Three-Side Dynamic Task Offloading for Smart Roads Enabled Vehicular Edge Computing

Yunpeng Wang, Quyuan Luo, Yilong Hui, Zhao Liu, and Changle Li*
State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, Shaanxi, 710071 China
*E-mail: clli@mail.xidian.edu.cn

2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall) | 978-1-7281-9484-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/VTC2020-Fall49728.2020.9348440

***Abstract*—Smart roads can achieve a comprehensive, real-time and accurate perception of road environment, which is of great significance for intelligent transportation systems (ITS). However, due to massive data needed to be computed, cloud computing usually imposes pressure on backhaul and produces high delay. In this context, mobile edge computing (MEC) provides a promising solution. Meanwhile, current researches of the task offloading based on MEC lack global considerations and ignore IoT devices along the roadside, so optimization on three-side is very necessary and worth researching. To this end, we consider a scenario of smart roads including vehicular terminals (VTs), IoT devices and MEC servers. And we formulate an optimization problem aiming at minimizing a weighted sum of the costs of energy consumption and time delay for users side and cost for MEC servers. On this basis, we propose a three-side dynamic joint task offloading and resource allocation (TDJORA) scheme. Moreover, considering that the optimization problem is a multi-objective optimization problem, we utilize a combination of the particle swarm optimization (PSO) algorithm and Pareto optimality to obtain the optimal solution. Simulation results show that our proposed TDJORA can realize reasonable task offloading and optimal resource allocation for three sides.**

***Index Terms*—Smart roads; vehicular edge computing; task offloading; particle swarm optimization**

I. Introduction

As a comprehensive transportation system that guarantees safety, saves energy, improves transportation efficiency, intelligent transportation systems (ITS) has aroused extensive attention in recent years [1], [2]. Faced with this background, smart roads [3] stand out for achieving a real-time and accurate perception of surroundings through the integration of some advanced technologies, e.g., cloud computing, big data and Internet of Things (IoT) [4], which can improve the efficiency of transportation and is of great significance for ITS. In the current ITS environment, a large number of computation-intensive and latency-critical applications, e.g., augmented reality, mobile gaming and speech recognition [5], [6], have also emerged in vehicular terminals (VTs). Moreover, amounts of data sensed by multiple vehicle-mounted sensors in real time is massive [7], which makes it difficult to process tasks efficiently and timely. At the same time, massive data obtained from real-time perception of the external environment for IoT devices along the roadside is also difficult to achieve full local processing because of limited local processing capacity and energy consumption.

Aiming at the problem that the local computing resources of the VTs and IoT devices along the roadside are limited and it is difficult to process massive data in real time, mobile edge computing (MEC) is born as a promising solution [8], [9]. By connecting MEC servers to a roadside unit (RSU), an MRSU with MEC functions is deployed along the roadside, which can provide wireless access to VTs and IoT devices. Thus, by offloading computing tasks from VT and IoT side to the MRSU side, the processing time delay and local processing energy consumption can be reduced significantly. However, in the resource-constrained multi-user scenario, the task offloading and resource allocation for VT and IoT side are very complicated. Therefore, it is indispensable to design a low-complexity and global task offloading and resource allocation scheme for multi-user and multi-MEC scenario.

The existing task offloading and resource allocation schemes based on MEC mainly include optimizing only one side, that is, optimizing only users or MEC side, and bilateral optimization, that is, taking into account both users and MEC side. On the one hand, Paymard *et al.* [10] jointly allocated uplink and downlink communication and computing resources for multi-user scenarios to minimize user-side delay. Zheng *et al.* [11] adopted the SemiMarkov Decision Process (SMDP) to formulate the resource allocation scheme to maximize the long-term expected total reward of the cloud. On the other hand, in [2], authors proposed the approach AVARAC, which mainly addresses the resource management problem in a vehicular cloud system to maximize the average reward of the system. In addition, Du *et al.* [12] proposed a strategy for optimizing the allocation of communication and computing resources on both sides of users and servers for the connected vehicles, which minimizes the cost of both parties.

Most of the existing joint communication and computing resource allocation schemes based on MEC only consider the performance of VT side or MEC server, which ignore the task offloading and resource allocation of IoT devices along the roadside and do not consider different types of tasks arrived. Meanwhile, for multi-objective problems, it is only transformed into a single-objective problem, and lacks global comprehensive consideration and optimization.

In this paper, we focus on jointly optimizing the task offloading decision and allocation of communication and computing resources, to minimize a weighted sum of the costs of energy consumption and time delay for VT and IoT side and cost for MRSU side according to different types of tasks arrived. Due to the NP-hard and multi-objective features for the joint optimization problem, we combine the three-dimensional (3D) Pareto front and particle swarm optimization (PSO) algorithm to solve the formulated problem.

Authorized licensed use limited to: SOUTHWEST JIAOTONG UNIVERSITY. Downloaded on June 02,2021 at 15:20:05 UTC from IEEE Xplore. Restrictions apply.

The rest of the paper is organized as follows. Section II presents the system model and Section III analyzes the problem to be solved. The details of our proposed TDJORA are shown in Section IV. Simulation results are shown in Section V and conclusions are discussed in Section VI.

II. System Model

A. Scenario Description

We consider a novel hybrid road scenario called smart roads where the road is divided into K segments, each of which is covered by an MRSU. As shown in Fig. 1, this scenario mainly has three important aspects, including VTs, IoT devices and MRSU. Suppose that IoT devices are deployed at 20m intervals and VTs move at an average speed of 20m/s, then an MRSU has a coverage distance of 500m. Simultaneously, for the sake of research, we consider a time slot. According to the above assumption, we can find that the vehicle can move 0.02m within one time slot. Consequently, the network can be considered to be quasi-static where VTs and wireless channels keep unchanged in each time slot but can vary in different time slots [12]. We denote the set and the number of VTs served by each MRSU as $\mathcal{N} = \{1, 2, ..., N\}$ and N, respectively. Meanwhile, let $\mathcal{M} = \{1, 2, ..., M\}$ and M be the set and number of IoT devices.

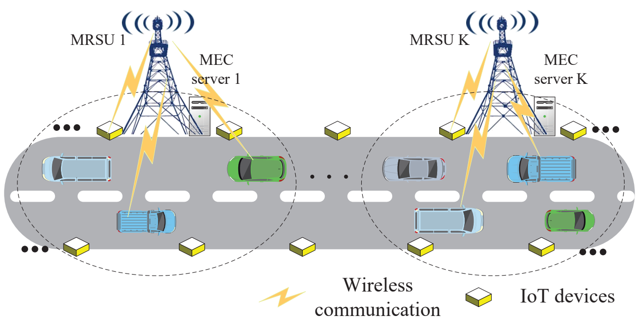


Fig. 1. Architecture of MRSU-enabled vehicular networks based on smart roads.

B. Computing Tasks and Processing Model

We denote the task of VT n and IoT m on slot t as $\Omega_n(t) = \{D_n(t), \lambda_n(t), A(t)\}$ and $\Omega_m(t) = \{D_m(t), \lambda_m(t), A(t)\}$, where $D_n(t)$ and $D_m(t)$ represent bits of computing tasks on slot t for VT side and IoT side, which are constrained by $0 < D_n(t) < D_n^{\max}$ and $0 < D_m(t) < D_m^{\max}$. And processing density of tasks (in CPU cycles/bit) for VT side and IoT side are denoted as $\lambda_n(t)$ and $\lambda_m(t)$ respectively. $A(t)$ represents the arriving service type, shown as

$$A(t) = (\alpha(t), \beta(t)) = \begin{cases} (1, 0), & \text{Delay-sensitive} \\ (0, 1), & \text{Power-sensitive} \\ (\frac{1}{2}, \frac{1}{2}), & \text{Otherwise} \end{cases} \quad (1)$$

When a large number of different types of tasks reach VT side and IoT side, they will consume different computing resources. Let $a_n(t)$ and $a_m(t)$ denote the offloading decision of VT n and IoT m on slot t respectively, whose value is between 0 and 1. $a_n(t) = 1$ and $a_m(t) = 1$ indicate the tasks are all offloaded to MRSU side to process, and $a_n(t) = 0$ and $a_m(t) = 0$ indicate the tasks are all processed locally. Otherwise, part of the tasks are processed locally, and part of them are offloaded to MRSU side. In each slot t, if tasks are executed locally, the power consumption of VT n and IoT m is $p_n(t) = k_v f_n(t)^3$ and $p_m(t) = k_I f_m(t)^3$, where k_v and k_I are a constant coefficient related to the CPU chip architecture [13]. $f_n(t)$ and $f_m(t)$ denote the local processing capability (in CPU cycles/s) of VT n and IoT m, which are limited by $0 \le f_n(t) \le f_n^{\max}(t)$ and $0 \le f_m(t) \le f_m^{\max}(t)$.

C. Communication Model

We consider the communication between VTs and MRSU to use the wireless channel for uplink and downlink data transmission, which is similar to the communication between IoT devices and MRSU. Let $W_{\rm VT}^{\rm up}$ and $W_{\rm VT}^{\rm Dn}$ be the uplink and downlink wireless channel bandwidth between VTs and MRSU, respectively. And let $W_{\rm I}^{\rm up}$ and $W_{\rm I}^{\rm Dn}$ be the uplink and downlink wireless channel bandwidth between IoT devices and MRSU, respectively. By blocking communication bandwidth resources, uplink communication resource blocks for VTs and IoT devices are denoted as $\mathbf{B}_{\rm VT}^{\rm up} = \{1, 2, \ldots, B_{\rm VT}^{\rm up}\}$ and $\mathbf{B}_{\rm I}^{\rm up} = \{1, 2, \ldots, B_{\rm I}^{\rm up}\}$, respectively. The downlink resource blocks are expressed as $\mathbf{B}_{\rm VT}^{\rm Dn} = \{1, 2, \ldots, B_{\rm VT}^{\rm Dn}\}$ and $\mathbf{B}_{\rm I}^{\rm Dn} = \{1, 2, \ldots, B_{\rm I}^{\rm Dn}\}$, respectively. $\rho_{\rm n}^{b^{\rm up}}(t)$ represents the uplink binary resource block assignment indicator of VTs, where $\rho_{\rm n}^{b^{\rm up}}(t){=}\,1$ means that the resource block $B_{\rm VT}^{\rm up}$ will be assigned to the user VT and otherwise, $\rho_{\rm n}^{b^{\rm up}}(t){=}\,0$.

The transmission power of VTs to offload the computing tasks to the MRSU in uplink and the transmission power of the MRSU to send the executed task of VTs in downlink are denoted by $p_{\rm n}^{b^{\rm up}}(t)$and $p_{\rm n}^{b^{\rm Dn}}(t)$, respectively. We denote the uplink and downlink channel power gains between the MRSU and VTs as $G_{\rm n}^{b^{\rm up}}(t)$ and $G_{\rm n}^{b^{\rm Dn}}(t)$. And the uplink and downlink interference signal power between VTs and MRSU are denoted by $I_{\rm n}^{b^{\rm up}}(t)$ and $I_{\rm n}^{b^{\rm Dn}}(t)$. Moreover, the uplink and downlink summation of the received additive white Gaussian noise (AWGN) power are expressed as $(\sigma^{b^{\rm up}}(t))^2$ and $(\sigma^{b^{\rm Dn}}(t))^2$. In this way, the maximum uplink transmission rate of each resource block for VT n in slot t can be given by

$$r_n^{b^{\rm up}}(t) = \overline{W}_{\rm VT}^{\rm up}\log_2\left(1 + \frac{\rho_{\rm n}^{b^{\rm up}}(t)p_{\rm n}^{b^{\rm up}}(t)G_{\rm n}^{b^{\rm up}}(t)}{I_{\rm n}^{b^{\rm up}}(t) + (\sigma^{b^{\rm up}}(t))^2}\right) \quad (2)$$

where $\overline{W}_{\rm VT}^{\rm up} = W_{\rm VT}^{\rm up}/B_{\rm VT}^{\rm up}$, which denotes the bandwidth of each uplink resource block for VTs and $r_{\rm n}^{\rm up}(t) = \sum_{b^{\rm up}=1}^{B_{\rm n}^{\rm up}} r_{\rm n}^{b^{\rm up}}(t)$, which denotes the uplink data transmission rate for all communication resource blocks. In this context, we can obtain the uplink energy consumption for tasks produced by VTs, which is given by

$$E_{\rm n}^{\rm up}(t) = \left(\sum_{b^{\rm up}=1}^{B_{\rm VT}^{\rm up}} p_{\rm n}^{b^{\rm up}}(t)\right)T_{\rm n}^{\rm up}(t) \quad (3)$$

where $T_{\rm n}^{\rm up}(t) = D_n(t)/r_{\rm n}^{\rm up}(t)$, which indicates the time required for VTs to offload local tasks to MRSU and $D_n(t)$ denotes the amount of tasks that VTs need to offload to MRSU.

Similarly as above, the maximum uplink transmission rate of each resource block for IoT m in slot t can be given by

$$r_{\rm m}^{b^{\rm up}}(t) = \overline{W}_{\rm I}^{\rm up}\log_2\left(1 + \frac{\rho_{\rm m}^{b^{\rm up}}(t)p_{\rm m}^{b^{\rm up}}(t)G_{\rm m}^{b^{\rm up}}(t)}{I_{\rm m}^{b^{\rm up}}(t) + (\sigma^{b^{\rm up}}(t))^2}\right) \quad (4)$$

Authorized licensed use limited to: SOUTHWEST JIAOTONG UNIVERSITY. Downloaded on June 02,2021 at 15:20:05 UTC from IEEE Xplore. Restrictions apply.

where $\overline{W}_{\mathrm{I}}^{\mathrm{up}}=W_{\mathrm{I}}^{\mathrm{up}}/B_{\mathrm{I}}^{\mathrm{up}}$ and $r_{\mathrm{m}}^{\mathrm{up}}(t)=\sum\limits_{b^{\mathrm{up}}=1}^{B_{\mathrm{I}}^{\mathrm{up}}} r_{\mathrm{m}}^{b^{\mathrm{up}}}(t)$.

Besides this, the uplink energy consumption for tasks produced by IoT devices is given by

$$E_{\mathrm{m}}^{\mathrm{up}}(t)=(\sum_{b^{\mathrm{up}}=1}^{B_{\mathrm{I}}^{\mathrm{up}}} p_{\mathrm{m}}^{b^{\mathrm{up}}}(t))T_{\mathrm{m}}^{\mathrm{up}}(t) \tag{5}$$

where $T_{\mathrm{m}}^{\mathrm{up}}(t)=D_m(t)/r_{\mathrm{m}}^{\mathrm{up}}(t)$, which indicates the time required for IoT devices to offload local tasks to MRSU and $D_m(t)$ denotes the amount of tasks that IoT devices need to offload to MRSU.

D. Cost Model

The cost of VT and IoT side consists of two parts, energy consumption and time delay caused by task processing and transmission.

1) ***Cost of VTs:*** On the one hand, in local processing, the monetary cost of VT n can be given by $COST_n^{\mathrm{local}}(t)=k_v\omega_v D_n(t)\lambda_n(t)(f_n(t))^2$, where ω_v (in \$/J) is a human-determined weight coefficient, which is used to convert energy consumption into money and depends on the human-sensitiveness on money and energy consumption [13]. VT n needs to pay for different costs and fees if offloading tasks to remote processing:(i) The cost of transmitting per bit data in uplink transmission that VT n needs to pay to MRSU, where $\Phi_n(t)$ is the price of transmitting per bit data. (ii) The energy consumption of VT n in data uplink transmission. (iii) The cost of processing tasks of the virtual machine (VM) for MRSU side. So the monetary cost of VT n in task offloading is $a_n(t)\Phi_n(t)D_n(t)+a_n(t)\omega_v E_{\mathrm{n}}^{\mathrm{up}}(t)+a_n(t)\phi_v D_n(t)\lambda_{\mathrm{VM}}(t)$. Thus, the cost (in \$) of each VT n in slot t is given by

$$\begin{aligned}&COST_n(t)=(1-a_n(t))k_v\omega_v D_n(t)\lambda_n(t)(f_n(t))^2\\&+a_n(t)[\Phi_n(t)D_n(t)+\omega_v E_{\mathrm{n}}^{\mathrm{up}}(t)+\phi_v D_n(t)\lambda_{\mathrm{VM}}(t)]\end{aligned} \tag{6}$$

On the other hand, considering that tasks can be processed locally or offloaded to the MRSU side, the time delay mainly includes local processing delay, uplink transmission delay, MRSU processing delay, and downlink transmission delay. Among them, local processing delay is denoted by $T_{\mathrm{n}}^{\mathrm{local}}$, which can be given as

$$T_{\mathrm{n}}^{\mathrm{local}}=\tfrac{(1-a_n(t))\lambda_n(t)D_n(t)}{f_n(t)} \tag{7}$$

When part of tasks are offloaded to the MRSU side for processing, uplink transmission time delay for VT side can be obtained according to formula (4), which is denoted by $T_{\mathrm{n}}^{\mathrm{up}}(t)$. And task processing time delay on the MRSU side is given by

$$T_{\mathrm{MRSU-V}_n}^{\mathrm{local}}=\tfrac{a_n(t)\lambda_{\mathrm{VM}}(t)D_n(t)}{f_{\mathrm{VM}}(t)} \tag{8}$$

As such, the total delay in processing tasks for VT side can be expressed as:

$$\mathrm{Delay}_n=\max\{T_{\mathrm{n}}^{\mathrm{local}}(t),T_n^{\mathrm{offload}}(t)\} \tag{9}$$

In the above formula, $T_n^{\mathrm{offload}}(t)=T_{\mathrm{MRSU-V}_n}^{\mathrm{local}}+T_{\mathrm{n}}^{\mathrm{up}}(t)+T_{\mathrm{n}}^{\mathrm{Dn}}(t)$, which refers to processing and transmission delays caused by task offloading.

2) ***Cost of IoT Devices:*** In local processing for IoT side, the monetary cost of IoT m can be given by $COST_m^{\mathrm{local}}(t)=k_I\omega_I D_m(t)\lambda_m(t)(f_m(t))^2$, where the definition of ω_I (in \$/J) is the same as ω_v [13]. When the amount of data acquired by perception is too large to process in real time, task offloading is also required. Similarly, the monetary cost of IoT m in task offloading is $a_m(t)\Phi_m(t)D_m(t)+a_m(t)\omega_I E_{\mathrm{m}}^{\mathrm{up}}(t)+a_m(t)\phi_I D_m(t)\lambda_{\mathrm{VM}}(t)$. Thus, the cost (in \$) of each IoT m in slot t is given by

$$\begin{aligned}&COST_m(t)=(1-a_m(t))k_I\omega_I D_m(t)\lambda_m(t)(f_m(t))^2\\&+a_m(t)[\Phi_m(t)D_m(t)+\omega_I E_{\mathrm{m}}^{\mathrm{up}}(t)+\phi_I D_m(t)\lambda_{\mathrm{VM}}(t)]\end{aligned} \tag{10}$$

At the same time, local processing time delay for IoT side can be given by

$$T_{\mathrm{m}}^{\mathrm{local}}=\tfrac{(1-a_m(t))\lambda_m(t)D_m(t)}{f_m(t)} \tag{11}$$

The processing delays for tasks from the IoT side handled by the MRSU side is expressed as

$$T_{\mathrm{MRSU-I}_m}^{\mathrm{local}}=\tfrac{a_m(t)\lambda_{\mathrm{VM}}(t)D_m(t)}{f_{\mathrm{VM}}(t)} \tag{12}$$

As a result, the total delay in processing tasks for IoT side can be given by

$$\mathrm{Delay}_m=\max\{T_m^{\mathrm{local}}(t),T_m^{\mathrm{offload}}(t)\} \tag{13}$$

In the above formula, $T_m^{\mathrm{offload}}(t)=T_{\mathrm{MRSU-I}_m}^{\mathrm{local}}+T_m^{\mathrm{up}}(t)+T_m^{\mathrm{Dn}}(t)$, which refers to processing and transmission delays caused by task offloading.

3) ***Cost of MRSU:*** The MRSU side is used to handle tasks from VT side and IoT side. On the one hand, the cost of task processing is paid by VT side and IoT side. On the other hand, the MRSU side needs to pay for energy consumption cost, the downlink communication cost and occupying VM cost for processing tasks from VT side and IoT side. We denote $\gamma(t)$ as the electricity bills (in \$) for running a VM. So, the cost (in \$) of MRSU in slot t is given by

$$\begin{aligned}&COST_{\mathrm{MRSU}}(t)=a_n(t)\omega_v E_{\mathrm{n}}^{\mathrm{Dn}}(t)+a_n(t)D_n(t)\gamma(t)\\&+a_n(t)\omega_v k_{\mathrm{MRSU}}D_n(t)\lambda_{\mathrm{VM}}(t)(f_{\mathrm{VM}}(t))^2\\&+a_m(t)\omega_I k_{\mathrm{MRSU}}D_m(t)\lambda_{\mathrm{VM}}(t)(f_{\mathrm{VM}}(t))^2\\&+a_m(t)\omega_I E_{\mathrm{m}}^{\mathrm{Dn}}(t)+a_m(t)D_m(t)\gamma(t)\end{aligned} \tag{14}$$

III. Problem Formulation and Transformation

A. Three-Side Problem Formulation

In this section, we propose three optimization problems for VTs, IoT devices and MRSU respectively, which aim at minimizing their respective cost under system constraints, including energy consumption and time delay for task processing. Simultaneously, in order to unify the two indices, we introduce a coefficient ξ, which represents the weight on the time delay relative to energy consumption for task processing in the total cost of system [14]. Thus, the three-side optimization problem is formulated as follows.

1) **VT Side Optimization Problem**

For VT side optimization problem, we need to optimize the offloading decision $a_n(t)$, the local CPU processing frequency $f_n(t)$ and the uplink binary resource block assignment indicator $\rho_{\mathrm{n}}^{b^{\mathrm{up}}}(t)$ under given constraints. So, the VT side

Authorized licensed use limited to: SOUTHWEST JIAOTONG UNIVERSITY. Downloaded on June 02,2021 at 15:20:05 UTC from IEEE Xplore. Restrictions apply.

optimization problem is formulated as follows:

$$\begin{aligned}
\min_{\substack{a_n(t),f_n(t),\\ \rho_n^{b^{\text{up}}}(t),f_{\text{VM}}(t)}} \quad & \frac{1}{T}\sum_{n=1}^{N_t}\sum_{t=1}^{T}\{\alpha COST_n(t)+\xi\beta\text{Delay}_n(t)\}\\
s.t.\text{C-V1}: & \ 0\le a_n(t)\le 1, \forall t\in T, n\in\mathcal{N}\\
\text{C-V2}: & \ 0\le D_n(t)\le D_n^{\max}, \forall t\in T, n\in\mathcal{N}\\
\text{C-V3}: & \ 0\le f_n(t)\le f_n^{\max}(t), \forall t\in T, n\in\mathcal{N}\\
\text{C-V4}: & \ \rho_{\text{n}}^{b^{\text{up}}}(t)\in\{0,1\}, \forall t\in T, n\in\mathcal{N}, b^{\text{up}}\in\mathbf{B}_{\text{VT}}^{\text{up}}\\
\text{C-V5}: & \ 0\le f_{\text{VM}}(t)\le f_{\text{VM}}^{\max}(t), \forall t\in T
\end{aligned}$$

where C-V1 represents computation offloading decisions, whose value is between 0 and 1; C-V2 denotes data size of computing tasks; C-V3 is the CPU processing frequency constraint for each VT; C-V4 indicates the uplink binary resource block assignment indicator, whose value is 0 or 1; C-V5 is the VM processing frequency constraint for MRSU side.

2) **IoT Side Optimization Problem**

Similarly, for IoT side optimization problem, we need to optimize the offloading decision $a_m(t)$, the local CPU processing frequency $f_m(t)$ and the uplink binary resource block assignment indicator $\rho_{\text{m}}^{b^{\text{up}}}(t)$ under given constraints. So, the IoT side optimization problem is formulated as follows:

$$\begin{aligned}
\min_{\substack{a_m(t),f_m(t),\\ \rho_{\text{m}}^{b^{\text{up}}}(t),f_{\text{VM}}(t)}} \quad & \frac{1}{T}\sum_{m=1}^{M_t}\sum_{t=1}^{T}\{\alpha COST_m(t)+\xi\beta\text{Delay}_m(t)\}\\
s.t.\text{C-I1}: & \ 0\le a_m(t)\le 1, \forall t\in T, m\in\mathcal{M}\\
\text{C-I2}: & \ 0\le D_m(t)\le D_m^{\max}, \forall t\in T, m\in\mathcal{M}\\
\text{C-I3}: & \ 0\le f_m(t)\le f_m^{\max}(t), \forall t\in T, m\in\mathcal{M}\\
\text{C-I4}: & \ \rho_{\text{m}}^{b^{\text{up}}}(t)\in\{0,1\}, \forall t\in T, m\in\mathcal{M}, b^{\text{up}}\in\mathbf{B}_{\text{I}}^{\text{up}}\\
\text{C-I5}: & \ 0\le f_{\text{VM}}(t)\le f_{\text{VM}}^{\max}(t), \forall t\in T
\end{aligned}$$

where C-I1 represents computation offloading decisions, whose value is between 0 and 1; C-I2 denotes data size of computing tasks; C-I3 is the CPU processing frequency constraint for each IoT device; C-I4 indicates the uplink binary resource block assignment indicator, whose value is 0 or 1; C-I5 is the VM processing frequency constraint for each MRSU.

3) **MRSU Side Optimization Problem**

Moreover, MRSU side optimization aims at minimizing the average cost of MRSU by jointly optimizing the offloading decision $a_n(t)$ for VT side and $a_m(t)$ for IoT side, the local CPU processing frequency $f_{\text{VM}}(t)$ and the downlink binary resource block assignment indicator $\rho_{\text{n}}^{b^{\text{Dn}}}(t)$ and $\rho_{\text{m}}^{b^{\text{Dn}}}(t)$ under given constraints. So, the MRSU side optimization problem is formulated as follows:

$$\begin{aligned}
\min_{\substack{a_n(t),a_m(t),\rho_{\text{n}}^{b^{\text{Dn}}}(t),\\ \rho_{\text{m}}^{b^{\text{Dn}}}(t),f_{\text{VM}}(t)}} \quad & \frac{1}{T}\sum_{t=1}^{T}COST_{\text{MRSU}}(t)\\
s.t.\text{C-M1}: & \ 0\le a_n(t)\le 1, 0\le a_m(t)\le 1, \forall t\in T\\
\text{C-M2}: & \ 0\le f_{\text{VM}}(t)\le f_{\text{VM}}^{\max}(t), \forall t\in T, n\in\mathcal{N}, m\in\mathcal{M}\\
\text{C-M3}: & \ \rho_n^{b^{\text{Dn}}}(t), \rho_m^{b^{\text{Dn}}}(t)\in\{0,1\}, \forall t\in T, n\in\mathcal{N}, m\in\mathcal{M}
\end{aligned}$$

where C-M1 represents computation offloading decisions for VT side and IoT side; C-M2 is the CPU processing frequency constraint for each VM for MRSU side; C-M3 indicates the downlink binary resource block assignment indicator for VT side and IoT side, whose value is 0 or 1.

B. Problem Transformation

Our goal is to minimize the tripartite weighted sum of time delay and cost of VT, IoT, and MRSU side based on different types of services arrived. When the tasks are offloaded to the MRSU side for processing, considering that the final calculation result is often a numerical value, the data amount of which is small, so the energy consumption of downlink transmission can be ignored. And we take one time slot as the research object, so the three-side optimization problem is formulated as follows according to (15), (16) and (17):

$$\begin{aligned}
(\mathcal{P}) \quad & \min_{\substack{a_n(t),f_n(t),\\ \rho_{\text{n}}^{b^{\text{up}}}(t),f_{\text{VM}}(t)}} \sum_{n=1}^{N_t}\{\alpha COST_n(t)+\xi\beta\text{Delay}_n(t)\}\\
& \min_{\substack{a_m(t),f_m(t),\\ \rho_{\text{m}}^{b^{\text{up}}}(t),f_{\text{VM}}(t)}} \sum_{m=1}^{M_t}\{\alpha COST_m(t)+\xi\beta\text{Delay}_m(t)\}\\
& \min_{a_n(t),a_m(t),f_{\text{VM}}(t)} COST_{\text{MRSU}}(t)\\
& s.t.\text{(C-V1)-(C-V5)}, \text{(C-I1)-(C-I4)}
\end{aligned}$$

IV. Our Proposed TDJORA

A. PSO Based on 3D Pareto Optimality

Particle swarm optimization (PSO) algorithm is derived from the study of bird predation behavior, whose main idea is to find the optimal solution through cooperation and information sharing between individuals in a swarm. We adopt PSO algorithm to solve the optimization problem because of its simplicity and ease of implementation. Firstly, initialize a group of random particles, then find the optimal solution through iteration. In each iteration, the particle updates itself by tracking two values, *pbest* and *gbest*. After finding these two optimal values, the particle updates its velocity and position using the equation (15).

$$\begin{aligned}
v_i &= \omega\times v_i + c_1\times rand()\times(pbest_i - x_i)\\
&\quad + c_2\times rand()\times(gbest_i - x_i)\\
x_i &= x_i + v_i
\end{aligned} \tag{15}$$

where ω represents a dynamic inertia weight, whose value changes continuously with iteration. Moreover, c_1 and c_2 serve as learning factor. $pbest_i$ and $gbest_i$ indicate local optimal solution and global optimal solution respectively. Since the problem to be solved is a multi-objective optimization problem, we introduce the Pareto optimality to find an optimal 3D Pareto front. The Pareto optimal solution is to find as many non-dominated solutions as possible to ensure that the interests of all parties are not damaged.

B. Proposed Three-Side Dynamic Joint Task Offloading and Resource Allocation (TDJORA) Scheme

Considering different types of services and each side of the entire system, we propose a three-side dynamic joint task offloading and resource allocation scheme, which minimizes a weighted sum of the costs of energy consumption and time delay for VT and IoT side and cost for MRSU side. At the same time, the optimization problem is solved by PSO algorithm based on 3D Pareto optimality so as to obtain optimal task offloading decision and resource allocation scheme.

Authorized licensed use limited to: SOUTHWEST JIAOTONG UNIVERSITY. Downloaded on June 02,2021 at 15:20:05 UTC from IEEE Xplore. Restrictions apply.

V. Simulation and Result Discussion

A. Simulation Settings

In our simulations, we set up a freeway model based on MATLAB. The major parameters are included in TABLE I.

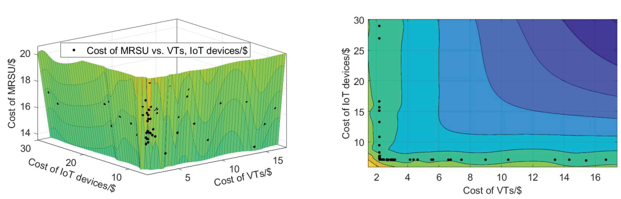
TABLE I: Simulation Parameters

Parameter	Value
Number of VTs $\mathcal{N}$	$\{20, 30, 40, 50, 60\}$
Number of IoT devices $\mathcal{M}$	$\{40, 80, 120, 160, 200\}$
Input data size D_n, D_m	$0.1 - 1M$
Processing density of VT λ_n	$1000(cycles/bit)$
Processing density of IoT λ_m	$10000(cycles/bit)$
Processing density of MRSU $\lambda_{\rm VM}$	$100(cycles/bit)$
Max. local processing capability for VT $f_n^{\max}$	$1.4(Gcycles/s)$
Max. local processing capability for IoT $f_m^{\max}$	$0.8(Gcycles/s)$
Effective switched capacitance k_v	10^{-27}
Effective switched capacitance at MRSU $k_{\rm MRSU}$	10^{-29}
Uplink bandwidth $W_{\rm VT}^{\rm up}, W_{\rm I}^{\rm Up}$	$10MHz$
Downlink bandwidth $W_{\rm VT}^{\rm Dn}, W_{\rm I}^{\rm Dn}$	$10MHz$
Transmit power of VT and IoT $p_{\rm n}^{b^{\rm up}}$, $p_{\rm m}^{b^{\rm up}}$	$1W$
Power gains $G_{\rm n}^{b^{\rm up}}$ and $G_{\rm m}^{b^{\rm up}}$	1
Number of uplink and downlink RBs $B_{\rm VT}^{\rm up}, B_{\rm VT}^{\rm Dn}$	$20 * N$
AWGN power $(\sigma^{b^{\rm up}})^2$ and $(\sigma^{b^{\rm Dn}})^2$	$-100dBm$
price for transmitting per bit data in uplink	$1.16 * 10^{-10}$(\$/bit)
price for transmitting per bit data in downlink	$0.5 * 10^{-10}$(\$/bit)
Energy-money weight coefficient ω_v, ω_I	$2.44 * 10^{-4}$(\$/J)
Price for MRSU to process tasks	$3 * 10^{-10}$(\$/cycle)
Max. processing capability $f_{\rm VM}^{\max}$ of MRSU	$300(Gcycles/s)$
Size of the swarms s	200
Inertia weight factor w	$0.4 - 0.8$
Learning factor θ_1 and θ_2	1.5

B. Simulation Result and Discussion

We perform simulations mainly for delay-sensitive and power-sensitive services in terms of our objective, a weighted sum of costs of time delay and energy consumption for VT and IoT side and cost for MRSU side. By finding the Pareto optimal solution to the optimization problem, the respective goals are minimized and the 3D Pareto front is obtained.

As is shown in Fig. 2 (a), the optimal solution for joint task offloading and resource allocation is on the 3D Pareto front, which mainly describes the distribution of the optimal solution that satisfies the minimization of the cost of all parties. Moreover, it can be seen from Fig. 2 (b) that the cost between VT side and IoT side is inversely proportional, which also indicates that in the case of limited resources on the MRSU side, there is a competitive relationship between VTs and IoT devices. And the distribution of the optimal solution is consistent with the two-dimensional Pareto flat, which can also prove the validity and correctness of the 3D Pareto front.



(a) The 3D pareto front (b) The top view of 3D pareto front

Fig. 2. An illustration of optimal global 3D Pareto front

VI. Conclusion

In this paper, we have studied a three-side optimization framework based on smart roads in an MRSU-enabled vehicular network. Meanwhile, the problem of jointly task offloading and resource allocation is formulated as an NP-hard problem, whose goal is to minimize a weighted sum of the costs for user side and cost for MRSU side. Moreover, we have proposed a TDJORA scheme and have solved the optimization problem by using an PSO algorithm combined with Pareto optimality. Through extensive simulations, we have analyzed the performances of TDJORA by 3D Pareto front, which shows our proposed TDJORA can realize reasonable task offloading and optimal resource allocation according to different types of services arrived.

VII. Acknowledgment

This work was supported by the National Natural Science Foundation of China (U1801266, 61901341), National Key R&D Program of China (2019YFB1600100), Key R&D Program of Shaanxi (2018ZDXM-GY-038, 2018ZDCXL-GY-04-02), the Youth Innovation Team of Shaanxi Universities, and the Science and Technology Projects of Xi'an, China (201809170CX11JC12).

References

[1] R. I. Meneguette, A. Boukerche and A. H. M. Pimenta, “AVARAC: An Availability-Based Resource Allocation Scheme for Vehicular Cloud,” in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 10, pp. 3688-3699, 2019.

[2] Q. Luo, C. Li, T. H. Luan and W. Shi, ”EdgeVCD: Intelligent Algorithm-Inspired Content Distribution in Vehicular Edge Computing Network,” in IEEE Internet of Things Journal, vol. 7, no. 6, pp. 5562-5579, 2020.

[3] Z. Yuan, Talking about smart road and building intelligent road network[J]. Science and Economic Guide, vol. 27, no. 2, pp. 33-34, 2019.

[4] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari and M. Ayyash, “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,” in IEEE Communications Surveys and Tutorials, vol. 17, no. 4, pp. 2347-2376, 2015.

[5] B. Gu and Z. Zhou, “Task Offloading in Vehicular Mobile Edge Computing: A Matching-Theoretic Framework,” in IEEE Vehicular Technology Magazine, vol. 14, no. 3, pp. 100-106, 2019.

[6] M. Sookhak, F. R. Yu, Y. He, et al., “Fog vehicular computing: Augmentation of fog computing using vehicular cloud computing,” IEEE Vehicular Technology Magazine, vol. 12, no. 3, pp. 55-64, 2017.

[7] K. Chan, T. Dillon and E. Chang, “An intelligent particle swarm optimization for short-term traffic flow forecasting using on-road sensor systems,” in IEEE Transactions on Industrial Electronics, vol. 60, no. 10, pp. 4714-4725, 2013.

[8] K. Zhang, Y. Mao, S. Leng, S. Maharjan and Y. Zhang, “Optimal delay constrained offloading for vehicular edge computing networks,” 2017 IEEE International Conference on Communications (ICC), Paris, pp. 1-6, 2017.

[9] Y. Mao, C. You, J. Zhang, K. Huang and K. B. Letaief, ”A Survey on Mobile Edge Computing: The Communication Perspective,” in IEEE Communications Surveys and Tutorials, vol. 19, no. 4, pp. 2322-2358, 2017.

[10] P. Paymard, S. Rezvani and N. Mokari, “Joint task scheduling and uplink/downlink radio resource allocation in PD-NOMA based mobile edge computing networks,” in PHYSICAL COMMUNICATION, vol. 32, no. 6, pp. 160-171, 2018.

[11] K. Zheng, H. Meng, P. Chatzimisios, L. Lei and X. Shen, “An SMDP-Based Resource Allocation in Vehicular Cloud Computing Systems,” in IEEE Transactions on Industrial Electronics, vol. 62, no. 12, pp. 7920-7928, 2015.

[12] J. Du, F. R. Yu, X. Chu, J. Feng and G. Lu, “Computation Offloading and Resource Allocation in Vehicular Networks Based on Dual-Side Cost Minimization,” in IEEE Transactions on Vehicular Technology, vol. 68, no. 2, pp. 1079-1092, 2019.

[13] Y. Kim, J. Kwak, and S. Chong, Dual-side optimization for cost-delay tradeoff in mobile edge computing, in IEEE Transactions on Vehicular Technology, vol. 67, no. 2, pp. 1765-1781, 2018.

[14] K. Zhang, X. Gui and D. Ren, ”Joint Optimization on Computation Offloading and Resource Allocation in Mobile Edge Computing,” IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, pp. 1-6, 2019.

Authorized licensed use limited to: SOUTHWEST JIAOTONG UNIVERSITY. Downloaded on June 02,2021 at 15:20:05 UTC from IEEE Xplore. Restrictions apply.