Towards Energy-Aware Caching for Intelligent Connected Vehicles

Hongjia Wu, **Jiao Zhang**, **Zhiping Cai***, **Fang Liu***, **Yangyang Li** and **Anfeng Liu**

***Abstract*—With the widespread application of infotainment services in intelligent connected vehicles (ICV), network traffic has grown exponentially, bringing huge burden and energy consumption to ICV network. Edge caching, which enables edges (e.g., vehicles or roadside units) with cache storages, is a promising technology to alleviate this problem. In this paper, in terms of the hybrid communication mode of vehicle to vehicle (V2V) and vehicle to roadside unit (V2R), an energy aware caching scheme for infotainment services is proposed. Considering the geographical distribution of vehicles and roadside units as well as the size of transmission content, the energy consumption model in ICV network is formulated to implement the optimal selection of cache nodes. Then the selection of cache node in ICV network is transformed into the optimal stopping problem and solved by the optimal stopping theory. Finally, we propose a new algorithm for optimal energy efficiency cache node selection (OEECS). Simulation results show that the proposed OEECS can obtain higher energy saving and lower average access latency than other baseline schemes.**

***Index Terms*—Intelligent connected vehicles, infotainment services, edge caching, energy consumption, optimal stopping theory.**

I. Introduction

With the rapid development of the Internet of things (IoT) [1], [2] and artificial intelligence (AI) technologies [3], the intelligent connected vehicles (ICV) [4] are emerging and gaining enormous popularity. Gartner has predicted [5] that by 2020, in order to transmit and share information, around one in five vehicles (i.e., more than 250 million) on the road will be globally connected to the Internet. The resulting large-scale data generation challenges the performance of ICV network, which may lead to more energy consumption and affect the experience of ICV users [6]. A promising approach is to use edge caching [7], [8] to react to these challenges in vehicle networks. The cache service model of ICV can receive the inspiration from the mobile edge computing (MEC) architecture [9]–[12] to cache content at the edge of the network, so as to provide better services for vehicle users.

Nowadays, some caching schemes adopt mobility prediction [13]–[15] and cooperative caching [16], [17] to reduce network latency and increase network throughput. The others use additional tools [18], [19], such as caching assistants or unmanned aerial vehicle (UAV) to improve the quality of service (QoS), but with additional overhead. The utilization of the above caching methods mainly focus on the improvement of latency for security applications [20]. They can speed up users' access to real-time information about road conditions and route navigation, thereby reducing congestion and accidents. However, with the enrichment of people's requirement, the infotainment services [21] in ICV are getting more attention. The emerging services can provide a variety of entertainment and leisure information services, such as video, news, tourist attractions query and more, aiming to improve users' travel convenience and experience. Due to the high popularity of infotainment services in ICV, it will be frequently requested, which leads to the increase of network traffic and huge energy consumption. Moreover, due to the mobility of vehicles, the network cannot guarantee the stable transmission of infotainment with low energy consumption. In addition, driven by the construction of green networks, electric vehicles are becoming more and more popular, which enables the energy consumption problem in ICV networks more prominent and cannot be ignored [22]. In view of the above problems, the design of an energy efficient dynamic caching scheme for infotainment services is of great significance in ICV network.

In this work, we focus on energy saving in ICV network. An energy aware caching scheme for ICV is proposed, aiming to optimize energy consumption while guaranteeing the user quality of service. The main contributions of this paper are as follows:

- From the perspective of vehicle users, a cache integration framework considering end-edge-cloud collaboration is proposed.
- According to the optimal stopping theory, we transform the dynamic cache node selection problem into an optimal stopping problem, which takes the mobility of nodes and the network energy consumption into account.
- Considering the limited energy, an optimal energy efficient cache node selection algorithm for infotainment services is proposed, which is capable of improving the energy conservation of ICV network.

The remainder of this paper is organized as follows: In

Hongjia Wu and Zhiping Cai (corresponding author) are with the College of Computer, National University of Defense Technology, Changsha, Hunan, CN 410073 Email: zpcai@nudt.edu.cn.

Jiao Zhang is with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha, CN.

Fang Liu (corresponding author) is with the School of Design, Hunan University, Changsha, Hunan, CN 410073 e-mail: liufang06@gmail.com.

Yangyang Li is with National Engineering Laboratory for Public Safety Risk Perception and Control by Big Data, Mobile Internet Development and Research Center China Academy of Electronics and Information Technology, Beijing, CN.

Anfeng Liu is with School of Computer Science and Engineering, Central South University, Changsha, CN.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2980954, IEEE Internet of Things Journal

Section II, we introduce the application of caching in ICV network. We construct an energy consumption model in Section III. To minimize the energy consumption of the system, an energy aware caching scheme is proposed in Section IV. In Section V, we present and analyze simulation results. We finally conclude the paper in Section VI.

II. Related Works

Nowadays, the existing works in caching are focused on hops [23]–[25], mobility prediction [13]–[15], roadside units (RSU) [26]–[28], cooperative caching [16], [17] and auxiliary [18], [19]. The most representative caching strategy was Leave Copy Down (LCD) [23], which has the simple idea of selecting the next hop of the content source as the cache placement point. The authors in [24] proposed a Random cache placement algorithm. The Random strategy was improved on the basis of LCD, realizing random equal probability selection. Psaras et al. proposed a Probcache strategy [25], where the cache nodes were selected with weighted probabilities. In this case, the nodes closer to the users are more likely to become cache placement points. Based on these classic caching ideas, new caching strategies are constantly being proposed. Mahmood et al. [13] proposed a mobility-aware probabilistic caching scheme by predicting the probability for content required at each edge node. They studied how to efficiently stream data to connected vehicles on roads covered by edge nodes. The authors [14] proposed a non-cooperative caching scheme which used the user mobility pattern and daily demand to minimize the total network latency. Considering the mobility of vehicles, Tan et al. [15] designed resource allocation strategies to improve the performance and cost-effectiveness of vehicle networks.

Since, the prediction of mobility faces much uncertainty, and the stability of performance can be improved by selecting RSU as edge caching node. The authors [26] solved the files allocation problem in three algorithms, such as the optimal, sub-optimal, and greedy methods respectively. They addressed the content delivery problem by caching popular files with large storage capacity in RSUs, while ignoring the number and limited capacity of RSUs. In order to improve the throughput of vehicle network, a caching scheme was proposed by Bitaghsir [27], which deployed some storage-capable RSUs on the street to store the content. Ma et al. [28] proposed a caching allocation policy which jointly considered the caching at the vehicular layer and RSU layer, aiming to minimize average latency while meeting service quality requirements. However, the above works only consider a single edge caching node, resulting in a limited amount of cached data, thus the collaboration of multiple edge caching nodes becomes necessary. Cooperative content caching between moving vehicles was introduced in [16]. Attia et al. analyzed the performance of cooperative content caching in vehicular ad hoc networks. A distributed cooperative caching scheme [17] was proposed, in which RSUs in an area periodically shared their contents so as to update their cache locally. In addition, network performance can be effectively improved by using additional tools. Abdelhamid et al. [18] proposed a caching scheme called caching-assisted data delivery, which introduced a lightweight Road Cache Point as a caching assistant on the road. Chen et al. [19] proposed a novel cache-enabled UAV framework in CRANs which effectively deployed cache-enabled UAVs, aiming to maximize users' QoS. In [26] and [27], although the caching aids were employed, the additional device overhead was ignored.

The above researches mainly consider latency sensitive information, and focus on the optimization of network latency or throughput. The characteristics of the related caching strategies are shown in TABLE I. On one hand, few caching schemes pay attention to infotainment which has no strict information requirements for latency. On the other hand, most authors tend to focus too much on latency and ignore the importance of energy consumption in the network. Different from these works, we propose an energy-aware caching scheme for infotainment services in a V2V and V2R hybrid mode. The goal is to minimize network energy consumption under the constraint of maximum latency.

III. System Model And Problem Formulation

This section gives a detailed description of the system model. We first present the network application scenario of ICV, and then introduce the network energy consumption model based on the hybrid communication mode.

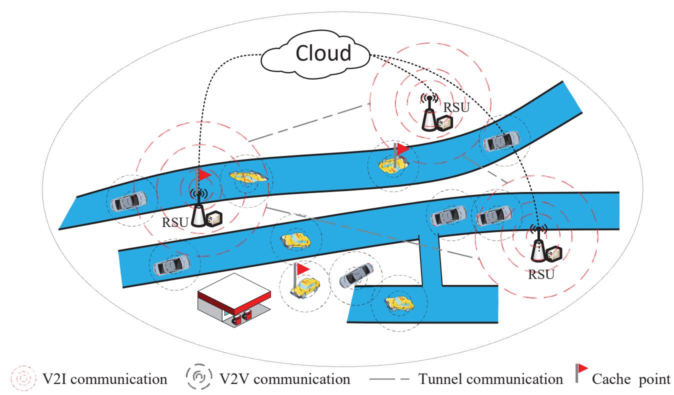

Fig. 1: The ICV network architecture.

A. Network Scenario

We consider a highway traffic scenario, as shown in Fig. 1. In the highway, vehicles are assumed to travel in platoon and RSUs are deployed uniformly along the road with the same coverage [28]. For this scenario, we propose a cache system architecture, which consists of three layers: cloud (data center), edge (RSU nodes), and end (vehicle nodes). Some vehicles have caching capability (candidate cache node) and the rest have no caching capability (vehicle users). The candidate cache nodes can be decided whether to be a cache node by the vehicle users within its communication range. Each RSU has a powerful cache server, and the cloud decides whether to use them as a cache point or not. For vehicles and RSUs, if the content has been stored, they can be reused. In this paper, the global positioning system (GPS) is used to precisely measure the geographic location of vehicles and the content with large data volume is divided into fine-grained blocks.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2980954, IEEE Internet of Things Journal

TABLE I: Characteristics of the related caching strategies.

Characteristic	Reference	Optimization goal	Main parameters	Constraint
Mobility prediction	[6]	Cache throughput	Content service time	Network latency
	[7]	Network latency	Cache size	No clear constraints
	[8]	Network cost	Packet size	Hard service deadline
Cooperative caching	[9]	Outage performance	Time slot	Network latency
	[10]	Cache hit ratio	Cache capacity	Movement speed
Auxiliary	[11]	Network cost	Human-centric information	Network latency
	[12]	Energy	Content popularity Time solt	
Hops	[16]	Average hit distance	Cache size	No clear constraints
	[17]	Caching redundancy	Cache capacity	Network latency
	[18]	Hop reduction	Node number	
Roadside units (RSU)	[19]	Average downloading time	Storage capacity and vehicle speed	No clear constraints
	[20]	Cache hit ratio	Vehicle speed and the number of RSUs	Network latency
	[21]	Network latency	RSU's local information	Quality of experience

B. Energy Consumption Model

We adopt V2V and V2R hybrid communication mode, and the nodes access the network through Orthogonal Frequency Division Multiplexing (OFDM) [29]. The set of vehicle users and candidate cache nodes is denoted as $M = \{1, 2...m\}$ and $N = \{1, 2...n\}$, respectively. RSUs use tunnels to communicate with each other.

Let $\phi_i(t)$ be the transmitted signal from user i to user j, then the received signal at user j can be written as [30]

$$S_j(t) = a_{ij}\phi_i(t) + \omega_i(t), \tag{1}$$

where a_{ij} is the channel attenuation factor. $\omega_i(t) \sim N(0, 1)$ is the random white Gaussian noise with mean 0 and variance σ^2. The highest signal to noise ratio (SNR) of the communication link can be expressed as

$$f = G_T/\sigma^2, \tag{2}$$

where G_T is the transmitted signal power at node i. Through the wireless channel transmission, the corresponding power of received signal at node j can be obtained as

$$G_R = G_T a_{ij}^2 = f\sigma^2 a_{ij}^2, \tag{3}$$

where $a_{ij} = \lambda/d_{i\to j}$. $d_{i\to j}$ is the distance between node i and node j and λ is a constant. Thus G_R can be further extended as

$$G_R = \frac{f\sigma^2\lambda^2}{d_{i\to j}^2}. \tag{4}$$

According to the Shannon-Hartley formula [31], in order to transmit the cache content, the uplink transmission rate [32], [33] between nodes can be denoted as

$$R = Blog_2\left(1 + \frac{G_T H}{\sigma^2}\right), \tag{5}$$

where B is the bandwidth. H stands for the channel gain between the nodes.

When the content of c MB file is transferred, the transmitted energy consumed by node i and the received energy consumed by node j are respectively defined as

$$e_i = \frac{c}{R}G_T, \tag{6}$$

$$e_j = \frac{c}{R}G_R = \frac{cf\sigma^2\lambda^2}{Rd_{i\to j}^2}. \tag{7}$$

Combining (6) and (7), we can obtain the total energy consumption for a content transmission between nodes.

$$\begin{aligned} e_{i\to j} &= \frac{c}{R}(G_T + G_R) \\ &= \frac{c}{R}(G_T + \frac{f\sigma^2\lambda^2}{d_{i\to j}^2}). \end{aligned} \tag{8}$$

In conclusion, we find that the energy consumed for communication between nodes is proportional to the size of the transmitted content. However, the transmission rate and distance between two nodes have an inversely effect on the communication energy consumption.

C. Optimal Stopping Theory

The optimal stopping theory is based on the continuous observation of random variables, and the decision maker chooses an appropriate moment to take a given behavior with the goal of maximizing the reward [34]. The optimal stopping rule problem is defined by the following two types of objects:

(1) Suppose the random variable sequence obeys joint distribution: $X_1, X_2, \cdots$;

(2) Sequence of reward functions can be denoted as: $y_0, y_1(x_1), y_1(x_1, x_2), \cdots, y_\infty(x_1, x_2, \cdots)$.

The associated stopping rules are detailed as [35]: After observing $X_1 = x_1, X_2 = x_2, \cdots, X_n = x_n\ (n = 1, 2, \cdots)$, the decision maker chooses to stop observing and accepts the known reward $y_n(x_1, \cdots, x_n)$, or to continue observing X_{n+1}. If no satisfied node is observed, the decision maker accepts the constant y_0. This rule enables the decision maker to select the optimal stopping time $n (0 \leq n \leq \infty)$ to maximize the expected reward $E[Y_n]$. Among them, $Y_n = y_n(x_1, \cdots, x_n)$ is the random reward of stopping at time n.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2980954, IEEE Internet of Things Journal

D. Problem Transformation

In this paper, we continuously detect the distance between the first candidate cache node and each vehicle user within its communication range. Then the energy consumption sequence of the first candidate cache node is obtained. Next, the vehicle users continue to detect the second candidate cache node as described above. Finally, the energy consumption corresponding to the second candidate cache node is subtracted from that of the first one, and the value is compared with the energy saving expectation solved by the optimal stopping theory. Based on the comparison results, the vehicle user group determines whether the second candidate cache node is selected as the cache node. Follow the above approach until the optimal cache node is found.

The vehicle users choose the optimal time to stop the detection based on the continuous observations of energy consumption sequence. Under the premise of meeting the maximum latency of the network, the optimal cache placement point is selected to maximize the energy saving. Therefore, the cache node selection problem can be transformed into an optimal stopping problem under the constraint of maximum latency.

In fact, the vehicle users need to decide whether to continue the detection based on the current detected energy consumption and the energy saving expectation. If the decision makers stop the detection, the current candidate cache node is selected as the cache node, or the opportunity is passed and the next candidate cache node is detected. This is an optimal stopping strategy problem, and the corresponding relationship between this problem and the optimal stopping problem is shown in Fig. 2.

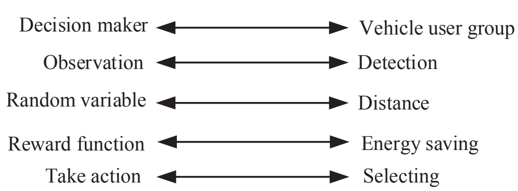

Fig. 2: Optimal stopping elements in cache node selection.

Suppose that the energy consumed by the content transmission constitutes an independent and identically distributed random variable $\{P_{i\to j}\}$. Y_n represents the energy saving of selecting other nodes compared with the first selected node at random. In order to make the model more realistic, it is assumed that each real-time detection consumes a certain amount of energy, which is defined as P_c. So, Y_n can be expressed as

$$Y_n = \sum_{i=1}^{m} P_{i\to 1} - \sum_{i=1}^{m} P_{i\to n} - nmP_c, \tag{9}$$

where m is the number of detection nodes, and mP_c is the total energy consumed by m nodes in one detection. In the detection process, when n candidate nodes are detected, the algorithm's stopping time is also expressed as n.

Assume that the detection node group is taken as a whole, Y_n can be further expressed as

$$\begin{aligned} Y_n &= m(e_{i\to 1} - e_{i\to n}) \\ &= m(\tfrac{c}{R}(G_T + \tfrac{f\sigma^2\lambda^2}{d_{i\to 1}^2}) \\ &\quad -\tfrac{c}{R}(G_T + \tfrac{f\sigma^2\lambda^2}{d_{i\to n}^2}) - nP_c) \\ &= m(\tfrac{cf\sigma^2\lambda^2}{R}\left(1/d_{i\to 1}^2 - 1/d_{i\to n}^2\right) - nP_c) \\ &= my_n, \end{aligned} \tag{10}$$

where y_n represents the energy saving of one detection node

$$\begin{aligned} y_n &= \frac{cf\sigma^2\lambda^2}{R}\left(1/d_{i\to 1}^2 - 1/d_{i\to n}^2\right) - nP_c \\ &= \frac{cf\sigma^2\lambda^2}{R} x_n - nP_c, \end{aligned} \tag{11}$$

where x_n is denoted as

$$x_n = 1/d_{i\to 1}^2 - 1/d_{i\to n}^2. \tag{12}$$

Then, in order to find the optimal stopping time n^* that can obtain the best expected reward, we aims to maximize the expectation of Y_n. The energy saving expectation maximization problem with maximum latency constraint is described as follows

$$\begin{aligned} &max\ E\left(Y_n\right) \\ s.t.\ & n \in N, m \in M \\ & 0 \le P_c \le P_{max},\, P_{i\to j} > 0 \\ & 0 < \tfrac{c}{R} \le T_{max} \end{aligned} \tag{13}$$

where, T_{max} is the maximum time latency allowed by the network. P_{max} is the maximum additional energy consumption required for detection.

When the maximum energy saving is achieved, the optimal stopping time n^* and the cache node j^* can be determined as

$$\begin{aligned} \{n^*, j^*\} = arg\, max E[Y_n], \\ n, j \,\epsilon\, N \end{aligned} \tag{14}$$

where, j^* is determined by the corresponding n^*.

The maximum reward function of the energy saving problem is shown in equation (13). $d_{i\to 1}$ is a fixed value and $d_{i\to n}$ is a random variable, assuming that it obeys uniform distribution.

To sum up, the selection of cache nodes in the ICV network has been transformed into the optimal stopping problem.

IV. An Energy-Aware Caching Scheme

In this section, we propose an optimal energy efficient cache node selection algorithm based on the optimal stopping theory. The objective of our algorithm is to find the optimal cache nodes with lower energy consumption.

A. Problem Solution

Before solving the problem, we first prove that there is an optimal solution for the energy saving expectation maximization problem.

Proposition 1: Equation (9) has an optimal stopping rule.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2980954, IEEE Internet of Things Journal

According to [36], the optimal stopping rule exists when the following two conditions are satisfied.

$$\begin{aligned} &A1: E\{\sup_n Y_n\} < \infty, \\ &A2: \limsup_{n\to\infty} Y_n \le Y_\infty. \end{aligned} \tag{15}$$

According to the definition of Y_n, we can easily find $limsup_{n\to\infty}Y_n{\le}{-}\infty$, and $Y_\infty{=}{-}\infty$, so $limsup_{n\to\infty}Y_n \le Y_\infty$ and A2 is proved. Meanwhile, for any $n\epsilon N_+, sup_n Y_n < \infty$, so $E\left\{sup_n Y_n\right\} < \infty$ must satisfy the condition, and A1 is proved.

When $X_n < W^*$, the candidate cache node needs to be replaced, and the detection node group continues to detect other nodes. Otherwise, the detection node group should stop detecting and select the current candidate cache node as the caching node. Therefore, the stopping rule can be changed to

$$n^* = \min\{n \geqslant 1 : \left(1/d_{i\to 1}^2 - 1/d_{i\to n}^2\right) \geqslant W^*\}, \tag{16}$$

where, the energy saving expectation W^* satisfies the following condition

$$W^*{=}E[\max(1/d_{i\to 1}^2, W^*)] - P_c. \tag{17}$$

According to the optimal stopping formula of Theorem 3.1 in [37], the solution method of stopping rule W^* is

$$\begin{aligned} W &= E[\max(1/d_{i\to 1}^2, W^*)] - P_c \\ &= \int_0^{W^*} W^* \cdot dF(x) + \int_{W^*}^{l} x \cdot dF(x) - P_c. \end{aligned} \tag{18}$$

Meanwhile, it can also be expressed as

$$W = \int_0^l W^* \cdot dF(x). \tag{19}$$

Combining (18) with (19), we can get

$$\begin{aligned} &\textstyle\int_0^l W^* \cdot dF(x) = \int_0^{W^*} W^* \cdot dF(x) + \int_{W^*}^l x \cdot dF(x) - P_c \\ &\Rightarrow \textstyle\int_0^{W^*} W^* \cdot dF(x) + \int_{W^*}^l x \cdot dF(x) - \int_0^l W^* \cdot dF(x) = P_c \\ &\Rightarrow \textstyle\int_{W^*}^l (x - W^*) dF(x) = P_c, \end{aligned} \tag{20}$$

where, F is the distribution function of x_n, which is assumed to be uniform distribution in this paper. For the discrete random variable x_n, we have

$$E(x_n - W^*)^+{=}P_c, \tag{21}$$

thus, the optimal solution to the problem would be obtained as

$$W^* = E(x_n) - P_c. \tag{22}$$

Finally, the decision rule $N(W)$ can be determined as

$$N(W) = \begin{cases} 0, & W < W^* \\ 1, & W \geqslant W^*, \end{cases} \tag{23}$$

where, $N(W)$=0 means to continue detection. Otherwise, $N(W)$=1 means to stop detection, and select the current node as the cache node.

Intuitively, W^* is the critical value of the optimal stopping rule. It is affected by two network parameters: the communication distance between different nodes and the energy required for detection. According to formulas (12) and (22), reducing the distance between nodes and the energy consumption for detection will make W^* larger. The larger the W^* is, the greater the reward the selected cache node can bring.

B. Optimal Energy Efficiency Cache Node Selection Algorithm

According to the stochastic rule of the optimal stopping theory, we randomly select the first candidate cache node v_1. The detection group composed of vehicle users within the communication range detects the distance sequence to v_1 and obtains the corresponding energy consumption. The next candidate cache node v_j is randomly selected, and the detection method is the same as the first node. Second, the energy saving is then calculated by comparing the energy consumption detected at two different nodes. If the energy saving obtained is greater than or equal to the maximum expected energy saving W^*, then node v_j is selected as the cache node and we stop detecting. If no node satisfying the condition is found within the limited detection times, the first node v_1 is selected as the cache node by default. Algorithm 1 provides a formal description of the optimal energy efficiency cache node selection algorithm (OEECS).

Algorithm 1: The OEECS Algorithm

Input: (c, m, r, P_c, R, n, T_{max})
Initialize: $j = 0$;
for $1 \le j \le n$ **do**
  Detecting random variables d_j;
  $stop \leftarrow false$; /* detecting */
  if $(1/d_1^2 - 1/d_j^2) \ge W^* \cap \left(\frac{c}{R} \le T_{max}\right)$ **then**
    $S_j = E_1 - E_j$; /* calculate energy saving */
    if $(S_j > W^*)$ **then**
      $point \leftarrow v_j$;
      $stop \leftarrow true$;
    else
      break;
    end if
  else
    $j = j + 1$;
  end if
end for
if $(!stop \&\& j > n)$ **then**
  $stop \leftarrow true$; /*no satisfied node can be found.*/
  $point \leftarrow v_1$;
end if
Output: $point$ /* cache node */

We assume that the maximum detection times for the optimal cache node is n. In the detection process, the worst case is that no better node is found than the first randomly selected node, and the number of comparisons is n. Therefore, the time complexity of the algorithm is $O\left(n\right)$, indicating that the algorithm has high execution efficiency. The more specific algorithm flow is shown in Fig. 3.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2980954, IEEE Internet of Things Journal

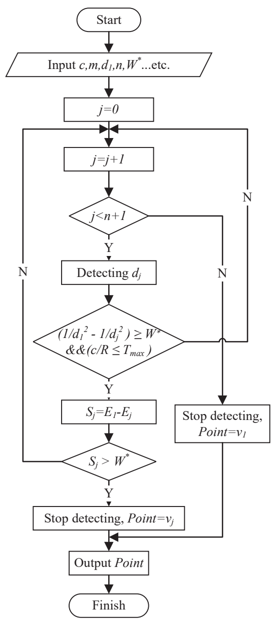

Fig. 3: The flow chart of OEECS.

We divide the ICV network into two main selection sets: 1) edge layer: RSU set; 2) end layer: vehicle set. In this paper, OEECS algorithm is mainly implemented on the end layer, and the number of iterations is selected according to the specific number of nodes. After each iteration, a cache node is selected. At the same time, due to the mobility of the vehicle, the cache node will be re-selected periodically.

The OEECS strategy consists of four main processes that enable users to successfully obtain the content they need. When a vehicle user requests content, the user first accesses the cache node within its communication range in the 'end' layer. If the cache hits, the content is returned directly to the user. Second, the user accesses the nearest RSU node. If the RSU has the content requested by the user, the RSU returns the content directly to the user. Third, the RSU makes a request to the cache node in the 'edge' layer. If the cache hits, the content is returned to the user via the RSU. Finally, the user requests the content from the cloud center, and the cloud returns the content directly to the user.

V. Simulation Results And Discussions

In this section, we perform extensive simulations using MATLAB, a commercial mathematics software produced by MathWorks to evaluate our proposed algorithm. We consider a ICV network covered with a $200m \times 200m$ area, in which three roadsites and several vehicles are randomly scattered over the region. Through parameter settings in TABLE II, we provide a comprehensive simulation to compare the performance of the proposed OEECS algorithm with LCD, Random and Probcache in terms of energy saving, cache efficiency and average access latency. In addition, the delivery success rate of the OEECS is tested and analyzed. The total number of simulations is 10,000.

The three strategies used for comparison in this paper are: (1) Leave Copy Down (LCD): this strategy greatly reduces redundancy on the basis of the classic full cache (LCE), and only selects the next hop of the hit node as the cache node; (2) Random: the cache nodes are selected by random equal probability; (3) Probcache: the node closer to the request node has higher probability to be selected as the cache one.

TABLE II: Simulation parameters.

Symbol	Parameters	Value
m	Number of vehicle users	$100 \sim 1000$
P_c	Detection energy consumption	$1 \sim 10$ J
R	Transmission rate	$5 \sim 50$ MB/s
c	Content size	$10 \sim 100$ MB
σ^2	Noise power	1 W
f	Noise ratio	1 dB
G_T	Transmission power	$1 \sim 3$ W
r	Coverage radius	$2 \sim 20$ KM

A. Energy Saving

Energy saving (ES) is a widely used performance evaluation criterion, calculated by $\sum_{j=1}^{n} Y_j/n$. The parameter Y_j represents the energy saved by different nodes compared with the content obtained by the first randomly selected node.

The effect of P_c on energy saving is shown in Fig. 4. We can see that Probcache and Random are greatly affected by P_c, and there is a certain degree of fluctuation due to the characteristics of their own strategies. OEECS and LCD are less affected by P_c. Among them, the LCD is least affected by P_c. Since the next node of the hit point is selected every time, its detection time is always 1. Due to the method of LCD is designed simply, its energy saving effect is not ideal. For Random, it is a random selection of equal probability, which has a certain probability of selecting detection time, so it is more affected by P_c than LCD. For Probcache, the selection probability is proportional to the number of hops between the current node and the content source, where P_c is also proportional to the number of hops. Therefore, compared with LCD, Probcache is more affected by P_c. It is obvious that OEECS has the highest performance-to-price ratio. Due to the flexibility of detection times and the characteristics of energy consumption optimization, OEECS is less affected by P_c, and the energy saving effect is always higher than the other three strategies.

In Fig. 5, the impact of the number of vehicle users m on energy saving is investigated. As m increases, the four curves show an increasing trend, of which OEECS has the largest increase. By contrast, LCD has the least energy saving, due to the fact that LCD reduces the distance by only one hop. Random and Probcache select cache nodes based on probability and have greater uncertainty, making up for singleness and invariance. so the energy saving achieved by the two schemes is better than LCD. It can be seen from the comparison that OEECS has the greatest energy saving. Because it aims to reduce energy consumption, and consider the total energy saving of all cache nodes from a global perspective. When the number of m increases, the utilization rate of cache nodes increases, which contributes to the energy saving. LCD, Random and Probcache do not consider cache

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2980954, IEEE Internet of Things Journal

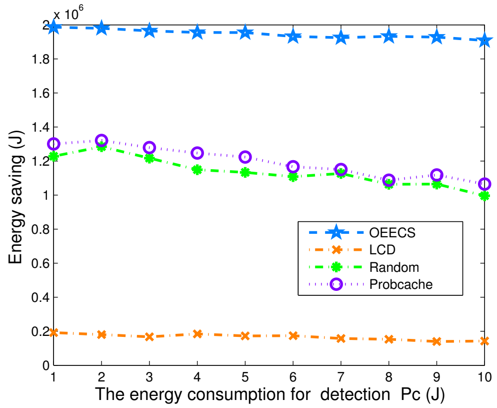

Fig. 4: The effect of P_c on energy saving: m=1000, c=50.

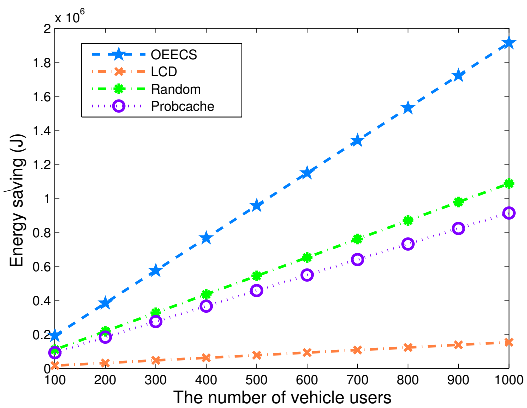

Fig. 5: The effect of the number of vehicle users m on energy saving: c=50, P_c=10.

placement from a global perspective, ignoring the importance of node distribution.

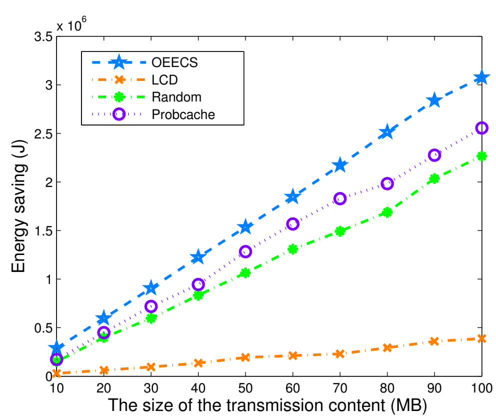

Fig. 6: The effect of c on energy saving: m=1000, P_c=10.

Fig. 6 shows that the larger the content size c is, the more energy is consumed during transmission. In different schemes, OEECS achieves the greatest energy-saving growth rate. In OEECS, c is an important parameter in the reward function. The energy saving increases almost linearly as the content size c grows. Probcache selects the node with high centrality as the cache node, which effectively reduces the transmission distance and is superior to Random and LCD in energy saving. However, Probcache is based on weighted probability selection and there is some uncertainty. Compared with OEECS and Probcache, the growth rate of Random and LCD is slightly lower. Since the selection of cache nodes for Random and LCD is uncertain and exclusive, its effect on energy saving is not prominent.

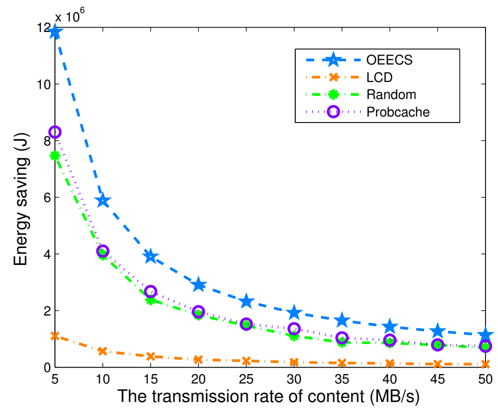

Fig. 7: The effect of R on energy saving: m=1000, c=50, P_c=10.

As shown in Fig. 7, when R is gradually increased, the effect of energy saving is getting worse. When R is very small, Probcache saves more energy than Random. However, as R increases, the values of Random and Probcache get closer and closer. After $R = 20$, they almost overlap. It can be explained that when R is large enough, the efficiency of content transmission is high, the role of the cache becomes smaller, and the advantages of the strategy are difficult to reflect. In contrast, LCD has the smallest energy saving and reduction rate. The results show that when the transmission speed is high enough, the advantage of caching strategy is not obvious. Although the energy saving of OEECS has a downward trend, it still saves the most energy.

To sum up, among the four cache strategies, only OEECS has stronger adaptability to network environment changes and can obtain more energy saving.

B. Cache Efficiency

The cache efficiency (CE) is another widely used performance evaluation criterion that refers to the energy saved by transmitting per unit of content, calculated by $\sum_{j=1}^{n} Y_j / \sum_{j=1}^{n} c_j$. Y_j is the same as described in ES, and the parameter c_j is the size of each transmitted content.

In Fig. 8, we can see that the cache efficiency of the four strategies fluctuates with the increase of the transmitted content size. Although c is constantly changing, OEECS always maintains a high cache efficiency without significant fluctuation. It has good stability for the energy saved by the unit content. Compared with OEECS and LCD, Probcache and Random are greatly affected by c, and some degree of fluctuation is generated. Fluctuations are due to the randomness of the network and the nature of the strategy itself. However,

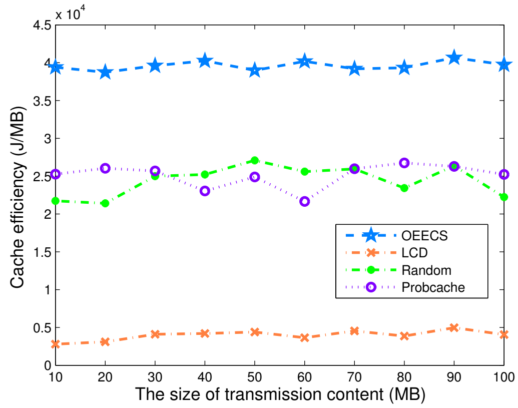

Fig. 8: The effect of c on cache efficiency: m=1000, P_c=10.

since they do not consider the network energy consumption from a global perspective, they are easily affected when the network environment changes, and the performance stability cannot be guaranteed. The OEECS is superior to the other three strategies, demonstrating its availability and effectiveness in selecting cache nodes.

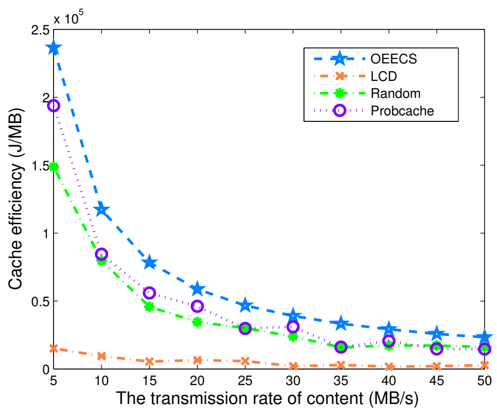

Fig. 9: The effect of R on cache efficiency: m=1000, c=50, P_c=10.

Fig. 9 shows that the cache efficiency of these four strategies is affected by the content transmission rate. The best cache efficiency is OEECS, and the worst is LCD. However, as R increases, the gap between the four strategies gradually narrows. The OEECS saves the most unit content energy due to the goal of energy consumption optimization, which enables better selection of the best cache node and then energy consumption is effectively saved. Due to the probabilistic selection characteristics of Probcache and Random, both of them are fluctuating. Because Probcache is more targeted than Random, its cache efficiency is almost higher than Random. With the increase of R, the cache efficiency of LCD decreases slowly, and it fluctuates slightly because of the randomness of the network. However, due to the single choice constraint of LCD, it is less effective in energy saving of unit content.

In summary, among the four strategies, OEECS has an advantage in terms of cache efficiency and performance stability.

C. Average Access Latency

The average access latency (AAL) is another widely used performance evaluation criterion that refers to the time it takes for a node to get content, calculated by $\sum_{j=1}^{n} c_j \cdot d_{i\to j}/\sum_{j=1}^{n} R_j$. Where c_j is the same as described in CE, the parameter R_j is the content transmission rate, and the parameter $d_{i\to j}$ is the distance between the two different nodes.

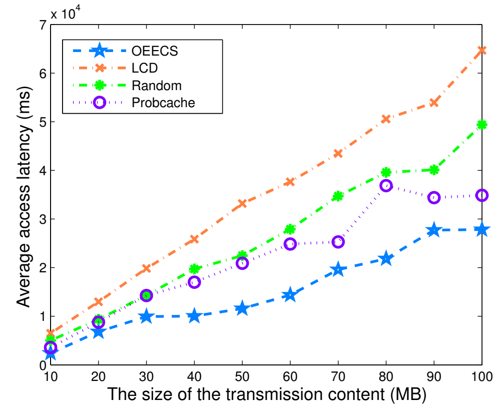

Fig. 10: The effect of c on average access latency: m=1000, P_c=10.

As shown in Fig. 10, the average access latency increases as c grows. It can be explained that the larger the c is, the more content is transferred across the network. Therefore, the longer latency is required to get the content. The LCD grows almost linearly, and is most affected by c. It shows that LCD has little effect on reducing access latency, which is attributed to the singularity of the LCD strategy itself. The average access latency of Probcache and Random is between LCD and OEECS. This shows that these two strategies can effectively reduce the access latency to some extent and shorten the distance between the node and the required content. Since the cache node in Probcache is selected according to the probability generated by multiple factors weighting, the effect is superior to Random. Among them, OEECS is slowly rising, which is the least affected by c. Due to the fact that the OEECS aims to optimize energy consumption, which is proportional to the distance between nodes. Under the same conditions, the distance is reduced and the latency is correspondingly reduced.

In Fig. 11, we can see that the average access latency decreases as R increases. In the case of other parameters remaining unchanged, the larger the R is, the less time it takes to retrieve the content. As R increases, the reduction rate of average access latency of LCD is the greatest, it shows that the main factor of the average access latency reduction is R, and its caching strategy is not prominent. In contrast, OEECS has the smallest decline. The results show that OEECS plays an important role in reducing the average access latency. The decline of Probcache and Random is relatively comparable, between LCD and OEECS. At the beginning, when R is small, the average access latency of Probcache is significantly lower than Random. However, as R increases, the difference between the two strategies are getting smaller. This shows that when

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2980954, IEEE Internet of Things Journal

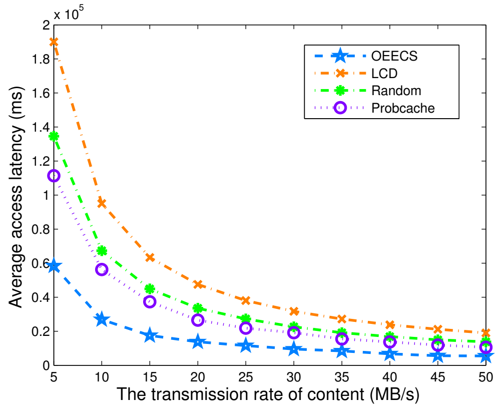

Fig. 11: The effect of R on average access latency: m=1000, c=50, P_c=10.

the network spreads very well, the effect of reducing access latency using Probcache and Random is similar.

Through analysis and comparison, OEECS has better objectivity and can further reduce network latency in a good network environment.

D. Delivery Success Rate

The delivery success rate (DSR) is the probability of finding the best cache node within a limited detection times, calculated by $\sum_{j=1}^{n} n_j^*/n$. The parameter n_j^* is the time the node needed to be changed to find the best cache node.

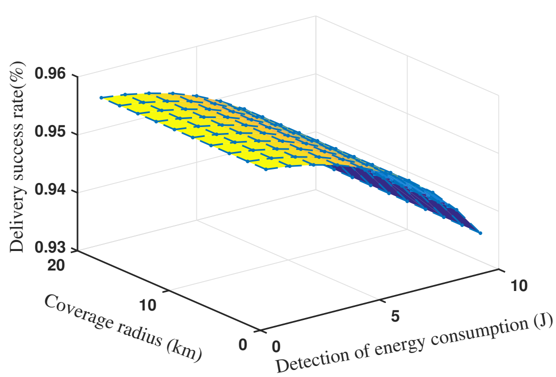

Fig. 12: The effect of P_c and coverage radius r on delivery success rate.

As shown in Fig. 12, when P_c is a fixed value, the DSR remains almost unchanged as the r increases. This shows that DSR is less affected by r. It can be explained by the fact that the OEECS selects the cache node based on the result of comparison with the energy saving expectation, regardless of the communication coverage. When r is a fixed value, the DSR decreases slowly as P_c increases. The reason is that P_c is the extra energy that needs to be consumed for each detection. In terms of energy saving, the increase of P_c will reduce the energy saving effect. The smaller the energy saved, the lower the probability of detecting a node that satisfies the condition, leading to the reduction of DSR. According to the analysis, the OEECS does not fluctuate significantly under the condition of the whole parameters change and always maintain a good delivery success rate.

VI. Conclusion

In this paper, an energy-aware caching strategy for infotainment services is proposed in the ICV network. First, we propose a cache node selection problem, which aims to put the content in the best position to reduce network energy consumption. Then, in order to solve this problem, we construct a network energy consumption model and transform the problem into an optimal stopping problem. Finally, we propose a optimal energy efficient cache node selection algorithm based on optimal stopping theory. The simulation results show that the proposed algorithm has a significant improvement in network performance compared with the baseline strategies, especially in energy saving.

VII. Acknowledgement

This work was supported by The National Key Research and Development Program of China (2018YFB1800202, 2016YFB1000302, SQ2019ZD090149, 2018YFB0204301.), NSFC(61832020, 61702569), Key-Area Research and Development Program of Guang Dong Province(2019B010107001).

References

[1] A. H. Ngu, M. Gutierrez, V. Metsis, S. Nepal, and Q. Z. Sheng, "Iot middleware: A survey on issues and enabling technologies," *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 1–20, 2017.

[2] H. Li, K. Ota, and M. Dong, "Learning iot in edge: Deep learning for the internet of things with edge computing," *IEEE network*, vol. 32, no. 1, pp. 96–101, 2018.

[3] G. Yu, Z. Cai, S. Wang, H. Chen, F. Liu, and A. Liu, "Unsupervised online anomaly detection with parameter adaptation for kpi abrupt changes," *IEEE Transactions on Network and Service Management*, 2019.

[4] L. Guo, M. Dong, K. Ota, Q. Li, T. Ye, J. Wu, and J. Li, "A secure mechanism for big data collection in large scale internet of vehicle," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 601–610, 2017.

[5] R. Van der Meulen and J. Rivera, "Gartner says by 2020, a quarter billion connected vehicles will enable new in-vehicle services and automated driving capabilities," *Gartner) Retrieved September*, vol. 18, p. 2015, 2015.

[6] H. Zhang, Q. Zhang, J. Liu, and H. Guo, "Fault detection and repairing for intelligent connected vehicles based on dynamic bayesian network model," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2431–2440, 2018.

[7] L. T. Tan, R. Q. Hu, and Y. Qian, "D2d communications in heterogeneous networks with full-duplex relays and edge caching," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4557–4567, 2018.

[8] J. Xu, K. Ota, and M. Dong, "Saving energy on the edge: In-memory caching for multi-tier heterogeneous networks," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 102–107, 2018.

[9] Y. Liu, Z. Zeng, X. Liu, X. Zhu, and M. Z. A. Bhuiyan, "A novel load balancing and low response delay framework for edge-cloud network based on sdn," *IEEE Internet of Things Journal*, 2019.

[10] M. Huang, W. Liu, T. Wang, A. Liu, and S. Zhang, "A cloud-mec collaborative task offloading scheme with service orchestration," *IEEE Internet of Things Journal*, 2019.

[11] F. Liu, G. Tang, Y. Li, Z. Cai, X. Zhang, and T. Zhou, "A survey on edge computing systems and tools," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1537–1562, 2019.

[12] H. Chen, J. Yu, F. Liu, Z. Cai, and J. Xia, "Archipelago: A medical distributed storage system for interconnected health," *IEEE Internet Computing*, 2019.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2980954, IEEE Internet of Things Journal

[13] A. Mahmood, C. Casetti, C. F. Chiasserini, and P. Giaccone, "Mobility-aware edge caching for connected cars," in *Conference on Wireless On-demand Network Systems & Services*, 2015.

[14] Y. AlNagar, S. Hosny, and A. A. El-Sherif, "Towards mobility-aware proactive caching for vehicular ad hoc networks," *arXiv preprint arXiv:1810.01172*, 2018.

[15] T. L. Thanh, "Mobility-aware edge caching and computing framework in vehicle networks: A deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 1–1, 2018.

[16] O. Attia and T. A. ElBatt, "On the role of vehicular mobility in cooperative content caching," *2012 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pp. 350–354, 2012.

[17] S. Akhavanbitaghsir and A. Khonsari, "Cooperative caching for content dissemination in vehicular networks," *International Journal of Communication Systems*, vol. 31, no. 3, p. e3534, 2018.

[18] S. Abdelhamid, H. S. Hassanein, and G. Takahara, "On-road caching assistance for ubiquitous vehicle-based information services," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 12, pp. 5477–5492, 2015.

[19] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1046–1061, 2017.

[20] A. Jalooli, S. Min, and X. Xu, "Delay efficient disconnected rsu placement algorithm for vanet safety applications," in *Wireless Communications & Networking Conference*, 2017.

[21] C. F. Chiasserini, F. Malandrino, and M. Sereno, "Advertisement delivery and display in vehicular networks," *IEEE Vehicular Technology Magazine*, vol. 12, no. 3, pp. 1–1, 2017.

[22] S. Zhang, Y. Luo, K. Li, and V. Li, "Real-time energy-efficient control for fully electric vehicles based on an explicit model predictive control method," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 6, pp. 4693–4701, 2018.

[23] N. Laoutaris, H. Che, and I. Stavrakakis, "The lcd interconnection of lru caches and its analysis," *Performance Evaluation*, vol. 63, no. 7, pp. 609–634, 2006.

[24] Z. Zhang, H. Ma, and L. Liu, "Cache-aware named-data forwarding in internet of things," in *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, IEEE, 2015.

[25] I. Psaras, K. C. Wei, and G. Pavlou, "Probabilistic in-network caching for information-centric," in *Edition of the Icn Workshop on Information-centric Networking*, 2012.

[26] 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 *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1207–1212, IEEE, 2015.

[27] S. A. Bitaghsir and A. Khonsari, "Modeling and improving the throughput of vehicular networks using cache enabled rsus," *Telecommunication Systems*, no. 2, pp. 1–14, 2018.

[28] J. Ma, J. Wang, G. Liu, and P. Fan, "Low latency caching placement policy for cloud-based vanet with both vehicle caches and rsu caches," in *2017 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, IEEE, 2017.

[29] J. Zhang, X. Hu, Z. Ning, E. C. H. Ngai, L. Zhou, J. Wei, J. Cheng, B. Hu, and V. C. M. Leung, "Joint resource allocation for latency-sensitive services over mobile edge computing networks with caching," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4283–4294, 2019.

[30] N. Xia, M. Tang, J. G. Jiang, D. Li, and H. W. Qian, "Energy efficient data transmission mechanism in wireless sensor networks," in *International Symposium on Computer Science & Computational Technology*, 2008.

[31] O. Rioul and J. Magossi, "On shannon's formula and hartley's rule: Beyond the mathematical coincidence," *Entropy*, vol. 16, no. 9, pp. 4892–4910, 2014.

[32] S. Yousefi, E. Altman, R. El-Azouzi, and M. Fathy, "Analytical model for connectivity in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3341–3356.

[33] S. M. Abuelenin and A. Y. Abul-Magd, "Empirical study of traffic velocity distribution and its effect on vanets connectivity," in *2014 International Conference on Connected Vehicles and Expo (ICCVE)*, pp. 391–395, IEEE, 2014.

[34] M. H. Davis and I. Karatzas, "A deterministic approach to optimal stopping," *Probability, Statistics and Optimisation (ed. FP Kelly). NewYork Chichester: John Wiley & Sons Ltd*, pp. 455–466, 1994.

[35] Y. Peng, G. Wang, and N. Wang, "Energy-efficient transmission strategy by using optimal stopping approach for mobile networks," *Mobile Information Systems*, vol. 2016, 2016.

[36] H. Zhang, Q. Du, P. Ren, and Z. Wang, "Social stability enhanced mobile d2d relay networks: An optimal stopping approach," *2017 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2017.

[37] G. Liang, *Optimal stopping problems and their applications*. PhD thesis, University of Warwick, 2019.

Hongjia Wu received the B.S. and M.S. degrees from Liaoning Technical University and Guangxi University, China, in 2016 and 2018, respectively. She is currently working toward the Ph.D. degree with the College of Computer from National University of Defense Technology (NUDT), Changsha, China. Her research interests include scheduling and resource allocation in dispersed computing.

Jiao Zhang received the Ph.D. degree from National University of Defense Technology (NUDT), Changsha, China, in 2019. She is currently a Post-Doctoral Researcher with the College of Electronic Science and Engineering from National University of Defense Technology. Her research interests include mobile edge computing and resource allocation in heterogeneous networks.

Zhiping Cai received the B.Eng., M.A.Sc., and Ph.D. degrees in computer science and technology from the National University of Defense Technology (NUDT), China, in 1996, 2002, and 2005, respectively. He is a full professor in the College of Computer, NUDT. His current research interests include artificial intelligence, network security and big data. He is a senior member of the CCF and a member of the IEEE.

Fang Liu received the PhD degree in Computer Science and Technology from National University of Defense Technology in 2005. She is a professor and PhD supervisor at Hunan University (HNU) . Her main research interests include edge computing and storage system.

Yangyang Li received the B.S. degree from the Nanjing University of Information and Technology, in 2009, and the Ph.D. degree in computer science from the Beijing University of Posts and Telecommunications, in 2015. He is currently a Senior Engineer at the National Engineering Laboratory for Public Safety Risk Perception and Control by Big Data (PSRPC). His research interests include the mobile internet, content security, and edge computing.

Anfeng Liu is a Professor of School of Computer Science and Engineering of Central South University, China. He is also a Member (E200012141M) of China Computer Federation (CCF). He received the M.Sc. and Ph. D degrees from Central South University, China, 2002 and 2005, both in computer science. His major research interest is wireless sensor network. afengliu@mai.csu.edu.cn.