IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 29, 2017, accepted May 22, 2017, date of publication June 9, 2017, date of current version July 24, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2714191

An Efficient Cache Strategy in Information Centric Networking Vehicle-to-Vehicle Scenario

WEICHENG ZHAO[1], YAJUAN QIN[1], (Member, IEEE), DEYUN GAO[1], (Member, IEEE), CHUAN HENG FOH[2], (Senior Member, IEEE), AND HAN-CHIEH CHAO[3], (Senior Member, IEEE)

[1]School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China
[2]Institute for Communication Systems, Department of Electrical and Electronic Engineering, University of Surrey, Surrey GU1 2UX, U.K.
[3]Department of Electrical Engineering, National Dong Hwa University, Hualien 97401, Taiwan

Corresponding author: Deyun Gao (gaody@bjtu.edu.cn)

This work was supported in part by the 973 Program under Grant 2013CB329100, in part by NSFC under Grant 61271201, Grant 61232017, and Grant 61272504, and in part by the National High Technology of China through 863 Program under Grant 2015AA016101.

ABSTRACT Information centric networking (ICN) has been recently proposed as a prominent solution for content delivery in vehicular ad hoc networks. By caching the data packets in vehicular unused storage space, vehicles can obtain the replicate of contents from other vehicles instead of original content provider, which reduces the access pressure of content provider and increases the response speed of content request. In this paper, we propose a community similarity and population-based cache policy in an ICN vehicle-to-vehicle scenario. First, a dynamic probability caching scheme is designed by evaluating the community similarity and privacy rating of vehicles. Then, a caching vehicle selection method with hop numbers based on content popularity is proposed to reduce the cache redundancy. Moreover, to lower the cache replacement overhead, we put forward a popularity prediction-based cooperative cache replacement mechanism, which predicts and ranks popular content during a period of time. Simulation results show that the performance of our proposed mechanisms is greatly outstanding in reducing the average time delay and increasing the cache hit ratio and the cache hit distance.

INDEX TERMS Information centric networking, vehicular ad-hoc networks, cache policy, community similarity, privacy rating, content popularity.

I. INTRODUCTION

With the improvement of technology and increasing demands for applications, new interests have emerged in the delivery of popular content service for Vehicular Ad Hoc Networks (VANET). Vehicles equipped with a variety of On Board Unit(OBU) exchange information to support safety, traffic efficiency and infortainment applications [1]. Such vehicular applications require the distribution of large amounts of data among heterogeneous users under intermittent connectivity in harsh signal propagation, high mobility and sparse roadside facility conditions [2], [3]. The host-centric IP-based protocols of current Internet, designed with the end-to-end connectivity principle in mind, barely work under such settings [4].

Information Centric Networking (ICN) has been recently proposed as a prominent solution for content delivery service situation [5]. ICN uses a content-centric communication method to replace the existing IP address-centric one with a publish/subscribe and content requestor driven-based paradigm [6]. It retrieves a given content directly using the content name instead of referring to the IP address of the node storing the content [7]. Meanwhile, the in-network caching feature of the ICN allows the routers in the network to cache content. It shortens the response time of content requestors accessing the cached content and reduces the access pressure of the content provider [8].

In-network caching is the main strategy of Information Centric Networking framework in which caches spread out in the network can be used to store the most popular contents [9]. By caching the data packets in the vehicular unused storage space, vehicles can obtain the replicate of provider's contents from neighbor vehicles instead of a original content. When a large number of users request the same content in a time, or vehicles can not directly access to the base station under the partial coverage constraints, cache policy significantly decreases the burden of original content server and improves the content request response speed [10].

Compared to ICN cache strategy in Internet, there are three special features for applying the ICN cache in VANETs with a Vehicle-to-Vehicle(V2V) scenario. First, with the consideration of their privacy and selfishness, drivers of the vehicles may take a reluctant role to obey the rules of a cache sharing policy [11]. Secondly, the high-speed movement of vehicles brings dynamic topology changes, which greatly increases the complexity of the cache strategy [12]. Finally, the vehicle's storage space and bandwidth resource are relatively weak compared to the base stations and routers, cache redundancy of the strategy should be decreased [13]. In this paper, we propose a Community Similarity and Population-based Cache Policy (CSPC) in an ICN V2V scenario. Firstly, a dynamic probability caching scheme is designed by evaluating the community similarity and privacy rating of vehicles. A caching vehicle selection method with hop numbers based on content popularity is introduced further to reduce the cache redundancy. Additionally, we put forwards a Popularity Prediction-based Cooperative Cache Replacement (PPCCR) to lower the replacement overhead.

The rest of the paper is organized as follows. We introduce the related work in Section II. The main framework of the cache mechanism for VANETs is given in Section III. Section IV shows the details of our proposed cache policy. The simulation results and performance evaluation are discussed in Section V. We conclude the contribution in Section VI finally.

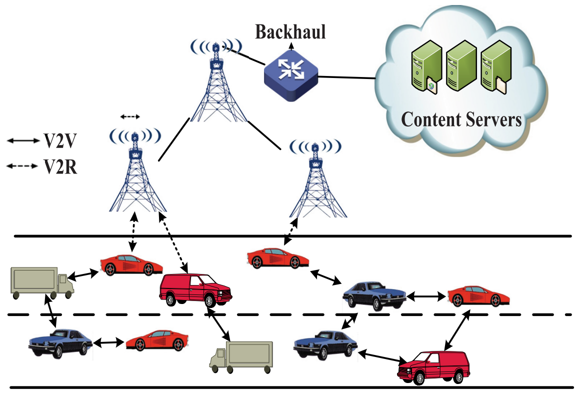

FIGURE 1. Vehicular network architecture.

II. RELATED WORK

ICN architecture is originally proposed by Van Jacobson as a research project initiated by Palo Alto Research Center (PARC) [14]. In a typical ICN scenario, each node maintains three data structures: Pending Interest Table (PIT), Content Store (CS) and Forwarding Information Base (FIB) [15]. Meanwhile, there are two message types in ICN: interest packet and data packet. The interest packet is initiated by the content requester for retrieving the content, and the content provider replies the data packet as a response to the interest packet [16].

Experts and scholars have done some researches for in-network cache in the ICN architecture [17]. In [18], each content router in the probabilistic caching policy caches the content with a fixed probability p. That is, the content router randomly generates a probability after receiving the data packet. If the probability is lower than the set probability p, the content can be cached. Otherwise the content will not be cached. Therefore, the more the number of requests for a content, the greater the chance of caching in the content router. In [19], the SPRA strategy (Storage Planning and Replica Assignment) was proposed to use a trait-explicit cooperative caching policy. It classifies the content into different popularity level and advertises the level of cached content among the content routers. It can reduce content request time and balance network traffic. However, the detail communication mechanism is not very clear between the content routers. In [20], the authors proposed a path-explicit cooperative caching strategy based on neighbor search and admission control. Neighbor content routers interact with cached content to improve cache hit rates, while admission control forces cached content not to be cached by other routers to reduce cache redundancy.

All above work do not consider and explore the characteristics of VANETs. In [21], the authors proposed a Vehicle-to-Infrastructure scenario cache policy in VANETs. They present an Integer Linear Programming (ILP) formulation of the problem of optimally distributing contents in the network nodes while considering the available storage capacity and the available link capacity to maximize the probability that a user retrieves desired content. However, with the poor-quality wireless links and the mobility, vehicles can not directly access to the base station reliably all the time, and a Vehicle-to-Vehicle cache policy is needed.

In [22], the authors proposed a ICN-based COoperative Caching solution (ICoC) for a V2V scenario to improve the quality of experience (QoE) of multimedia streaming services. It enhances information-centric networking caching by leveraging two novel social cooperation schemes, namely partner-assisted and courier-assisted. While, ICoC can only apply to high-speed road scenes, which requires that the arrival rate of the vehicle be strictly Poisson distribution, and it lacks generality to extend to the Urban road scenes. To overcome these shortcomings, we propose the CSPC mechanism to reduce the cache redundancy and improve the content request cache hit.

III. NETWORK ARCHITECTURE

The network architecture is shown in Fig. 1. The vehicles can connect to roadside facilities through a variety of networks, such as accessing to base station with 4G/5G cellular network, Road Side Unit (RSU) with 802.11P protocol and other possible facilities. Vehicles also can share and exchange date packet via cellular network. To assist the vehicle in requesting and downloading various contents stored in the content cloud, the base stations are connected to each other through a wired or wireless connection to the backbone network. The base station can forward and deliver these content, which also cache some popular content for other vehicles and roadside units to provide more convenient downloading service. In this paper, we mainly focus on the caching strategy in V2V scenario. When vehicles can not directly access to the base station under the partial coverage constraints, vehicles can obtain the replicate of provider's contents from neighbor vehicles by caching the contents in the vehicular unused storage space.

To better understand how caching scheme works in this architecture, we briefly describe the basic communication mechanism of the ICN framework. In ICN, content retrieval relies on two types of messages: interest packet and data packet. Interest packet is used to request the content and carries the content name and other information. Data packet carries the content name, content and signature and other information. When a user attempts to obtain a service content, an interest packet is sent to the network, and the content source sends out the corresponding data packet after receiving the interest packet. The contents in our schemes are real-time media streaming, audio and video services, multimedia communication, etc. Those frequently requested contents are called popular content. In V2V mode, when a vehicle a tries to obtain the content, it will send an interest packet which contains the name of the service content to the surrounding vehicle, and neighbor vehicles will carry and forward the interest packet until the corresponding data packet is found out. Vehicle b carrying the corresponding data packet forwards the data packet to the vehicle a by some route strategy, which is not the focus in this paper. In the process of forwarding data packets, the vehicle should obey a cache policy to decide whether to cache this data packet.

IV. COMMUNITY SIMILARITY AND CONTENT POPULATION-BASED CACHE POLICY

A. VEHICLE ADAPTIVE DYNAMIC PRIVACY RATING

Considering the selfishness and privacy, some vehicles may take a reluctant role in obeying network protocol such as sharing service protocol by manually modifying the vehicle's program [23]. In [24], we have already proposed a incentive method to encourage vehicles participating in the bandwidth and storage space sharing service. In this paper, we present a dynamic privacy rating based on evaluating the vehicles' contribution in caching communication performance.

As shown in Fig. 2, the vehicles are divided into two categories: the public vehicle set and the private vehicle set. The public vehicle set Θ_{repub} includes taxis, buses and other members. The selfishness of these vehicles is weak and the cache contribution of the public vehicle is the highest. The private vehicle set is different and vehicle drivers more or less worry about the privacy leaks.

The private vehicle set can be divided into the following three categories. In the private vehicle set $\Theta_{vehicle1}$, the drivers are willing to try new things. Their selfishness are weak and

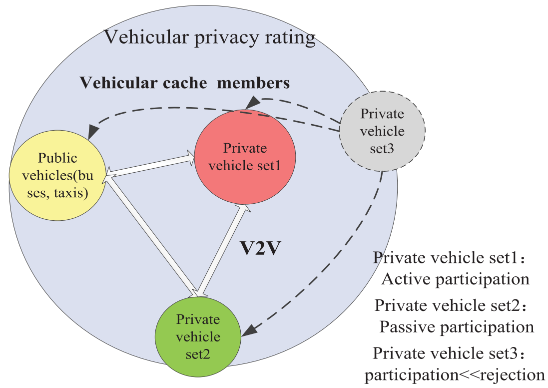

FIGURE 2. Classification of vehicle privacy.

they participate in content caching and sharing communication same as the public vehicle set. In the private vehicle set $\Theta_{vehicle2}$, the drivers are willing to try new things, but they also worry about their own privacy leaking. They may participate in the content cache and may also refuse to cache and forward the content. In the private vehicle set $\Theta_{vehicle3}$, the drivers tend to be more conservative, and there is a great probability to refuse to cache and forward the content.

The vehicle drivers can set their own privacy rating initially when connecting to the vehicular network. Adaptive dynamic rating adjustment is performed according to the number of times the networked vehicles participate in the cache sharing service during a period i. The ratio of content delivery participation F_a^i for vehicle a is:

$$F_a^i = \frac{N_{accp}}{N_{total}} \tag{1}$$

where, N_{total} represents the number of communications of the vehicle a in the i period, and N_{accp} represents the number of times the vehicle a actively participates in vehicle communication. According to the participation degree of the vehicle, we can dynamic adjust the privacy level of vehicle a and evaluate the sets of vehicle as follows:

$$a \in \begin{cases} \Theta_{vehicle1} \mathrm{F}_a^i \geq W_{vehicle1}; \\ \Theta_{vehicle2} \mathrm{F}_a^i < W_{vehicle1} \mathrm{and} \mathrm{F}_a^i \geq W_{vehicle2}; \\ \Theta_{vehicle3} \mathrm{F}_a^i < W_{vehicle2}; \end{cases} \tag{2}$$

Some parameters and notations are given in Table 1. The parameter $W_{vehicle1}$ is a threshold in which the vehicle can join the private vehicle set $\Theta_{vehicle1}$. When $\mathrm{F}_a^i \geq W_{vehicle1}$, we think the participation of vehicle is high enough, and the vehicle can be classified as a private vehicle set $\Theta_{vehicle1}$. The parameter $W_{vehicle2}$ is a transfer threshold in which the vehicle can join the private vehicle set $\Theta_{vehilce2}$. When $\mathrm{F}_a^i \geq W_{vehicle2}$ and $\mathrm{F}_a^i < W_{vehicle1}$, we believe that the vehicle has a certain degree of forwarding participation, the vehicle can be classified as a private vehicle set $\Theta_{vehicle2}$. Otherwise, the vehicle can be classified as a private vehicle set $\Theta_{vehicle3}$.

TABLE 1. Notations.

Symbol	Meaning
F_a^i	The successful content deliver ratio of vehicle a in i period
N_{accp}	Counts of successful content deliver
N_{total}	Counts of total content deliver
$\Theta_{vehicle1}$	Privacy rating Set of vehicle1
$\Theta_{vehicle2}$	Privacy rating Set of vehicle2
$\Theta_{vehicle3}$	Privacy rating Set of vehicle3
$W_{vehicle1}$	Threshold for set vehicle1
$W_{vehicle2}$	Threshold for set vehicle2
$W_{vehicle3}$	Threshold for set vehicle3
$\Pi(a)$	Privacy rating utility values of vehicle a
$S^{\alpha}_{(i,\theta)}$	Content similarity utility
$S^{\beta}_{(i,j)}$	Vehicle moving similarity utility
$Prob(U_{CSPC})$	Content cache probability for CSPC method
ω_α	Weight of $S^{\alpha}_{(i,\theta)}$
ω_β	Weight of $S^{\beta}_{(i,j)}$
ω_χ	Weight of vehicle privacy rating
M	Total level of content popularity
$F_{Level}(P_i)$	Popularity level for content i
$Interval(P_i)$	Cache interval for content i
$Interval(P_i^a)$	Cache interval for content i in vehicle

We can obtain privacy utility values based on different vehicle privacy sets $\Pi\ (a)$čž

$$\Pi\ (a) = \begin{cases} \Pi_{repub}, & a \in \Theta_{repub}; \\ \Pi_{\Theta_{vehicle1}}, & a \in \Theta_{vehicle1}; \\ \Pi_{\Theta_{vehicle2}}, & a \in \Theta_{vehicle2}; \\ \Pi_{\Theta_{vehicle3}}, & a \in \Theta_{vehicle3}; \end{cases} \tag{3}$$

where $\Pi_{\Theta_r epub}$, $\Pi_{\Theta_v ehicle1}$, $\Pi_{\Theta_v ehicle2}$, $\Pi_{\Theta_v ehicle3}$ indicate the privacy utility values of the vehicle sets Θ_{repub}, $\Theta_{vehicle1}$, $\Theta_{vehicle2}$ and $\Theta_{vehicle3}$ respectively.

B. COMMUNITY SIMILARITY-BASED PROBABILISTIC CACHE STRATEGY

Due to the high mobility resulting in the dynamic topology, the vehicles sometimes cannot directly access to the base station or roadside unit to obtain service content, so we design a Community Similarity-based Probabilistic Cache strategy in V2V scenario. Community similarity is divided into moving similarity and content similarity. The moving similarity indicates the link stability of two vehicles. The higher the moving similarity of vehicles, the longer the link stays stable. Meanwhile, the content similarity indicates those vehicle drivers have similar hobbies. The higher content similarity indicates the higher probability that the vehicles will cache the content.

When a data packet is forwarded by the vehicle a, we first determine whether to cache the data packet by assessing the content similarity. In the ICN, content naming policy has three types: flattening, hierarchical and hybrid naming rules. As shown in Fig. 3, a tennis video can be named as the following format {Videos, Sports, Tennis, Andy Murray, Olympic Final 2016}.

To illustrate our method, we use name component θ ($\theta \in N$) to represent content's attribute. When a content

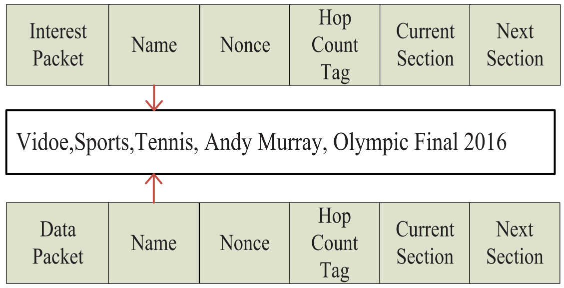

FIGURE 3. Name format of the information centric networking.

θ is transmitted through the vehicle a, we can compare the content similarity by observing contents' name component. If the vehicle a has received many same contents, the content similarity is high. Otherwise, the content similarity is low. Considering that the interest of the vehicle may change with time, the content similarity weights are different under different time periods. We evaluate the content similarity $S^{\alpha}_{(i,\theta)}$ by the following formula:

$$S^{\alpha}_{(i,\theta)} = R_{(\theta,T)} \otimes \frac{1}{T} = \int_0^T R_{(\theta,\tau)} \times \frac{1}{T-\tau} d_\tau \tag{4}$$

Here, we can use a continuous integral for the discrete data packets. At time τ, there are only two states for the received data packet in the vehicle: existing or not existing. So, we use $R_{(\theta,\tau)}$ to represent whether the vehicle has received a packet that contained component θ at the moment τ. If it has received, the value is 1, otherwise the value is 0. If we considered $R_{(\theta,\tau)}$ as a digital pulse at a moment, then $\int_0^T R_{(\theta,\tau)} \times \frac{1}{T-\tau}$ can be considered as residual strength that name component θ during continuous time period.

Secondly, we evaluate the moving similarity. Normally, the movement of vehicle has the regional characteristic. We use moving similarity $S^{\beta}_{(i,j)}$ to characterize this physical relationship between vehicle a and vehicle b.

$$S^{\beta}_{(a,b)} = \frac{\int_0^T f_{(a,b)}(t)dt}{T} \tag{5}$$

In this formula, we consider both the meeting time and the meeting frequency of vehicle a and vehicle b. $f_{(a,b)}(t)$ characterizes the pulse intensity of vehicle a and b at time t. $f_{(a,b)}(t) = 1$ indicates that the two vehicles are meeting and they are in the neighborhood communication range. $\frac{\int_0^T f_{(a,b)}(t)dt}{T}$ represents the time during which these two vehicles can connect during the time window T.

Therefore, when the vehicle a obtains the content component θ, the cache probability utility function can be calculated as follows:

$$Prob(U_{CSPC}) = \omega_\alpha S^{\alpha}_{(a,\theta)} + \omega_\beta S^{\beta}_{(a,b)} + \omega_\chi \Pi(a) \tag{6}$$

where $\omega_\alpha, \omega_\beta, \omega_\chi$ are the utility weights, and $\omega_\alpha + \omega_\beta + \omega_\chi = 1$. When the new data packet arrives, the vehicle computes the utility function $ProbU_{CSPC}$ accordingly and caches it with $ProbU_{CSPC}$ as the probability.

C. CACHING MECHANISM WITH HOP NUMBERS BASED ON CONTENT POPULARITY RATING

Compared to fixed facilities and routers, the vehicle's cache spaces and bandwidth resources are relatively weak. We propose a dynamic caching method with hop numbers based on content popularity to reduce the cache redundancy. Furthermore, we improve it according to the privacy rating of the vehicle.

In our proposed mechanism, we add an "Interval" field into the Data packet. The value of "Interval" represents the hop numbers and the content will be cached once for every other those hops. For example, if the value of the "Interval" is 2, the content will be cached every other two hops. The value of "Interval" relates to the content popularity, which can be calculated based on the request times of content over a period of time. Obviously, popular content has a small value of "Interval". According to the analysis in [25], the distribution characteristic of the content request conforms to the Zipf-Mandelbrot distribution (referred as Zipf distribution). The probability quality function of this distribution is shown as follows.

$$P(i) = \frac{(i+\theta)^{-\alpha}}{\sum_{x=1}^{N} x + \theta^{-\alpha}} \tag{7}$$

where i is the ordinal number of content popularity, N is the number of requests, θ is the parameters of Zipf distribution, and α is the exponential parameter of distribution.

In the above formula, we can see that most of the requests are concentrated in a small part of the contents with high popularity. Thus, we classify the contents into different levels with different popularity to reduce cache redundancy. The content popularity level $F_{Level}(P_i)$ is calculated in the following formula.

$$F_{Level}(P_i) = \begin{cases} 1 & \log P_i \geq \log P_H - \dfrac{\log \frac{P_H}{P_L}}{M} \\ M - \left\lceil \dfrac{M * \log \frac{P_i}{P_L}}{\log \frac{P_H}{P_L}} - 1 \right\rceil & else \\ M & \log P_i < \log P_L + \dfrac{\log \frac{P_H}{P_L}}{M} \end{cases} \tag{8}$$

where, P_H is the probability of the highest content popularity, P_L is the probability of the lowest content popularity, M is the total number of content popularity level, and P_i is the popularity probability of content i. $\log P_H - \frac{\log \frac{P_H}{P_L}}{M}$ and $\log P_L + \frac{\log \frac{P_H}{P_L}}{M}$ are the highest level and lowest level of the popularity threshold. When P_i is larger than the highest-level popularity threshold, the content i belongs to the first level. Likewise, if P_i is smaller than the lowest-level popularity threshold, the content i belongs to the last level M.

We optimize the value of "interval" according to the above classification result. Since the Zipf distribution is a power-law distribution, we give out the "interval" between the different levels according to the content popularity in the following formula.

$$Interval(P_i) = F_{Level}(P_i) - F_{Level}(P_H) \tag{9}$$

where $F_{Level}(P_H)$ is the content popularity level of the highest pobility P_H, which generally is 1. The maximum value of the $Interval(P_i)$ equals to $M - 1$, the minimum value of the $Interval(P_i)$ is 0.

Vehicle sets Θ_{repub} and $\Theta_{vehicle1}$ are more involved in the cache sharing. These information can be used to reduce the value of "interval". Thus, the "Interval" value $Interval(P_i^a)$ in these vehicles can be amended as follows.

$$\begin{aligned} Interval(P_i^a) &= \max(interval(P_i) + \lambda, 0), \\ &\lambda < 0, a \in \{\Theta_{repub} \cup \Theta_{vehicle1}\} \end{aligned} \tag{10}$$

where λ is a negative integer. Meanwhile, vehicle set $\Theta_{vehicle3}$ less participates in the cache sharing. $Interval(P_i^a)$ can be amended as follows

$$Interval(P_i^a) = Interval(P_i) + \gamma, \quad \gamma > 0, \ a \in \Theta_{vehicle3} \tag{11}$$

where γ is a positive integer. The improved scheme can reduce the cache on the vehicle storage space, while allowing the most useful contents are cached.

When the data packet is forwarded hop-by-hop, the vehicle will cache once for every other "Interval" hops. The "Interval" value $Interval(P_i^a)$ can be calculated according to the mechanisms introduced in this subsection. Then, the data packet will be cached with the caching probability $Prob(U_{CSPC})$ calculated in Eq. 6.

D. POPULARITY PREDICTION-BASED COOPERATIVE CACHE REPLACEMENT IN VEHICLES

In this subsection, we use the Popularity Prediction-based Cooperative Cache Replacement to decrease the cache replacement overhead. We divide the time into continuous periods. A period contains K requests. For a content i, we sort the contents by request frequency. And we use the linear fitting method to predict the content popularity in the $N + 1$ period according to the result of the previous N periods. To simplify the calculation, we use the binary linear regression model as follows.

$$p_i = \beta_{i0} + \beta_i 1 x_{i1} + \beta_i 2 x_{i2} + \mu_i \tag{12}$$

where p_i is an expect of the popularity of content i, μ_i is a random error term, which obeys a normal distribution with zero mean and variance σ^2, that is $\mu_i \sim N(0, \sigma_i^2)$. $\beta_{i0}, \beta_{i1}, \beta_{i2}$ are the undetermined coefficients. Assuming $x_{i1} = 1, 2, 3, \ldots, n$, $x_{i2} = 1^2, 2^2, 3^2, \ldots, n^2$ as a binary linear model, the normal equation matrix coefficient is in the following.

$$A = \begin{Bmatrix} 1 & 1 & 1 \\ 1 & 2 & 2^2 \\ 1 & 3 & 3^2 \\ \vdots & \vdots & \vdots \\ 1 & n-1 & (n-1)^2 \\ 1 & n & n^2 \end{Bmatrix} \tag{13}$$

Let $Y_i = (y_{i1}, y_{i2}, y_{i3}, \ldots y_{in})^T$, where Y_i is the request number of content i in n period. Let $Y = \{Y_i , i = 1, 2, \ldots\}$, where Y denotes the set of request numbers for all the contents. We assume $\beta = \{\beta_{i0} , \beta_{i1} , \beta_{i2}\}^T$. Since the random error term μ_i and the variable x_{i1}, x_{i2} are irrelevant, we have the equation as follows.

$$\begin{aligned} Y_1 &= \beta_{10} + \beta_{11}x_{11} + \beta_{12}x_{12} \\ Y_2 &= \beta_{20} + \beta_{21}x_{21} + \beta_{22}x_{22} \\ &\cdots\cdots \\ Y_n &= \beta_{n0} + \beta_{n1}x_{n1} + \beta_{n2}x_{n2} \end{aligned} \tag{14}$$

The above formula can be simplified as follows.

$$\begin{aligned} Y &= X\beta = A\beta \\ \Leftrightarrow \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 2^2 \\ \vdots & \vdots & \vdots \\ 1 & n & n^2 \end{bmatrix} \begin{bmatrix} \beta_{10} & \beta_{20} & \cdots & \beta_{n0} \\ \beta_{11} & \beta_{21} & \cdots & \beta_{n1} \\ \beta_{12} & \beta_{22} & \cdots & \beta_{n2} \end{bmatrix} \end{aligned} \tag{15}$$

If the matrix on both sides are multiplied by A transposition A^T at the same time, the normal equation is:

$$A^T Y = A^T A\beta \tag{16}$$

We assume the inverse matrix of matrix $A^T A$ is $(A^T A)^{-1}$. Let the expect of the β_i be $\bar{\beta}_i = (\bar{\beta}_{i0}, \bar{\beta}_{i1}, \bar{\beta}_{i2})^T$. Then the coefficient matrix $\bar{\beta}$ can be expressed as follows.

$$\bar{\beta}=A^T A^{-1} A^T Y \tag{17}$$

For the content i, the predictability of its content popularity can be expressed as follows.

$$\begin{aligned} \bar{p}_i &= (1, n+1, (n+1)^2)\bar{\beta}_i \\ &= \bar{\beta}_{i0} + (n+1)\bar{\beta}_{i1} + (n+1)^2\bar{\beta}_{i2} \end{aligned} \tag{18}$$

In the above formula, $\bar{p}_i$ is the popularity of the content i in the $n+1$ periods, where $\bar{\beta}_{i0}$, $\bar{\beta}_{i1}$, $\bar{\beta}_{i2}$ are the evaluation factors of the content in the $n+1$ periods.

Normally, the existing CS does not have enough space to cache all contents. When the remaining cache capacity is less than the requirements of the new arrived data packet, the proposed PPCR mechanism deletes the lower popularity content in the CS to cache it based on the its size. The initial content popularity value of the new packet is set to $(P_H + P_L)/2$. However, compared with the Internet, vehicles usually play a consumer instead of a provider in the ICN scenario. The contents cached in the vehicles is not enough to reflect the initial value of content popularity. Furthermore, we propose an improved collaborative cache replacement policy based on popular content prediction (PPCCR). When a content provider or a copy owner returns a data packet, it fills the current content popularity result into the packet, which provides an initialized assignment of content popularity. As shown in Fig. 4, we add the popular content statistics field to the packet for the forwarder to initiate the popularity assignment.

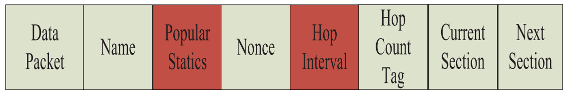

FIGURE 4. Improved data packets in named data networking.

TABLE 2. Simulation parameters.

Parameters	Values
Network size(km)	3×3
Maximum speed of vehicles: (m/s)	13.9
Transmission range: R(m)	300
Content source	50
Data packet	100,000
Cache size	0.5%, 1%, 2%, 4%
Parameter of Zipf distribution	0.6, 0.85, 1.0, 1.2
$(\omega_\alpha, \omega_\beta, \omega_\chi)$	(1/3,1/3,1/3), (1/4,2/4,1/4) (2/4,1/4,1/4), (1/4,1/4,2/4)
(λ, γ)	(1,-1), (2,-1), (1,-2), (0,0)
Cache Replacement Policy	LRU,LFR,PPCR,PPCCR
Simulation time(s)	500

V. SIMULATION RESULTS AND PERFORMANCE ANALYSIS

In this section, we evaluate the proposed caching strategies by conducting the simulations on OMNeT++ simulator [26].

A. SIMULATION ENVIRONMENT

We simulate a simple urban scenario by using SUMO (Simulation of Urban Mobility) software [27] to generate a corresponding map including 4 horizontal roads and 4 vertical roads. The size of the simulation scene is 3KM * 3KM. Simulation parameters are summarized in the Table 2.

Three metrics are used to evaluate our proposed cache policy with other caching strategies:

1) Average Time Delay: The average waiting time between the request packet sent by the content requester and the data packet received in the request packet.
2) Cache Hit Ratio: The ratio of the number of cache hits to the total number of request messages.
3) Cache Hit Distance: The distance required for the request message to be forwarded to the location of the corresponding data packet, usually in hops. The small cache hit distance represents the ability to quickly get a copy of the data packet.

B. CSPC PERFORMANCE ANALYSIS

In the simulation, three cache strategies, LCE (Leave Copy Everywhere) [14], Prob(p) (Caching with Probability) [28] and LCD (Leave Copy Down) [18] are implemented to verify the performance of the cache policies. In LCE cache strategy, every forwarded packet will be cached and we use it as a basic one, that is, the lower bound of performance. Prob(p) uses a fixed probability p to decide whether to cache contents. In the simulation, we choose fixed probability $p = 0.5$. LCD means leaving the copies in the downstream router. Also, we implement the LRU (Least Recently Used) and LFU (Least Frequently Used) cache replacement strategies. When the cache space is full, the LRU strategy replaces the least recently used data, and the LFU replaces the least frequently used data. The other key parameters are set as follows: $(\omega_\alpha, \omega_\beta, \omega_\chi) = (1/3, 1/3, 1/3)$, $(\lambda, \gamma) = (1, -1)$.

To simulate the difference of the vehicle's cache participation, we use four different kinds of vehicles, which discard the received data packet with different fixed probabilities for four privacy levels. These probabilities are 0% for Θ_{repub}, 10% for $\Theta_{vehicle1}$, 20% for $\Theta_{vehicle2}$, and 40% for $\Theta_{vehicle3}$, respectively. The proportions of the number of four kinds of vehicles are 1:1:1:1.

Fig. 5 shows the performance under different cache strategies when the cache size is increased from 0.5% to 4%. The Zipf distribution parameter α is set to 1. As shown in the figure, with the increase of cache size, the average time delay of all cache strategies are shortened, the cache hit distance is decreased and the cache hit ratio is increased. When the cache size is gradually increased, the cached contents are increased in the network. These cached data packets can satisfy directly to subsequent requests for the same content, thereby increasing the cache hit rate. Obviously, the average time delay and cache hit distance will be reduced and shorten without obtaining the content from the original content source.

We compare the average time delay of different cache strategies based on LRU and LFU. As shown in Fig. 5(a) and Fig. 5(b), when the cache size is 2%, compared with the cache strategy LCE, Prob(0.5) and LCD, the average time delay of the CSPC cache strategy is reduced by 41.38%, 35.85%, and 23.91% respectively with the LRU replacement strategy. With the LFU replacement strategy, the average time delay of CSPC cache strategy is reduced by 47.55%, 41.61% and 25.39% respectively. As shown in Fig. 5(c) and Fig. 5(d), when the cache size is 2%, compared with the cache strategy LCE, Prob (0.5) and LCD with the LRU replacement strategy, the cache hit rate of CSPC strategy increases by 122.28%, 67.49%, 61.90% respectively, and the cache hit distance has shortened of 26.31%, 27.96% and 9.55% respectively.

Fig. 6 shows the average time delay for different cache strategies with increasing the parameter α of the Zipf distribution from 0.6 to 1.2. Here, the cache size is 2%. As shown in the figure, the average time delay of all strategies is shortened as the parameter of the Zipf distribution increases. Because the user's preference for the content, i.e. the content popularity, is subject to the Zipf distribution. When the parameter α

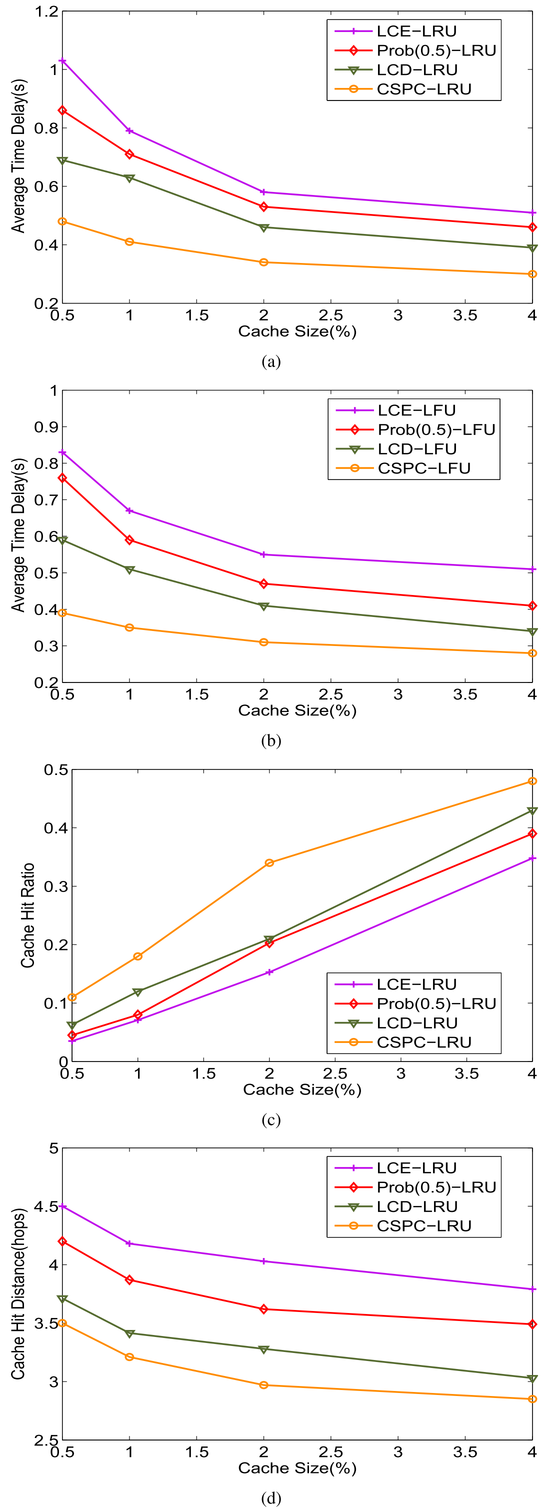

FIGURE 5. The effect of Cache size for different cache strategies. (a) Average time delay with LRU. (b) Average time delay with LFU. (c) Cache hit ratio. (d) Cache hit distance.

of the Zipf distribution increases, the user's requests will be more concentrated. The higher the popularity of the content, the more the number of corresponding requests. These popular contents are cached with higher probability for longer

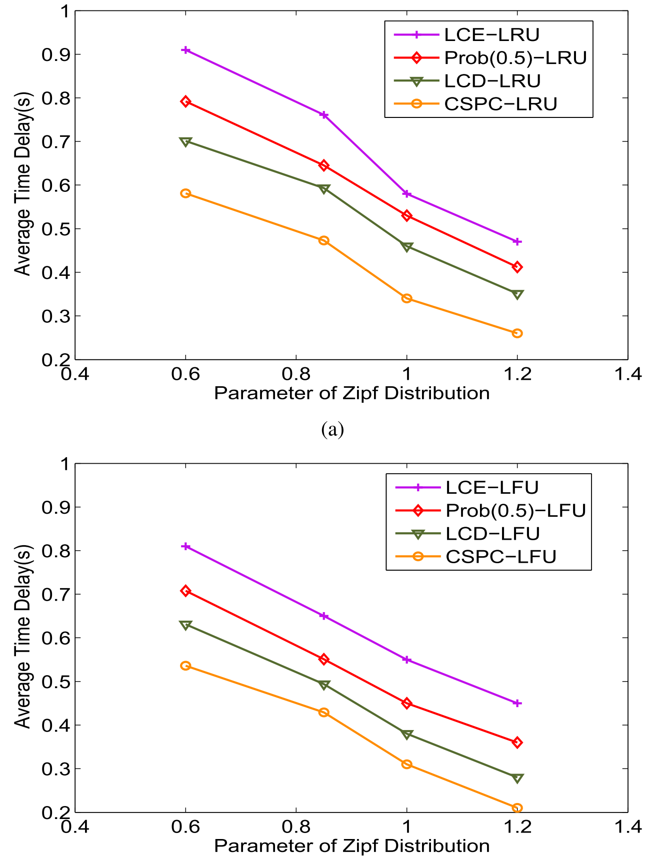

FIGURE 6. The effect of Parameter of Zipf Distribution for different cache strategies. (a) Average time delay with LRU. (b) Average time delay with LFU.

time, thus the corresponding requests can be more quickly responded to. From Fig. 6(a)), when the parameter $\alpha = 0.85$, with the LRU replacement strategy, the average time delay of CSPC policy reduces by 37.84%, 26.67%, 20.23% compared with the cache strategy LCE, Prob (0.5) and LCD respectively. With the LFU replacement strategy, the average time delay of CSPC policy reduces by 34.05%, 22.14%, 13.15% respectively as shown in Fig. 6(b)).

As shown in Fig. 7, we evaluate the performance of the PPCR and PPCCR cache replacement policies mentioned in Section IV-D. Besides of the LRU and LFU mechanisms, we have also implemented these two mechanisms in the simulation software. The difference between PPCR and PPCCR is that the popularity prediction in the PPCR is only based on the vehicle itself, and the content popularity is initialized as the average value. In the PPCCR, the source node adds the popular information field to the data packet, which provides the initial assignment for the popularity statistics of the forwarding vehicle. It is a cooperative alternative caching mechanism. By the way, here we use the cache strategy CSPC and set $\alpha = 1$.

Fig. 7 shows the performance of four mechanisms. With the increase of cache size, the PPCCR mechanism has obvious advantage compared with LRU and LFU. The cache hit ratio of the PCCR mechanism is 29.42% and 15.73% higher than that of the LRU and LFU respectively from Fig. 7(a). The cache hit distance is reduced by 19.18% and 13.69% respectively from Fig. 7(b). The average time delay is reduced by 27.68% and 21.97% respectively shown in Fig. 7(c). Compared with the PPCR, the PPCCR mechanism is better, especially under small cache size. Take Fig. 7(c) as an example, when the cache size is 0.5%, 1%, 2% and 4%, the average time delay of the PPCCR is shorter than that of the PPCR by 13.91%, 8.39%, 2.31%, 2.53% respectively. This is because, when the cache size is small, the cache content replacement is more frequent. Some popular content may be replaced due to the the low initial value of content popularity in PPCR, while PPCCR cache method can make content popularity initial value More in line with the real situation, thereby enhancing the cache performance.

C. THE SELECTION OF CSPC PARAMETERS

In this subsection, we select the main parameters involved in the proposed caching mechanism calculation, and use simulation to analyze their impacts on the performance of the mechanism. The cache strategy used in the simulation is CSPC, and the cache replacement strategy is LRU. The Zipf distribution parameter $\alpha = 1$.

For convenience of description, we use $COE(F(x))$ to represent the series of weight parameters in a certain function expression $F(x)$. For example, the CSPC caching mechanism utility value coefficient series is $COE(U_{CSPC}) = (\omega_\alpha, \omega_\beta, \omega_\chi)$. We take four groups of values to evaluate their impacts on the experimental performance, namely $COE(U_{CSPC})$ is set to $\{(1/3, 1/3, 1/3), (1/4, 2/4, 1/4), (2/4, 1/4, 1/4), (1/4, 1/4, 2/4)\}$ respectively. We also select four sets of parameters $COE(Interval) = (\lambda, \gamma)$ as $\{(1, -1), (2, 1), (1, -2), (0, 0)\}$ respectively to assess its impacts on experimental performance.

Fig. 8 shows the effect of different $COE(U_{CSPC})$ on the performance of the CSPC cache mechanism with the variety of cache size. Here, the parameter $COE(Interval)$ is set to $(1, -1)$. To evaluate the impacts of the three coefficients on performance, we use $COE(U_{CSPC}) = (1/3, 1/3, 1/3)$ as the reference series. As shown in Fig. 8(a), the average time delay in the different series decreases with the increase of the cache size, and the gap with the reference series is also reduced. When the cache size is 0.5%, the series $(1/4, 1/4, 2/4)$ has the best performance of average time delay, which is 11.17% less than the reference series. The following is the series $(1/4, 2/4, 1/4)$ and the average time delay increases by 10.41%. The last series $(2/4, 1/4, 1/4)$ increases the average time delay by 11.87%. When the cache size is 4%, compared with the reference series, the series $(1/4, 2/4, 1/4)$ has the shortest of average time delay, which decreases by 4.38%, while the series $(1/4, 1/4, 2/4)$ decreases by 0.86% and the series $(2/4, 1/4, 1/4)$ increases by 10.27% follows.

Fig. 8(b) shows, with the increase of cache size, the cache hit distance under different $COE(U_{CSPC})$ parameters will

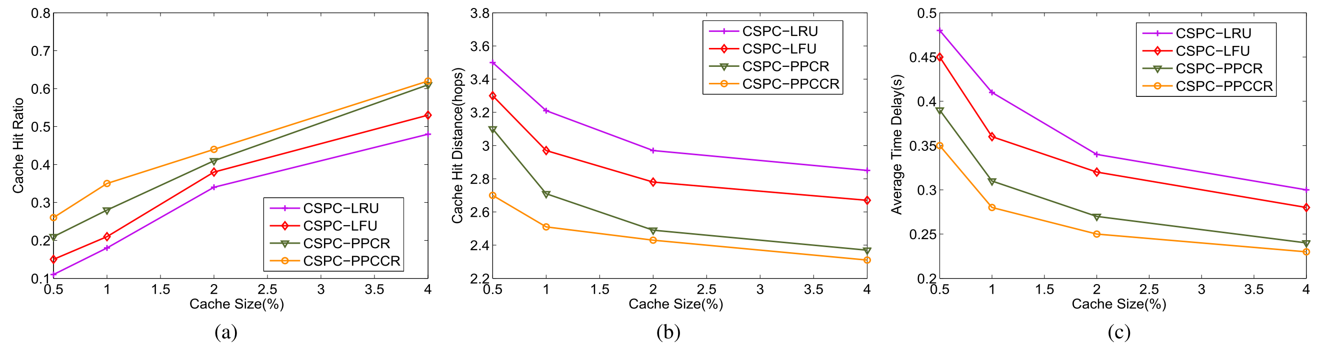

FIGURE 7. **The effect of of cache size for different cache replacement policy. (a) Cache hit ratio. (b) Cache hit distance. (c) Average time delay.**

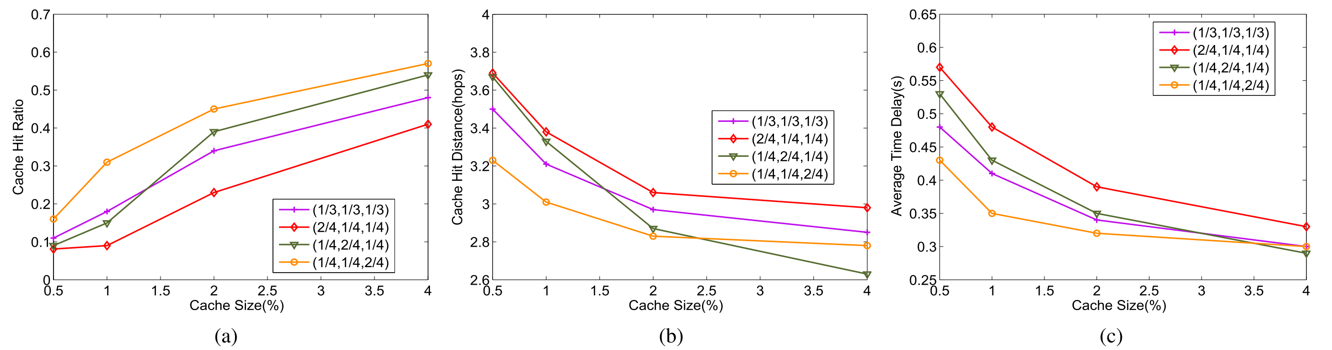

FIGURE 8. **The effect of of cache size for different $COE(U_{CSPC})$. (a) Cache hit ratio. (b) Cache hit distance. (c) Average time delay.**

be shortened, and the gap between reference series is gradually reduced. When the cache size is 0.5%, the series with the lowest cache hit distance is $(1/4, 1/4, 2/4)$. It is 8.17% shorter than that of the reference series. The series $(1/4, 2/4, 1/4)$ and $(2/4, 1/4, 1/4)$ are increased by 4.85% and 5.42% respectively. Different from Fig. 8(a), the cache hit distance of the series $(1/4, 2/4, 1/4)$ is better than the reference series when the cache size is 2%.

Fig. 8(c) shows, with the increase of cache size, cache hit rate of different series will increase, and the gap between reference series is gradually reduced. Similar to Fig. 8(a), when the cache size is 0.5%, the series with the highest cache hit rate is $COE(U_{CSPC}) = (1/4, 1/4, 2/4)$, and it is 45% higher than that of the reference series. The Series $(1/4, 2/4, 1/4)$ and series $(2/4, 1/4, 1/4)$ decreased by 19.19% and 26.37% lower respectively. When the cache size is 4%, the series with the highest cache hit ratio is still $(1/4, 1/4, 2/4)$, and it is 18.73% higher than that of the reference series. The series $(1/4, 2/4, 1/4)$ increases by 12.51% and the series $(2/4, 1/4, 1/4)$ decreases by 15.67%.

Fig. 9 shows the impacts of different $COE(Interval)$ parameters on the cache performance when the cache size changes. Here, the parameters $COE(U_{CSPC})$ are set to $(1/3, 1/3, 1/3)$. As shown in Fig. 9, with the increase in cache size, the average time delay decreases, the cache hit distance reduces, and the cache hit rate increases. The series $COE(U_{CSPC}) = (2, -1)$ has the best performance, followed by $(1, -2), (1, -1), (0, 0)$. As shown in Fig. 9(a), when the cache size is 1%, compared to the series $(1, -2)$, $(1, -1)$ and $(0, 0)$, the average time delay of the series (2, -1) reduces by 7.90%, 14.73% and 27.66% respectively. The cache hit distance reduces by 6.43%, 9.22% and 15.9% as shown in Fig. 9(b). And the cache hit ratio increases by 95.58%, 127.76% and 73.33% respectively as shown in Fig. 9(c).

The different values of the formula coefficients $COE(U_{CSPC})$ represent the contributions of different parameters to the cache probability. For example, a larger ω_χ in the series $COE(U_{CSPC}) = (1/4, 1/4, 2/4)$ represents a greater impact on the cache probability from the privacy rating. Generally, ω_β and ω_χ have more impacts on the cache performance than ω_α. When the cache size is small, the contribution of ω_χ is higher than that of ω_β. When the cache space is increased, the contribution of the ω_β is approached to ω_χ, or even better than ω_β. Obviously, in the case of limited cache size, the number of times a vehicle participates in caching sharing plays a important role in caching mechanism. As the cache size increases, the higher the degree of similarity between cache locations, the better the cache performance. The cache hit distance is geographically affected by the position of the vehicle, thus the more similar the moving path between vehicles, the smaller the cache hit distance. Similarly, when $COE(Interval) = (2, -1)$, the performance of the cache mechanism is optimal, and the contribution of λ to the cache performance is higher than γ. So, allowing

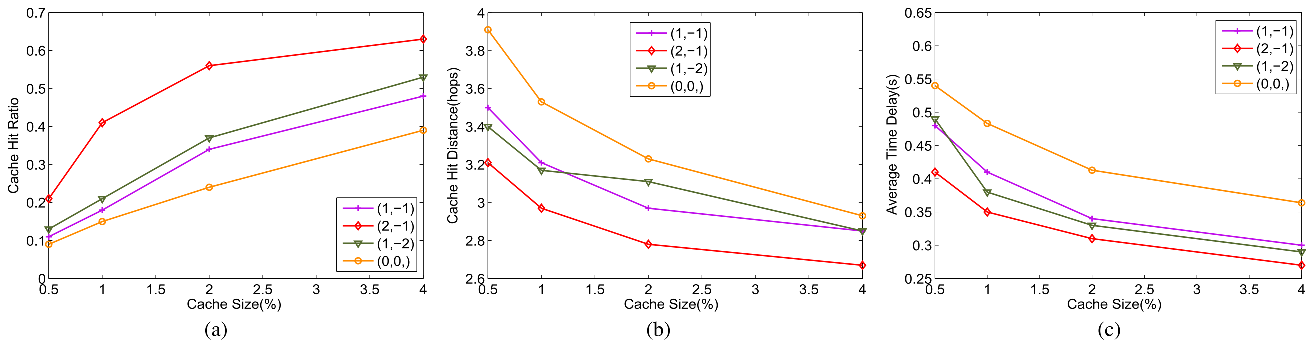

FIGURE 9. **The effect of of cache size for different *COE*(*Interval*). (a) Cache hit ratio. (b) Cache hit distance. (c) Average time delay.**

more "reliable" vehicles to cache more popular content can effectively improve the cache performance. Meanwhile, $COE(U_{CSPC}) = (0, 0)$ achieves the lowest performance.

VI. CONCLUSIONS

In-network caching can decrease the burden of original content source and improve the response speed in vehicular ICN framework. By caching the data packets in vehicular unused storage space, the vehicles can obtain the cached content from other vehicles instead of original content provider. In this paper we propose the CSPC cache policy in an ICN vehicle-to-vehicle (V2V) scenario. It evaluates the community similarity and privacy rating of vehicles, and selects the caching vehicle based on content popularity to reduce the cache redundancy. Additionally, we put forwards the PPCCR cache replacement mechanism to decrease the overhead. The PPCCR predicts and ranks popular contents in a period of time, then adds content popularity statistics field into the data packets. The simulation results show that the prosed schemes reduce the average time delay, increases the cache hit ratio and reduces the cache hit distance compared with several typical cache mechanisms and cache replacement strategies.

REFERENCES

[1] M. Amadeo, C. Campolo, and A. Molinaro, "Priority-based content delivery in the Internet of vehicles through named data networking," *J. Sensor Actuat. Netw.*, vol. 5, no. 4, pp. 1–17, 2016.

[2] M. Mangili, F. Martignon, S. Paris, and A. Capone, "Bandwidth and cache leasing in wireless information-centric networks: A game-theoretic study," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 679–695, Jan. 2017.

[3] T. D. Lagkas, D. G. Stratogiannis, and P. Chatzimisios, "Modeling and performance analysis of an alternative to IEEE 802.11e hybrid control function," *Telecommun. Syst.*, vol. 52, no. 4, pp. 1961–1976, Apr. 2013.

[4] S. H. Ahmed, S. H. Bouk, M. A. Yaqub, D. Kim, H. Song, and J. Lloret, "CODIE: Controlled data and interest evaluation in vehicular named data networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 3954–3963, Jun. 2016.

[5] M. Amadeo, C. Campolo, and A. Molinaro, "Information-centric networking for connected vehicles: A survey and future perspectives," *IEEE Commun. Mag.*, vol. 54, no. 2, pp. 98–104, Feb. 2016.

[6] M. A. Yaqub, S. H. Ahmed, S. H. Bouk, and D. Kim, "Interest forwarding in vehicular information centric networks: A survey," in *Proc. ACM Symp. Appl. Comput.*, Pisa, Italy, Apr. 2016, pp. 724–729.

[7] F. Modesto and A. Boukerche, "A novel service-oriented architecture for information-centric vehicular networks," in *Proc. 19th ACM Int. Conf. Modeling, Anal. Simulation Wireless Mobile Syst.*, Malta, Nov. 2016, pp. 136–139.

[8] T. Lagkas and P. Chatzimisios, "AWPP: A new scheme for wireless access control proportional to traffic priority and rate," *EURASIP J. Wireless Commun. Netw.*, vol. 2011, no. 1, pp. 1–11, 2011.

[9] J. A. Khan and Y. Ghamri-Doudane, "SAVING: Socially aware vehicular information-centric networking," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 100–107, Aug. 2016.

[10] R. Ding, T. Wang, L. Song, Z. Han, and J. Wu, "Roadside-unit caching in vehicular ad hoc networks for efficient popular content delivery," in *Proc. IEEE Wireless Commun. Netw. Conf.*, New Orleans, LA, USA, Mar. 2015, pp. 1207–1212.

[11] Z. Yan, S. Zeadally, and Y.-J. Park, "A novel vehicular information network architecture based on named data networking (NDN)," *IEEE Internet Things J.*, vol. 1, no. 6, pp. 525–532, Dec. 2014.

[12] C. Bian, T. Zhao, X. Li, and W. Yan, "Boosting named data networking for efficient packet forwarding in urban VANET scenarios," in *Proc. 21st IEEE Int. Workshop Local Metropolitan Area Netw.*, Beijing, China, Apr. 2015, pp. 1–6.

[13] G. Grassi, D. Pesavento, G. Pau, R. Vuyyuru, R. Wakikawa, and L. Zhang, "VANET via named data networking," in *Proc. IEEE Conf. Comput. Commun. Workshops*, Toronto, ON, Canada, Apr. 2014, pp. 410–415.

[14] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proc. 5th Int. Conf. Emerging Netw. Experim. Technol.*, Rome, Italy, Dec. 2009, pp. 1–12.

[15] Z. ul Abidin Jaffri, Z. Ahmad, and M. Tahir, "Named data networking (NDN), new approach to future Internet architecture design: A survey," *Int. J. Informat. Commun. Technol.*, vol. 2, no. 3, pp. 155–164, 2013.

[16] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Commun. Mag.*, vol. 50, no. 7, pp. 26–36, Jul. 2012.

[17] M. Zhang, H. Luo, and H. Zhang, "A survey of caching mechanisms in information-centric networking," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, pp. 1473–1499, 3rd Quart., 2015.

[18] N. Laoutaris, H. Che, and I. Stavrakakis, "The LCD interconnection of LRU caches and its analysis," *Perform. Eval.*, vol. 63, no. 7, pp. 609–634, Jul. 2006.

[19] V. Sourlas, P. Flegkas, G. S. Paschos, D. Katsaros, and L. Tassiulas, "Storage planning and replica assignment in content-centric publish/subscribe networks," *Comput. Netw.*, vol. 55, no. 18, pp. 4021–4032, Dec. 2011.

[20] W. Wong, L. Wang, and J. Kangasharju, "Neighborhood search and admission control in cooperative caching networks," in *Proc. IEEE Global Commun. Conf.*, Dec. 2012, pp. 2852–2858.

[21] G. Mauri, M. Gerla, F. Bruno, M. Cesana, and G. Verticale, "Optimal content prefetching in NDN vehicle-to-infrastructure scenario," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2513–2525, Mar. 2017.

[22] W. Quan, C. Xu, J. Guan, H. Zhang, and L. A. Grieco, "Social cooperation for information-centric multimedia streaming in highway VANETs," in *Proc. IEEE 15th Int. Symp. World Wireless, Mobile Multimedia Netw.*, Jun. 2014, pp. 1–6.

[23] T. Chen, L. Zhu , F. Wu, and S. Zhong, "Stimulating cooperation in vehicular ad hoc networks: A coalitional game theoretic approach," *IEEE Trans. Veh. Technol.*, vol. 60, no. 2, pp. 566–579, Feb. 2011.

[24] W. Zhao, Y. Qin, and Y. Cheng, "An efficient downloading service of large popular files in VANET based on 802.11p protocol," *Int. J. Distrib. Sensor Netw.*, vol. 1, no. 1, pp. 1–8, Jul. 2015.

[25] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in *Proc. 18th Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, vol. 1. Mar. 1999, pp. 126–134.

[26] D. Eckhoff, C. Sommer, and F. Dressler, "On the necessity of accurate IEEE 802.11P models for IVC protocol simulation," in *Proc. IEEE 75th Veh. Technol. Conf. (VTC Spring)*, May 2012, pp. 1–5.

[27] *SUMO: Simulation of Urban Mobility*. [Online]. Available: http://sumo.dlr.de/index.html

[28] S. Arianfar, P. Nikander, and J. Ott, "Packet-level caching for information-centric networking," in *Proc. ACM SIGCOMM*, New Delhi, India, Sep. 2010, pp. 1–5.

WEICHENG ZHAO received the B.S. degree from Beijing Jiaotong University in 2011, where he is currently pursuing the Ph.D. degree with the School of Electronic and Information Engineering. His research interests are vehicular networking, information centric networking, and Internet of Things.

YAJUAN QIN received the B.S. and M.S. degrees in electrical engineering from the University of Electronic Science and Technology of China in 1985 and 1988, respectively, and the Ph.D. degree in communication engineering from the Beijing University of Posts and Telecommunications in 2003. In 2002, she was a Research Associate with CRL, Japan. In 2003, she joined the School of Electronic and Information Engineering, Beijing Jiaotong University, where she is currently a Full Professor. Her research interests are in the areas of computer networks and wireless communications.

DEYUN GAO received the B.Eng. and M.Eng. degrees in electrical engineering and the Ph.D. degree in computer science from Tianjin University, China, in 1994, 1999, and 2002, respectively. He spent one year as a Research Associate with the Department of Electrical and Electronic Engineering, The Hong Kong University of Science and Technology. He then spent three years as a Research Fellow with the School of Computer Engineering, Nanyang Technological University, Singapore. In 2007, he joined the faculty of Beijing Jiaotong University as an Associate Professor with the School of Electronics and Information Engineering and was promoted to a Full Professor in 2012. In 2014, he was a Visiting Scholar with the University of California at Berkeley, USA. His research interests are in the area of Internet of Things, vehicular networks, and next-generation Internet.

CHUAN HENG FOH (SM'09) received the M.Sc. degree from Monash University, Australia, in 1999, and the Ph.D. degree from the University of Melbourne, Australia, in 2002. After his Ph.D., he spent six months as a Lecturer with Monash University, Australia. In 2002, he joined Nanyang Technological University, Singapore, as an Assistant Professor until 2012. He is currently a Senior Lecturer with the University of Surrey. He has authored or co-authored over 100 refereed papers in international journals and conferences. His research interests include protocol design and performance analysis of various computer networks, including wireless local area and mesh networks, mobile ad hoc and sensor networks, Internet of Things, 5G networks, and data center networks. He is currently an Associate Editor for the IEEE Access, the IEEE Wireless Communications, and the *International Journal of Communications Systems*. He is the Vice-Chair (Europe/Africa) of the IEEE Technical Committee on Green Communications and Computing.

HAN-CHIEH CHAO received the M.S. and Ph.D. degrees in electrical engineering from Purdue University in 1989 and 1993, respectively. He is a joint appointed Chair Professor with the Department of Computer Science and Information Engineering, Electronic Engineering of National Ilan University, I-Lan, Taiwan, and Fujian University of Technology. He has been serving as the President for National Dong Hwa University since 2016. He was the Director of the Computer Center for Ministry of Education, Taiwan, from 2008 to 2010. His research interests include high speed networks, wireless networks, IPv6-based networks, digital creative arts, e-Government, and digital divide. He is the Editor-in-Chief of the *IET Networks* and the *Journal of Internet Technology*. He is the founding Editor-in-Chief of the *International Journal of Internet Protocol Technology* and the *International Journal of Ad Hoc and Ubiquitous Computing*. He is also the Associate Editor of the IEEE *Network Magazine* and the IEEE *Wireless Communications Magazine*.