Multi-Objective Optimization for Resource Allocation in Vehicular Cloud Computing Networks

Wenting Wei, *Member, IEEE*, Ruying Yang, Huaxi Gu, *Member, IEEE*, Weike Zhao, Chen Chen, *Senior Member, IEEE*, and Shaohua Wan, *Senior Member, IEEE*

***Abstract*— Modern transportation is associated with considerable challenges related to safety, mobility, the environment and space limitations. Vehicular networks are widely considered to be a promising approach for improving satisfaction and convenience in transportation. However, with the exploding popularity among vehicle users and the growing diverse demands of different services, ensuring the efficient use of resources and meeting the emerging needs remain challenging. In this paper, we focus on resource allocation in vehicular cloud computing (VCC) and fill the gaps in the previous research by optimizing resource allocation from both the provider's and users' perspectives. We model this problem as a multi-objective optimization with constraints that aims to maximize the acceptance rate and minimize the provider's cloud cost. To solve such an NP-hard problem, we improve the nondominated sorting genetic algorithm II (NSGA-II) by modifying the initial population according to the matching factor, dynamic crossover probability and mutation probability to promote excellent individuals and increase population diversity. The simulation results show that our proposed method achieves enhanced performance compared to the previous methods.**

***Index Terms*— Multi-objective optimization, vehicular cloud computing, resource allocation, NSGA-II.**

I. Introduction

THE Internet of Things (IoT) has attracted considerable attention in recent years as the presumed future Internet in which all physical things can exchange and communication with each other by connecting to a network. As a significant part of IoT, the vehicular network takes moving vehicles as information perception objects. Due to the new generation of information and communication technology, information interaction between vehicles and all other things (i.e., vehicles, people, roads and service platforms) can be realized. This great progress provides a safe, comfortable and efficient driving experience and desired traffic services to users by improving the intelligent driving level and transportation efficiency [1], [2]. However, due to the increasing number of vehicles and growing diverse demands of different services, achieving efficient use of resources remains challenging, and methods must be developed that meet the emerging needs of vehicle IoT [3].

As a vehicle operation information platform, the vehicular network covers massive information for various fields, including intelligent transportation system, freight transportation, vehicle management and emergency rescue. In such a complex system, dynamic network structures and growing amount of vehicle nodes pose a challenge with respect to supporting resources (bandwidth, computing, storage) and quality of service (QoS) constraints (delay, jitter, packet loss rate, etc.) [4], [5]. Furthermore, a single vehicle is no longer able to support dynamic resource demand. Vehicular cloud computing (VCC) [6] is a potential solution for providing scalable computing resources and various application services.

Owing to advances in cloud computing and the development of artificial intelligence, a variety of cloud-based applications such as real-time communication, multimedia entertainment and location-based services have emerged [7]–[9]. In these applications, information is collected and acquired by smart sensors and actuators that are eventually connected to the Internet [10]. Cloud computing enables applications to access computing power, storage space, and information services on demand, providing users with flexible and scalable services, and may effectively reduce the waste of resources.

Unfortunately, with the exploding vehicle traffic and the extensive cloud applications, the limited space and computing capacity of vehicular devices pose a great challenge to the vehicular network [11], [12]. Due to space limitations, the hardware systems of vehicles must have a small volume and a low cost to meet the requirements of some complicated applications for a single vehicle [13]. Therefore, sharing computing and storage resources between the vehicles and roadside

Manuscript received 5 December 2020; revised 26 February 2021 and 1 April 2021; accepted 22 April 2021. Date of publication 3 August 2021; date of current version 5 December 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFE0202800, Grant 2020YFB1807500, and Grant 2018YFC0831200; in part by the National Natural Science Foundation of China under Grant 62072360, Grant 61571338, Grant 61672131, and Grant 61901367; in part by the Natural Science Foundation of Shaanxi Province for Distinguished Young Scholars under Grant 2020JC-26; in part by the Youth Innovation Team of Shaanxi Universities; in part by the Key Laboratory of Embedded System and Service Computing (Tongji University) under Grant ESSCKF2019-05; in part by the Ministry of Education; in part by the Xi'an Science and Technology Plan under Grant 20RGZN0005; in part by the Xi'an Key Laboratory of Mobile Edge Computing and Security under Grant 201805052-ZD3CG36; in part by the Fundamental Research Funds for the Central Universities under Grant 31412111303 and Grant 31512111310; and in part by the Open Project from the State Key Laboratory for Novel Software Technology, Nanjing University, under Grant KFKT2019B17. The Associate Editor for this article was H. Lu. *(Corresponding authors: Huaxi Gu; Shaohua Wan.)*

Wenting Wei, Ruying Yang, Huaxi Gu, Weike Zhao, and Chen Chen are with the State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China (e-mail: hxgu@xidian.edu.cn).

Shaohua Wan is with the School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan 430073, China (e-mail: shaohua.wan@ieee.org).

Digital Object Identifier 10.1109/TITS.2021.3091321

infrastructure is increasingly important. Furthermore, due to the diversity of applications and the dynamic nature of the traffic load on the cloud computing platform, load imbalance between the nodes occurring during the supply period will be even harder to address and may thus become a bottleneck that greatly affects the overall performance of the cloud computing system. Therefore, the major problem is to optimize resource allocation reasonably through the appropriate algorithm.

To date, most works have focused on optimization of energy, cost, overhead and other metrics from the provider's perspective. For instance, the authors in [14] proposed a roadside unit cloud resource management scheme that minimizes the cost of service deployment, reconfiguration overhead and infrastructure routing delay. In [15], the authors proposed an infinite-horizon semi-Markov decision process (SMDP) scheme for vehicular cloud systems to achieve energy and time efficiency.

Although resource allocation in vehicular networks is well developed, most works on this topic do not consider simultaneous minimization of the blocking probability and cost from both the provider's and users' perspectives, particularly focusing on efficient use of resources with load balance. To fill this gap, in this paper, we focus on a resource allocation scheme with low blocking probability and cost for vehicular cloud systems in order to meet the requirements of the requests and make full use of vehicle resources. To implement our idea, we model resource allocation in a VCC system as a multi-objective optimization with constraints where the characteristics of the central cloud and vehicle cloud such as the deadlines of requests and connection duration between vehicles are considered in the constraint conditions.

To solve such an NP-hard problem, we develop an improved version of the nondominated sorting genetic algorithm II (NSGA-II), named AC-INSGA, by modifying the initial population, crossover probability and mutation probability. Specifically, the initial population is generated according to the matching factor (MF) rather than by random generation. The dynamic crossover probability and mutation probability are designed to protect excellent individuals, increase the diversity and avoid local optimization.

The rest of our paper is organized as follows. Section II reviews some related work. Section III presents the proposed multi-objective constraint programming resource allocation model. Section IV introduces the AC-INSGA algorithm to solve the proposed model. Section V highlights and analyzes the simulation results. The conclusions of this paper are summarized in Section VI.

II. Related Work

The emergence of vehicular networks has played a vital role in the state-of-the-art transportation applications. Its performance heavily relies on the cloud resource allocation in VCC. Hence, allocation of resources to optimize specific objectives by efficient algorithms has been widely explored in literatures.

Typically, the resource allocation problem for VCC can be formulated as a mixed-integer programming problem along with predefined resource constraints [16]. The majority of works in the literature designed low-complexity heuristics to obtain approximate solutions in acceptable time. To guarantee the efficient utilization of the vehicular on-board computing resources and minimize the average response time of the offloaded missions, Fei *et al.* [17] designed a low-complexity modified genetic algorithm GA-based task scheduling scheme. Based on linear programming-based optimization and binary particle swarm optimization, Zhu *et al.* [18] proposed a dynamic task allocation framework to maintain a trade-off between service latency and quality loss. Considering the deadline of requests and the capacities of vehicles, Mekki *et al.* [19] solved a vehicular fog resource allocation problem using NSGA-II.

Several studies [15], [20], [21] formulated the resource allocation for VCC as a Markov decision process (MDP), which is a mathematical framework for dynamic control systems with discrete state spaces. In contrast to [15], Meneguette *et al.* [22] presented an SMDP-based resource allocation approach in order to maximize the utilization of allocated idle resources. Lin *et al.* [23] additionally considered heterogeneous vehicles and RSUs and finally demonstrated great performance in terms of power and time consumption costs. Inspired by its success in solving decision-making problems theoretically modeled as maximizing the final cumulative return in an MDP, reinforcement learning (RL) was employed in resource allocation. To reduce the service discovery and consumption delays, Arkian *et al.* [24] employed a traditional RL algorithm of Q-learning to select a service provider among participating vehicles and introduced three different queuing strategies to solve this resource allocation problem. Peng and Shen [25] exploited deep RL to transform and solve two proposed multi-dimensional resource optimization problems to maximize the number of offloaded tasks with satisfied QoS requirements.

Game theory is an effective method for designing distributed mechanisms in which the users of mobile devices can locally make decisions based on strategic interactions and realize mutually satisfactory resource allocation schemes. Yu *et al.* [26] represented the resources in vehicle and roadside clouds in the form of virtual machines and proposed a noncooperative game approach for the allocation of cloud resources considering efficiency, QoS and fairness. In their other work [27], a coalition game model based on two-sided matching theory was introduced to promote the sharing of bandwidth and computing resources among cloud service providers. Zhou *et al.* [28] developed a two-stage computation resource allocation and task assignment approach by combining contract theory and matching theory with the objective of maximizing the expected utility of the base station. To utilize idle resource within parked electric vehicles for the execution of resource-hungry tasks of nearby mobile users' application demands, Birhanie *et al.* [29] formulated a stochastic theoretical game to minimize the energy and delay overhead.

Despite the existence of some related works in the literature, the problem of reducing the blocking probability while achieving a low cost related to the efficiency of resource utilization and user experience has received much less attention. The duration of connection between the requesting vehicle and the vehicle that provides the resources is another significant factor in the VCC system. In contrast to the traditional static cloud computing system, the VCC system has some unique features, particularly the mobility of vehicles [30], [31]. A mobile vehicle cloud consists of a set of moving vehicles. The connection may break during transmission, making maintenance of reliable data dissemination difficult [32]. Thus, it has a significant impact on the blocking probability. Furthermore, the cost and reward are handled by a simple linear computation in the literature. The rationality of this approach remains to be verified.

In this paper, we attempt to solve the above limitations. A multi-objective constraint programming resource allocation scheme is proposed that aims to minimize the blocking probability while reducing the cost. According to the unique features of the VCC system, a series of constraints are implemented. By using the AC-INSGA algorithm, the nondominated front is obtained. The requesters can choose the most suitable solution in the nondominated front according to their own requirements. The resources in our scheme refer to computing resources. Since the vehicles are produced by different manufacturers, they have inherently different computing resources [15]. To address the heterogeneity, virtualization is developed to abstract the resources. The virtualized resources can be shared in the VCC system. In this paper, the vehicle and the central cloud are assumed to have different numbers of virtualized resource units (RUs).

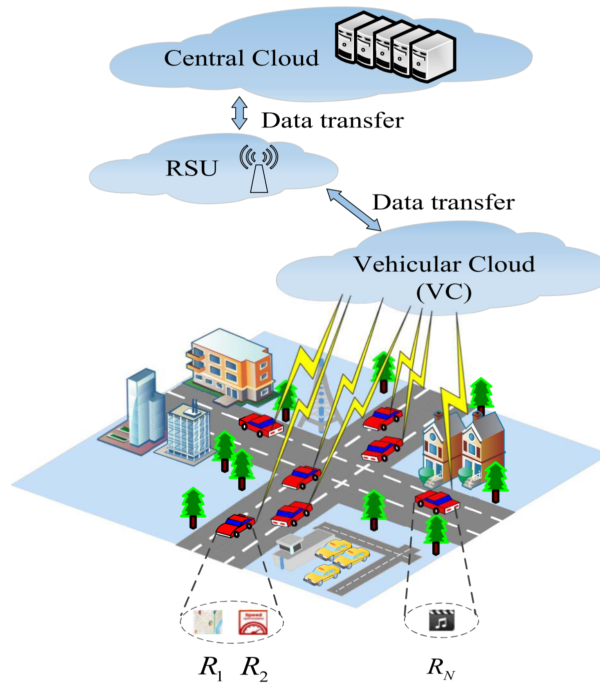


Fig. 1. Resource allocation for vehicular cloud computing.

III. System Model

In this section, we first introduce the network model with a three-tier cloud architecture and present the resource allocation process in the VCC system. To achieve our optimization objective, we formulate a multi-objective optimization model to jointly minimize the blocking probability and cost.

A. System Framework

The framework of resource allocation in a VCC system is illustrated in Fig.1. As shown in Fig. 1, a group of vehicles constitutes a dynamic vehicle cloud serving as the low layer. They share resources with each other in this layer. RSUs form the middle layer. The traditional cloud that consists of large server clusters serves as the top layer.

When a request (e.g., cloud-based route planning, cloud-based multimedia entertainment, cloud-based video and audio communications) arrives, RSUs process this request in the vehicle cloud and decide whether to transfer it to the central cloud according to both its own requirements and the available resources in the vehicle cloud. Although the resources in the central cloud are deemed to be infinite, the cost of transmitting the requests to the central cloud and then processing them in the central cloud is higher than the cost of the corresponding use of the vehicle cloud, and the processing response time is longer than that of the vehicle cloud.

B. Network Model and Notations

Consider M vehicles carrying N service requests passing through the vehicular cloud service range. We suppose that the number of available RUs of vehicle j is c_j^v. Let c_i^{cc}, c_i^{vc} denote the number of RUs required for request i that is processed in the central cloud or vehicle cloud. For ease of reference, the important notations in this article are shown in Table I.

To handle requests from vehicles, the vehicle cloud must choose among two process modes. In one, the vehicle cloud allocates a number of RUs in the resource pool to compute the requested service, which consumes a little power. In the other, the vehicle cloud transfers requests to the central cloud, which has powerful computing resources and handles service requests immediately. For different requests, the choice of mode must consider the requested resources, delays, benefits, costs, etc.

C. Problem Formulation

In this paper, our goal is to jointly reduce the blocking probability and cost. Such a multi-objective optimization problem consists of objective functions and constraints. The main objective function can be described mathematically as follows:

1) Objective Function: Blocking Probability: When the resources provided by the VCC system are insufficient, request requirements are unsuccessfully serviced. We define the blocking probability of requests as equation (1).

$$BP = \frac{N - (N_{cc}^{suc} + N_{vc}^{suc})}{N}, \tag{1}$$

where N is the number of requests. N_{cc}^{suc} is the number of requests processed in the central cloud, and N_{vc}^{suc} is the number of requests processed in the vehicle cloud.

2) Objective Function: Cost of Allocation: The provider's cost in the resource allocation process can be calculated by

TABLE I
NOTATIONS

Symbol	Description
N	Number of requests.
M	Number of vehicles.
N_{cc}^{suc}	Number of requests processed in the central cloud.
N_{vc}^{suc}	Number of requests processed in the vehicle cloud.
$\mathrm{S_{cc}}$	Cost per RU of processing requests in the central cloud.
$\mathrm{S_{vc}}$	Cost per RU of processing requests in the vehicle cloud.
c_i^{cc}	Number of RUs required for request i processed in the central cloud.
c_i^{vc}	Number of RUs required for request i processed in the vehicle cloud.
$R_{i,in}^{cc}$	Input size of request i processed in the central cloud.
$R_{i,out}^{cc}$	Output size of request i processed in the central cloud.
$R_{i,in}^{vc}$	Input size of request i processed in the vehicle cloud.
$R_{i,out}^{vc}$	Output size of request i processed in the vehicle cloud.
c_j^v	Number of available RUs of vehicle j.
c_{ij}^{vc}	Number of RUs of vehicle j allocated for request i.
$\gamma_{\mathrm{cc}}/\gamma_{\mathrm{vc}}$	Transmission rate in the central cloud/vehicle cloud.
η_i	Processing time of request i in the central cloud.
τ_i	Processing time of request i in an RSU.
d_i	Deadline of request i.
s_{ij}^0	Current distance between vehicle i and vehicle j.
s	Distance threshold.
t_j	Duration of connection between requesting vehicle and vehicle j.
v_j	Velocity of vehicle j.
α	Velocity change factor.
Δv_j	Velocity change of vehicle j.
(x_j^0, y_j^0)	Current position coordinates of vehicle j.
$v_j^{\min}$	Minimal velocity of vehicle j.
$v_j^{\max}$	Maximal velocity of vehicle j.

equation (2).

$$C = \mathrm{S_{cc}} \times \frac{1}{\gamma_{\mathrm{cc}}} \times \sum_{i=1}^{N_{cc}^{suc}} (R_{i,in}^{cc} + R_{i,out}^{cc}) + \mathrm{S_{vc}} \times \frac{1}{\gamma_{\mathrm{vc}}} \times \sum_{i=1}^{N_{vc}^{suc}} (R_{i,in}^{vc} + R_{i,out}^{vc}), \quad (2)$$

where $\mathrm{S_{cc}}$ and $\mathrm{S_{vc}}$ are the prices per unit time in the central cloud and the vehicle cloud, respectively. $R_{i,in}^{cc}$ and $R_{i,out}^{cc}$ are the input and output sizes of request i that is processed in the central cloud, respectively. $R_{i,in}^{vc}$ and $R_{i,out}^{vc}$ are the input and output sizes of request i that is processed in the vehicle cloud, respectively. γ_{cc} and γ_{vc} are the transmission rates in the central cloud and the vehicle cloud, respectively. The blocking probability is minimized in equation (1), and the cost is minimized in equation (2). In addition, several constraints should be satisfied.

3) Main Objective Function:

$$\textbf{minimize } (BP, C), \quad (3)$$

where BP is the blocking probability, and C indicates the cost of the resource allocation process in the vehicle cloud and the central cloud. The objective has two terms: minimizing the blocking probability and minimizing the cost of resource allocation.

4) Constraints:

a) Transmission and processing of requests in the central cloud (CC):

$$\frac{R_{i,in}^{cc}}{\gamma_{\mathrm{cc}}} + \frac{c_i^{cc}}{\mu^{\mathrm{cc}}} + \frac{R_{i,out}^{cc}}{\gamma_{\mathrm{cc}}} + \tau_{cc} \le d_i, \quad (4)$$

where $R_{i,in}^{cc}$ and $R_{i,out}^{cc}$ denote the input and output sizes of request i that is processed in the CC, respectively. μ^{cc} is the computing service rate of the request in the CC if only one RU is allocated. τ_{cc} is the processing time in an RSU when the request is transferred to the CC. The response time to request i is denoted d_i. Equation (4) guarantees that processing and transmission should be completed by the deadline of the request if the request is allocated to the CC.

b) Processing requests in the vehicle cloud (VC):

$$c_{ij}^{vc} \le c_j^v. \quad (5)$$

$$\sum_{i=1}^{N_{vc}^{suc}} c_i^{vc} \le \sum_{j=1}^{M} c_j^v. \quad (6)$$

$$\sum_{j=1}^{N_{vc}^{suc}} c_{ij}^{vc} \le c_j^v, \; j = 1, 2, \ldots, M. \quad (7)$$

When request i is allocated to vehicle j, the available resources of vehicle j should satisfy the resource requirement of request i, as described by equation (5). Equation (6) ensures that the total resources of all requests allocated to the VC cannot exceed the available resources in the VC. Equation (7) ensures that the total resources of the requests allocated to vehicle j cannot exceed the available resources of vehicle j.

$$\frac{R_{i,in}^{vc}}{\gamma_{\mathrm{vc}}} + \frac{c_i^{vc}}{\mu^{\mathrm{vc}}} + \frac{R_{i,out}^{vc}}{\gamma_{\mathrm{vc}}} + \tau_{vc} \le d_i. \quad (8)$$

where μ^{vc} is the computing service rate of the request in the VC if only one RU is allocated. $R_{i,in}^{vc}$ and $R_{i,out}^{vc}$ denote the input size and output size of request i that is processed in the VC, respectively. τ_{vc} represents the processing time in an RSU when the request is transferred to the VC. Equation (8) ensures that processing and transmission are completed by the deadline of the request if the request is allocated to the VC.

$$\begin{cases} s_{ij}^0 + t_j \times \left|\left[v_j \pm \alpha \times \Delta v_j\right] - \left[v_i \pm \alpha \times \Delta v_i\right]\right| \leq \mathrm{s} \\ s_{ij}^0 = \sqrt{(x_j^0 - x_i^0)^2 + (y_j^0 - y_i^0)^2}. \end{cases} \tag{9}$$

$$\begin{cases} v_j^{\min} \leq v_j \leq v_j^{\max} \\ v_j + \Delta v_j \leq v_j^{\max} \\ v_j - \Delta v_j \geq v_j^{\min}. \end{cases}$$

$$\begin{cases} v_i^{\min} \leq v_i \leq v_i^{\max} \\ v_i + \Delta v_i \leq v_i^{\max} \\ v_i - \Delta v_i \geq v_i^{\min}. \end{cases} \tag{10}$$

Due to the movement of the vehicles, the connection between the requesting vehicle and the vehicle that provides resources may break during transmission. The connection time among the vehicles must be predicted. When the distance between the vehicles is greater than the threshold, the connection is considered to be broken, and the request is blocked. The connection times between the requesting vehicle and other vehicles are predicted by equation (9). The speed of the vehicles must meet the constraints described by equation (10).

$$\frac{R_{i,out}^{vc}}{\gamma_{\mathrm{vc}}} \leq t_j. \tag{11}$$

Equation (11) ensures that the download is completed before the connection is broken.

c) The number of RUs:

$$\begin{cases} c_i^{cc} \geq 0 \\ c_i^{vc} \geq 0. \end{cases} \tag{12}$$

Equation (12) means that the resources required for requests should be positive.

IV. Heuristic Algorithm

In this section, we propose the AC-INSGA algorithm, which is a modification of NSGA-II, to solve the proposed mathematical model effectively and choose the optimal solution with a tradeoff between the two goals.

A. Population Initialization

In the population initialization process, an MF is defined to measure the capability of the vehicle to provide the resources. Instead of traditional random generation, the initial population is generated based on the MF of the vehicles. The dynamic crossover probability and mutation probability are set to increase the population diversity and avoid local optimization.

Definition 1: Relative average velocity

$$\psi_j = \frac{\left|\frac{1}{n_{sam}}\sum_{i=1}^{n_{sam}} v_j^i - \mu\right|}{\mu}, \tag{13}$$

where ψ_j is the relative average rate of vehicle j. n_{sam} denotes the number of samples of vehicle j. v_j^i is the velocity of sample i of vehicle j, and μ is the average velocity of neighbor vehicles. The relative average velocity is defined as the difference between the average velocity of vehicle j and other neighbor vehicles in equation (13). The vehicle with stable velocity should be selected as the resource provider.

Definition 2: Adjacent node degree

$$\begin{cases} D_j = \dfrac{M_j}{M} \\ M_j = \left\{num\left\{v_i \left|\left|v_j - v_i\right| \leq \Delta \mathrm{v}\right.\right\}\right\}, \end{cases} \tag{14}$$

where D_j is the adjacent node degree of vehicle j. The difference in the velocity is used to identify the real adjacent nodes of vehicle j. Δv denotes the velocity threshold. If the difference in the velocity between the vehicle and vehicle j is less than the threshold, then the vehicle is considered to be a real adjacent node of vehicle j. M_j is the number of real adjacent nodes of vehicle j. In equation (14), the adjacent node degree is defined as the proportion of the number of real adjacent nodes to the total number of vehicles. A large adjacent node degree implies high stability and long duration of the connections.

Definition 3: Resource capacity

$$C_j = \frac{c_j^v}{\sum_{i=1}^{M} c_i^v}, \tag{15}$$

where C_j is the resource capacity of vehicle j. In equation (15), the resource capacity is defined as the ratio of the number of available RUs of vehicle j to the total number of available RUs in the VC. The resource capacity evaluates the amount of resources of the vehicle.

Definition 4: Matching factor

An MF is defined to measure the probability that the vehicles successfully provide resources for requests based on the relative average velocity, adjacent node degree and resource capacity. The MF can be calculated as follows.

$$MF_j = \gamma_1 \psi_j + \gamma_2 D_j + \gamma_3 C_j, \tag{16}$$

where MF_j is the MF of vehicle j. γ_1, γ_2 and γ_3 are the weight coefficients, and $\gamma_1 + \gamma_2 + \gamma_3 = 1$. By considering the abovementioned relative average velocity, adjacent node degree and resource capacity, the MF is proposed to evaluate the likelihood that the vehicles successfully provide resources for the requests as described in equation (16).

In AC-INSGA, a chromosome represents a resource allocation result. A chromosome has N genes. Each gene has $M+1$ values. The gene value $p = 0$ in the chromosome represents that the request is transferred to the CC. The gene value $p = 1, \cdots, M$ in the chromosome represents that the request is allocated to vehicle p. According to the MF, the generated probability of each gene value is defined in equation (17).

$$P_p = \begin{cases} \dfrac{\overline{MF}}{\sum_{j=1}^{M} MF_j + \overline{MF}}, & p = 0 \\ \dfrac{MF_p}{\sum_{j=1}^{M} MF_j + \overline{MF}}, & p = 1, \ldots, M, \end{cases} \tag{17}$$

where P_p is the generated probability of gene value p. $\overline{MF}$ is the average value of the MFs of all vehicles. When initializing the population, the gene values in the chromosome are generated according to equation (17).

B. Crossover and Mutation

To assess the quality of the chromosomes, a fitness function should be defined to represent the achievement of all objectives. We define the fitness function ζ as equation (18).

$$\begin{cases} \zeta_k(i) = \dfrac{1}{f_k(i)} \\ \zeta(i) = \displaystyle\sum_{k=1}^{m} \zeta_k(i), \end{cases} \tag{18}$$

where m is the number of objective functions. $\zeta(i)$ is the fitness of individual i, and $f_k(i)$ is the value of objective function k of individual i.

Two chromosomes are randomly selected from the mating pool. A random number between (0,1) is generated. If the random number is less than the crossover probability, then the gene position is randomly selected as the crossover point. We designed a crossover probability function according to the fitness of individuals, given as equation (19), in which a greater fitness corresponds to a lower setting of the crossover probability.

$$P_c = \begin{cases} \dfrac{\mathrm{a}_1\left(1 - \sum\limits_{k=1}^{m} \beta_k \frac{\zeta_k(i)}{\zeta_k^{\max}}\right)}{1 - \sum\limits_{k=1}^{m} \beta_k \frac{\zeta_k^{avg}}{\zeta_k^{\max}}}, & \zeta_k(i) \geq \zeta_k^{avg} (k = 1, \ldots, m); \\ \mathrm{a}_2, & \zeta s_k(i) < \zeta_k^{avg} (k = 1, \ldots, m); \\ \dfrac{m_l}{m}\mathrm{a}_2, & else, \end{cases} \tag{19}$$

where a_1 and a_2 are the parameters in the crossover probability function, whose values are set as constants between 0 and 1. β_k is the weight coefficient, and $\beta_1 + \cdots + \beta_k + \cdots + \beta_m = 1$. $\zeta_k(i)$ is the kth fitness, corresponding to the objective function k of individual i. ζ_k^{avg} is the average of the kth fitness, and $\zeta_k^{\max}$ is the maximum of the kth fitness. m_l is the number of objective functions in which the corresponding fitness values are less than the average fitness value. If all fitness values of individual i are less than the corresponding averages, then a large crossover probability is set. If all fitness values of individual i are greater than the corresponding averages, then a small crossover probability is set. Moreover, a greater fitness corresponds to a lower setting of the crossover probability. Otherwise, the crossover probability is set according to the number of objective functions such that the corresponding fitness values are less than the average fitness value. By setting the crossover probability in this manner, excellent individuals can be protected to enter the next generation.

$$D_g = \sum_{k=1}^{m} \alpha_k \left[\frac{1}{N_{pop}} \sum_{i=1}^{N_{pop}} \left(\frac{f_k(i)}{f_k^{\max}} - \frac{f_k^{avg}}{f_k^{\max}}\right)^2 \right], \tag{20}$$

Algorithm 1 AC-INSGA Algorithm for Multi-Objective Optimization

Require: The number of generations T_{max}
Ensure: Nondominated front S_p

```
Initialize parent population P_0;
for i ∈ P_0 do
    Calculate the objective function values f_k(i);
    Calculate the nondomination rank F_i and the crowding-distance n_d;
end for
for T = 1 to T_max do
    Select i ∈ P_0 as candidates x_c based on F_i and n_d;
    Calculate P_c and P_m according to equations (19) and (20);
    Get offspring population P_1 from x_c based on P_c and P_m;
    Merge P_0 and P_1 as P_2;
    for i ∈ P_2 do
        Calculate f_k(i);
        Calculate F_i and n_d;
        Record F_i as F = (F_1, F_2, ··· , F_l);
    end for
    Get parent population from P_2 based on F_i and n_d;
end for
Select the individuals F_1 to form S_p.
```

where D_g is the difference in objective function values. N_g is the population size. α_k is the weight coefficient, and $\alpha_1 + \cdots + \alpha_k + \cdots + \alpha_m = 1$. A number between (0,1) is generated, and the mutation position is randomly selected. If the generated number is less than the mutation probability P_m, then the value of the selected mutation position is changed to one of the other optional values. When the difference among the objective function values of the individuals is small, a large mutation probability is set. Otherwise, a small mutation probability is set. The difference between the objective function values is defined in equation (20). By setting the mutation probability in this manner, a local optimum is avoided.

To obtain a child solution, the parents are initially selected from the current parent population using tournament selection. The comparison criteria in the tournament selection are again based on nondominance comparison using the Pareto objective. For every generation, individuals are selected from the parent population as candidates based on the nondomination rank and the crowding distance. Once the parents are selected using tournament selection, crossover and mutation operators are used on the selected parents to produce the offspring solutions. The detailed steps of the AC-INSGA algorithm are shown in Algorithm 1.

V. Performance Evaluation

In this section, we briefly present the experimental environment and parameters. In the experiment, we used two baseline algorithms to evaluate the efficiency of our proposal, including GA and a part of the FOS algorithm [33]. The optimization objective of the GA scheme is to minimize either the blocking

TABLE II
LIST OF IMPORTANT PARAMETERS

Parameter	Value	Parameter	Value
N	200	c_j^v	[0,10]
M	50	s_{ij}^0	[0,300]
S_{cc}	2.5	c_i^{cc}	[1,3]
S_{vc}	2	c_i^{vc}	[1,3]
v_j	[60,120]	α	0.1
γ_{cc}	1.8	$v_j^{\min}$	60
γ_{vc}	2	$v_j^{\max}$	120

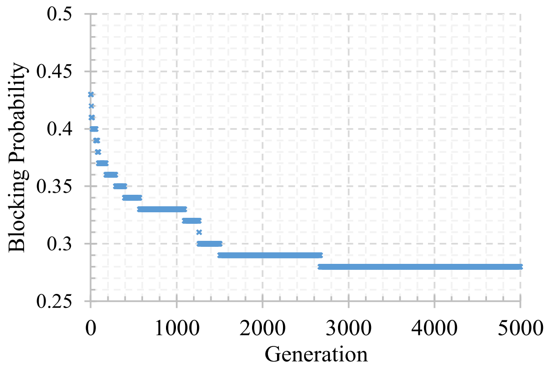


Fig. 2. Convergence curve for minimizing blocking probability.

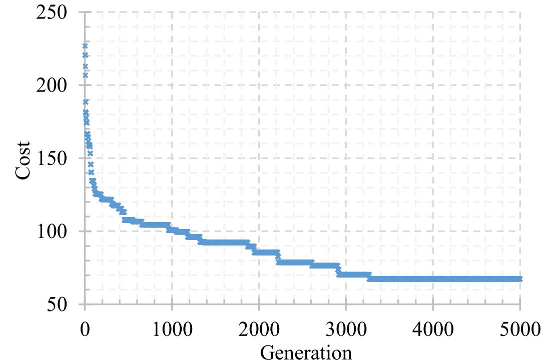


Fig. 3. Convergence curve for minimizing cost.

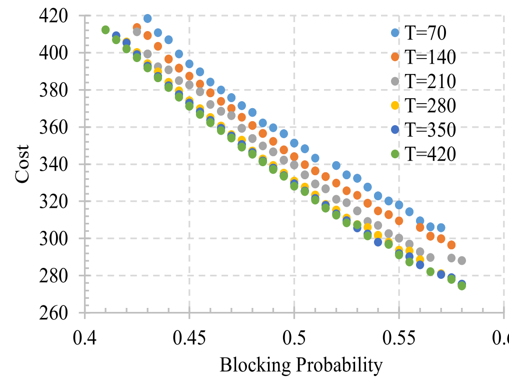


Fig. 4. Pareto fronts under different iterations.

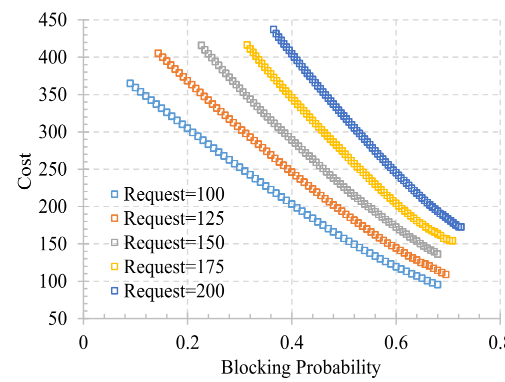


Fig. 5. Pareto fronts under different request numbers.

probability or cost as an optimization goal, and it was solved by the traditional GA.

A. Experimental Setting

We test the AC-INSGA algorithm using MATLAB as the simulator. All simulations are carried out on an Intel Core i5-7300HQ @2.50 GHz processor with 8 GB of RAM. $M = 50$ vehicles and $N = 200$ requests are set at the initial moment, and other important parameters are listed in Table II.

B. Convergence and Pareto Solution

Convergence is an important factor for algorithm performance. To observe the convergence of the AC-INSGA algorithm, we first record the optimal values in each generation of the population during evolution. Fig.2 and Fig.3 show the convergence curve of the objective functions. The figures show that the objective functions flatten out with increasing iteration number and finally converge in approximately 3000 iterations. The results show that our proposed method is convergent.

In addition, the Pareto optimal solution is mapped from the decision variable space to the target function space. Fig.4 shows the nondominant fronts under different numbers of iterations T. With an increase in the number of iterations, the optimal solution set continuously approaches the lower left corner and finally reaches the nondominant front. This figure shows that the Pareto optimal solutions of our algorithm are uniformly distributed along the Pareto frontier that achieves a tradeoff between the cost and blocking probability.

As shown in Fig.5, the nondominant fronts change with the number of requests and finally move to the top right. The cost and blocking probability increase with increasing number of requests. However, the Pareto frontier is still maintained, verifying the effectiveness of the AC-INSGA algorithm.

C. Performance Comparison of Our Optimal Objective

When the resources provided by the VCC system are insufficient, the request requirements may not be satisfied. Additionally, the cost of the resource allocation process is vital for evaluation of the efficiency of the VCC system. Thus, we use our two optimal objectives, the blocking probability and cost, as the metrics to evaluate the performance of the AC-INSGA algorithm and baselines.

As depicted in Fig.6-Fig.7, we compare the performance in terms of the blocking probability and cost for AC-INSGA, GA and FOS. We note that the blocking probability and

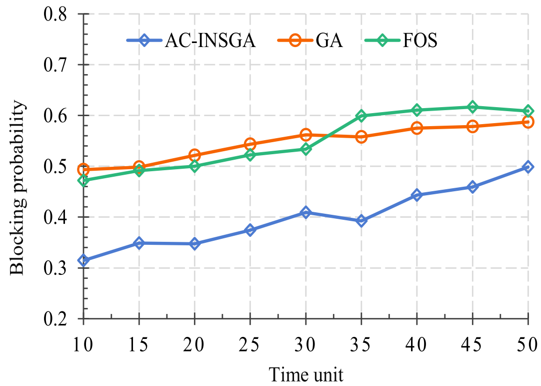


Fig. 6. Blocking probability vs. time unit.

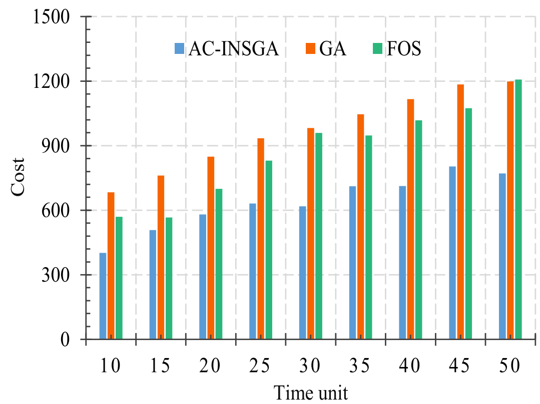


Fig. 7. Cost vs. time unit.

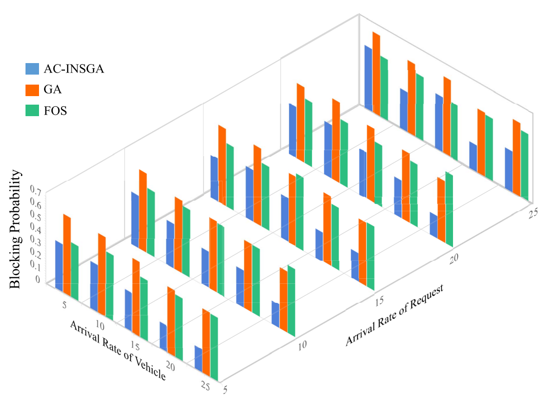


Fig. 8. Blocking probability vs. arrival rate of vehicles and requests.

cost increase with increasing number of time units under all circumstances. However, compared with GA and FOS, AC-INSGA reduces the blocking probability and cost by an average of 28% and 31%, respectively and displays a significant reduction in the blocking probability (by 2 orders of magnitude).

To test the AC-INSGA algorithm under different traffic loads, we plot three-dimensional histograms in Figs.8 - 9, where blocking probability and cost are the function of the arrival rate of the vehicles and requests. We note that the

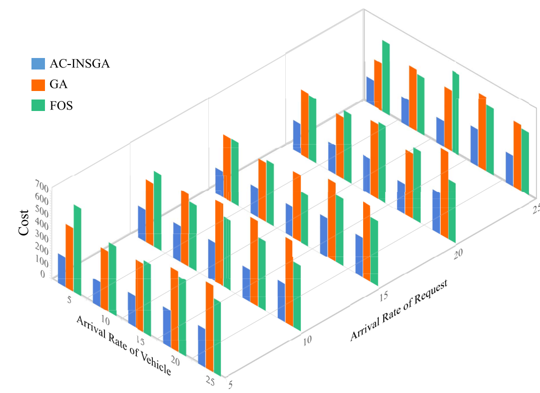


Fig. 9. Cost vs. arrival rate of vehicles and requests.

blocking probability and cost of AC-INSGA are clearly lower than those of the comparison algorithms. This is because the GA targets only the resource constraints and does not use the elite strategy. Under the fixed crossover and mutation probability, it easily becomes trapped in a local optimization. In the FOS algorithm, the selection of candidate vehicles is random, and the intelligent optimization algorithm is not used to select the optimal solution. It considers only the demand of the current request and not that of other requests and possible future requests. Therefore, the blocking probability and cost are higher than those obtained by the AC-INSGA algorithm.

VI. Conclusion

This paper proposed a multi-objective programming resource allocation scheme in the VCC system. Minimization of blocking probability and cost are set as the objective functions in this scheme. For constraints, the resource constraint, characteristics of the CC and VC, deadline of each request and connection duration between vehicles are considered. The AC-INSGA algorithm is used to solve our proposed scheme. During population initialization, the MF is defined to evaluate the vehicle capability to provide resources. In addition, the dynamic crossover probability is designed to allow excellent individuals to enter the next generation. The dynamic mutation probability is designed to increase the diversity and avoid local optimization. In the simulation, the convergence of the algorithm is proved. Nondominated fronts with different numbers of iterations and requests are obtained. Furthermore, our simulations show that the proposed algorithm achieves better performance than the previously used algorithms.

However, there remains some space to improve our work. First, we focus only on experience-driven resource allocation without considering real-time road traffic information. Several works [9], [34] have explored traffic flow detection and prediction in cloud computing to provide intelligent guidance for relieving traffic jams. Therefore, further extension to data-driven resource allocation with the help of traffic flow prediction is left for our further study. Another limitation is that we do not evaluate the algorithms with real trace data in Vehicle Networks. In the future, we will discuss and optimize our model under real application scenarios and extend our experimental results through trace-driven simulation in order to verify the effectiveness of our algorithm. Furthermore, some novel computing paradigms, such as transparent computing [35], may be considered in the future to provide safe, cross-platform and real-time services for heterogeneous smart terminals in the intelligent vehicular computing Networking era.

References

[1] C. Chen, X. Liu, H.-H. Chen, M. Li, and L. Zhao, "A rear-end collision risk evaluation and control scheme using a Bayesian network model," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 264–284, Jan. 2019.

[2] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang, and Y. Zhou, "Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2377–2396, 4th Quart., 2015.

[3] C. Chen, L. Liu, T. Qiu, K. Yang, F. Gong, and H. Song, "ASGR: An artificial spider-Web-based geographic routing in heterogeneous vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1604–1620, May 2019.

[4] T. Mekki, I. Jabri, A. Rachedi, and M. ben Jemaa, "Vehicular cloud networks: Challenges, architectures, and future directions," *Veh. Commun.*, vol. 9, pp. 268–280, Jul. 2017.

[5] E. Lee, E.-K. Lee, M. Gerla, and S. Oh, "Vehicular cloud networking: Architecture and design principles," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 148–155, Feb. 2014.

[6] M. Whaiduzzaman, M. Sookhak, A. Gani, and R. Buyya, "A survey on vehicular cloud computing," *J. Netw. Comput. Appl.*, vol. 40, pp. 325–344, Nov. 2013.

[7] C. Chen, J. Hu, T. Qiu, M. Atiquzzaman, and Z. Ren, "CVCG: Cooperative V2 V-aided transmission scheme based on coalitional game for popular content distribution in vehicular ad-hoc networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 12, pp. 2811–2828, Dec. 2019.

[8] Y. Dai, K. Zhang, S. Maharjan, and Y. Zhang, "Edge intelligence for energy-efficient computation offloading and resource allocation in 5G beyond," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12175–12186, Oct. 2020.

[9] C. Chen, Z. Liu, S. Wan, J. Luan, and Q. Pei, "Traffic flow prediction based on deep learning in Internet of vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3776–3789, Jun. 2020.

[10] J. Feng, F. Richard Yu, Q. Pei, X. Chu, J. Du, and L. Zhu, "Cooperative computation offloading and resource allocation for blockchain-enabled mobile-edge computing: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6214–6228, Jul. 2020.

[11] C. Chen, B. Liu, S. Wan, P. Qiao, and Q. Pei, "An edge traffic flow detection scheme based on deep learning in an intelligent transportation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1840–1852, Mar. 2021.

[12] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular edge computing and networking: A survey," *Mobile Netw. Appl.*, vol. 3, pp. 1–24, 2020.

[13] Y. Dai, K. Zhang, S. Maharjan, and Y. Zhang, "Deep reinforcement learning for stochastic computation offloading in digital twin networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 4968–4977, Jul. 2021.

[14] M. A. Salahuddin, A. Al-Fuqaha, M. Guizani, and S. Cherkaoui, "RSU cloud and its resource management in support of enhanced vehicular applications," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2014, pp. 127–132.

[15] K. Zheng, H. Meng, P. Chatzimisios, L. Lei, and X. Shen, "An SMDP-based resource allocation in vehicular cloud computing systems," *IEEE Trans. Ind. Electron.*, vol. 62, no. 12, pp. 7920–7928, Dec. 2015.

[16] T. Bahreini, M. Brocanelli, and D. Grosu, "Energy-aware resource management in vehicular edge computing systems," in *Proc. IEEE Int. Conf. Cloud Eng. (IC2E)*, Apr. 2020, pp. 49–58.

[17] F. Sun *et al.*, "Cooperative task scheduling for computation offloading in vehicular cloud," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 11049–11061, Nov. 2018.

[18] C. Zhu *et al.*, "Folo: Latency and quality optimized task allocation in vehicular fog computing," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4150–4161, Jun. 2019.

[19] T. Mekki, R. Jmal, L. Chaari, I. Jabri, and A. Rachedi, "Vehicular fog resource allocation scheme: A multi-objective optimization based approach," in *Proc. IEEE 17th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2020, pp. 1–6.

[20] H. M. Birhanie, M. A. Messous, S.-M. Senouci, E.-H. Aglzim, and A. M. Ahmed, "MDP-based resource allocation scheme towards a vehicular fog computing with energy constraints," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–6.

[21] L. Hou, K. Zheng, P. Chatzimisios, and Y. Feng, "A continuous-time Markov decision process-based resource allocation scheme in vehicular cloud for mobile video services," *Comput. Commun.*, vol. 118, pp. 140–147, Mar. 2018.

[22] R. I. Meneguette, A. Boukerche, A. H. M. Pimenta, and M. Meneguette, "A resource allocation scheme based on semi-Markov decision process for dynamic vehicular clouds," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[23] C.-C. Lin, D.-J. Deng, and C.-C. Yao, "Resource allocation in vehicular cloud computing systems with heterogeneous vehicles and roadside units," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3692–3700, Oct. 2018.

[24] H. R. Arkian, R. E. Atani, and A. Pourkhalili, "A cluster-based vehicular cloud architecture with learning-based resource management," in *Proc. IEEE 6th Int. Conf. Cloud Comput. Technol. Sci.*, Dec. 2014, pp. 162–167.

[25] H. Peng and X. Shen, "Deep reinforcement learning based resource management for multi-access edge computing in vehicular networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 4, pp. 2416–2428, Oct. 2020.

[26] R. Yu, Y. Zhang, S. Gjessing, W. Xia, and K. Yang, "Toward cloud-based vehicular networks with efficient resource management," *IEEE Netw.*, vol. 27, no. 5, pp. 48–55, Sep. 2013.

[27] R. Yu *et al.*, "Cooperative resource management in cloud-enabled vehicular networks," *IEEE Trans. Ind. Electron.*, vol. 62, no. 12, pp. 7938–7951, Dec. 2015.

[28] Z. Zhou, P. Liu, J. Feng, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Computation resource allocation and task assignment optimization in vehicular fog computing: A contract-matching approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3113–3125, Apr. 2019.

[29] H. M. Birhanie, S.-M. Senouc, M. A. Messous, A. Arfaoui, and A. Kies, "A stochastic theoretical game approach for resource allocation in vehicular fog computing," in *Proc. IEEE 17th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2020, pp. 1–2.

[30] P. Ghosh, N. Roy, and S. K. Das, "Mobility-aware efficient job scheduling in mobile grids," in *Proc. 7th IEEE Int. Symp. Cluster Comput. Grid (CCGrid)*, May 2007, pp. 701–706.

[31] L. Liu *et al.*, "Blockchain-enabled secure data sharing scheme in mobile-edge computing: An asynchronous advantage Actor–Critic learning approach," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2342–2353, Feb. 2021.

[32] L. Liu, C. Chen, T. Qiu, M. Zhang, S. Li, and B. Zhou, "A data dissemination scheme based on clustering and probabilistic broadcasting in VANETs," *Veh. Commun.*, vol. 13, pp. 78–88, Jul. 2018.

[33] H. Zhang, Q. Zhang, and X. Du, "Toward vehicle-assisted cloud computing for smartphones," *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5610–5618, Dec. 2015.

[34] C. Chen, X. Liu, T. Qiu, and A. K. Sangaiah, "A short-term traffic prediction model in the vehicular cyber–physical systems," *Future Gener. Comput. Syst.*, vol. 105, pp. 894–903, Apr. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X17311846

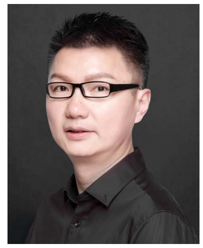
[35] Y. Zhang, S. Duan, D. Zhang, and J. Ren, "Transparent computing: Development and current status," *Chin. J. Electron.*, vol. 29, no. 5, pp. 793–811, Sep. 2020.

Wenting Wei (Member, IEEE) received the M.E. and Ph.D. degrees in telecommunication and information systems from Xidian University, in 2014 and 2019, respectively. Since 2019, she has been working with the State Key Laboratory of ISN, Xidian University. Her main research interests include data center networking, network virtualization, cloud computing, and intelligent transportation.

Ruying Yang received the M.E. degree in communication engineering from Xidian University, Xi'an, China, in 2019. Her major is telecommunication and information systems. Her research interests include vehicular networks and cloud computing.

Huaxi Gu (Member, IEEE) is currently a Professor with the State Key Laboratory of ISN, Xidian University. He is also the Leader of the Youth Innovation Team of Shaanxi Universities. He is leading a project as the principal investigator funded by the National Key Research and Development Program of China. He is also the principal investigator for one key, two general, and one youth project funded by the National Natural Science Foundation. He has published over 200 journal articles and conference papers, with his research interests being in the areas of networking technologies, network on chip, and optical interconnect. He served as a TPC Member of GLOBECOM, ICC, and PDCAT, and a Technical Reviewer for multiple journals, including IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI), IEEE TRANSACTIONS ON CLOUD COMPUTING, and IEEE/OSA JOURNAL OF LIGHTWAVE TECHNOLOGY.

Weike Zhao is a Junior Student with the School of Telecommunications Engineering, Xidian University, China. She was awarded the National Scholarship in 2020. Her major is information engineering. Her research interests include vehicular networks and data center networking.



Chen Chen (Senior Member, IEEE) received the B.Eng., M.Sc., and Ph.D. degrees in telecommunication from Xidian University, Xi'an, China, in 2000, 2006, and 2008, respectively. He was a Visiting Professor with the Department of EECS, University of Tennessee, and the Department of CS, University of California. He is currently a Professor with the Department of telecommunication, Xidian University, where he is also a member of the State Key Laboratory of Integrated Service Networks. He is also the Director of the Xi'an Key Laboratory of Mobile Edge Computing and Security and Intelligent Transportation Research Laboratory, Xidian University. He has authored/coauthored two books, over 100 scientific articles in international journals and papers in conference proceedings. He has contributed to the development of five copyrighted software systems and invented over 100 patents. He is also a Senior Member of China Computer Federation (CCF) and a member of ACM and Chinese Institute of Electronics. He serves as the general chair, the PC chair, and the workshop chair or a TPC member of a number of conferences.

Shaohua Wan (Senior Member, IEEE) received the joint Ph.D. degree from the School of Computer, Wuhan University, and the Department of Electrical Engineering and Computer Science, Northwestern University, USA, in 2010. Since 2015, he has been holding a post-doctoral position with the State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology. From 2016 to 2017, he was a Visiting Professor with the Department of Electrical and Computer Engineering, Technical University of Munich, Germany. He is currently an Associate Professor with the School of Information and Safety Engineering, Zhongnan University of Economics and Law. He is an author of over 100 peer-reviewed research articles and books. His main research interests include deep learning for the Internet of Things and edge computing.