{s_i^x} = \sum_{k=1}^{\infty} \sum_{i=1}^{N} P_i \{ K_1 = k \} \cdot \{ f^x(N) \cdot 1 + (1 - f^x(N))^{k-1} \} \cdot \sum_{k=1}^{\infty} \sum_{i=1}^{N} P_i \{ K_1 = k \} \cdot (1 - f^x(N))^{k}
\]

Obtaining content from vehicles that are a two-hop distance away from the request vehicle

Assumed that the distance from relay vehicle to request vehicle and content source vehicle is \( 3x \), and the probability of content success transmission from relay vehicle to request vehicle and content source vehicle is also equal. When \( s_i^x > s_i^y \), the condition of content success transmission is \( \frac{r^x+3x}{s_i^x-s_i^y} \cdot V^x \geq W \) and the feasible region is

\[
G_3 = \{ (s_i^x, s_i^y) \mid s_i^x - s_i^y \leq \frac{3x \cdot V^x}{W} \}
\]

When \( s_i^x < s_i^y \), the condition of content success transmission is \( \frac{r^x+3x}{s_i^x-s_i^y} \cdot V^x \geq W \) and the feasible region is

\[
G_4 = \{ (s_i^x, s_i^y) \mid s_i^x - s_i^y \leq \frac{3x \cdot V^x}{W} \}
\]

According to Formula (3), the probability that a vehicle on lane \( i \) can obtain content from another relay vehicle is derived as follows:

\[
p_{i}^{\text{two}} = p\{s_i^x > s_i^y\} \cdot \int_{1}^{\infty} f(s_i^x, s_i^y) \, ds + p\{s_i^x < s_i^y\} \cdot \int_{1}^{s_i^x} f(s_i^x, s_i^y) \, ds
\]

The probability that the desired content fails to be cached by vehicles that are a one-hop distance away from the request vehicle is as follows:

\[
P_{i}^{\text{one}} = \sum_{k=1}^{\infty} \sum_{i=1}^{N} P_i \{ K_1 = k \} \cdot (1 - f^x(N))^{k}
\]

\[
= e^{-2\lambda Dr^x f^x(N)} - e^{-2\lambda Dr^o}
\]

Let \( K_2 \) denote the number of vehicles that are a two-hop distance away from the request vehicle. The probability that the desired content manages to be cached by vehicles that are a two-hop distance away from the request vehicle is as follows:
The desired content will be transferred to the request vehicle by a vehicle that is a two-hop distance away. According to Formulas (3), (17), and (18), the probability that a request vehicle on lane \( i \) can obtain desired content from the vehicle that is a two-hop distance away from itself is derived as the following:

\[
P_{i,\tau}^{\text{nei}} = P_{i,\tau}^{\text{ns}} \cdot P_{i,\tau}^{\text{en}} \cdot (p_{i}^{\text{fus}})^2
\]

(16)

\( \circ \) Obtaining content from vehicles that are a one-hop or two-hop distance away from the request vehicle on the front opposite lane

Let \( K_3 \) denote the number of vehicles that are a one-hop or two-hop distance away from the request vehicle in the same lane. The probability that the desired content fails to be cached by those \( K_3 \) vehicles is as follows:

\[
P_{i,\tau}^{\text{sa}} = \sum_{k=1}^{\infty} P_i \{ K_3 = k \} \cdot (1 - f^\tau(N))^k
\]

(17)

\[
= e^{-4\lambda_i D p^\tau f^\tau(N)} - e^{-4\lambda_i D p^\tau}
\]

Let \( K_4 \) denote the number of vehicles with one-hop or two-hop distance away from request vehicle on front opposite lane, the probability that desired content manages to be cached at those \( K_4 \) vehicles is as following

\[
P_{i,\tau}^{\text{op}} = \sum_{k=1}^{\infty} P_{i+1} \{ K_4 = k \} \cdot (1 - f^\tau(N))^k
\]

(18)

Assuming that the request vehicle needs to obtain the desired content from content source vehicle \( V_c \) on opposite lane \( i + I \), \( V_c \) chooses \( K^5 \) vehicles that are a one-hop distance away from \( V_c \) to average cache content chunks. As depicted in Section 3.2, those \( K^5 \) vehicles selected to cache content chunks will meet the request vehicle with a probability of 1. Assuming that the speed of any two vehicles, respectively, in the same and opposite lane are \( s^y_i \) and \( s^y_{i+1} \), the condition of content chunk success transmission between the two vehicles is \( 2\sigma^2 V_N \frac{s^y_i + s^y_{i+1}}{s^y_i + s^y_{i+1}} \leq \frac{W}{\tau} \) and the feasible region is \( G_5 = \{(s^x_i, s^y_{i+1}) | s^x_i + s^y_{i+1} \leq 2\sigma^2 V_N K^5 \} \).

According to Formulas (3), (17), and (18), probability that a request vehicle on lane \( i \) can obtain desired content from a vehicle that is a one-hop or two-hop distance away from itself on front opposite lane is derived as the following:

\[
P_{i,\tau}^{\text{oppp}} = P_{i,\tau}^{\text{sa}} \cdot P_{i,\tau}^{\text{op}} \cdot \left[ \int_{G_5} f(s^x_i, s^y_{i+1}) ds \right]^{K_5}
\]

(19)

where \( K^5 \) denotes the number of vehicles selected to cache content chunks, and \( V_a \) denotes the average transmission rate between vehicles.

Based on what is discussed above, the probability that a vehicle on lane \( i \) can obtain desired content without the participation of an RSU is as follows:

\[
P_{i,\tau}^{\text{not-rsu}} = P_{i,\tau}^{\text{self}} + P_{i,\tau}^{\text{nei}} + P_{i,\tau}^{\text{oppp}}
\]

(20)

\( \circ \) Obtaining content directly from an RSU

The authors of [22] indicated that the number of vehicles on the lane obeys the Poisson Point Process. Thus, the average driving distance of a vehicle in the coverage area of RSU was assumed to be \( R^\text{rsu} \). Assuming that the RSU serve and the request vehicles within the
coverage area of itself have an equal probability, the probability that RSU transfers desired content to a request vehicle is as follows:

\[
P_r = \frac{R^\text{max}}{\lambda^\text{rsu} \cdot W}
\]  

(21)

Assuming that the average rate that an RSU transfers content to a vehicle is \( V^{\text{per}} \), the condition of request vehicle successfully obtains the desire content from an RSU is \( s_i \leq \frac{R^\text{rsu} \cdot V^{\text{per}}}{W} \). According to Formulas (4) and (21), the probability that the request vehicle directly obtains the desired content from an RSU is derived as the following:

\[
P_{i,rsu}^{\text{dir}} = p_r \cdot F\left( \frac{R^\text{rsu} \cdot V^{\text{per}}}{W} \right) = \frac{R^\text{max}}{\lambda^\text{rsu} \cdot W} \cdot F\left( \frac{R^\text{rsu} \cdot V^{\text{per}}}{W} \right)
\]  

(22)

Obtaining content from cluster with content cached by an RSU

Assuming that \( R_m \) proactively caches content at cluster \( C_{\text{m},i+1} \) and chooses \( k \) vehicles from \( C_{\text{m},i+1} \) to respectively cache content chunks based on the caching optimization algorithm depicted in Section 3.2, the condition that RSU successfully transfers content chunks to a vehicle is \( s_i \leq \frac{R^\text{rsu} \cdot V^{\text{per}} \cdot k}{2W} \), and the prediction probability of those \( k \) vehicles constitute is set as \( \{ p_{i,rsu}^{\text{pre}} \}_{i=1}^k \). According to Formulas (3), (4), and (21), the probability that the request vehicle can successfully obtain the desired content from \( C_{\text{m},i+1} \) is derived as the following:

\[
P_{i,rsu}^{\text{indir}} = \left[ p_r \cdot F\left( \frac{R^\text{rsu} \cdot V^{\text{per}} \cdot k}{2W} \right) \right] \cdot \left[ \prod_{j=1}^k f(s_i, s_{i+1}^y) \right] \cdot \prod_{i=1}^k p_{i,rsu}^{\text{pre}}
\]  

(23)

\( G6 = \{(s_i^x, s_{i+1}^y) | s_i^x + s_{i+1}^y \leq \frac{2R^\text{rsu} \cdot V^{\text{per}} \cdot k}{W} \} \) denotes the feasible region of \( s_i^x \) and \( s_{i+1}^y \), \( s_i^x \) and \( s_{i+1}^y \) denote the speed of request vehicle and the vehicle from \( C_{\text{m},i+1} \).

Because the number of vehicles on the lane obeys the Poisson Point Process [22], the probability that a vehicle is in the coverage area of an RSU is

\[
P^{\text{in}} = \frac{2 \cdot R^\text{rsu}}{E}
\]  

(24)

According to Formulas (22)–(24), the probability that a vehicle can obtain desired content with the participation of the RSU is derived as the following:

\[
P_{i,\tau}^{\text{rsu}} = P_{i,\tau}^{\text{in}} \cdot P_{i,\tau}^{\text{dir}} + (1 - P_{i,\tau}^{\text{in}}) \cdot P_{i,\tau}^{\text{indir}}
\]  

(25)

The average probability that a vehicle on lane \( i \) can successfully obtain desired content ranked \( \tau - th \) is derived as the following:

\[
P_i^{\text{ave}} = \sum_{\tau=1}^{l} f_\tau \cdot [P_{i,\tau}^{\text{not,rsu}} + (1 - P_{i,\tau}^{\text{not,rsu}}) \cdot P_{i,\tau}^{\text{rsu}}]
\]  

(26)

4. Discussion

4.1. Parameter Settings

In order to evaluate the performance of proposed strategy, we used SUMO to simulate an urban road environment, and vehicles moved along the urban road like real scenes. The arrival rate of vehicle was 600 vehicles/h, and the maximum speed of vehicle was 80 km/h. An interface, named TraCI (Traffic Control Interface), was used to connect SUMO to Python. The 3-hour moving information of vehicles was collected by Python through the interface. Python used the moving information to perform the algorithm simulation and to compare the simulation results with those described by the authors of [19]. In our simulation, the channel model is expressed as \( \beta \cdot P_i \cdot r^{-\alpha} \) with a path loss exponent \( \alpha \) of 4,
channel fading gain $\beta$ of $10^{-2}$, channel bandwidth $B$ of 1.5 MHz, and power of Gaussian noise $P_n$ of $-110$ dBm. The transmission rate of the network can be derived by the Shannon formula. The average rate between vehicles was set as 9 Mbps, and the average between vehicle and RSU was set as 20 Mbps. Other simulation parameters are shown in Table 2. The Average Delay was used to describe the average time duration between the vehicle send requests and when the vehicle obtains the desired content. The Request Success Ratio was used to describe the average ratio that can be derived by dividing the total number of requests by the number of successful requests.

Table 2. Simulation parameters.

| Parameter                     | Value  |
|-------------------------------|--------|
| The number of content         | 200    |
| The size of content (Mbit)    | 600    |
| The parameter of Zipf Distribution | 1     |
| The transmission power of vehicle (mW) | 200   |
| The transmission power of RSU (mW) | 2000  |
| The coverage range of vehicle (m) | 50    |
| The coverage range of RSU (m)  | 100    |
| The length of lane (m)        | 2000   |
| The width of lane (m)         | 7      |

4.2. Results and Analysis

Figures 7 and 8 show the variations of the Average Delay and the Request Success Ratio in the case of different content sizes. As illustrated in the graphs, the request delay of vehicles increased and the probability of content success transmission decreased as the content size increased. This is because larger content requires more time to be transferred by V2V or I2V, and the transmission process fails more easily. Compared with the strategy described by the authors of [19], our strategy has a lower request delay and higher probability of content success transmission, and the gap between the two strategies increases as content size increases. Because our strategy caches content chunks at multiple vehicles in a cluster, request vehicles can obtain content chunks from vehicles in the cluster one by one, which increases the transmission duration between vehicles. Our strategy also adopts the multiuser multichannel transmission mode in the communication process between clusters to solve the problem of the communication duration being too short to complete the content transmission. Figure 8 also demonstrates that the theory value and the simulation value of the Request Success Ratio were similar to each other. The former was slightly higher than the latter, and the gap increased as content size increased. This is because we assumed that the process of the request vehicle obtaining content chunks from different relay vehicles in the cluster was independent in case 5 of our theory analysis. However, as the content size increases, the process between the request vehicle and relay vehicle may wait to begin until the last process is over, which make the theory value higher than the simulation value.

Figures 9 and 10 show the variations of the Average Delay and the Request Success Ratio in the case of different vehicle caching sizes. Vehicle caching size is used to describe the maximum capability of the vehicle's caching space. As depicted in the figures, the request delay of vehicles decreased and the probability of content success transmission increased as the caching size increased. This is because increasing the caching size enlarges the probability that a vehicle can find desired content in neighboring vehicles. Compared with the strategy described by the authors of [19], our strategy has a lower request delay and higher probability of content success transmission. Because our strategy caches content chunks at multiple vehicles in a cluster, request vehicles can obtain content chunks from vehicles in the cluster one by one, which increases the transmission duration between vehicles. Our strategy also adopts the multiuser multichannel transmission mode in the communication process between clusters to solve the problem of the communication
duration being too short to complete the content transmission. The probability that a vehicle can obtain content with participation of RSU decreases as vehicle caching size increases, which weakens the advantages of our strategy, so the gap in performance between those two strategies decreases as caching size increases. Figure 10 also indicates that the theory value and the simulation value of the Request Success Ratio were similar to each other. The former was slightly higher than the latter, and the gap increased as the vehicle caching size increased. This is because the probability that a vehicle will obtain content from neighboring vehicles increases as vehicle caching size increases. We assumed that the relay vehicle situated in the middle position between the request vehicle and content source vehicle in case ② of our theory analysis, which makes the theory value higher than the simulation value.

![Figure 7](image1.png)

**Figure 7.** The relationship between the Average Delay and content size.

![Figure 8](image2.png)

**Figure 8.** The relationship between the Request Success Ratio and content size.
Figures 11 and 12 show the variations of the Average Delay and the Request Success Ratio in the case of different Zipf distribution parameters. As depicted in the graphs, the request delay of vehicles decreased, and the probability of content success transmission increased as the parameter increased. This is because increasing the Zipf distribution parameter increases the probability that a vehicle can find desired content in neighboring vehicles. Compared with the strategy described by the authors of [19], our strategy possesses a lower request delay and higher probability of content success transmission. Because our strategy caches content chunks at multiple vehicles in a cluster, request vehicles can obtain content chunks from vehicles in the cluster one by one, which increases the transmission duration between vehicles, besides. Our strategy also adopts the multiuser multichannel transmission mode in the communication process between clusters to solve the problem of the communication duration being too short to complete content transmission. The probability that a vehicle can obtain content with the participation of RSU decreased as the
Zipf distribution parameter increased, which weakened the advantages of our strategy, so the gap in performance between those two strategies decreased as the parameter increased. Figure 12 also illustrates that the theory value and the simulation value of the Request Success Ratio were similar to each other. The former was slightly higher than the latter, and the gap increased as the Zipf distribution parameter increased. This is because the probability that a vehicle will obtain content from neighboring vehicles increases as the Zipf distribution parameter increases. We assumed that the relay vehicle situated in the middle position between the request vehicle and the content source vehicle in case of our theory analysis, which makes the theory value higher than the simulation value.

Figure 11. The relationship between Average Delay and Zipf distribution parameter.

Figure 12. The relationship between Request Success Ratio and Zipf distribution parameter.
5. Conclusions

In this paper, we proposed the idea of predicting the moving trajectory of a cluster based on the vehicle behavior prediction in intersection, whereby the driving behavior of vehicles in a cluster determines the moving behavior of the cluster. On this basis, RSU divides the content into multiple chunks and proactively caches those chunks at multiple relay vehicles in a cluster that is about to meet the request vehicle. By letting the request vehicle obtain content chunks from relay vehicles one by one, our strategy enlarges the communication duration. Our paper also optimizes the number of chunks to maximize the probability that the request vehicle will successfully obtain content from the cluster. Besides, our paper adopts the multiuser multichannel transmission mode in the communication process between clusters. Our simulation results demonstrate that the proposed strategy can improve the performance of vehicular network.

Our paper uses vehicle cluster in a trajectory-predicted approach to deliver content. However, there are still several issues, for example, the caching optimization algorithm fails to consider the factor of distance. Future works should explore this issue.

Author Contributions: Conceptualization: S.Y. and Q.Z.; methodology: S.Y.; software: S.Y.; validation: S.Y. and Q.Z.; formal analysis: S.Y.; investigation: S.Y.; resources: Q.Z.; data curation: S.Y.; writing: S.Y.; visualization: S.Y.; supervision: Q.Z.; project administration: Q.Z.; funding acquisition: Q.Z. Both authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China (61971239), (92067201), (61631020) and Postgraduate Research & Practice Innovation Program of Jiangsu Province (KYCX20_0810).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and codes presented in this study are available from the corresponding author by request.

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

References

1. Cheng, X.; Yang, L.; Shen, X. D2D for Intelligent Transportation Systems: A Feasibility Study. *IEEE Trans. Intell. Transp. Syst.* 2015, 16, 1784–1793. [CrossRef]
2. Campanile, L.; Iacono, M.; Levis, A.H.; Marulli, F.; Mastroianni, M. Privacy Regulations, Smart Roads, Blockchain, and Liability Insurance: Putting Technologies to Work. *IEEE Secur. Priv.* 2021, 19, 34–43. [CrossRef]
3. Campanile, L.; Iacono, M.; Marulli, F.; Mastroianni, M. Designing a GDPR compliant blockchain-based IoV distributed information tracking system. *Inf. Process. Manag.* 2021, 58, 1–23. [CrossRef]
4. Jacobson, V.; Smetters, D.K.; Thornton, J.D.; Plass, M.F.; Briggs, N.H.; Braynard, R.L. Networking named content. *Commun. ACM.* 2009, 55, 117–124. [CrossRef]
5. Laoutaris, N.; Che, H.; Stavrakakis, I. The LCD interconnection of LRU caches and its analysis. *Perform. Eval.* 2006, 63, 609–634. [CrossRef]
6. Laoutaris, N.; Syntila, S.; Stavrakakis, I. Meta Algorithms for Hierarchical Web Caches. In Proceedings of the IEEE International Conference on Performance, Computing, and Communications, Phoenix, AZ, USA, 15–17 April 2004; IEEE: Piscataway, NJ, USA, 2004; pp. 445–452.
7. Psaras, I.; Chai, W.K.; Pavlou, G. Probabilistic in-Network Caching for Information-Centric Networks. In Proceedings of the 2nd edition of the ICN Workshop on Information-Centric Networking, Helsinki, Finland, 17 August 2012; Association for Computing Machinery (ACM): New York, NY, USA, 2012; pp. 55–60.
8. Grewe, D.; Wagner, M.; Frey, H. PeRCelVE: Proactive Caching in ICN-Based VANETs. In Proceedings of the IEEE Vehicular Networking Conference, Columbus, OH, USA, 8–10 December 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–8.
9. Alnagar, Y.; Hosny, S.; El-Sherif, A.A. Towards Mobility-Aware Proactive Caching for Vehicular Ad hoc Networks. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference Workshop (WCNCW), Marrakech, Morocco, 15–18 April 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6.
10. Khelifi, H.; Luo, S.L.; Nour, B.; Sellami, A.; Moungla, H.; Naït-Abdesselam, F. An Optimized Proactive Caching Scheme Based on Mobility Prediction for Vehicular Networks. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, 9–13 December 2018; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6.
11. Yao, L.; Chen, A.; Deng, J.; Wang, J.; Wu, G. A Cooperative Caching Scheme Based on Mobility Prediction in Vehicular Content Centric Networks. *IEEE Trans. Veh. Technol.* **2018**, *67*, 5435–5444. [CrossRef]

12. Zhao, Z.; Guardalben, L.; Karimzadeh, M.; Silva, J.; Braun, T.; Sargento, S. Mobility Prediction-Assisted Over-the-Top Edge Prefetching for Hierarchical VANETs. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 1786–1801. [CrossRef]

13. Zhang, F.; Xu, C.; Zhang, Y.; Ramakrishnan, K.K.; Mukherjee, S.; Yates, R.; Nguyen, T. EdgeBuffer: Caching and Prefetching Content at the Edge in the MobilityFirst Future Internet Architecture. In Proceedings of the 16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Boston, MA, USA, 14–17 June 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–9.

14. Hou, L.; Lei, L.; Zheng, K.; Wang, X. A Q-Learning based Proactive Caching Strategy for Non-safety Related Services in Vehicular Networks. *IEEE Internet Things J.* **2018**, *6*, 4512–4520. [CrossRef]

15. Gerla, M.; Tsai, J.T.-C. Multicluster, mobile, multimedia radio network. *Wirel. Netw.* **1995**, *1*, 255–265. [CrossRef]

16. Basu, P.; Khan, N.; Little, T.D.C. A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks. In Proceedings of the 21st International Conference on Distributed Computing Systems Workshops, Mesa, AZ, USA, 16–19 April 2001; Association for Computing Machinery: New York, NY, USA, 2001; pp. 413–418.

17. Chatterjee, M.; Das, S.K.; Turgut, D. WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks. *Clust. Comput.* **2002**, *5*, 193–204. [CrossRef]

18. Huang, W.; Song, T.; Yang, Y.; Zhang, Y. Cluster-Based Cooperative Caching with Mobility Prediction in Vehicular Named Data Networking. *IEEE Access* **2019**, *7*, 23442–23458. [CrossRef]

19. Huang, W.; Song, T.; Yang, Y.; Zhang, Y. Cluster-Based Selective Cooperative Caching Strategy in Vehicular Named Data Networking. In Proceedings of the 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN), Shenzhen, China, 15–17 August 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 7–12.

20. Fang, S.; Fan, P. A Cooperative Caching Algorithm for Cluster-Based Vehicular Content Networks with Vehicular Caches. In Proceedings of the 2017 IEEE Globecom Workshops (GC Wkshps), Singapore, 4–8 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–6.

21. Hu, B.; Fang, L.; Cheng, X.; Yang, L. In-Vehicle Caching (IV-Cache) Via Dynamic Distributed Storage Relay (D2SR) in Vehicular Networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 843–855. [CrossRef]

22. Dacey, M.F. Some Properties of the Superposition of a Point Lattice and a Poisson Point Process. *Econ. Geogr.* **2016**, *47*, 86–90. [CrossRef]

23. Wei, M.; Jin, W.; Shen, L. A Platoon Dispersion Model Based on a Truncated Normal Distribution of Speed. *J. Appl. Math.* **2012**, *2012*, 1–13. [CrossRef]

24. Cha, M.; Kwak, H.; Rodriguez, P.; Ahn, Y.; Moon, S. Analyzing the Video Popularity Characteristics of Large-Scale User Generated Content Systems. *IEEE/ACM Trans. Netw.* **2009**, *17*, 1357–1370. [CrossRef]

25. Krajzewicz, D. Traffic Simulation with SUMO—Simulation of Urban Mobility. In *Fundamentals of Traffic Simulation*; Barceló, J., Ed.; Springer: Berlin/Heidelberg, Germany, 2010; pp. 269–293.

26. Kagalkar, A.; Raghuram, S. CORDIC Based Implementation of the Softmax Activation Function. In Proceedings of the 2020 24th International Symposium on VLSI Design and Test (VDAT), Bhubaneswar, India, 23–25 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–4.