A Contract-Based Incentive Mechanism for Resource Sharing and Task Allocation in Container-Based Vehicular Edge Computing

Siming Wang, Xumin Huang, Beihai Tan(✉), and Rong Yu

Guangdong Key Laboratory of IoT Information Processing,
Guangdong University of Technology, Guangzhou, China
bhtan@gdut.edu.cn

Abstract. Vehicular edge computing (VEC) has emerged as a promising paradigm to provide low-latency service by extending the edge computing to vehicular networks. To meet the ever-increasing demands of computation and communication resources, utilizing vehicles as augmented infrastructure for computation offloading is an appealing idea. However, due to the lack of effective incentive and task allocation mechanism, it is challenging to exploit vehicles as infrastructure for computation offloading. To cope with these challenges, we first propose a container-based VEC paradigm by using efficient, flexible and customized resources of the vehicles. Then, we present a contract-based incentive mechanism to motivate vehicles to share their resources with service requesters (SRs). The optimal contract items are designed for multiple types of vehicles while maximizing the expected utilities of the SRs. Numerical results demonstrate that the proposed contract-based incentive mechanism is efficient compared with conventional schemes.

Keywords: Container-based vehicular edge computing · Resource sharing · Task allocation · Contract-based incentive mechanism

1 Introduction

With the rapid advance of the Internet of Vehicles (IoV), smart vehicles with vehicular networks access have experienced ever-increasing growth in number and variety [1,2]. According to the recent report, nearly a quarter billion vehicles will be connected by 2020 [3]. These vehicles with different communication modes such as vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) are regarded as the important component of the future IoT-based infrastructure for providing various applications and services [4,5]. The emerging vehicular applications are computation-intensive and have low-latency requirements, such as augmented reality (AR), self-driving, and intelligent navigation service etc. [6]. Unfortunately, this poses huge challenges to the resource-limited vehicles to guarantee

B. Li et al. (Eds.): IoTaaS 2019, LNICST 316, pp. 116–129, 2020.
https://doi.org/10.1007/978-3-030-44751-9_11

the low-latency requirements and the quality of service (QoS). To handle the ever-increasing demands of computation resources, vehicular edge computing (VEC) supported by container-based technology constitutes a new computation offloading paradigm in vehicular networks and improves the system performance [7,8].

To deal with the ever-increasing demands of computation resources, it is promising to utilize vehicles as augmented infrastructure with container-based virtualization for computation offloading. Nowadays, smart vehicles are installed with powerful computing units, advanced communication devices and sensors [9]. It is practicable to leverage such large amount of on-board resources to perform computation offloading in vehicular networks [10]. Furthermore, compared with heavyweight virtual machines (VM), container-based virtualization technology is more available in vehicular networks. With the characteristics of short time implementation, efficient resource utilization and low maintenance cost, the onboard units (OBUs) customization can be implemented by container-based virtualization and offers high flexibility in platform management [11]. Existing studies have exploited container-based virtualization technologies in edge computing. The authors in [11] integrated light weighted virtualization with IoT edge networks. They presented container-based VEC and exploited task offloading among different vehicles. A parked vehicle edge computing with container-based virtualization was proposed to utilize the computing resources of parked vehicles [12]. The authors in [3] proposed a novel architecture for task selection and scheduling at the edge of network using container-as-a-service.

However, there are still many issues in the implementation of computation offloading in container-based VEC. First, existing studies assume that the vehicles serve as infrastructure for computation offloading voluntarily [13,14]. However, this assumption is not available in the practical situation. Because the vehicles are selfish, they may reject to contribute their onboard resource without any payment. The challenge is to design an efficient incentive mechanism to motivate vehicles to share their computation resources with SRs. Second, the information asymmetry between SRs and vehicles is necessary to be considered. The vehicles can be dishonest, and they are not willing to reveal their private information to others. The vehicles are intended to maximize their own payoffs and cheat SRs by charging more payment. Third, with multiple vehicles and multiple SRs, it is not easy to design an efficient task allocation scheme to reduce service delay and maximize the saved delay utility. Thus, it is necessary to design an efficient incentive mechanism to address the resource sharing and task allocation problem and overcome the asymmetric information scenario between vehicles and SRs.

To solve the challenges mentioned above, we propose a contract-based incentive mechanism to encourage the vehicles to share their onboard resources and help offload tasks from SRs. Furthermore, the proposed novel contract-based framework solves the task allocation problem. A set of resource-reward contract items is designed for maximizing the SRs utilities and stimulating each type of vehicle to accept the contract item that is intently designed for its type. The vehicles with different energy cost efficiency are classified into multiple types. The asymmetric information scenario between SRs and vehicles can be overcome by solving the contract-based optimization problem. Numerical results demonstrate that the proposed contract-based incentive mechanism is efficient compared with conventional schemes.

The rest of this paper is organized as follows. Section 2 introduces the system model with network entities in container-based VEC. The contract formulation and simplification is presented in Sect. 3. The solution is given in Sect. 4. Performance evaluation results are shown in Sect. 5 before the paper is concluded in Sect. 6.

2 System Model

2.1 Network Entities

The system model of container-based VEC is shown in Fig. 1. There are existing multiple vehicles and multiple SRs in VEC. Due to the limited computation resources, the potential SRs have great demands for computation resources in various applications, such as data mining, image processing and natural language translation. Each SR generates a task which can be offloaded to a vehicle. To offer low-latency services to the SRs, the vehicles with rich onboard resources can serve as infrastructure by offloading tasks from SRs. The vehicular containers deployed on OBUs share hardware infrastructure and host operation system. Compared with traditional virtual machines (VM), the main benefits of applying container-based virtualization in OBUs' resource customization include light weight, increased performance, higher efficiency, and no need for privilege

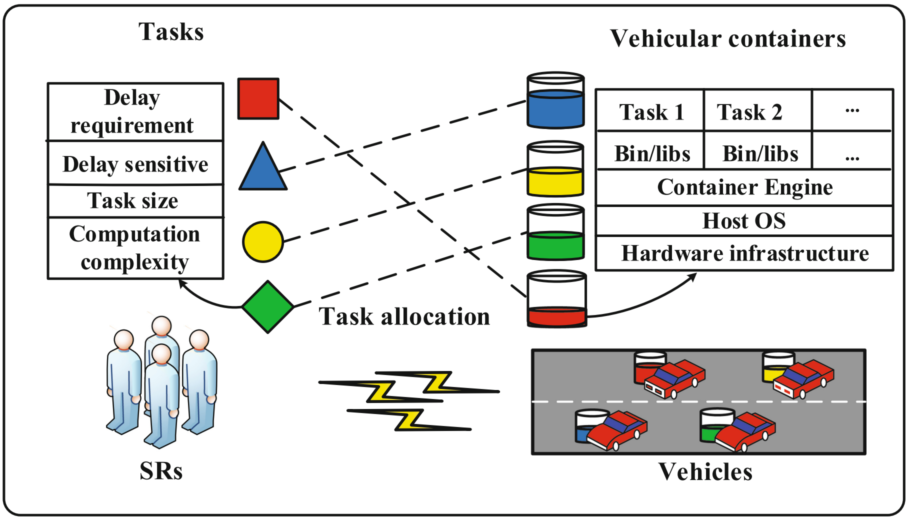

Fig. 1. System model of container-based VEC.

instruction trapping [3]. Each vehicle can accept multiple computation offloading tasks from the SRs, and the vehicular container will allocate computation resources to the tasks efficiently according to the demands of the SRs. In the container-based VEC paradigm, the SRs can reduce service delay by offloading computation tasks to proximate vehicles with rich computation resources. Each vehicle has freedom to decide whether to accept the tasks from the SRs according to its utility.

2.2 Utility Functions of SRs

The set of SRs is denoted as $\mathcal{I}$. The tasks from the SRs are described in four terms as $\alpha_i = \{T_i^{\max}, \lambda_i, s_i, \kappa_i\}, i \in \mathcal{I}$. Here, $T_i^{\max}$ denotes the maximum delay tolerance of the task, s_i denotes the size of the task, and κ_i represents the amount of required computation resource of unit size of task. The tasks are delay-sensitive, and the SRs can get utility $\lambda_i \Delta t_i$ by finishing the task α_i, where λ_i is the unit revenue of saved delay of task α_i and Δt_i is the saved delay in completing the task α_i compared to $T_i^{\max}$. We assume that an orthogonal spectrum is allocated to each vehicle, and thus we can ignore the co-channel interference among vehicles. For a single link between the i-th SR and the type-j vehicle, the signal-to-noise ratio (SNR) at the type-j vehicle can be represented by

$$\gamma_{i,j} = \frac{p_i d_{i,j}^{-\varepsilon} |h_{i,j}|^2}{N_0}, \tag{1}$$

where p_i is i-th SR's transmitting power, $d_{i,j}$ represents the transmission distance between the i-th SR and the type-j vehicle, ε represents the path-loss exponent, $h_{i,j}$ denotes the Rayleigh channel coefficient with a complex Gaussian distribution and N_0 is the power noise. The transmission time for uploading the task α_i to type-j vehicle can be denoted by

$$t_{i,j}^{up} = \frac{s_i}{r_{i,j}} = \frac{s_i}{B \log_2 (1 + \gamma_{i,j})}, \tag{2}$$

where $r_{i,j}$ is the transmission rate between the i-th SR and the type-j vehicle, B denotes the bandwidth of the link. Because of the fast mobility of the vehicles, the task uploading process will fail if the vehicles run out of the communication range of the SRs. According to [14], we denote the dwell time of type-j vehicle inside the communication range of i-th SR as $t_{i,j}^d$. The task uploading process will fail if $t_{i,j}^{up} > t_{i,j}^d$. We assume that the SRs are distributed along the road and their communication range as a circle with a diameter, and $t_{i,j}^d$ can be computed as

$$t_{i,j}^d = \frac{\widetilde{d}_{i,j}}{\overline{v}_j}, \tag{3}$$

where $\widetilde{d}_{i,j}$ is the distance between the type-j vehicle and the endpoint of the i-th SRs communication diameter in the vehicle heading direction [14] and $\overline{v}_j$

denotes the average velocity of type-j vehicle. If the task α_i is offloaded to type-j vehicle, the task execution time $t_{i,j}^{com}$ is denoted as

$$t_{i,j}^{com} = \frac{\kappa_i s_i}{f_{i,j}}, \tag{4}$$

where $f_{i,j}$ is the amount of computation resources contributed by the type-j vehicle. The saved delay in completing the task α_i compared to the maximum delay tolerance $T_i^{\max}$ is given as

$$\Delta t_{i,j} = T_i^{\max} - t_{i,j}^{com} - t_{i,j}^{up}, \tag{5}$$

We introduce a binary variable $x_{i,j}$ as follows,

$$x_{i,j} = \begin{cases} 1, & \text{if task}\, \alpha_i \text{ allocated to type-}j \text{ vehicle,} \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

Similar to [1]. The utility of i-th SR is defined as the revenue minus the payment, which is written by

$$U_i = \sum_{j \in \mathcal{J}} x_{i,j} \left(\lambda_i \Delta t_{i,j} - \pi_{i,j} \right). \tag{7}$$

where π_i^j is the reward paid by i-th SR for the task offloading to the type-j vehicle.

2.3 Utility Functions of Vehicles

If the tasks from SRs are offloaded to the vehicles, there will be energy cost when the vehicles executes the tasks. For the task $\alpha_i, i \in \mathcal{I}$, the utility of the type-j vehicle can be defined as

$$V_j = \sum_{i \in \mathcal{I}} x_{i,j} \left(\pi_{i,j} - e_j \kappa_i s_i \eta {f_{i,j}}^2 \right), \tag{8}$$

where e_j is the energy cost coefficient, η represents the constant determined by the switched capacitance of type-j vehicle and $e_j \kappa_i s_i \eta {f_{i,j}}^2$ denotes the energy cost of type-j vehicle when finishing the task α_i.

Definition 1. *Because the SRs are not aware of vehicles' private information, such as energy cost coefficient, the SRs can sort the vehicles into multiple discrete types. Based on (8), we define the type $\theta_{i,j}$ as follows*

$$\theta_{i,j} \triangleq \frac{1}{e_j \kappa_i s_i \eta}, \tag{9}$$

which suggests that the lower energy cost coefficient, the higher type of vehicles. The set of vehicles' types is denoted as $\Theta_i = \{\theta_{i,1}, \theta_{i,2}, ..., \theta_{i,J}\}, \forall i \in \mathcal{I}$. The types of vehicle are sorted in an ascending order and classified into J types, which are denoted by $\theta_{i,1} < ... < \theta_{i,j} < ... < \theta_{i,J}, \forall i \in \mathcal{I}$.

According to (9), the utility functions of vehicles can be rewritten as

$$V_j = \sum_{i \in \mathcal{I}} x_{i,j} \left(\pi_{i,j} - \frac{f_{i,j}^2}{\theta_{i,j}} \right). \tag{10}$$

2.4 Social Welfare

Based on (7) and (10), social welfare is defined as the summation of the utility functions of the SRs and the vehicles, which is denoted by

$$W = \sum_{i \in I} \sum_{j \in J} x_{i,j} \left(\lambda_i \Delta t_{i,j} - \frac{f_{i,j}^2}{\theta_{i,j}} \right). \tag{11}$$

The payment $\pi_{i,j}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}$ are cancelled out in the social welfare. The social welfare is the profit of saved delay minus the vehicles' energy cost, which is equivalent to optimize the whole system's efficiency, i.e., earning more profits from saved delay at the less cost of energy cost.

3 Problem Formulation

3.1 Contract Formulation

There exists an asymmetric information scenario between SRs and vehicles, the SRs can optimize their expected utilities by using the statistical distributions of vehicles' types from historical data. The SRs are only aware of the probability of the vehicles belong to type-$\theta_{i,j}$ from statistical data. We denote $\beta_{i,j}$ as the probability that the vehicles belong to the type-$\theta_{i,j}$, and $\sum_{j \in J} \beta_{i,j} = 1, \forall i \in \mathcal{I}$. By considering the heterogeneity among different vehicles, the SRs offer different contract items to multiple types of vehicle. The vehicles can accept or reject the offering contract items according to their utility functions. The contract-based optimization problem is to optimize the expected utilities of the SRs, which is formulated as

$$\begin{aligned}
&\max_{(x_{i,j}, f_{i,j}, \pi_{i,j})} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} N \beta_{i,j} U_i \\
&\text{s.t.} (12a)\ x_{i,j} \in \{0,1\}, \sum_{j \in \mathcal{J}} x_{i,j} \le 1, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \\
&(12b)\ \sum_{i \in \mathcal{I}} x_{i,j} f_{i,j} \le f_j^{\max}, \forall j \in \mathcal{J}, \\
&(12c)\ t_{i,j}^{up} \le t_{i,j}^{d}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \\
&(12d)\ \pi_{i,j} - \frac{{f_{i,j}}^2}{\theta_i^j} \ge \pi_{i,j} - \frac{{f_{i,j}}^2}{\theta_{i,j}}, \forall i \in \mathcal{I}, \forall j, k \in \mathcal{J}, \\
&(12e)\ \pi_{i,j} - \frac{{f_{i,j}}^2}{\theta_{i,j}} \ge 0, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}.
\end{aligned} \tag{12}$$

where N denotes the total number of vehicles, and $N\beta_{i,j}$ represents the number of the vehicles that belong to type-$\theta_{i,j}$. (12a) indicates that the variable $x_{i,j}$ is defined as a binary value, and one SR can be allocated with at most one vehicle. (12b) denotes the limit of computation capacity of the vehicle. The constraint

(12c) denotes the delay constraints of task uploading. (12d) is incentive compatibility (IC) constraints which ensure that the vehicles can optimize their utilities by choosing the contract items that are designed for their types. (12e) is individual rationality (IR) constraints which ensure that each type of vehicle's utility is positive.

3.2 Problem Simplification

It is difficult to solve the optimization problem (12) with so many complicated IC constraints and IR constraints which are not-convex and coupled among different types of vehicle. Since the optimization problem (12) is not a convex optimization problem, the complicated constraints in optimization problem should be simplified through following lemmas.

Lemma 1. *For any feasible contract* $(\pi_{i,j}, f_{i,j}), \forall i \in \mathcal{I}, \forall j,k \in \mathcal{J}$, $\pi_{i,j} > \pi_{i,k}$ if and only if *and* $\theta_{i,j} > \theta_{i,k}$, *and* $\pi_{i,j} = \pi_{i,k}$ if and only if $\theta_{i,j} = \theta_{i,k}$.

Proof. Please refer to [15].

Lemma 2. *For any feasible contract* $(\pi_{i,j}, f_{i,j}), \forall i \in \mathcal{I}, \forall j,k \in \mathcal{J}$, $\pi_{i,j} > \pi_{i,k}$ if and only if $f_{i,j} > f_{i,k}$ *and* $\pi_{i,j} = \pi_{i,k}$ if and only if $f_{i,j} = f_{i,k}$.

Proof. Please refer to [15].

Lemma 3. *Given that the IC constraints of all types of vehicle are satisfied, if the utility of the SR is maximized under asymmetric information scenario, the IR constraints of vehicles can be replaced by*

$$\pi_{i,1} - \frac{{f_{i,1}}^2}{\theta_{i,1}} = 0, \forall i \in \mathcal{I}, \tag{13}$$

Proof. From Definition 1, the types of vehicle satisfy $\theta_{i,1} < \theta_{i,2} < \cdots < \theta_{i,j} < \cdots < \theta_{i,J}, \forall i \in \mathcal{I}$. According to the IC constraints in (12d), we can obtain

$$\pi_{i,j} - \frac{{f_{i,j}}^2}{\theta_{i,j}} \geq \pi_{i,1} - \frac{{f_{i,1}}^2}{\theta_{i,j}} \geq \pi_{i,1} - \frac{{f_{i,1}}^2}{\theta_{i,1}} \geq 0, \tag{14}$$

If the IR constraint of type-1 vehicle is guaranteed, the IR constraints of all type of vehicles are satisfied. This completes the proof.

Lemma 4. *The IC constraints of vehicles can be reduced as the local downward incentive compatibility (LDIC):*

$$\pi_{i,j} - \frac{{f_{i,j}}^2}{\theta_{i,j}} \geq \pi_{i,j-1} - \frac{{f_{i,j-1}}^2}{\theta_{i,j}}, \forall i \in \mathcal{I}, \forall j \in \{2, ..., J\}, \tag{15}$$

and the local upward incentive compatibility (LUIC):

$$\pi_{i,j} - \frac{{f_{i,j}}^2}{\theta_{i,j}} \geq \pi_{i,j+1} - \frac{{f_{i,j+1}}^2}{\theta_i^j}, \forall i \in \mathcal{I}, \forall j \in \{1, ..., J-1\}, \tag{16}$$

Proof. Please refer to [16].

Lemma 5. *If the utility of SR is maximized, the IC constraints of vehicles can be reduced as*

$$\pi_{i,j} - \frac{{f_{i,j}}^2}{\theta_{i,j}} = \pi_{i,j-1} - \frac{{f_{i,j-1}}^2}{\theta_{i,j}}, \forall i \in \mathcal{I}, \forall j \in \{2, ..., J\}. \tag{17}$$

Proof. Please refer to [16].

According to Lemmas 3 and 5, the complicated IR and IC constraints can be reduced. Thus the optimization problem (12) can be rewritten as

$$\begin{aligned} &\max_{\{x_{i,j}, f_{i,j}, \pi_{i,j}\}} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} N\beta_{i,j} U_i \\ s.t.(18a)\ & x_{i,j} \in \{0,1\}, \sum_{j \in \mathcal{J}} x_{i,j} \le 1, \forall i \in \mathcal{I}, \\ &(18b) \sum_{i \in \mathcal{I}} x_{i,j} f_{i,j} \le f_j^{\max}, \forall j \in \mathcal{J}, \\ &(18c)\ t_{i,j}^{up} \le t_{i,j}^{d}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \\ &(18d)\ \pi_{i,j} - \frac{{f_{i,j}}^2}{\theta_{i,j}} = \pi_{i,j-1} - \frac{{f_{i,j-1}}^2}{\theta_{i,j}}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \\ &(18e)\ \pi_{i,1} - \frac{{f_{i,1}}^2}{\theta_{i,1}} = 0, \forall i \in \mathcal{I}, \\ &(18f)\ f_{i,1} \le ... \le f_{i,j} \le ... \le f_{i,J}, \forall i \in \mathcal{I}. \end{aligned} \tag{18}$$

4 Solution

We solve the optimization problem (18) by using a standard method. We first resolve the relaxed problem without monotonicity constraint (18f). The solutions are then verified whether to satisfy the monotonicity constraint (18f). By iterating the (18d) and (18e), we have

$$\begin{aligned} \pi_{i,j} &= \frac{{f_{i,1}}^2}{\theta_{i,1}} + \sum_{n=2}^{j} \frac{{f_{i,n}}^2 - {f_{i,n-1}}^2}{\theta_{i,n}} \\ &= \frac{{f_{i,j}}^2}{\theta_{i,j}} + \sum_{n=2}^{j} \left(\frac{1}{\theta_{i,n-1}} - \frac{1}{\theta_{i,n}} \right) {f_{i,n-1}}^2, \end{aligned} \tag{19}$$

where $\forall i \in \mathcal{I}, \forall j \in \{2, ..., J\}$. Substitute (18e) and (19) into optimization problem (18), and all $\pi_{i,j}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}$ can be removed from the optimization problem (18), which becomes

$$\begin{aligned}
&\max_{\{x_{i,j},f_{i,j}\}} \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} N\beta_{i,j}x_{i,j}\left[\lambda_i\left(T_i^{max}-\frac{\kappa_i s_i}{f_{i,j}}-\frac{s_i}{r_{i,j}}\right)\right.\\
&\left.-\sum_{j=1}^{J-1}\left(\frac{1}{\theta_i^j}\sum_{n=j}^{J}\beta_{i,n}-\frac{1}{\theta_{i,j+1}}\sum_{n=j+1}^{J}\beta_{i,n}\right){f_{i,j}}^2-\frac{\beta_{i,J}}{\theta_{i,J}}{f_{i,J}}^2\right]\\
&\text{s.t.}(20a)\; x_{i,j}\in\{0,1\},\sum_{j\in\mathcal{J}}x_{i,j}\le 1,\forall i\in\mathcal{I},\\
&\quad(20b)\;\sum_{i\in\mathcal{I}}x_{i,j}f_{i,j}\le f_j^{\max},\forall j\in\mathcal{J},\\
&\quad(20c)\; t_{i,j}^{up}\le t_{i,j}^{d},\forall i\in\mathcal{I},\forall j\in\mathcal{J},
\end{aligned} \tag{20}$$

First, we can use standard convex optimization tools in [17] to solve it to get ${f_{i,j}}^*$. Then ${\pi_{i,j}}^*$ can be calculated by (18e) and (19). After that, we need to check whether the solutions satisfy the monotonicity constraint (18f). If the solutions $\widehat{f_{i,j}}^*,\forall i\in\mathcal{I},\forall j\in\mathcal{J}$ satisfy the monotonicity constraint (18f), the solutions are our optimal solutions. However, if the solutions $\widehat{f_{i,j}}^*,\forall i\in\mathcal{I},\forall j\in\mathcal{J}$ do not satisfy the monotonicity constraint (18f), the solutions are infeasible solutions. Thus, we need to make some adjustments as follows. Since $U_i,\forall i\in\mathcal{I}$ are concave functions on $\widehat{f_{i,j}}^*,\forall i\in\mathcal{I},\forall j\in\mathcal{J}$, the infeasible solutions can be replaced by feasible solutions iteratively [18]. When there exists an infeasible solution $\left\{\widehat{f_{i,m}}^*,\widehat{f_{i,m+1}}^*,...,\widehat{f_{i,n}}^*\right\}$, set

$$ {f_{i,j}}^* = \arg\max_{\{f\}}\sum_{s=m}^{n}U_s, i\in\{m,m+1,...,n\}, \tag{21}$$

After obtaining the feasible solutions ${f_{i,j}}^*,\forall i\in\mathcal{I},\forall j\in\mathcal{J}$, we can derive the optimal price ${p_{i,j}}^*$ as follow

$$ {\pi_{i,j}}^* = \begin{cases} \frac{{f_{i,1}}^{*2}}{\theta_{i,1}}, & j=1,\\ \frac{{f_{i,j}}^{*2}}{\theta_{i,j}}+\sum_{n=2}^{i}\left(\frac{1}{\theta_{i,n-1}}-\frac{1}{\theta_{i,n}}\right){f_{i,n-1}}^{*2}, & j=\{2,3,...,J\}. \end{cases} \tag{22}$$

Then, ${x_{i,j}}^*$ can be calculated iteratively by the constraints (20a), (20b) and (20c). So far, we have derived the optimal contract $({f_{i,j}}^*,{\pi_{i,j}}^*),\forall i\in\mathcal{I},\forall j\in\mathcal{J}$ which can optimize the expected utilities of the SRs and satisfy the IR and IC constraints.

Table 1. Simulation parameters

Parameter	Setting
Radius of the SRs communication coverage	200 m
Number of SRs I	15
Number of vehicles' type J	10
Bandwidth of SRs B	10 MHz
Transmission power of SRs p_i	30 dBm
Noise power N_0	-114 dBm
Path loss exponent ε	3.4
Maximum delay tolerance T_i^{max}	8$-$10 s
Mapping from bit to cycles κ_i	$1 * 10^3 - 1.5 * 10^3$ cycle/bit
Size of task s_i	$3 * 10^6 - 4 * 10^6$ bit
Saved delay profit coefficient λ_i	0.1$-$1
Effective switched capacitance η	10^{-28}
Energy cost coefficient e_j	0.1$-$1
Velocity of vehicles $\bar{v}_j$	2$-$20 m/s
Maximum computation resource of vehicle f_j^{max}	2.5$-$3 GHz

5 Numerical Results

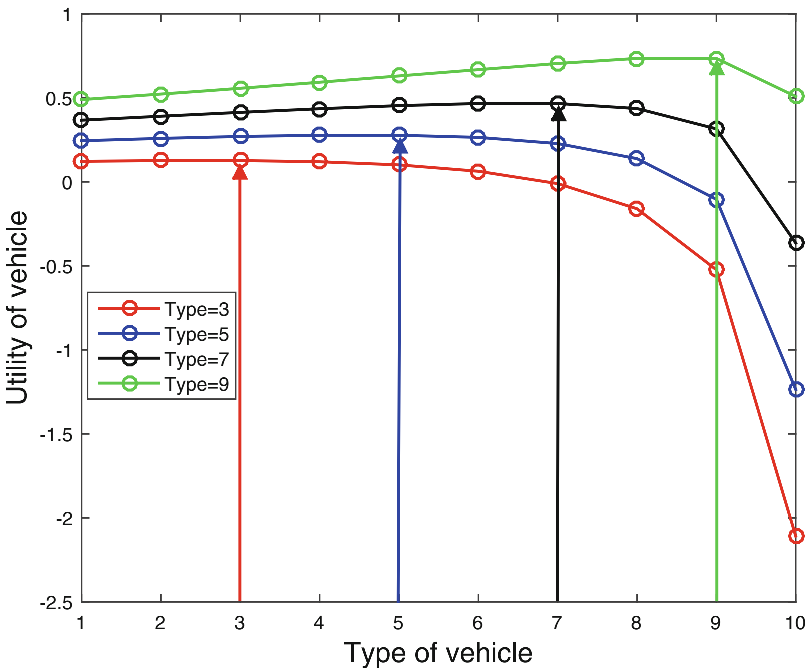

Fig. 2. Utility of vehicle versus type of vehicle.

We consider a two-lane two directional road randomly distributed with 10 vehicles and 15 SRs. We assume that the SRs are distributed along the road and their communication range as a circle with a diameter. The number of vehicle types is equal to the number of vehicles. Without loss of generality, the vehicle types are following a uniform distribution. In our simulation, we first give an analysis about the feasibility (IC and IR constraints) of the proposed contract-based incentive scheme. Second, we conduct the comparisons of social welfare with different types of vehicles. Finally, we compare the social welfare by varying the number of SR. For comparisons, the numerical results are performed by solving the problem by using the proposed contract-based scheme under asymmetric information scenario (CA), contract-based scheme under complete information scenario (CC) [16], Stackelgerg game scheme (SG) [19] and take-it-or-leave scheme (ToL) [20]. The system performance is being simulated using MATLAB with system parameters in Table 1.

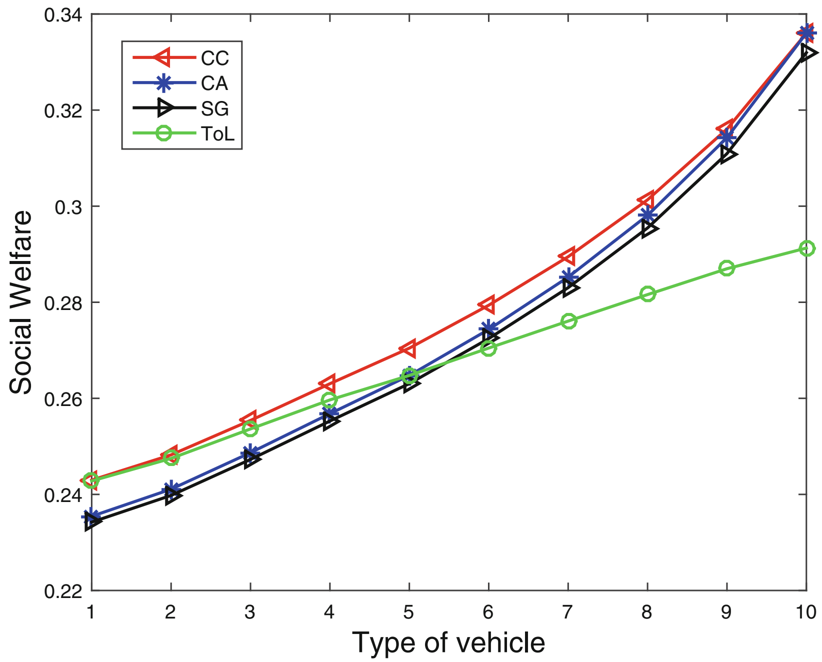

Fig. 3. Social welfare versus type of vehicle.

The feasibility of IR and the IC constraints of the proposed contract-based scheme under asymmetric information scenario is shown in Fig. 2. A SR (i.e., $i = 10$) is selected randomly. Figure 2 shows that the utilities of type-3, type-5, type-7, and type-9 vehicles when the vehicles select all the contract items $(\pi_i^j, f_i^j), i = 10, j \in \mathcal{J}$ offered by the i-th SR. We can observe that the utility of each type of vehicles can maximize its utility when the vehicles select the contract item that is designed for their types, which suggests the IC constraints are satisfied. Further, the utilities of vehicles are non-negative when they choose the best contract item that fits their corresponding types, which means that the IR constraints are guaranteed. After the vehicles select the contract items, the types of vehicles will be known by the SRs. Therefore, the information asymmetry between SRs and vehicles can be overcome.

We illustrate social welfare with respect to different types of vehicle under four schemes in Fig. 3. We set the threshold type of ToL scheme is 5. It can be seen from Fig. 3 that the social welfare achieved by all schemes increase with type of vehicle. It is profitable to employ higher type of vehicles to help execute computation tasks from SRs. The higher type of vehicles are with higher energy efficiency to share resources with SRs. Furthermore, the CC scheme achieves the maximum value of social welfare as the upper bound. The social welfare achieved by the proposed CA scheme is better than SG scheme and ToL scheme. Due to the information asymmetry between SRs and vehicles, SRs have no acknowledgment of the type of vehicle, the designed IC constraint-based CA scheme can only bring a approximate optimal social welfare, which is upper bounded by CC scheme under complete information scenario. The SRs are fully aware of the types of vehicle under complete information scenario and tries to extract all profit from the vehicles. The gap between social welfare and that of CA scheme increases along with the type of vehicles.

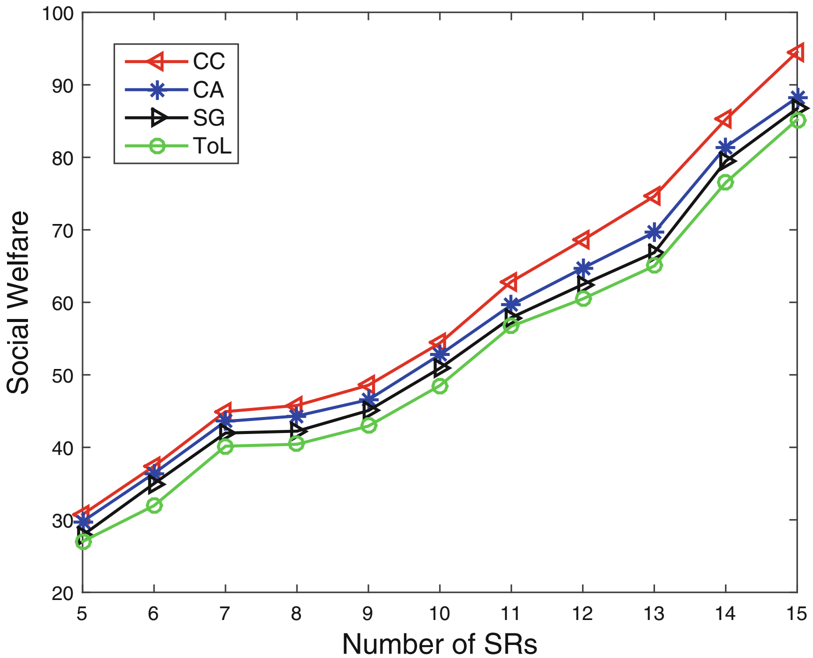

Fig. 4. Social welfare versus number of SRs.

Figure 4 shows the performance of social welfare as a function of the number of SRs under all four schemes. It can be observed from Fig. 4 that social welfare achieved by all four schemes increase with the number of SRs. As shown in Fig. 4, the CC scheme gives the highest performance of social welfare among all four schemes, followed by the CA scheme, SG scheme and ToL scheme. That because SRs extract revenue from the vehicles as much as possible under complete information scenario, and less benefits are left to the vehicles. While in CA scheme, vehicles have limited contract items to select from SRs under asymmetric information scenario. However, the vehicles have freedom to optimize their own utility in SG scheme, and they can reserve more benefits from SRs. Therefore, the social welfare achieved by contract-based scheme are better than that of SG scheme. ToL scheme achieves the lowest social welfare among four schemes. The reason is that any vehicle whose type larger than the threshold type will reject the contract offered by SRs. In this case, only the vehicles higher than the threshold type can achieve non-negative utilities.

6 Conclusion

In this paper, we propose a container-based VEC paradigm with efficient and flexible customization of vehicles' resources. Then, we present a contract-based incentive mechanism to motivate vehicles to share their computation resource and help offload tasks from SRs. The proposed novel contract-based framework solves the task allocation problem among multiple vehicles and multiple SRs. To overcome the asymmetric information scenario between the SRs and the vehicles, a set of resource-reward contract items are designed for maximizing the SRs expected utilities while ensuring the IR and IC constraints of the vehicles. Finally, numerical results show that the proposed contract-based incentive mechanism is more effective than the traditional schemes.

Acknowledgment. Beihai Tan is the corresponding author of this paper. The work is supported in part by program of NSFC under Grant no. 61971148, the Science and Technology Program of Guangdong Province under Grant no. 2015B010129001, and Natural Science Foundation of Guangxi Province under Grant 2018GXNSFDA281013.

References

1. Liwang, M., Dai, S., Gao, Z., Tang, Y., Dai, H.: A truthful reverse-auction mechanism for computation offloading in cloud-enabled vehicular network. IEEE Internet Things J. **6**, 4214–4227 (2019)
2. Yu, R., Zhang, Y., Gjessing, S., Xia, W., Yang, K.: Toward cloud-based vehicular networks with efficient resource management. IEEE Network **27**(5), 48–55 (2013)
3. Kaur, K., Dhand, T., Kumar, N., Zeadally, S.: Container-as-a-Service at the edge: trade-off between energy efficiency and service availability at fog nano data centers. IEEE Wirel. Commun. **24**(3), 48–56 (2017)
4. Li, X., Hu, B.-j., Chen, H., Li, B., Teng, H., Cui, M.: Multi-hop delay reduction for safety-related message broadcasting in vehicle-to-vehicle communications. IET Commun. **9**(3), 404–411 (2015)

5. Li, X., Hu, B.-J., Chen, H., Andrieux, G., Wang, Y., Wei, Z.-H.: An RSU-coordinated synchronous multi-channel MAC scheme for vehicular ad hoc networks. IEEE Access **3**, 2794–2802 (2015)
6. Dai, Y., Xu, D., Maharjan, S., Zhang, Y.: Joint load balancing and offloading in vehicular edge computing and networks. IEEE Internet Things J. **6**, 4377–4387 (2019)
7. Yang, C., Liu, Y., Chen, X., Zhong, W., Xie, S.: Efficient mobility-aware task offloading for vehicular edge computing networks. IEEE Access **7**, 26652–26664 (2019)
8. Liu, Y., Yu, H., Xie, S., Zhang, Y.: Deep reinforcement learning for offloading and resource allocation in vehicle edge computing and networks. IEEE Trans. Veh. Technol. **68**(11), 11158–11168 (2019)
9. Abdelhamid, S., Hassanein, H., Takahara, G.: Vehicle as a resource (VaaR). IEEE Network **29**(1), 12–17 (2015)
10. Huang, X., Yu, R., Liu, J., Shu, L.: Parked vehicle edge computing: exploiting opportunistic resources for distributed mobile applications. IEEE Access **6**, 66649–66663 (2018)
11. Morabito, R., Cozzolino, V., Ding, A.Y., Beijar, N., Ott, J.: Consolidate IoT edge computing with lightweight virtualization. IEEE Network **32**(1), 102–111 (2018)
12. Huang, X., Li, P., Yu, R.: Social welfare maximization in container-based task scheduling for parked vehicle edge computing. IEEE Commun. Lett. **23**(8), 1347–1351 (2019)
13. Arif, S., Olariu, S., Wang, J., Yan, G., Yang, W., Khalil, I.: Datacenter at the airport: reasoning about time-dependent parking lot occupancy. IEEE Trans. Parallel Distrib. Syst. **23**(11), 2067–2080 (2012)
14. Xu, C., Wang, Y., Zhou, Z., Gu, B., Frascolla, V., Mumtaz, S.: A low-latency and massive-connectivity vehicular fog computing framework for 5G. In: 2018 IEEE Globecom Workshops (GC Wkshps), pp. 1–6. IEEE (2018)
15. Bolton, P., Dewatripont, M.: Contract Theory. MIT Press, Cambridge (2005)
16. Hou, Z., Chen, H., Li, Y., Vucetic, B.: Incentive mechanism design for wireless energy harvesting-based Internet of Things. IEEE Internet Things J. **5**(4), 2620–2632 (2017)
17. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
18. Gao, L., Wang, X., Xu, Y., Zhang, Q.: Spectrum trading in cognitive radio networks: a contract-theoretic modeling approach. IEEE J. Sel. Areas Commun. **29**(4), 843–855 (2011)
19. Liu, T., Li, J., Shu, F., Tao, M., Chen, W., Han, Z.: Design of contract-based trading mechanism for a small-cell caching system. IEEE Trans. Wireless Commun. **16**(10), 6602–6617 (2017)
20. Zhou, Z., Liu, P., Feng, J., Zhang, Y., Mumtaz, S., Rodriguez, J.: Computation resource allocation and task assignment optimization in vehicular fog computing: a contract-matching approach. IEEE Trans. Veh. Technol. **68**(4), 3113–3125 (2019)