SPECIAL SECTION ON INNOVATION AND APPLICATION OF INTELLIGENT PROCESSING OF DATA, INFORMATION AND KNOWLEDGE AS RESOURCES IN EDGE COMPUTING

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 19, 2020, accepted March 7, 2020, date of publication March 17, 2020, date of current version March 26, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2980567

A Task Offloading Solution for Internet of Vehicles Using Combination Auction Matching Model Based on Mobile Edge Computing

SHI YANG

School of Information Engineering, Changchun University of Finance and Economics, Changchun 130122, China

e-mail: teacheryangshi@126.com

ABSTRACT From a global perspective, an Internet of Vehicles task offloading solution based on mobile edge computing is proposed, which satisfies the application requirements (high reliability) strictly. The average time for completing a task can be minimized with the reasonable task offloading solution. Firstly, we model the wireless network, the transmission time and the movement of vehicles. Besides, heterogeneous wireless network architecture is adopted, data centers are deployed at Small-cell Base Stations, Macro-cell Base Stations and Internet. Then considering the limitedness, heterogeneity and task diversity of resources, we utilize matching model based on combination auction to design the offloading model. Furthermore, the multi-round sequential combination auction mechanism is proposed, which equals the matching problem to the multi-dimensional grouping knapsack problem and uses dynamic programming to get the optimal match. This solution is based on virtual machine technology and voltage scaling technology in the task execution time model. Moreover, the computing resources can be measured by CPU frequency. We propose an optimization problem for the shortest average task completion time with limited resources. Finally, the effects of these parameters (such as the number of tasks per unit time, the amount of data offloaded and the number of CPU cycles) on the task execution efficiency are analyzed and compared with other algorithms by simulation experiments. Compared to existing schemes, simulation results show that the proposed algorithm can reduce system overhead and shorten task execution time effectively.

INDEX TERMS Internet of Vehicles, task offloading, heterogeneous wireless network architecture, mobile edge computing, combination auction, channel gain.

I. INTRODUCTION

The Internet of Vehicles (IoV) is a typical application of Internet of Things in the automotive industry. By equipping vehicles with various sensors and communication modules, it is regarded as a next-generation intelligent transportation system with great potential. In recent years, the automotive industry is undergoing critical and huge changes. Many new automotive applications, services and ideas have been proposed, such as autonomous driving, safe driving and intelligent transportation, digital services for transportation and logistics, intelligent navigation and entertainment office services [1]–[4]. At the same time, IoV in the construction of smart cities is increasingly becoming an important part of the network. Millions of roadside units (RSUs) and vehicles equipped with embedded devices can form IoV that integrates data communication, transmission and computing. New problems continue to appear with the expansion of IoV scale and the continuous upgrading of demand, we need to study and solve it urgently. Most of newly-developing automotive applications involve video or image processing technology, which requires strong data processing capabilities. For example, intelligent navigation services use Augmented Reality (AR) and real-time video streaming technologies in the future. And it considers data from sensors on themselves and surrounding vehicles to analyze the full range of traffic conditions at intersections that is shown on the vehicle's windshield. The service is very helpful to drivers at complex intersections. It can help them make choices in advance and reduce the burden. However, the sensor information of surrounding vehicles needs to be continuously processed in the process of providing services. The amount of data is very large and the

The associate editor coordinating the review of this manuscript and approving it for publication was Honghao Gao.

task is difficult to be processed separately on the vehicle system. Thus, the solution is the task is offloaded to cloud servers for processing.

Although mobile device computing power, running memory and other configurations are becoming more and more powerful, they are still insufficient for compute-intensive tasks. This inspires the development of mobile cloud computing (MCC) [5], [6]. Mobile devices offload tasks to the Internet cloud through core network of mobile operators in MCC, use its powerful computing and storage resources to perform tasks. Mobile edge computing (MEC) evolved from MCC and was first proposed by the European Telecommunications Standards Institute in 2014. It greatly reduces data processing time and energy consumption of mobile devices by deploying computing resources, network control functions and cached data near Small-cell Base Station (SBS) and Macro-cell Base Station (MBS) [7]. Due to the limited computing power of mobile devices, vehicles or users can offload computation-intensive tasks to network edge access points, such as base stations and wireless access points in MEC system. Besides, the tasks are processed by edge server that can greatly reduce the data transfer time compared with MCC. Moreover, MEC has the characteristics of close range, ultra-low latency, ultra-high energy efficiency and ultra-high reliability. At the same time, it is also a key technology for 5G [8], [9].

Task offloading means that vehicles offloads tasks that it cannot handle to the data center for processing. Resource allocation mainly refers to the allocation of wireless channels, wired channels, server computing and cache resources. Offloading tasks are closely related to the problem of resource allocation. And the issue of reasonable allocation of resources must be considered after making the task offloading decision. For example, when tasks are offloaded, how many radio sub-channels or time slots are allocated in the radio access network to transmit them, how much bandwidth a wired channel needs to be allocated in a back-haul network and how much computing resources the data center finally allocates to handle this task. These need to be properly planned in order to make efficient use of resources [10]. The reasonable allocation of resources determines the efficiency of task execution and users' service experience. The transmission bandwidth depends on the least bandwidth of multiple paths in the process of data offload. In general, wired transmission is easy to expand the bandwidth capacity by laying optical fibers, and it will not become a limiting factor for transmission rate. However, the wireless resources are relatively small, which easily becomes a bottleneck that limits transmission speed. Therefore, in this paper, it is assumed that the wired transmission rate in backhaul network is equal to the wireless transmission rate in access network. In the task offloading, we only consider the impact of wireless resource allocation on transmission time.

In the study of task offloading and resource allocation, many factors need to be considered for modeling task offloading of MEC: (1) To model the task, we need to consider whether it is a single-user or multi-user scenario, whether each user has multiple tasks, whether user task can be divided into multiple subtasks and whether the subtasks are interdependent. (2) The system architecture is also a factor that must be considered during task offloading. This paper considers heterogeneous wireless networks and the situation of that data centers are deployed in SBS, MBS and Internet. (3) The choice of wireless transmission mode: FDMA, TDMA, CDMA. (4) Whether the data center computing resources can be regarded as infinite. In addition, the following key challenges exist when the requirements of IoV and MEC technology are combined:

1) How to utilize the short-distance characteristics of edge server to make a reasonable task offloading decision, which based on the amount of data offloaded and the computing resources. And how to improve the efficiency of the system's task execution;

2) How to ensure service reliability and task execution efficiency in a heterogeneous wireless network environment;

3) To avoid the shortage of wireless and computing resources when user requesting peak periods.

Based on these challenges, this paper proposes an IoV task offloading solution for MEC, which based on combination auction matching model in heterogeneous wireless network. The main contributions of this paper are as follows:

(1) Heterogeneous wireless network is combined with MEC technology, task offloading, wireless and computing resource allocation issues are considered comprehensively. It is more comprehensive and closer to the actual situations.

(2) An offloading model is constructed to deploy multiple requesting vehicles and service nodes. Considering the limited resources, heterogeneity and task diversity, the offloading model is designed combination auction-based matching model. It further equals the matching problem to Multi-dimensional grouping knapsack problem and the optimal matching is obtained by dynamic programming. Finally, an optimization problem with the shortest average completion time is proposed with limited resources.

II. RELATED RESEARCH

In addition to data processing requirements, vehicle applications will have strict requirements on network bandwidth and task latency in the future. For example, driverless applications in IoV require 5 $\sim$ 10ms network delay; existing intelligent navigation using AR technology requires 20Mbit/s $\sim$ 200Mbit/s network speed. In order to prevent users from feeling disoriented or dizzy, the network latency must be less than 20ms. The on-vehicle application that obtains bird's-eye view of a crossroads requires approximately 40Mbit/s network rate and approximately 50ms network latency. The above applications have high requirements on delay and bandwidth and the existing mobile networks cannot provide satisfactory services obviously. Figure 1 shows the requirements for communication rate and delay of on-vehicle applications. From Figure 1, it can be seen that IoV is a typical "low-latency, high-bandwidth, high-reliability" application scenario in the future.

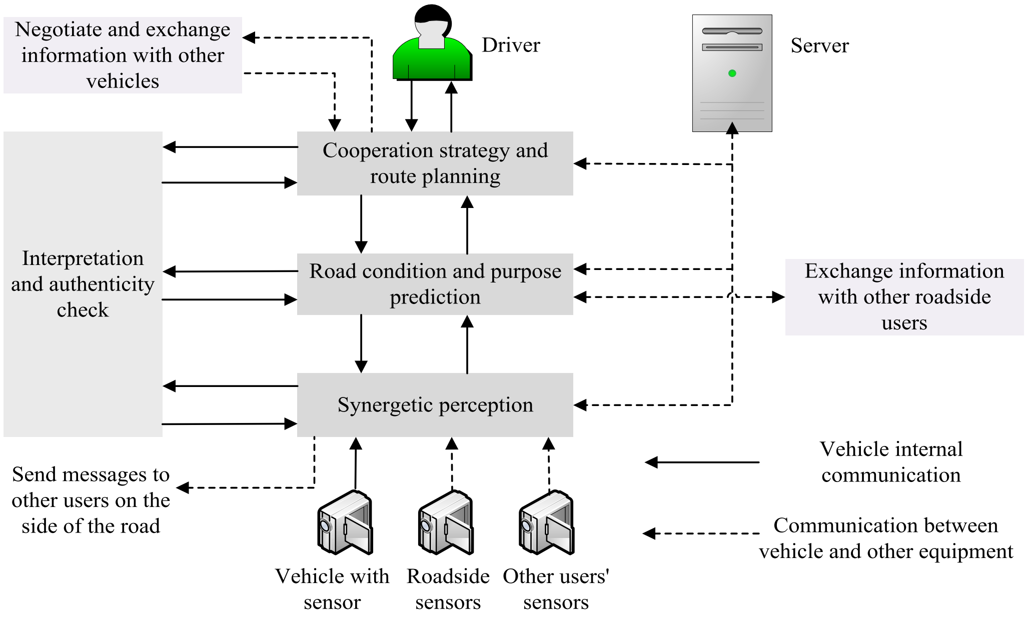
Negotiate and exchange information with other vehicles
Driver
Server
Cooperation strategy and route planning
Interpretation and authenticity check
Road condition and purpose prediction
Exchange information with other roadside users
Synergetic perception
Send messages to other users on the side of the road
Vehicle with sensor
Roadside sensors
Other users' sensors
Vehicle internal communication
Communication between vehicle and other equipment

FIGURE 1. The requirements for IoV application delay.

MEC evolves from MCC, which provides IT service environment and cloud computing capabilities on the wireless access network side near mobile users. In the MEC environment, users are closer to edge server, thus transmission delay caused by task offload is greatly reduced. And service requests can be responded at the edge, which can alleviate the burden on core network effectively. In the past two years, MEC draws lots of attention from researchers due to short distance, ultra-low latency and high bandwidth characteristics. Aiming at the task offloading and resource allocation, researchers proposed different solutions considering various requirements and application scenarios.

Due to too many factors that need to be considered in task offloading and resource allocation, it is difficult to take all factors into consideration. Therefore, the existing work simplified the modeling of task offloading to some extent. Some work only studied that tasks were offloaded to the edge server and proposed two task offloading models, namely, two-state offloading and partial task offloading models. Literature [11]–[13] discussed whether user tasks should be performed locally or in the cloud. Specifically, literature [11] taken into account the allocation of wireless resources for the purpose of energy saving. It assumed that the computing capacity of the server was a fixed constant and performed offloading by classifying different tasks. Besides, priorities are given to tasks based on the weighted sum of their latency, radio resource requirements and energy consumption. Literature [12] and [13] both aimed to minimize the weighted sum of energy consumption and delay. Literature [12] was considered more comprehensively since each user had multiple tasks. Literature [13] adopted the game theory to solve the optimization problem and proved the existence of Nash equilibrium. Literature [12] calculated the theoretical upper limit of server-side task processing and proved that his algorithm can approach the theoretical value very well. It transforms non-convex quadratic functions under quadratic constraints into separable semidefinite programming problems through relaxation. Literature [14] proposed a compromise solution. A task can be processed locally and then offloaded to the cloud to execute the rest. However, the task delay is only used as a reference condition and the delay of each task cannot be guaranteed in the above work. Literature [15] considered the task offload and the allocation of computing resources. Assuming that the wireless bandwidth is a fixed constant, the task execution cost is minimized while meeting strict time constraints of the task. Literature [16] used game theory to allocate the computing power of MEC servers under the best decision of each user (the user’s revenue is the largest), which can maximize the revenue of operators. Literature [17], [18] proposed the allocation of wireless channels and computing resources with meeting the delay, so that the energy consumption of users was minimized. Literature [19] used Markov decision model to allocate resources, which can ultimately reduce the delay, cannot guarantee the delay. The energy consumption was minimized under the constraints of delay and limited cloud computing resources in literature [20]. However, these works allocated resources based on cost or energy saving purposes. And there are few solutions based on efficient use of resources and high reliability. In the IoV scenario, energy consumption becomes a secondary factor, improving system reliability and considering execution efficiency are the most important issues.

In order to integrate with actual LTE network, there are also a few researches on the MEC system of heterogeneous wireless networks. Literature [21] proposed a wireless resource allocation scheme in the context of heterogeneous infinite

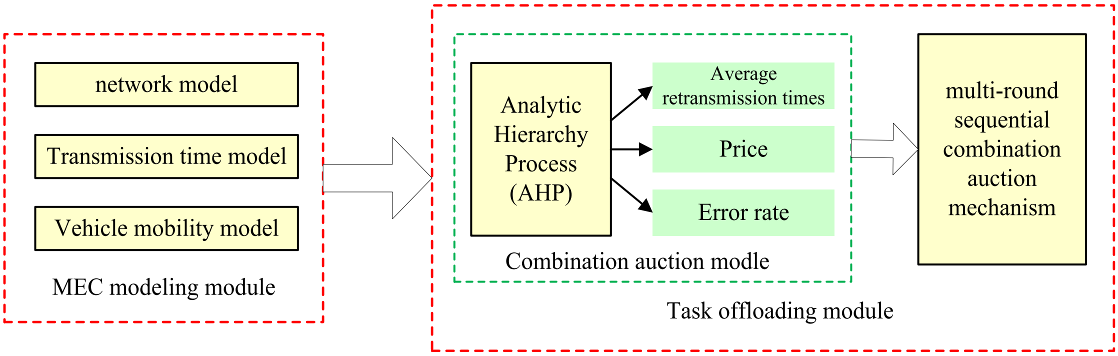
network model
Transmission time model
Vehicle mobility model
MEC modeling module
Analytic Hierarchy Process (AHP)
Average retransmission times
Price
Error rate
Combination auction modle
multi-round sequential combination auction mechanism
Task offloading module

FIGURE 2. Overall framework and problem modeling of the proposed solution.

networks. It increased the probability of successful execution with strict delay requirements by 40%. In [22], the interference between the MBS and the SBS is reduced by periodically pausing the MBS to transmit signals. And the wireless rate of multiple users is maximized. Literature [23] proposed a random self-organizing algorithm based on Markov chain and game theory to solve the problem of wireless resource allocation. The purpose of which is to minimize the operation cost. Literature [24] used time division multiple access and orthogonal frequency division multiple access technology to allocate time slots or sub-channels of wireless channels. It satisfied the task delay, which minimizes energy consumption of mobile users.

However, most of the existing work consider the finiteness and ignore the heterogeneity of resources. Resource heterogeneity is defined as the difference in demands between users. For example, vehicles have strict information reliability and timeliness requirements for services such as car navigation. However, they have higher throughput and fairness requirements for interactive services such as online games.

III. OVERALL FRAMEWORK AND PROBLEM MODELING OF THE PROPOSED SOLUTION

A. OVERALL FRAMEWORK

In order to provide vehicles with better quality of experience (QoE), we should consider how to get the best match between the service nodes and the requesting vehicles. Therefore, it is ensured that the economic benefit of requesting vehicles is increased and the economic benefit of service nodes are maximized on the basis that they all meet the budget equilibrium. The research work of this paper is: from a global perspective, we propose an IoV task offloading solution under the premise of strictly meeting the application requirements (high reliability). And the average time for completing a task can be minimized by a reasonable task offloading scheme. The overall framework is shown in Figure 2.

The average time for completing a task can be minimized by a reasonable task offloading scheme. Firstly, the wireless network, transmission time and vehicle movement are modeled. In the network model, each user has a task that needs to be unloaded. The task consists of three parameters: the amount of offloaded data, the number of CPU cycles required for calculation task and the maximum allowable completion time. In the transmission time model, it is assumed that vehicles and base stations communicate through TDMA. For vehicle movement modeling, we built an unloading model that deployed multiple requesting vehicles and multiple service nodes. Then considering the limited resources, heterogeneity and task diversity, the framework is modeled as combined auction model. According to the environment in which vehicles are located and the type of request task, this paper chooses the price, bit error rate and average retransmission times as the determining factors. The hierarchy can determine the priority order value ("satisfaction") of requesting nodes for service nodes according to the weighting coefficient of each factor. In this way, the best choice can be achieved.

Although a few works studied the MEC system of heterogeneous wireless networks, the research purpose is different from this paper. The research purpose of this paper is different from other literatures: In the IoV scenario, energy consumption is not a determinant factor for determining resource allocation. This paper aims to improve the efficiency of system task execution. However, task offloading schemes of many existing works considered energy consumption as a key factor.

B. NETWORK MODEL

In the system shown in Figure 3, the MBS is connected to Internet through core network in the cellular communication system. MEC servers are deployed at the MBS and the SBS. In this paper, it is assumed that the SBS is connected to the MBS in a wired manner. Because the interference between MBSs is small, we assume that a MBS and a MBS have n SBSs in the network coverage area. The SBS set is represented by $N = \{1, 2.3, \ldots, N\}$. There are K_n vehicles under the SBS n, and the vehicle set is represented by $K_n = \{1, 2.3, \ldots, K_n\}$. In this paper, we consider single-antenna vehicles and SBSs.

We assume that each vehicle has a computationally intensive and latency-intensive task to complete in a unit of time. Each vehicle can offload calculations to MEC servers via the connected SBS or MBS. Each vehicle offloads a task and the task offloaded by vehicle K_n is:

$$T_{k_n} = \left\{D_{k_n}, C_{k_n} T_{k_n}^{\max}\right\} \tag{1}$$

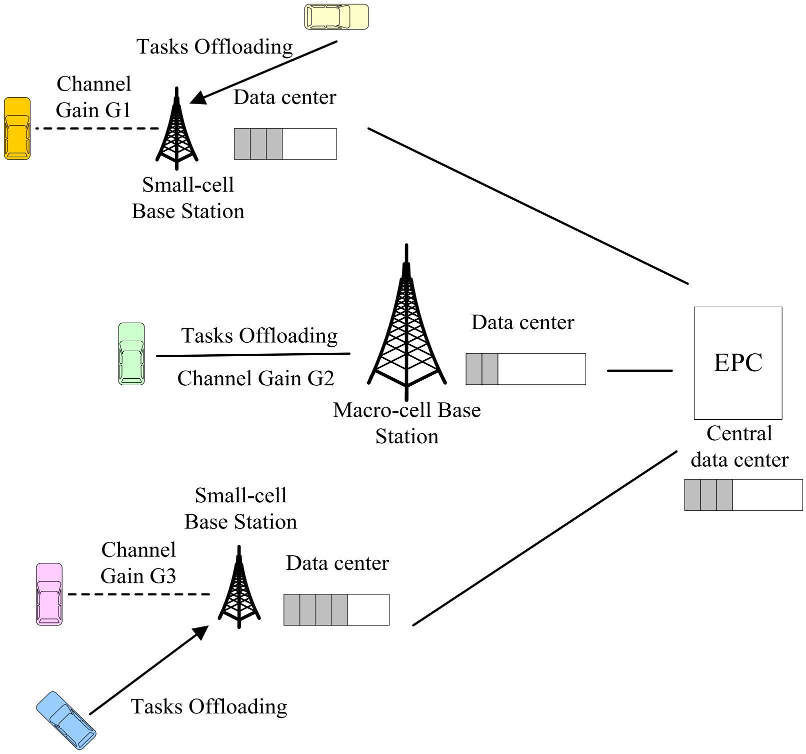

FIGURE 3. The diagram of resource usage in MEC system.

Specifically, D_{k_n} represents the amount of data offloaded by the task, C_{k_n} represents the number of CPU cycles required by server to process the task and $T_{k_n}^{\max}$ represents the maximum completion time allowed for the task. During the task unloading process, vehicles are constantly moving and the problem of handover to the base station may occur. In this paper, the calculation-intensive, ultra-low-latency task unloading is considered and $T_{k_n}^{\max}$ is less than tens of milliseconds. Therefore, it is assumed that no base station handover during task unloading process.

C. VEHICLE MOVEMENT MODEL

For vehicle movement, $\gamma(k)$ is the set of vehicle movement directions with time-tolerant data. In this paper, only the movement of roads and vehicles (or the data contained in it) along the south-north, east-west directions is considered. Therefore, $\gamma(k)$ includes 5 possible cases: north (N), south (S), east (E), west (W) and Stationary RSU direction (F), so $\gamma(k) \in \{N, S, E, W, F\}$.

In this paper, Manhattan mobility model is selected as mobility model of vehicles in the proposed network architecture. Suppose the density of vehicles and road intersections are ρ_{veh} and ρ_{int}. Respectively, the speed of vehicles' movement is v, the time of vehicles' random movement is t_{move} and the uniform distribution $U\left(0, \frac{1}{\rho_{\text{int}} v}\right)$ is satisfied. Therefore, the probability density function of vehicles' random movement time can be written as

$$f_{T_{move}}(t_{move}) = \begin{cases} \rho_{\text{int}} v, & t_{move} \in \left[0, \dfrac{1}{\rho_{int} v}\right] \\ 0, & others \end{cases} \tag{2}$$

For a vehicle arriving at a road intersection, let P_{wait} denote the probability that the vehicle needs to wait and random waiting time of the vehicle at intersection is denoted by t_{wait}. If the random waiting time satisfies a uniform distribution $U(0, T_{wait})$, the probability density function can be expressed as

$$f_{T_{wait}}(t_{wait}) = \begin{cases} \dfrac{1}{T_{wait}}, & t_{move} \in [0, T_{wait}] \\ 0, & others \end{cases} \tag{3}$$

Thus, the probability of the vehicle moving and stopping can be expressed as

$$P_{move} = \frac{\frac{1}{\rho_{\text{int}} v}}{\frac{1}{\rho_{\text{int}} v} + \frac{T_{wait} P_{wait}}{2}} = \frac{2}{2 + T_{wait} P_{wait} \rho_{int} v} \tag{4}$$

$$p_{shop} = 1 - P_{move} = \frac{T_{wait} P_{wait} \rho_{\text{int}} v}{2 + T_{wait} P_{wait} \rho_{\text{int}} v} \tag{5}$$

where: $1/\rho_{\text{int}} v$ is the time for the vehicle to move from one intersection to the next; $T_{wait}/2$ is the average waiting time at intersection; $1/\rho_{\text{int}} v + T_{wait} P_{wait}/2$ is the total time for the vehicle to move and wait.

In addition, it can be divided into two types as its neighbors for the vehicle A driving on the road: If vehicle B and vehicle A travel in the same direction and stay within communication range, vehicle B can be used as the vehicle A's Neighbors and establish a direct communication link. If vehicle A is stationary at the intersection and other vehicles waiting at the intersection or roadside RSU are within communication range, these vehicles and RSU can be neighbors of vehicle A.

And they can communicate with vehicle A Establish a wireless communication link. Except for these two cases, vehicle A will not be able to choose a neighbor to establish a wireless communication link with its neighbor in the connected vehicle.

D. TRANSMISSION TIME

In this paper, we consider that the MBS and SBS spectrum overlap, which means that there is interference between the two base stations. Compared with the interference between them, the interference between the SBSs is negligible. Because the SBS is deployed, the operators adjust transmit power of base station so that the coverage areas are less overlapping [25], [26]. This paper assumes that vehicles and base stations communicate through Time Division Multiplexing (TDM) technology and each vehicle offloads data using time slots allocated by the base station. In the future vehicle networking scenario, the task offload data is much larger than the download data. So only the channel allocation problem offloaded from vehicles to base stations is considered. Vehicle A is connected to its corresponding SBS and neighboring vehicle B is connected to its corresponding MBS. Then the wireless communication between vehicle A and B interferes with each other, so that the transmission rate decreases.

MEC servers are deployed at each SBS or MBS in the network. Vehicles can access the two base stations. Thus, the computing tasks are offloaded to MEC servers for execution [27]. Each vehicle can choose to access either the corresponding SBSs or MBSs. We use a_{k_n} to represent the time slot size used by vehicle k_n as a percentage of unit time, so there is $0 \leq a_{k_n} \leq 1$. Let $b_{k_n} \in \{0, 1\}$ denote the access choice of vehicles, $b_{k_n} = 0$ denote the SBS where vehicles accesses its range and $b_{k_n} = 1$ denote the MBS where vehicles accesses its range.

1) TRANSMISSION TIME WHEN ACCESSING SBSs

Assume that the bandwidth of SBSs is B_s. According to Shannon's theorem, the total transmission rate of SBSs is

$$R^s = B_s \log_2 \left(1 + \frac{G^S_{k_n} p^S}{1 + \sigma^2}\right) \tag{6}$$

Specifically, $G^S_{k_n}$ is the channel gain between SBS n and vehicle k_n, p^S is the transmission power when SBS n communicates with vehicle k_n, σ^2 is the wireless channel noise power, and I is the interference power between MBSs and SBSs. When vehicle k_n accesses SBS n, i.e., $b_{k_n} = 0$, the transmission rate between vehicles and base stations is

$$R^S_{k_n} = a_{k_n} R^s = a_{k_n} B_s \log_2 \left(1 + \frac{G^S_{k_n} p^S}{1 + \sigma^2}\right) \tag{7}$$

The corresponding transmission delay is

$$t^{ST}_{k_n} = \frac{D_{k_n}}{R^S_{k_n}} = \frac{D_{k_n}}{a_{k_n} R^s} \tag{8}$$

2) TRANSMISSION TIME WHEN ACCESSING MBSs

Similarly, assume that the MBS bandwidth is B_M. According to Shannon's theorem, the total transmission rate of MBSs is

$$R_M = B_M \log_2 \left(1 + \frac{G^M_{k_n} p^M}{1 + \sigma^2}\right) \tag{9}$$

Specifically, $G^M_{k_n}$ represents the channel gain between MBSs and vehicle k_n and p^M represents the transmission power when MBS communicates with vehicle k_n. σ^2 is the wireless channel noise power. When vehicle k_n enters MBSs, that is $b_{k_n} = 1$, the transmission rate is

$$R^M_{k_n} = a_{k_n} R^M = a_{k_n} B_M \log_2 \left(1 + \frac{G^M_{k_n} p^M}{1 + \sigma^2}\right) \tag{10}$$

The corresponding transmission delay is

$$t^{MT}_{k_n} = \frac{D_{k_n}}{R^M_{k_n}} = \frac{D_{k_n}}{a_{k_n} R^M} \tag{11}$$

IV. COMBINATION AUCTION MODEL BASED ON ANALYTIC HIERARACY PROCESS (AHP)

A. ANALYTIC HIERARACY PROCESS

According to the environment in which vehicles are located and the type of request tasks, this paper selects price, bit error rate and average retransmission times as the judging factors. The hierarchy can determine the priority ("satisfaction") of requesting nodes for serving nodes based on the weight of request tasks for each factor. So that the model reaches the optimal selection. The following describes the process of satisfaction analysis based on AHP in detail.

(1) Establishing hierarchical structure model: By decomposing the problem into target layer, criterion layer and solution layer, a complex decision problem with multiple indicators is solved. The hierarchical structure model established is shown in Figure 4.

(2) Construct judgment matrix: Compare the indicators with each other firstly. Judge its importance to the target layer, that is, the weight of each standard. Then a pairwise comparison is performed between the scheme layers to obtain the local weight of the indicator for each indicator. The criterion layer judgment matrix $M = \{m_{ij}\}$, and element m_{ij} represents the importance of the i-th index compared to the j-th index. Regarding m_{ij}, the best results are usually obtained on a scale of 1 to 9, see Table 1.

(3) Calculating weights: This paper uses the eigenvector method to calculate weights. Calculate the maximum eigenvalue $\lambda_{\max}$ of the judgment matrix and the eigenvector corresponding to the maximum eigenvalue. Normalize the eigenvectors to get weights. By multiplying local weights of each element with weights of the corresponding index, the final weight is obtained. That is, satisfaction.

(4) Consistency test: Due to the subjectivity of judgment matrix, the consistency is usually not fully satisfied. Therefore, the consistency ratio CR is used to detect judgment

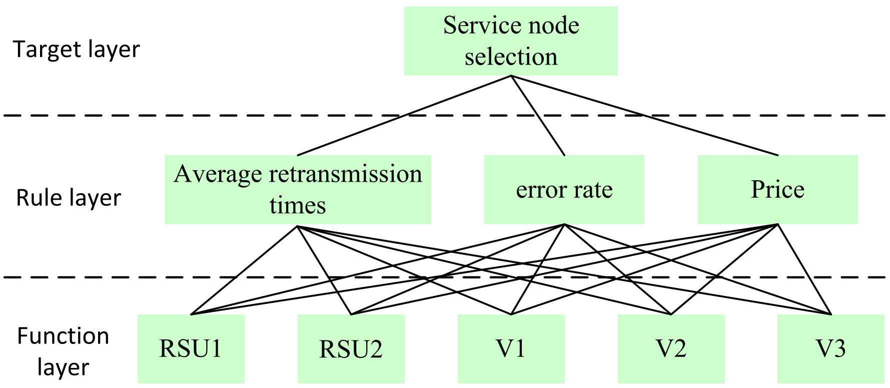

FIGURE 4. Structural model diagram of AHP.

TABLE 1. Bid winning degree and meaning of judgment matrix.

	Equal	Strong	Stronger	Very strong	Absolutely strong
m_{ij}	1	3	5	7	9

TABLE 2. Average random consistency index RI.

n	1	2	3	4	5
RI	0	0.61	0.93	1.09	1.27

errors. CR is the ratio of consistency index CI to the average random consistency index RI.

$$CI = (\lambda_{\max} - n)\big/(n-1) \tag{12}$$

$$CR = CI\big/RI \tag{13}$$

Here, n is the number of indicators. The values of RI are shown in Table 2.

B. PROBLEM STATEMENT

The service vehicle broadcasts its status periodically and the requesting vehicle to grasp service node information through wireless broadcast. The requesting vehicle issues request information to a service node that can increase its benefits. Define the benefit when the requesting vehicle to offload tasks to service nodes, as shown in equation (14):

$$u_i = w_k^i \left\{\mu \ln\left(\tau D_{ij} + \theta\right) + (1-\mu)\, e_0 D_{ij} - p_{ij}\right\} \tag{14}$$

where, w_k^i indicates that requesting vehicle benefits is related to satisfaction. The first item in brackets indicates the computational resource efficiency saved by offloading tasks, and the second item represents the computational resource efficiency gain saved by executing computing tasks at remote end. $p_{ij} = p_{ij}^c + p_{ij}^b$ represents the cost paid by p_{ij}^c to p_{ij}^c. μ is the unloading factor, and $0 < \mu < 1$, τ and θ are the coefficients. e_0 is the unit resource price defined by requesting vehicles, which indicates how much requesting vehicles prefers requesting resources. $D_{ij} = c_i + b_i$ represents the sum of resources obtained by vehicle v_i at service node g_j. Here, because the number of CPU cycles and bandwidth belong to different orders of magnitude, we use the form of data homogenization to express its size as a percentage of its respective total resources and then sum them.

The optimization goal is to maximize the benefits of service nodes while improving the efficiency of requesting vehicles under the constraints of system requirements and delays. The programming problem is as follows equation (15):

$$\left.\begin{aligned}
&\max U_j = \sum_{v_j \in V, g_j \in G, \rho_k \in N} \sum_{i=1}^{L} \rho_k \left(p_{ij} - \varphi D_{ij}\right)\\
&s.t.\ C1: \sum_{i=1}^{L} x_{ij} c_i \le C_j, \quad v_j \in V,\ g_j \in G\\
&C2: \sum_{i=1}^{L} x_{ij} b_i \le B_j, \quad v_j \in V,\ g_j \in G\\
&C3: r_{ij} \ge \zeta\\
&C4: t \le t_{\max}^c\\
&C5: \sum_{I=1}^{L} x_{ij} \le 1, \quad v_j \in V,\ g_j \in\\
&C6: \mu \frac{c_i}{c_0} \ln\left(\tau D_{ij} + \theta\right) + (1-\mu)\, e_0 D_{i,j}\\
&\quad \ge p_{ij}, \quad v_i \in V,\ g_j \in G
\end{aligned}\right\} \tag{15}$$

where, φ represents the unit resource price defined by service nodes, ρ_k represents the probability of accessing the k type request task. Constraints C1 and C2 indicate that the computing and wireless resources available to serving node g_j are limited; C3 is the transmission rate requirement, ζ is the minimum transmission rate threshold; C4 is the delay requirement and the total delay of entire offloading process must not be greater than $t_{\max}^c$; Indicates that a requesting vehicle is served by at most one service node, and one service node can serve multiple vehicles; C6 guarantees that the benefits will not be reduced when vehicle unloads tasks to service nodes.

C. VALUATION AND BID PRICES

Each requesting vehicle has an estimate of its requested resources that reflects the vehicle's preference for resources.

The evaluation functions of computing resources and wireless resources are defined as linear functions of the number of CPU cycles and bandwidth.

$$\left.\begin{array}{c}\dfrac{z_i^c(x)}{x}=\dfrac{z_i^c(y)}{y},\quad \forall v_i\in V\\ \forall x,y\in\{1,2,\cdots,C_i\}\end{array}\right\}\tag{16}$$

where, $z_i^c(x)$ and $z_i^c(y)$ represent requesting vehicle v_i's estimates of x and y CPU cycles respectively. In order to express the valuation function more rationally, we add satisfaction. Therefore, request vehicle v_i to estimate the computing resources of kth type task request as

$$z_i^c(x)=w_k^i e_{\max}^c x\tag{17}$$

where, $e_{\max}^c$ calculates the resource price for the largest unit that requesting vehicle v_i is willing to pay. Similarly, requesting vehicle v_i estimates the wireless resource of kth type task request as

$$z_i^b(x)=w_k^i e_{\max}^b x\tag{18}$$

where, $z_i^b(x)$ indicates that requesting vehicle v_i estimates the bandwidth of x, which is the maximum unit radio resource price that v_i is willing to pay. In summary, when v_i to issue the kth type task request, the estimated value of requesting resource $\{x,y\}$ is

$$z_i(x,y)=w_k^i\left(e_{\max}^c x+e_{\max}^b y\right)\tag{19}$$

This article uses a price increase auction, and the initial bidding price is half of the resource valuation.

V. MULTI-ROUND SEQUENTIAL COMBINATION AUCTION MECHANISM

We use combination auction model to build the system model, which is mainly composed of 4 factors: seller, commodity, buyer and decision maker. Here, the seller is service nodes with certain resources. The buyer intends to purchase products in order to request vehicles—combine resources and perform tasks. The decision maker is service nodes and determines the winning vehicle and the amount of money that it needs to pay to service nodes. The combination auction model is shown in Figure 5.

The auction model can find valid matching problems between requesting vehicles and service nodes. Service nodes that owns computing and wireless resources wants to lease these resources to requesting vehicles. And requesting vehicles intend to purchase resources from service nodes and complete the task's calculation. When a service node establishes a match with a requesting vehicle, the requesting vehicle should pay the service node. Moreover, payment is determined by the service node, so the offloading model is modeled as combination auction.

A. MULTI-ROUND SEQUENTIAL COMBINATION AUCTION OFFLOADING MECHANISM

When the service node determines that it has sufficient heterogeneous resources, provides services and calculations for requesting vehicles, the auction transaction is established after determining own benefits. The requesting vehicle pays corresponding fee to service nodes when it gets service.

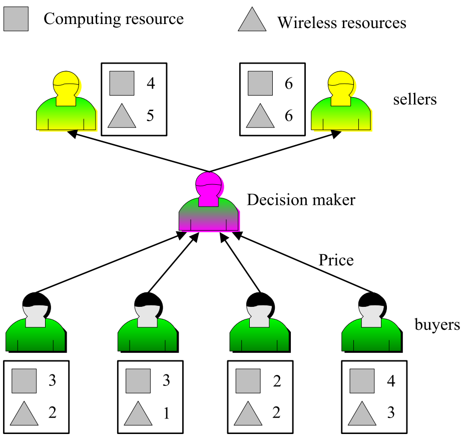

FIGURE 5. The combination auction model.

The highest round of auction is defined as M, that is, the number of service nodes. Requesting vehicles first use AHP to calculate satisfaction and determine the priority of release tasks. Then, they sequentially submit the resource requirements and bid price to the service nodes within communication range. After receiving the vector information from requesting vehicles, service nodes perform the winner decision step: Check its local wireless and computing resources, calculate its own benefits. In this step, a multi-dimensional packet backpack algorithm is used to determine requesting vehicle v_i served by service nodes. In each round of auctions, some requesting vehicles with lower bids will be added to unbid matrix F (the matching request vehicle set is not obtained). In order to be competitive, requesting vehicles in the unbid matrix increase bids in the next round price. The whole mechanism process satisfies three auction properties: personal rationality, balanced budget and user honesty.

The multi-round sequential combination auction offloading mechanism is detailed as follows:

(1) Hierarchical analysis method ranking: establish hierarchical model as shown in Figure 3 by quantifying each element. requesting vehicle v_i prioritizes service nodes as $W^i=\left\{w_1^i,w_2^i,\cdots w_M^i\right\}$ by using hierarchical ranking method.

In order to maximize the benefit of service nodes, these nodes need to be sorted. In this paper, the distance average method is used to determine: service nodes calculate the distance between requesting vehicles. Then we average the distances and finally they are serviced according to the average.

(2) Task bidding: Requesting vehicles sequentially provide requesting tasks to service nodes. After receiving requesting information, service nodes use the multi-dimensional packet backpack algorithm to decide which vehicles to serve. Vehicles that are currently not served by service nodes will be merged into the unbid matrix. These vehicles continue to be sorted in the next round and then bidding tasks. In order to

increase competitiveness, they will increase prices according to a certain gradient.

(3) Winner decision: Service node g_j receives requesting information from multiple vehicles. They need to choose a vehicle for service with limited resources and maximize their benefits. Requesting vehicles needs to pay service nodes at the same time. After requesting vehicles receive the information that service nodes can serve for it, it needs to judge their benefits. If the benefits are reduced, then this round of task offloading is abandoned.

In the unbid matrix, requesting vehicle $F = \{v_1, v_2, \cdots v_f\}$ provides a task offloading request $\{C_j, B_j\}$ to service node g_j with the resource status $\left\{c_{ij}, b_{ij}, p_{ij}^c, p_{ij}^b\right\}$. First, determine whether the first requesting vehicle is unloaded. If yes, that is, $x_{1j} = 1$, the problem turns into maximum service node capacity $\{C_j - c_{1j}, B_j - b_{1j}\}$ problem. If not, the problem is a multi-dimensional grouped knapsack problem with a capacity of $\{C_j, B_j\}$ to get the optimal decision. The matching problem is defined as knapsack problem. Here, a dynamic programming method is used to determine whether the i-th vehicle is unloaded. The idea is as follows:

Phase i: In the first i requesting vehicles, select several vehicles for unloading;

Status: Among the first i requesting vehicles, several vehicles are selected for offloading tasks to service nodes with remaining capacity of $\{C_j, B_j\}$ and make them maximize their benefit;

Decision: Whether or not the i-th requesting vehicle is unloaded, dynamic transfer equation is:

$$\begin{aligned} &U_j(i, j, c, b) \\ &\quad = \max\left\{U_j(i, j-1, c, b), U_j\left(i, j-1, c-c_{ij}, b-b_{ij}\right)+p_{ij}\right\} \end{aligned} \tag{20}$$

B. TASK EXECUTION TIME

In previous cloud computing models, server calculation delays can be negligible compared with communication transmission and local calculation delays [28], [29]. However, because its server resources are relatively small and offloading requests of vehicles are mostly computation-intensive tasks, the calculation time of edge servers cannot be ignored in MEC. The allocation of server computing resources is generally measured by CPU frequency. This can be achieved through virtual machine technology or Dynamic Frequency and Voltage Scaling (DFVS) technology [30]. Specifically, servers allocate different virtual machines to different vehicles, allowing independent computing. There are M servers in the data center and multiple virtual machines are virtualized on each server:

The computing resources in the system are represented by $C = \{C_S, C_M, C_C\}$, where C_S indicates the data center computing resources at SBSs, C_M indicates the data center computing resources at MBSs, and C_C indicates the data center computing resources at SBSs. We use $\varepsilon_{k_n} = \left\{e_{k_n}^S, e_{k_n}^M, e_{k_n}^C\right\}$ to indicate the calculation task offloading selection of vehicle k_n, where $e_{k_n}^S, e_{k_n}^M, e_{k_n}^C$ are all taken from 0 and 1. $e_{k_n}^S, e_{k_n}^M, e_{k_n}^C$ respectively indicate whether the task is offloaded to SBSs, MBSs, or data center at Internet. The value is 1 that indicates unloading to the data center here, otherwise the value is 0. Because tasks can only be offloaded to one data center for execution, the constraints satisfy:

$$e_{k_n}^S + e_{k_n}^M + e_{k_n}^C = 1 \tag{21}$$

When the sum of these three values is 0, it indicates that the task offloading failed. No one server can provide computing resources required for the task. After vehicle k_n task is unloaded to servers, the server performs calculation delay $t_{k_n}^E$ as

$$t_{k_n}^E = \frac{c_{k_n}}{f_{k_n}} \tag{22}$$

where f_{k_n} represents the CPU frequency allocated by the server to vehicle k_n.

C. COMPLETION TIME OF TASK OFFLOADING

When vehicle k_n is connected to SBS n and its tasks are offloaded to the server at SBSs for execution, that is, when $b_{k_n} = 1$ and $e_{k_n}^S = 0$. The total delay of task is

$$t_{k_n}^1 = t_{k_n}^{ST} + t_{k_n}^{SE} = \left(1 - b_{k_n}\right) e_{k_n}^S \left[\frac{D_{k_n}}{a_{k_n} R^S} + \frac{c_{k_n}}{f_{k_n}}\right] \tag{23}$$

where $t_{k_n}^{ST}$ represents the transmission delay when vehicles accesses SBSs, $t_{k_n}^{ST}$ task performs the calculation of delay in the data center at SBS.

When vehicle k_n is connected to SBS n, and its task is offloaded to the server at MBSs for execution, that is, when $b_{k_n} = 1$ and $e_{k_n}^C = 0$. The total delay of task is

$$\begin{aligned} t_{k_n}^2 &= t_{k_n}^{ST} + t_{k_n}^{ME} + T_{S,M} \\ &= \left(1 - b_{k_n}\right) e_{k_n}^C \left[\frac{D_{k_n}}{a_{k_n} R^S} + \frac{c_{k_n}}{f_{k_n}} + T_{S,M}\right] \end{aligned} \tag{24}$$

where $T_{S,M}$ represents the transmission delay of backhaul network from SBSs to MBSs. Because the two base stations are connected through a wire, the bandwidth can be considered very rich. We assume it to be a constant related to the transmission distance. $t_{k_n}^{ME}$ indicates that the task performs computational delay in the data center at MBS.

When vehicle k_n is connected to SBS n and its tasks are offloaded to the server at Internet for execution, that is, when $b_{k_n} = 1$ and $e_{k_n}^C = 1$, the total delay of task is

$$t_{k_n}^3 = t_{k_n}^{MT} + t_{k_n}^{ME} + T_{S,C} = b_{k_n} e_{k_n}^C \left[\frac{D_{k_n}}{a_{k_n} R^S} + \frac{c_{k_n}}{f_{k_n}} + T_{S,C}\right] \tag{25}$$

where $T_{S,C}$ represents the transmission delay from SBSs to Internet servers. It can also be considered as a constant related to the transmission distance. $t_{k_n}^{CE}$ indicates that the task performs computational delay in central data center at Internet.

When vehicle k_n is connected to MBSs and its task is offloaded to the server at MBSs for execution, that is, $b_{k_n} = 1$ and the total delay of task is the transmission delay when the vehicle accesses MBSs.

$$t^4_{k_n} = t^{MT}_{k_n} + t^{ME}_{k_n} = b_{k_n} e^M_{k_n}\left[\frac{D_{k_n}}{a_{k_n}R^S} + \frac{c_{k_n}}{f_{k_n}}\right] \tag{26}$$

When vehicle k_n is connected to MBSs and its task is offloaded to the server at Internet for execution, that is, when $b_{k_n} = 1$ and $e^C_{k_n} = 1$, the total delay of task is

$$t^5_{k_n} = t^{MT}_{k_n} + t^{ME}_{k_n} + T_{M,C} = b_{k_n} e^C_{k_n}\left[\frac{D_{k_n}}{a_{k_n}R^S} + \frac{c_{k_n}}{f_{k_n}} + T_{M,C}\right] \tag{27}$$

where $T_{S,C}$ represents the transmission delay between MBSs and Internet servers. It can also be considered as a constant related to the transmission distance. Task completion time t_{k_n} is

$$\begin{aligned} t_{k_n} = {} & (1-b_{k_n}) e^S_{k_n} t^1_{k_n} + (1-b_{k_n}) e^M_{k_n} t^2_{k_n} + (1-b_{k_n}) e^C_{k_n} t^3_{k_n} \\ & + b_{k_n} e^M_{k_n} t^4_{k_n} + b_{k_n} e^C_{k_n} t^5_{k_n} \end{aligned} \tag{28}$$

because $e^S_{k_n} + e^M_{k_n} + e^C_{k_n} = 1$, $b_{k_n} = 1$ and $e^S_{k_n} = 0$, the above formula can be further simplified as:

$$\begin{aligned} t_{k_n} = {} & (1-b_{k_n}) \frac{D_{k_n}}{a_{k_n}R^S} + b_{k_n}\frac{D_{k_n}}{a_{k_n}R^M} + \frac{c_{k_n}}{f_{k_n}} + (1-b_{k_n}) T_{S,M} \\ & + b_{k_n} e^M_{k_n} T_{S,C} + b_{k_n} e^C_{k_n} T_{M,C} \end{aligned} \tag{29}$$

VI. SIMULATION

The simulation of this paper is completed on MATLAB 2016a platform. The simulation results mainly include the influence of these parameters (such as the number of users per unit time, the amount of user task offload data, the number of CPU cycles required by user task) on the average completion time of task, and these results are compared with other algorithms.

A. SIMULATION SETTINGS

The density of vehicles in cities is 1000-3000 vehicle/km2; the density of vehicles in suburbs is 500-1000 vehicle/km2; the density of vehicles on highways is 100-500 vehicle/km2. The coverage area of base stations is different in different scenarios. This paper takes the urban environment as an example and selects a square area of 1000m×1000m. The number of vehicles in this area is about 1000-3000. In the IoV scenario, dense base stations need to be deployed due to the huge amount of offloaded and downloaded data. This paper assumes that the base station deployment scenario is shown in Figure 6. A MBS and 24 SBSs are deployed in a square area of 1000m×1000m. The MBS is located at the center of the area and its coverage area covers entire area. The entire square area is divided into four small areas; 24 SBSs are located in the center of 4 small areas. Its coverage area is its small area. Ideally, the coverage area of SBSs is a circle, not a square. So there are overlapping areas in the coverage of 24 SBSs. We assume that vehicles in overlapping areas can only choose SBSs that are closer to themselves. So the coverage area of SBSs can be regarded as a square.

TABLE 3. Wireless channel simulation parameters.

Parameter	Value
the transmission bandwidth of base station wireless B	400MHz
the transmit power of SBSs P^S	35dBm
the transmit power of MBSs P^M	46dBm
the power of Gaussian white noise σ^2	-147dBm
the interference power between SBSs and MBSs I	$100\sigma^2$
path fading χ	$127+30\times\log d$ (d is the distance between vehicles and base stations)

Vehicles in each small area are uniformly distributed. For the task T_{k_n} that vehicles needs to unload, the values of task parameters D_{k_n}, C_{k_n} and $T^{\max}_{k_n}$ all follow normal distribution. The total computing resource of data center at SBSs is 1000 GHz/s, at MBSs is 4000 GHz/s. And the total computing resource of data center on Internet is 16000GHz/s.

After referring to future 5G communication standards and literatures, the wireless channel simulation parameters in this paper are shown in Table 3. According to literature [23], the channel gain can be calculated from path fading = 1010 and the total base station rate can be calculated to be about 20 Gbit/s. This satisfies the requirements of 5G for base station speed. In addition, we assume that the cable transmission delay from SBS to MBS data center is 0.002s. The propagation delay from MBS to Internet data center is 0.1s. Hence the propagation delay from SBS to Internet data center is 0.102s.

There are much work of task offloading and resource allocation. This paper chooses two of them as references: Ref [12] algorithm, Ref [21] algorithm and Ref [23] algorithm.

B. ANALYSIS OF SIMULATION RESULTS

1) IMPACT OF TASK NUMBER PER UNIT TIME ON TASK OFFLOADING

Let u be the number of tasks per unit time in entire area. For user task T_{k_n}, the offloaded data D_{k_n} follows normal distribution with mean value 10MB and the number of CPU cycles required for calculation task follows normal distribution with average value 0.5 Gigacycles. The results are shown in Figure 7:

It can be seen from Figure 7 that when the number of users is small, the four algorithms can obtain a very low average task completion time. But when the number of tasks per unit time exceeds a critical value, the average task completion time will increase sharply. Figure 7 shows that the proposed algorithm grows slower than other three algorithms. This indicates that when user requests more, the proposed

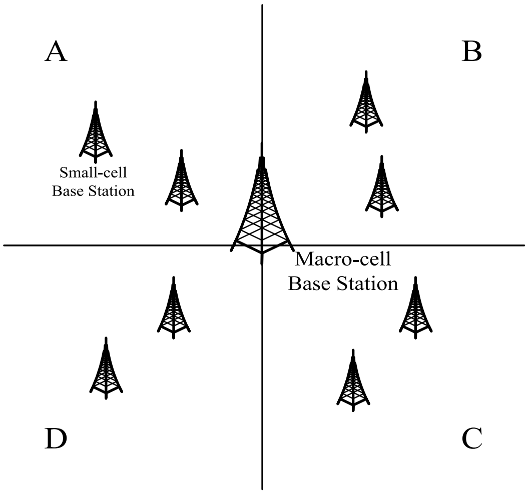

FIGURE 6. The scenario of base station deployment.

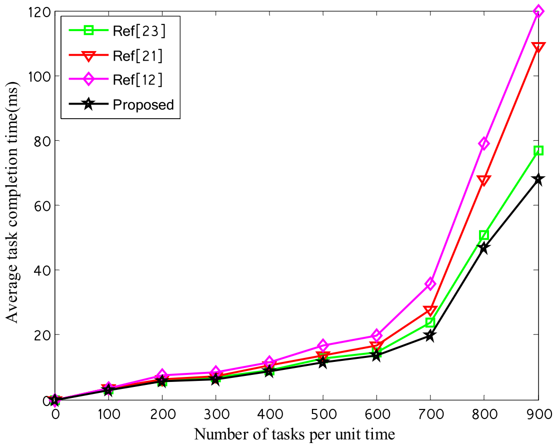

FIGURE 7. Impact of the number of users per unit time on average task completion time.

algorithm has higher task offloading and resource allocation efficiency and can complete tasks faster. When there are many users, the fastest growth is Ref [12] algorithm. The possible reason is that task offloading decision of Ref [12] algorithm is chosen randomly. Although the optimal resource allocation scheme is subsequently obtained, it can make up for the lack of random selection. However, it still reduces the system task execution efficiency compared with other algorithms. When the number of tasks per unit time is large enough, that is, vehicles are particularly congested the average completion time of the proposed algorithm is 33.7% less than Ref [12], 29.4% less than Ref [21] algorithm and 8.3% less than Ref [23] algorithm. It can be seen that the proposed algorithm is significantly better than other three algorithms.

2) IMPACT OF THE AMOUNT OF USER TASK OFFLOAD DATA ON AVERAGE TASK COMPLETION TIME

Assume that the number of users is 500. For user tasks, the offloaded data D_{k_n} follows normal distribution with mean

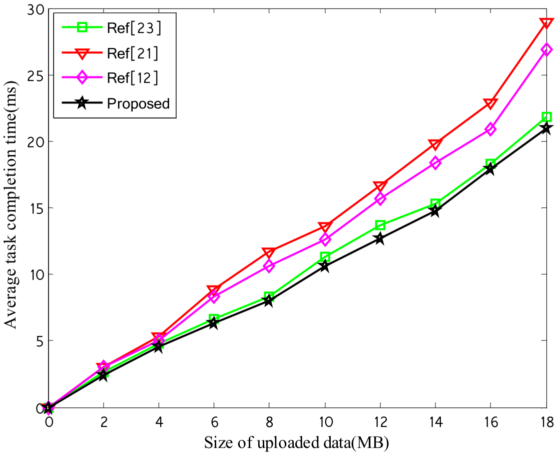

FIGURE 8. Impact of the amount of user task offload data on average task completion time.

value d and the number of CPU cycles required for calculation task follows normal distribution with mean value 0.5 Gigacycles. We analyze the impact of task offload data on average task completion time through simulation. The results are shown in Figure 8.

Observing the theoretical lower bound curve first, we find that the proposed algorithm is almost a straight line after d = 10MB and the slope is larger than before. Because users are evenly distributed, the number of users in each SBS may be between 10 and 30. The total transmission rate of each base station is about 20Gbit/s. Through calculation, we find that when d = 10MB, the base station transmission can basically complete the user's data during average task completion time. So we infer that when d > 10MB, the average task completion time is mainly determined by wireless transmission. Computational resources are sufficient, not a factor affecting average task completion time. It can be seen from Figure 8 that the proposed algorithm is very close to the theoretical value and is better than other three algorithms. When the amount of offloaded data is large, the average completion time of the proposed algorithm task is 19.6% less than Ref [12] algorithm, 24.8% less than Ref [21] algorithm and 4.3% less than Ref [23] algorithm. The proposed algorithm is significantly better than other three algorithms.

6.2.3 Impact of the number of CPU cycles required for user task execution on average task completion time

Assume that the number of users is 500. For user tasks, the offloaded data follows normal distribution with mean value 10MB. The number of CPU cycles required for calculation task C_{k_n} follows normal distribution with mean value c. Through simulation, we analyze the impact of CPU cycle number on average completion time of the task. The results are shown in Figure 9.

It can be seen from Figure 9 that when c < 0.4 Gigacycles, the average completion time obtained by the four algorithms is almost the same. This is because the data center computing resource parameters are set relatively large and the

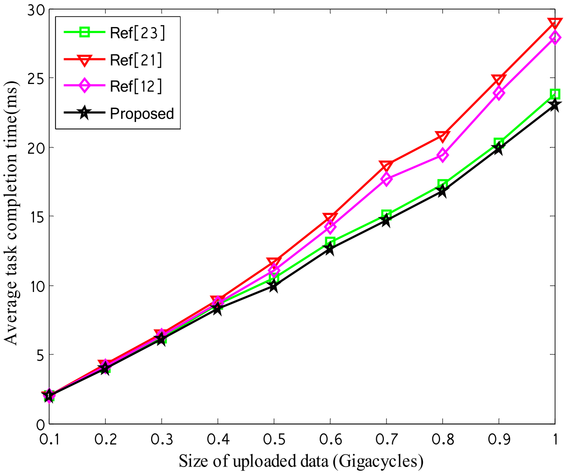

FIGURE 9. Impact of the number of CPU cycles required for user task execution on average task completion time.

computing resources are relatively abundant. Similar results can be obtained in the end where the task is executed. When it continues to increase and user tasks need to consume more computing resources. The advantages and disadvantages of the four algorithms begin to show. When c = 1.0 Gigacycles, the average completion time of the proposed algorithm and Ref [21] algorithm is 4.9ms apart. The average task completion time of the proposed algorithm is 20.0% less than that of Ref [21] algorithm.

VII. CONCLUSON

In the network model, each user has a task that needs to be offloaded. Besides, the task consists of three parameters: the amount of offloaded data, the number of CPU cycles required for calculation task and the maximum allowable completion time. This paper builds a network scenario with multiple MECs and multiple requesting vehicles. The service nodes are equipped with limited wireless and computing resources. It is assumed that vehicles and base station communicate by TDMA in proposed transmission time model. In the task execution time model, computing resources can be measured by CPU frequency based on virtual machine technology and DFVS technology. Finally, this paper considers different offloading decisions comprehensively and gives the completion time of tasks. In addition, an optimization problem with the minimum average completion time is also proposed. Therefore, we focus on ultra-low-latency task offloading, it is assumed that the handover of vehicles access base station does not occur during offloading process. In the follow-up work, the task model is expanded to general task offloading. In the future work, we will consider the impact of delay caused by the handover of base stations and the efficiency of system resource use comprehensively.

Moreover, the algorithm performance analysis of this paper is based on MATLAB simulation, and there is a certain distance from engineering implementation. In the future work, we hope to build the MEC system architecture on NS3 platform based on SDN and NFV technologies. And we use the proposed algorithm in this paper to implement software control over task offloading and resource allocation.

REFERENCES

[1] V. Sharma, "An energy-efficient transaction model for the blockchain-enabled Internet of vehicles (IoV)," *IEEE Commun. Lett.*, vol. 23, no. 2, pp. 246–249, Feb. 2019.

[2] H. Gao, W. Huang, and X. Yang, "Applying probabilistic model checking to path planning in an intelligent transportation system using mobility trajectories and their statistical data," *Intell. Automat. Soft Comput.*, vol. 25, no. 3, pp. 547–559, 2019.

[3] J. Wang, C. Jiang, Z. Han, Y. Ren, and L. Hanzo, "Internet of vehicles: Sensing-aided transportation information collection and diffusion," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 3813–3825, May 2018.

[4] H. Gao, Y. Xu, Y. Yin, W. Zhang, R. Li, and X. Wang, "Context-aware QoS prediction with neural collaborative filtering for Internet-of-Things services," *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2019.2956827.

[5] X. Ma, H. Gao, H. Xu, and M. Bian, "An IoT-based task scheduling optimization scheme considering the deadline and cost-aware scientific workflow for cloud computing," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, p. 249, Dec. 2019, doi: 10.1186/s13638-019-1557-3.

[6] J. Ahn, J. Lee, S. Park, and H.-S. Park, "Power efficient clustering scheme for 5G mobile edge computing environment," *Mobile Netw. Appl.*, vol. 24, no. 2, pp. 643–652, Apr. 2019.

[7] E. Schiller, N. Nikaein, E. Kalogeiton, M. Gasparyan, and T. Braun, "CDS-MEC: NFV/SDN-based application management for MEC in 5G systems," *Comput. Netw.*, vol. 135, pp. 96–107, Apr. 2018.

[8] H. Gao, Y. Duan, L. Shao, and X. Sun, "Transformation-based processing of typed resources for multimedia sources in the IoT environment," *Wireless Netw.*, Nov. 2019, doi: 10.1007/s11276-019-02200-6.

[9] H.-C. Hsieh, C.-S. Lee, and J.-L. Chen, "Mobile edge computing platform with container-based virtualization technology for IoT applications," *Wireless Pers. Commun.*, vol. 102, no. 1, pp. 527–542, Sep. 2018.

[10] G. Liga, A. Alvarado, E. Agrell, and P. Bayvel, "Information rates of next-generation long-haul optical fiber systems using coded modulation," *J. Lightw. Technol.*, vol. 35, no. 1, pp. 113–123, Jan. 1, 2017.

[11] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan, and Y. Zhang, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896–5907, 2016.

[12] M.-H. Chen, B. Liang, and M. Dong, "Joint offloading decision and resource allocation for multi-user multi-task mobile cloud," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–6.

[13] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, Apr. 2015.

[14] F. Samie, V. Tsoutsouras, L. Bauer, S. Xydis, D. Soudris, and J. Henkel, "Computation offloading and resource allocation for low-power IoT edge devices," in *Proc. IEEE 3rd World Forum Internet Things (WF-IoT)*, Dec. 2016, pp. 7–12.

[15] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[16] K. Zhang, Y. Mao, S. Leng, A. Vinel, and Y. Zhang, "Delay constrained offloading for mobile edge computing in cloud-enabled vehicular networks," in *Proc. 8th Int. Workshop Resilient Netw. Design Modeling (RNDM)*, Sep. 2016, pp. 288–294.

[17] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[18] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Joint allocation of computation and communication resources in multiuser mobile cloud computing," in *Proc. IEEE 14th Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jun. 2013, pp. 26–30.

[19] Y. Liu, M. J. Lee, and Y. Zheng, "Adaptive multi-resource allocation for cloudlet-based mobile cloud computing system," *IEEE Trans. Mobile Comput.*, vol. 15, no. 10, pp. 2398–2410, Oct. 2016.

[20] C. You and K. Huang, "Multiuser resource allocation for mobile-edge computation offloading," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[21] T. Zhao, S. Zhou, X. Guo, and Z. Niu, ''Tasks scheduling and resource allocation in heterogeneous cloud for delay-bounded mobile edge computing,'' in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–7.

[22] S. Singh and J. G. Andrews, ''Joint resource partitioning and offloading in heterogeneous cellular networks,'' *IEEE Trans. Wireless Commun.*, vol. 13, no. 2, pp. 888–901, Feb. 2014.

[23] T. Z. Oo, N. H. Tran, W. Saad, D. Niyato, Z. Han, and C. S. Hong, ''Offloading in HetNet: A coordination of interference mitigation, user association, and resource allocation,'' *IEEE Trans. Mobile Comput.*, vol. 16, no. 8, pp. 2276–2291, Aug. 2017.

[24] C. You, K. Huang, H. Chae, and B.-H. Kim, ''Energy-efficient resource allocation for mobile-edge computation offloading,'' *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.

[25] J.-N. Lee, M. D. Kim, and T. J. Kim, ''A dual-polarized small base station antenna having a high T_x/R_x isolation characteristics for 900 MHz small-cell MIMO systems,'' *J. Electromagn. Waves Appl.*, vol. 32, no. 8, pp. 1018–1028, May 2018.

[26] H. Shao, H. Zhao, and Y. Sun, ''Resource allocation for heterogeneous wireless networks: A robust layered game learning solutions,'' *J. Electron. Inf. Technol.*, vol. 39, no. 1, pp. 38–44, 2017.

[27] J. L. D. Neto, S.-Y. Yu, D. F. Macedo, J. M. S. Nogueira, R. Langar, and S. Secci, ''ULOOF: A user level online offloading framework for mobile edge computing,'' *IEEE Trans. Mobile Comput.*, vol. 17, no. 11, pp. 2660–2674, Nov. 2018.

[28] H. Gao, W. Huang, and Y. Duan, ''Research on cost-driven services composition in an uncertain environment,'' *J. Internet Technol.*, vol. 20, no. 3, pp. 755–769, 2019.

[29] C. You, K. Huang, and H. Chae, ''Energy efficient mobile cloud computing powered by wireless energy transfer,'' *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757–1771, May 2016.

[30] Q. Fettes, M. Clark, R. Bunescu, A. Karanth, and A. Louri, ''Dynamic voltage and frequency scaling in NoCs with supervised and reinforcement learning techniques,'' *Computer*, vol. 52, no. 9, pp. 4–5, Sep. 2019.

SHI YANG received the master's degree in computer software and theory from the Changchun University of Science and Technology, in 2012. He is currently an Associate Professor with the Changchun University of Finance and Economics. His research interests include cloud computing, the Internet of Things, and edge computing.