This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2690961, IEEE Internet of Things Journal



# Resource Allocation in Vehicular Cloud Computing Systems with Heterogeneous Vehicles and Roadside Units

Chun-Cheng Lin, *Member, IEEE*, Der-Jiunn Deng*, *Member, IEEE*, and Chia-Chi Yao

***Abstract*—Vehicular cloud computing (VCC) system coordinates the vehicular cloud (consisting of vehicles' computing resources) and the remote cloud properly to provide in-time services to users. Although pervious works had established the models for resource allocation in the VCC system based on semi-Markov decision processes (SMDP), few of them discussed heterogeneity of vehicles and influences of roadside units (RSUs). Heterogeneous vehicles made by different manufacturers may be equipped with different amount of computing resources; and furthermore, RSU can enhance the computing capability of VCC. Therefore, this work proposes an SMDP model for VCC resource allocation that additionally considers heterogeneous vehicles and RSUs, and an approach for finding the optimal strategy of VCC resource allocation. The two additional features significantly elaborate the SMDP model, and demonstrate different results from the original model. Simulation shows that the resource allocation in the VCC system can be captured by the proposed model, which performs well in terms of long-term expected values (consisting of consumption costs of power and time), under various parameter settings.**

***Index Terms*—Intelligent transportation system, vehicular cloud computing, semi-Markov decision processes, VANET**

## I. Introduction

Many countries have strived for possible practical applications of intelligent transportation systems (ITS). By ITS, the government can reduce risks of accidents on roads, and the transportation industry can provide sound service quality and operational efficiency through advanced information transmission. As traffic safety and logistics efficiency have received much attention, lots of research has focused on advances in the technologies of vehicular networks integrated with cloud computing and communications [1].

Vehicular ad hoc networks (VANET) were proposed for providing the abilities of network connectivity and real-time information sharing among vehicles. Furthermore, VANET has been integrated with sensors and roadside units (RSU) for communications with vehicles on roads to promote traffic safety [2], [3]. Recently, vehicular cloud computing (VCC) systems have been proposed for integrating VANET with cloud computing technologies. The abilities of computation and communications of VCC systems provide practicability and convenience to users on roads, so that the related research has received a lot of attention recently [4], [5].

In VANETs, each vehicle is equipped with a vehicle equipment (VE), which performs like a small computer with a networking interface [6]. Through VEs, vehicles can communicate with other VEs, RSUs, sensors, and even other vehicles. By interchanging respective known information and the information collected from sensors, the driving safety can be promoted, and the time consumed in driving to the destination can be reduced. Introducing cloud computing systems makes the abilities of VEs and RSUs be almost equivalent with small computers [7], [8]. Such a VCC network provides not only the ability of transmitting data, but also the computing services, e.g., the shortest path to the destination, the best driving speed, and so on.

In addition to the information collected by VEs and RSUs, VCC systems transmit service requests and results in two ways: vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V) [9], [10]. V2I allows vehicles to transmit their respective data to the vehicular networking platform through RSUs, and also to download the required data through RSUs [11]. V2V allows nearby vehicles to transmit data to each other directly without RSUs so that redundant power consumption and transmission delay time can be saved [12]. Therefore, data sharing among vehicles on roads can be achieved efficiently through the above two transmission ways, to improve transportation safety and reduce traffic accidents [13], [14].

In general, based on functionality, VCC systems are divided into four parts: Computation-as-a-Service (CaaS), Network-as-a-Service (NaaS), Storage-as-a-Service (SaaS), and Sensing-as-a-Service ($S^2$aaS) [6]. This work focuses on CaaS, in which the VE equipped in each vehicle provides a limited computing resource, like a small computer. Especially, during rush hour, connectivity among VEs of vehicles can constitute a network with a strong computing resource, bringing lots of benefits. Early research on VCC systems considered applying RSUs' computing resources to complete service requests from vehicles on roads, and focused on allocating limited resources to achieve multiple purposes on road safety. For instance, the work in [15] proposed a game-theoretical approach for the system. Later, some research considered connectivity of VEs

C.-C. Lin and C.-C. Yao are with Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu 300, Taiwan. E-mails: cclin321@nctu.edu.tw, okokoooppp@gmail.com.

D.-J. Deng is with Department of Computer Science and Information Engineering, National Changhua University of Education, Changhua 500, Taiwan. E-mail: djdeng@cc.ncue.edu.tw.

* D.-J. Deng is the corresponding author of this paper.

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2690961, IEEE Internet of Things Journal



of vehicles with clouds for resource sharing.

This work integrates both RSU's and vehicles' computing resources in the VCC system to provide services. Additionally, most previous works assumed homogeneous vehicles in V2I and V2V frameworks. However, in reality, vehicles made by different manufactures may be equipped with VEs with different amount of computing resources. Therefore, this work establishes a model for resource allocation in the VCC system with heterogeneous vehicles and RSUs based semi-Markov decision processes (SMDP) [16], which allocates either vehicular cloud (consisting of vehicle' computing resources) or remote clouds to handle vehicle's service requests.

The SMDP is one of the Markov decision processes (MDPs). MDPs aim to find the optimal decision-making strategy to affect a system with a dynamic stochastic process, in order to maximize a long-term value. MDPs are categorized into three types: in the discrete-time MDP, transition times of the system model and the timing points to make decisions to control the system are fixed and known; in the continuous-time MDP, these timing points are not fixed and can be arbitrary at any continuous time point; in the SMDP, these timing points are not fixed either, but are arbitrary at discrete time points.

In VCC systems, the times between vehicle arrivals and between vehicle departures are not fixed, i.e., the transition times in the system are not fixed. In addition, optimal decisions are made immediately for each service request from vehicles, i.e., at arbitrary discrete times. Therefore, the SMDP is applied to model the concerned VCC system. In this VCC system, heterogeneous vehicles meet the reality but their provided resources are more uncertain; whereas RSUs can provide relatively stable and reliable computing resources. This work assumes that different types of vehicles provide different amount of computing resources and arrive at the VCC system with different probability distributions, so that the SMDP model becomes more complex.

## II. Related Works

Previous works on vehicular networks focused on coping with traffic congestion on roads or preventing emergency conditions, e.g., watering roads and vehicle malfunction. For instance, the work in [8] considered insufficient positioning capability of GPS, and hence improved RFID to assist in positioning, to establish a VANET environment that can position vehicles more precisely. The works in [2], [3], [17] considered deployment of sensors and RSUs on roads in VANETs, with minimal deployment cost and number of transmission hops.

On mobile clouds, the work in [18] introduced a model of mobile devices connected to mobile clouds, and investigated cloud storage services on mobile clouds. The work in [19] adopted the SMDP to address the resource allocation problem in mobile cloud computing systems. The work in [20] proposed a model with mobile clouds and vehicular networks in which each vehicle can upload photos and videos of traffic conditions to the cloud, and share them with other vehicles.

The mobile clouds proposed in [7] were integrated to VANET, and the work in [8] established a VCC system framework. The previous works on VCC systems are divided into five categories [5]: security and privacy, data aggregation, energy efficiency, interoperability, and resource management. The work in [21] investigated the problem of safety and privacy arising from practical applications of VCC systems, and proposed some strategies for addressing the problems of privacy leaks of road users and authentication of high-speed moving vehicles. The work in [22] integrated trajectory data of taxis in real world to establish a model that can evaluate and forecast the serviceability of mobile vehicular cloudlet.

Some research focused on resource management in VCC systems. The work in [15] considered a cloud consisting of RSUs to help resource management and allocation between RC and VC. The [23] considered the V2V in the VCC system to be applied in the scheduling model of resource sharing. With the concept of V2V in the VCC system, the work in [16] defined a centralized VCC system that allocates computing resources of homogenous vehicles. The work in [24] adopted SMDP to analyze video streaming in heterogeneous cognitive vehicular networks to establish a resource allocation model to promote quality of service (QoS). The work in [25] adopted SMDP to establish the optimal resource allocation strategy to allocate machine-type communication gateways in a software-defined networking framework for internet of things.

The SMDP has also been applied to address various problems. For instance, the work in [26] proposed a partially observable SMDP to establish an asset maintenance model to reduce uncertainty of asset states and the rate at which deterioration takes place. The work in [27] proposed an SMDP to minimize the network signaling and processing load of cellular network coverage range. The works of [16], [24] and [25] adopted SMDPs to solve resource allocation problems of various types of cloud networks.

## III. System Framework

### A. Cloud architecture

This work tailor the system proposed in [16], [28] to propose our VCC system, as illustrated in Fig. 1. The cloud architecture in the system is divided into two layers: remote cloud (RC) and vehicular cloud (VC). The RC consists of multiple powerful computing resources so that any service can be handled efficiently through RC computation. Different from previous settings, this work considers heterogeneous vehicles in the VC. The VC is constituted by a number of VEs of heterogeneous vehicles within a certain range and a number of RSUs with computing resources. To quantize the computing resources in the VC, the computing resources provided by vehicles and RSUs are measured in terms of number of resource units (RU). These RUs from a resource pool, which is controlled and allocated by a centralized VC system. The computing ability of VC depends on the number of RUs in the resource pool.

Because of heterogeneity of vehicles concerned in this work, the VEs in theses heterogeneous vehicles provide different numbers of RUs. For instance, in Fig. 1, suppose that each black vehicle has a weaker computing ability to provide only one RU to the VC; whereas each blue vehicle (which may cost

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2690961, IEEE Internet of Things Journal



more expensive) has a strong computing ability to provide 2 RUs. Suppose that each RSU provides 2 RUs. Then, 2 RSUs and 5 vehicles (in which 2 in blue and 3 in black) in Fig. 1 provide 11 RUs in the VC resource pool.

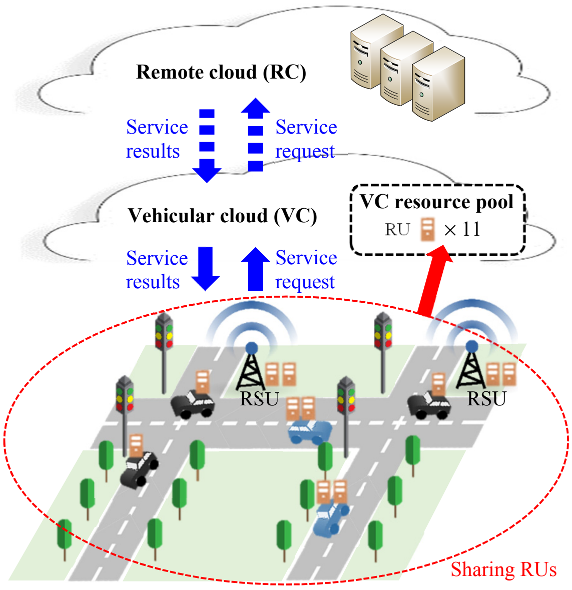


Fig. 1. Illustration of the system framework.

### B. Handling the service requests from vehicles

Consider that a vehicle passes through the VC service range, and makes a service request to the VC. Then, there are two cases for VC responses. In the first case, to handle this request directly, the VC allocates a number of RUs in the resource pool to compute the requested service. The number of the allocated RUs directly influences the time efficiency of completing this request. In the second case, the VC transfers this request to RC to handle it (see Fig. 1). In general, the RC has a powerful computing resource, and hence, it can finish this service request immediately. It is supposed that the RC almost does not consume any time. Although the VC does not use any RU computing, it consume a little power and time.

### C. Assumptions and notations

The power of RSUs is supported by wired electricity, so RSUs are supposed to have an infinite lifetime, and so is the VCC system. The service range size of the VCC system depends on geographical distribution of the vehicles dynamically connected to the VC. Consider that number and positions of RSUs in the VC are fixed, and the connectivity between the VC and RSUs is certain. Hence, the service range of the VC is based on its connectivity to RSUs.

Let $R$ denote the number of RSUs in the VCC system. Suppose that each RSU can provide $T_R$ RUs. For maintaining the stability of operating the VCC system, the number of vehicles served by the VCC system is assumed to have an upper bound, which is denoted by $K$. Let $M$ denote the number of RUs available in the VC resource pool currently. Number $M$ changes with time, because it is decided by the number of vehicles dynamically passing through the VCC service range.

The number of vehicles served by the VCC system is dynamic. The previous work assumed that both numbers of arrivals and departures of homogeneous vehicles follow Poisson distributions. However, this work considers heterogeneity of vehicles, i.e., vehicles are of different types. Hence, this work assumes that both numbers of arrivals and departures of vehicles of each type follow Poisson distributions. Let $\lambda_j$ (resp., $\mu_j$) denote the number of arrivals (resp., departures) of vehicles of the type that provides $j$ RUs to the VC. Similarly, numbers of arrivals and departures of service requests per vehicle also follow Poisson distributions. And, the two numbers are denoted by $\lambda_p$ and $\mu_p$, respectively. Note that subscript $p$ is a notation, not an index.

Therefore, the average time consumed in finishing a service request by an RU is $1/\mu_p$. This work assumes that the number of the service requests that are finished at each unit time grows exponentially with the number of the RUs that handles these requests. For instance, the number of the service requests that are finished by 2 RUs is $2\mu_p$; and the average time consumed in finishing a service request by two RUs is $1/(2\mu_p)$.

### D. System state setting

Before applying the SMDP to solve the concerned problem, we need to establish a representation for states of the VCC system. The set of all these states is represented as follows:

$$S=\left\{s \mid s=\left(n_1,n_2,\ldots\ldots,n_{N_R},V_1,V_2,\ldots\ldots,V_{S_R},e\right)\right\},$$

which consists of the following three components:

1. Number of the service requests each of which is handled by $i$ RUs in the VC resource pool currently is denoted by $n_i$. This work restricts that $i$ must be an integer and be no greater than the number of RUs in the VC resource pool, i.e., $i \le M$. In addition, when the number of RUs to handle a service request exceeds some threshold $N_R$, the time of computing this service cannot be reduced any longer. That is, we have the following constraints:
$$i \in \{1,2,\ldots,\min\{N_R,M\}\} \tag{1}$$
Since those numbers when $M > N_R$ do not influence the computing time, they are not needed to be included in the state set. Hence, the system state $s$ includes only $N_R$ numbers as follows: $(n_1,n_2,\ldots,n_{N_R})$.
2. Because of heterogeneity of vehicles, vehicles are divided into different types according to the number of RUs to be provided. Hence, let the number of the vehicles of the type that provides $j$ RUs and are connected to the VC currently be denoted by $V_j$. Suppose that the number of RUs that can be provided by a vehicle is $S_R$. Then, the system state $s$ includes the following $S_R$ numbers: $(V_1,V_2,\ldots,V_{S_R})$. Recall that $M$ denotes the number of RUs available in the VCC system; and $R$ denotes the number of RSUs, each of which can provide $T_R$ RUs. Hence, we have the following relation: $T_R \cdot R+\sum_{j=1}^{S_R} j\cdot V_j = M$. In addition, recall that $K$ denotes the maximal number of vehicles that the VCC system can serve. Hence, we can obtain the upper bound

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2690961, IEEE Internet of Things Journal



of $M$ (when $K$ vehicles of the type that can provide the maximal $S_R$ RSUs) as described as follows:

$$R \le M \le R \cdot T_R + K \cdot S_R \quad (2)$$

3. Let the set of all possible events that occur in the VCC system be denoted by $E$. These events include: arrival of a vehicle of the type that provides $j$ RUs (denoted by $B_j$), departure of a vehicle of the type that provides $j$ RUs (denoted by $B_{-j}$), receiving a service request (denoted by $A$), and finishing a service that was handled by $i$ RUs (denoted by $D_i$). Hence, set $E$ is expressed as follows:

$$E = \{B_1, B_2, ......, B_{S_R}, B_{-1}, B_{-2}, ......, B_{-S_R}, A, D_1, D_2, ......, D_{N_R}\} \quad (3)$$

Therefore, the system state $s$ includes an event $e \in E$.

The notations used in the system are summarized in Table I.

TABLE I
The notations used in the system framework.

| Notation | Meaning |
|---|---|
| $R$ | Number of the RSUs in the VCC system. |
| $K$ | Maximal number of vehicles that the VCC system can serve. |
| $M$ | Number of RUs in the VC resource pool. |
| $N_R$ | The maximal number of RUs that can be allocated to a service request. |
| $S_R$ | The maximal number of RUs that can be provided by a vehicle. |
| $\lambda_j$ | Number of arrivals of the vehicles of the type that provides $j$ RUs to the VC. |
| $\mu_j$ | Number of departures of the vehicles of the type that provides $j$ RUs to the VC. |
| $\lambda_p$ | Number of arrivals of service requests per vehicle. |
| $\mu_p$ | Number of departures of service requests per vehicle. |
| $S$ | Set of states in the VCC system. |
| $A_s$ | Set of the actions that can be taken in the VCC system. |
| $n_i$ | Number of the service requests each of which is handled by $i$ RUs in the VC resource pool currently. |
| $V_j$ | Number of the vehicles of the type that provides $j$ RUs. |
| $e$ | An event that occurs in the VCC system. |
| $B_j$ | Arrival of a vehicle that provides $j$ RUs. |
| $B_{-j}$ | Departure of a vehicle that provides $j$ RUs. |
| $A$ | The event of receiving a service request. |
| $D_i$ | The event of finishing a service that was handled by $i$ RUs. |
| $a$ | An action that is taken under a state. |

### E. Difference of this system from the previous system

The differences of the proposed system from the previous SMDP system in [16] are listed as follows:

- Different from the previous SMDP model without RSUs, the proposed VC resource pool additionally includes the RUs provided by RSUs (i.e., $T_R \cdot R$ RUs provided by $R$ RSUs).
- Different from the previous SMDP model that assumed arrivals and departures of all vehicles to follow the same Poisson distribution, the proposed model in this work considers different Poisson distributions for heterogeneous vehicle types, respectively (i.e., $\lambda_j$ and $\mu_j$ for each vehicle type $j$).
- Different from the previous SMDP model that considered a sum of RUs provided by all vehicles, the representation of a state in this work is extended to include the amount of RUs of multiple vehicle types (i.e., $V_j$ in each vehicle type $j$).
- The setting of heterogeneous vehicles drastically increases the number of states for the VCC system, so that the resource allocation of this system becomes much complex than the original model.

## IV. The Proposed SMDP

This work applies the SMDP to find a strategy of resource allocation in the VCC system so as to save the most long-term expected cost of consuming power and time. The flowchart of the proposed SMDP is depicted in Fig. 2, which is explained as follows. Initially, parameters are set, and the set of all states (including service requests, number of vehicles of different types, and events) in the VCC system is constructed, as described in the previous section.

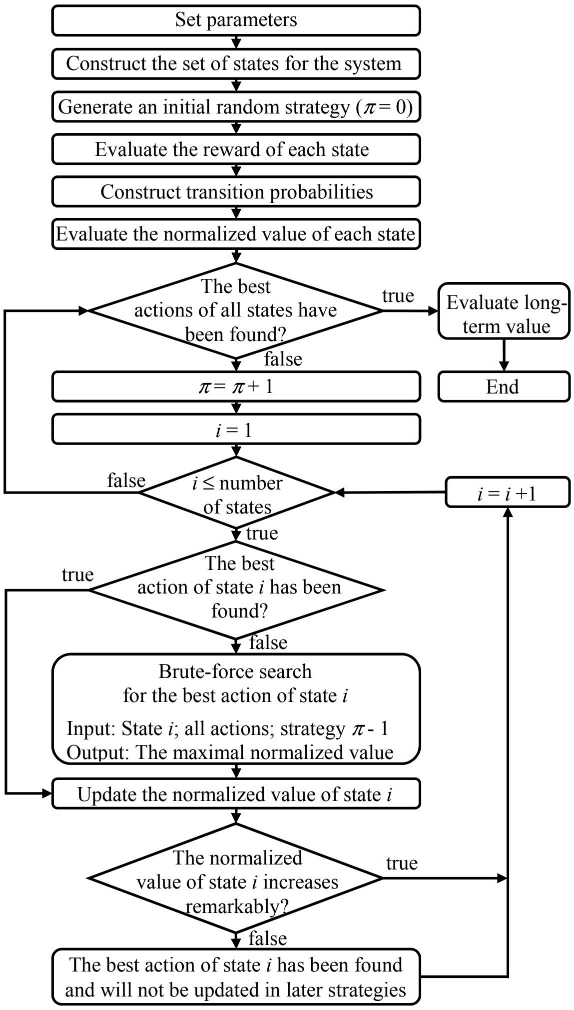


Fig. 2. Flowchart of the proposed SMDP.

For each state in the VCC system, we have to determine an *action* for whether RUs are allocated and how many RUs are allocated. When each state has a corresponding action, actions of all states constitute a *strategy*, by which, in practice, we can know what action should be taken at any time in the VCC system. Therefore, the SMDP first generates an initial random strategy, i.e., the action for each state is taken randomly. Let $\pi$

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

denote the number of strategies that have been searched. Hence, $\pi = 0$ for the initial strategy.

To evaluate performance of a strategy, the SMDP computes a *reward* of each state which reflects the saved cost of consuming power and time after the action of the state in the strategy is executed. However, it is not enough to apply the reward to evaluate performance of a strategy, because the reward is concerned with only the action of the current state, without the later actions. Therefore, the SMDP computes not only a reward but also a *value* (which reflects both current and future actions).

Before computing the value of each state, we need to first compute the transition probabilities for all states of the VCC system. With these probabilities, the expected value of the future state can be calculated, and a normalization procedure for this value can be conducted.

Next, the SMDP enters an iterative improvement process to find the strategy (i.e., the best actions for all states) with the maximal normalized value (i.e., to save the most cost of consuming power and time in a long run). It repeats the following procedure until the best actions of all states have been found. Consider a new strategy, and increase the number of strategies by one (i.e., $\pi = \pi + 1$). For each of such a strategy, we find the best action of each state $i$. Note that a status for each state $i$ is used to record whether the best action of state $i$ has been found. If the best action of state $i$ has not been found, we apply a brute-force search, which considers all actions of state $i$ and the previous strategy (i.e., $(\pi - 1)$-th strategy) to find the action with the maximal normalized value of state $i$. Then, update the normalized value of state $i$ in the $\pi$-th strategy. If the normalized value of state $i$ increases remarkably (meaning that a better normalized value could be found in later iterations), we do nothing and consider the next state; otherwise (meaning that no better normalized value could be found), the status of state $i$ records that the best action of state $i$ has been found and will not be updated in later strategies.

After obtaining the best strategy through the above process, we base the transition probabilities to evaluate the long-term value, used as the performance of this approach.

The rest of this section gives the details on main components of the SMDP, including setting actions, evaluating transition probabilities, evaluating the reward, evaluating the value, and stop condition. First, all actions under all possible states are defined in (4). With these actions, the SMDP is to find the optimal action under each state. Then, transition probabilities between states are established in (5)–(10). With these probabilities, the expected transition of each state from its all possible former states can be realized. Then, a reward function to reflect the objective of the optimal actions is calculated from (11)–(16), and an expected value function (considering transition probabilities) to be optimized is defined in (17)–(22). Finally, stop conditions of the SMDP are given in (23).

### A. Setting actions

In the SMDP, each state is associated with an action. In the VCC system, the action of allocating the number of RUs is taken according the current event $e \in E$. The set of actions $A_s$ is represented as follows:

$$A_s = \begin{cases} \{-1\}, & \text{if } e \in \{B_j, B_{-j}, D_i \mid i \in \{1,2,\ldots,N_R\}, j \in \{1,2,\ldots,S_R\}\}; \\ \{0,1,\ldots,N_R\}, & \text{if } e = A. \end{cases} \quad (4)$$

That is, if event $e$ is one of arrival of a vehicle of the type that provides $j$ RUs ($B_j$), departure of a vehicle of the type that provides $j$ RUs ($B_{-j}$), and finishing a service that was handled by $i$ RUs ($D_i$), then no RUs are allocated, and the number –1 is used to record this action. If event $e$ is the arrival of a service request ($A$), then the action is the number of RUs allocated to handle this service request, and the number is bounded by $N_R$. Note that the action of allocating 0 RU to this service request means that the VC does not allocate any RU to handle this request, but transfers it to the RC.

### B. Evaluating transition probabilities

In the SMDP for the VCC system, each state is not independent with the states prior and posterior to this state. And, occurrence of each state must be influenced by the prior state and action. Hence, the transition probability is defined as the probability of transiting from the current state $s$ to the next state $s'$ after taking action $a$. Let $\sigma(s, a)$ be the number of events of taking action $a$ under state $s$; and $\tau(s, a)$ be the service time from the current state $s$ to the next state in case of taking action $a$ under state $s$. It is obvious that the two numbers are inverses of each other, i.e., $\sigma(s, a)^{-1} = \tau(s, a)$. Different form previous works, heterogeneity of vehicles leads to much difference on the number of vehicle arrivals at the VCC system. Based on vehicular heterogeneity, $\sigma(s, a)$ can be calculated as follows:

$$\sigma(s,a) = \tau(s,a)^{-1} = \begin{cases} (\sum_{j=1}^{S_R} V_j + 1)\cdot\lambda_P + \sum_{j=1}^{S_R}\lambda_j + \sum_{j=1}^{S_R}\mu_j + \sum_{j=1}^{N_R} j\cdot n_j\cdot\mu_p, & \text{if } e = B_j, a = -1; \\ (\sum_{j=1}^{S_R} V_j - 1)\cdot\lambda_P + \sum_{j=1}^{S_R}\lambda_j + \sum_{j=1}^{S_R}\mu_j + \sum_{j=1}^{N_R} j\cdot n_j\cdot\mu_p, & \text{if } e = B_{-j}, a = -1; \\ \sum_{j=1}^{S_R} V_j\cdot\lambda_P + \sum_{j=1}^{S_R}\lambda_j + \sum_{j=1}^{S_R}\mu_j + \sum_{j=1}^{N_R} j\cdot n_j\cdot\mu_p + i\cdot\mu_p, & \text{if } e = A, a = i \in \{0,1,\ldots,N_R\}; \\ \sum_{j=1}^{S_R} V_j\cdot\lambda_P + \sum_{j=1}^{S_R}\lambda_j + \sum_{j=1}^{S_R}\mu_j + \sum_{j=1}^{N_R} j\cdot n_j\cdot\mu_p - i\cdot\mu_p, & \text{if } e = D_i, a = -1. \end{cases} \quad (5)$$

where $e$ is the event occurring under state $s$.

Based on the classical probability theory, $\sigma(s, a)$ is the denominator of the transition probability. Next, the transition probability $P(s' \mid s, a = -1)$ of transiting from state $s$ to state $s'$ in case of taking action $a$ is calculated as follows:

1) For state $s = (n_1, n_2, \ldots\ldots, n_{N_R}, V_1, V_2, \ldots\ldots, V_{S_R}, B_j)$,

$$P(s' \mid s, a=-1) = \begin{cases} \dfrac{(\sum_{j=1}^{S_R} V_j + 1)\lambda_p}{\sigma(s,a)}, & \text{if } s' = (n_1, n_2, \ldots, n_{N_R}, V_1, V_2, \ldots, V_{S_R}, A); \\ \dfrac{n_i i \mu_p}{\sigma(s,a)}, & \text{if } s' = (n_1, n_2, \ldots, n_{N_R}, V_1, V_2, \ldots, V_{S_R}, D_i); \\ \dfrac{\lambda_j}{\sigma(s,a)}, & \text{if } s' = (n_1, n_2, \ldots, n_{N_R}, V_1, V_2, \ldots, V_{S_R}, B_j); \\ \dfrac{\mu_j}{\sigma(s,a)}, & \text{if } s' = (n_1, n_2, \ldots, n_{N_R}, V_1, V_2, \ldots, V_{S_R}, B_{-j}); \\ \dfrac{\lambda_k}{\sigma(s,a)}, & \text{if } k \neq j, s' = (n_1, n_2, \ldots, n_{N_R}, V_1, V_2, \ldots, V_{S_R}, B_k); \\ \dfrac{\mu_k}{\sigma(s,a)}, & \text{if } k \neq j, s' = (n_1, n_2, \ldots, n_{N_R}, V_1, V_2, \ldots, V_{S_R}, B_{-k}). \end{cases} \quad (6)$$

2) For state $s = (n_1, n_2, \ldots\ldots, n_{N_R}, V_1, V_2, \ldots\ldots, V_{S_R}, B_{-j})$,

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

$$P(s'|s,a=-1)=\begin{cases} \dfrac{(\sum_{j=1}^{S_R} V_j - 1)\lambda_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},A); \\ \dfrac{n_i i\mu_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},D_i); \\ \dfrac{\lambda_j}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_j); \\ \dfrac{\mu_j}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_{-j}); \\ \dfrac{\lambda_k}{\sigma(s,a)}, & \text{if } k\neq j, s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_k); \\ \dfrac{\mu_k}{\sigma(s,a)}, & \text{if } k\neq j, s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_{-k}). \end{cases} \quad (7)$$

3) For state $s=(n_1,n_2,......,n_{N_R},V_1,V_2,......,V_{S_R},A)$,

$$P(s'|s,a=0)=\begin{cases} \dfrac{\sum_{j=1}^{S_R} V_j\lambda_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},A); \\ \dfrac{n_i i\mu_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},D_i); \\ \dfrac{\lambda_j}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_j); \\ \dfrac{\mu_j}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_{-j}). \end{cases} \quad (8)$$

$$P(s'|s,a=i)=\begin{cases} \dfrac{\sum_{j=1}^{S_R} V_j\lambda_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},A); \\ \dfrac{i(n_i+1)\mu_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},D_i); \\ \dfrac{mn_m\mu_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},D_m); \\ \dfrac{\lambda_j}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_j); \\ \dfrac{\mu_j}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_{-j}). \end{cases} \quad (9)$$

4) For state $s=(n_1,n_2,......,n_{N_R},V_1,V_2,......,V_{S_R},D_i)$,

$$P(s'|s,a=-1)=\begin{cases} \dfrac{\sum_{j=1}^{S_R} V_j\lambda_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},A); \\ \dfrac{i(n_i-1)\mu_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},D_i); \\ \dfrac{mn_m\mu_p}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},D_m); \\ \dfrac{\lambda_j}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_j); \\ \dfrac{\mu_j}{\sigma(s,a)}, & \text{if } s'=(n_1,n_2,...,n_{N_R},V_1,V_2,...,V_{S_R},B_{-j}). \end{cases} \quad (10)$$

### C. Evaluating the reward

A feature of the SMDP is to consider a finite number of states but infinite time. A reward for running the VCC system in a long run is defined as the expected cost of consuming power and time. Following the setting of [16], when an action $a$ is taken so that the VCC system transits from a state $s$ to another state, a reward $r(s, a)$ can be obtained as follows:

$$r(s,a)=k(s,a)-g(s,a) \quad (11)$$

where $k(s, a)$ and $g(s, a)$ are the revenue and expected cost, respectively, and are explained in detail in the following.

After an action $a$ is taken under a state $s$, revenue $k(s, a)$ is the difference of the maximal cost of consuming power and time from the real cost in the VCC system [16]. However, because of vehicular heterogeneity concerned in this work, six cases for $k(s, a)$ are caused as follows:

$$k(s,a)=\begin{cases} \begin{bmatrix} \omega_e\cdot\beta_e\cdot(E_l-P_l\cdot\delta_1) \\ +\omega_d\cdot\beta_d\cdot(D_l-\dfrac{1}{i\cdot\mu_p}-\delta_1)-\gamma\cdot\delta_1 \end{bmatrix}, & \text{if } e=A, a=i\in\{1,2,...,N_R\}; \\ I-\gamma\cdot(\delta_1+\delta_2), & \text{if } e=A, a=0; \\ 0, & \text{if } e=D_i, a=-1; \\ 0, & \text{if } e=B_j, a=-1; \\ 0, & \text{if } e=B_{-j}, \sum_{i=1}^{N_R} i\cdot n_i<(M-j+1), \; a=-1; \\ -\xi, & \text{if } e=B_{-j}, \sum_{i=1}^{N_R} i\cdot n_i\geq(M-j+1), \; a=-1. \end{cases} \quad (12)$$

where $\omega_e$ and $\omega_d$ are the weights of power and time, respectively, consumed when handling the service request; $\beta_e$ and $\beta_d$ are the ratios of transforming one unit of power and time into the revenue, respectively; $E_l$ and $D_l$ are the maximal power and time that are consumed when handling the service request; $P_l$ is the power consumed for one unit of time when handling the service request in reality; $\delta_1$ and $\delta_2$ are the time consumed in transmitting and computing between VE and VC, and between VC and RC; $I=\omega_e\cdot\beta_e\cdot(E_l-P_l\cdot\delta_1)+\omega_d\cdot\beta_d\cdot(D_l-\delta_1-\delta_2)$.

In what follows, the expected cost $g(s, a)$ is explained in detail. During the continuous time when the VCC transits from a state to the next state, $g(s, a)$ represents the cost caused by handling the service by RUs, and is expressed as follows:

$$g(s,a)=c(s,a)\cdot\tau(s,a) \quad (13)$$

where

$$c(s,a)=\sum_{i=1}^{N_R} i\cdot n_i \quad (14)$$

represents the number of the RUs that have been allocated and are in service in case of taking action $a$ under state $s$; and $\tau(s, a)$ represents the expected service time from the current state $s$ to the next state $s'$ in case of taking action $a$.

Because arrivals and departures of vehicles and service requests are assumed to follow Poisson distributions, we assume that the time from a state $s$ to the next state $s'$ follows an exponential distribution, with the following cumulative distribution function:

$$F(t|s,a)=1-e^{-\sigma(s,a)t}, \text{ for } t>0 \quad (15)$$

Consider that a discount factor $\alpha$ for continuous time is added to $\sigma(s, a)^{-1} = \tau(s, a)$ for meeting the practical condition. Therefore, the discount reward $r(s, a)$ is rewritten as follows:

$$r(s,a)=k(s,a)-c(s,a)/[\alpha+\sigma(s,a)], \quad (16)$$

### D. Evaluating the value

The objective of the SMDP for resource allocation in the VCC system is to find a strategy that maximizes the long-term expected discount value [29], i.e., $\pi: S\rightarrow A_s$. That is, if such a

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

strategy can be found, we consider that the VCC system is at some state. When some event occurs, the VCC system can apply the action defined in this strategy under the current state to achieve a maximal long-term expected discount value as follows:

$$v_{\alpha}^{\pi^*}(s) = \operatorname*{Max}_{\pi} E_s^{\pi}\left[\sum_{n=1}^{\infty} e^{-\alpha\sigma_n} r(s_n, a_n) \mid s_0 = s\right] \tag{17}$$

where $s_0$ is the initial state of the VCC system; $s_n$ and $a_n$ are the $n$-th state and the applied action, respectively; $\sigma_n$ is the times of event occurrences in average in a unit of time under state $s_n$.

After considering the transition probabilities and the discount rate $\lambda$ for the value, the above equation is rewritten by Bellman equation as follows:

$$v(s) = \operatorname*{Max}_{a \in A_s}\left[r(s,a) + \lambda \cdot \sum_{s' \in S} p(s' \mid s,a) \cdot v(s')\right] \tag{18}$$

Note that in the above equation, the unknown value $v(s)$ in the current strategy $\pi$ on the left side of the equality is computed based on the known values $v(s')$ in the previous strategy $\pi - 1$ on the right side of the equality. However, when computing the value in the initial strategy (i.e., the first strategy), we let the unknown values on the two sides of the equity to be the same. By doing so, we can obtain a simultaneous equation system where values of all states are unknown, and then we solve this system to obtain the initial values.

In addition to the reward caused by taking an action $a$ under the current state $s$ in the VCC system, the value is the cost of power and time saved by the state transition. The discount factor $\lambda$ in Equation (18) has a similar concept of the continuous-time discount factor $\alpha$, but $\lambda$ is used for discounting the future value of the VCC system to the present. Let $\lambda = \sigma(s, a) / (\alpha + \sigma(s, a))$ [16]. For easily observing the experimental results of continuous-time SMDP, each term in Equation (18) is normalized as follows:

$$\tilde{r}(s,a) = r(s,a) \cdot \frac{\alpha + \sigma(s,a)}{\alpha + y}, \tag{19}$$

$$\tilde{\lambda} = \frac{y}{\alpha + y}, \tag{20}$$

$$\tilde{p}(s' \mid s,a) = \begin{cases} 1 - \dfrac{[1 - p(s \mid s,a)]\sigma(s,a)}{y}, & s' = s \\ \dfrac{[1 - p(s' \mid s,a)]\sigma(s,a)}{y}, & s' \neq s \end{cases} \tag{21}$$

where $y = K \cdot \lambda_p + \sum_{j=1}^{S_R} \lambda_j + \sum_{j=1}^{S_R} \mu_j + K \cdot N_R \cdot \mu_p$ . With the above three equations, Equation (19) can be normalized as follows:

$$\tilde{v}(s) = \operatorname*{Max}_{a \in A_s}\left[\tilde{r}(s,a) + \tilde{\lambda} \sum_{s' \in S} \tilde{p}(s' \mid s,a) \tilde{v}(s')\right] \tag{22}$$

### E. Stop condition

The SMDP iteratively finds the best action of each state until the values in the current and the previous strategies do not have remarkable difference (by Banach fixed-point theorem). The stop condition is given as follows:

$$\| \tilde{v}^{\pi+1} - \tilde{v}^{\pi} \| < \varepsilon(1 - \tilde{\lambda}) / 2\tilde{\lambda}$$

### F. Difference of the proposed model from the previous model

Differences of the proposed model from the previous model [16] are given as follows:

- This work introduces RSUs, so that the number of RUs in the VCC system ($M$) must be no less than the number of RSUs ($R$). Hence, Equation (12) is different from the previous work.
- Heterogeneous vehicles with different numbers of RUs causes changes in the equations of rewards and values in Equations (16) and (18).
- Arrivals and departures of vehicles of the type that provides $j$ RUs follow Poisson distributions with means $\lambda_j$ and $\mu_j$, respectively, so that Equations (5) – (10) are different from the previous work.

## V. Experimental Implementation and Results

This section first gives the experimental design, and then gives the experimental results under various parameter settings.

### A. Experimental design

In addition to extending the parameter settings in [16], this work further considers the parameter settings for heterogeneous vehicles. The experiment considers two vehicle types (i.e., $S_R = 2$), which provide 1 and 2 RUs, respectively. The experimental parameters used in this work are given as follows: $K = 3 \sim 13$, $S_R = 2$, $N_R = 3$, $R = 1 \sim 5$, $\lambda_p = 1 \sim 9$, $\mu_p = 8$, $\lambda_1 = 4 \sim 8$, $\lambda_2 = 3 \sim 7$, $\mu_1 = 4 \sim 8$, $\mu_2 = 3 \sim 7$, $\omega_e = 0.5$, $\omega_d = 0.5$, $\beta_e = 2$, $\beta_d = 2$, $E_l = 20$, $D_l = 20$, $P_l = 4$, $\gamma = 2$, $\delta_1 = 2$, $\delta_2 = 5$, $\alpha = 0.1$, $\xi = 18$.

The plots of the average value of running 20 times of the SMDP versus number of RSUs (i.e., $R$) under different arrivals of service requests per vehicle ($\lambda_p$) are given in Fig. 3. The other parameters in this experiment are set as follows: $K = 3$, $\lambda_1 = 4$, $\lambda_2 = 3$, $\mu_1 = 4$, and $\mu_2 = 3$. From Fig. 3, when number of RSUs increases, the average value increases exponentially, i.e., the cost of consuming power and time in the VCC system are reduced. In addition, when $\lambda_p$ is smaller, the average value is larger, which is reasonable because less service requests consumes less power and time.

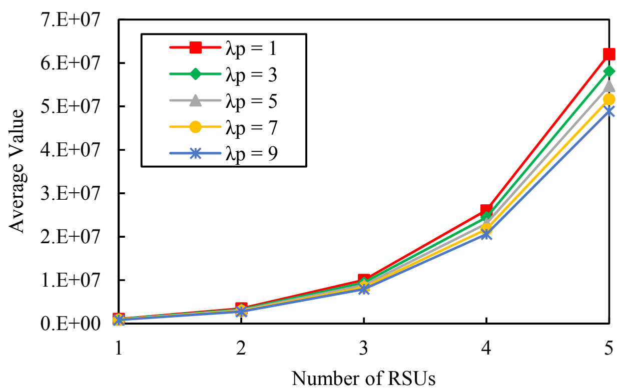


Fig. 3. Plot of average value of running 20 times of the SMDP versus number of RSUs under different arrivals of service requests per vehicle.

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2690961, IEEE Internet of Things Journal



The plots of the average value of running 20 times of the SMDP versus number of RSUs under various combinations of arrivals and departures of vehicles of the type that provides one RU (i.e., $\lambda_1$ and $\mu_1$, respectively) are given in Fig. 4. Other parameters are set as follows: $K = 3$, $\lambda_p = 1$, $\lambda_2 = 3$, $\mu_p = 8$, $\mu_2 = 3$. From Fig. 4, we have the same conclusion that when number of RSUs increases, the average value increases exponentially. When $\lambda_1$ and $\mu_1$ are smaller, the average value is larger. It is reasonable because smaller $\lambda_1$ and $\mu_1$ imply fewer vehicles in the VCC system and consume less cost.

The results under different combinations of ($\lambda_1$, $\mu_1$, $\lambda_2$, $\mu_2$) are shown in Fig. 5. Other parameters are set as follows: $K = 3$, $R = 1$, $\lambda_p = 1$, and $\mu_p = 8$. From Fig. 5, when the arrivals and departures of the vehicle type that provides 2 RUs (i.e., $\lambda_2$ and $\mu_2$) are large, the average value decreases, because more vehicles in the VCC system could lead to more service requests. We can obtain the same conclusion for $\lambda_1$ and $\mu_1$ by comparing the five curves in Fig. 5.

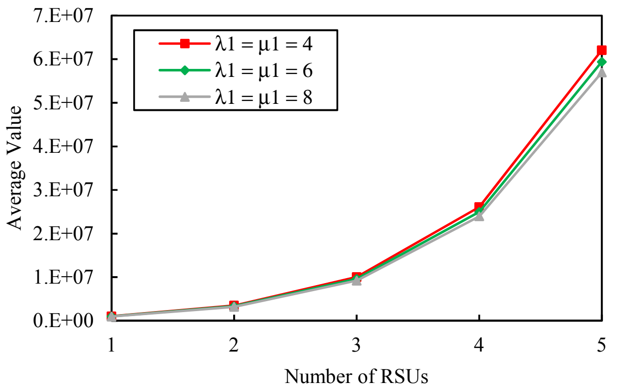


Fig. 4. Plots of the average value of running 20 times of the SMDP versus number of RSUs under various combinations of arrivals and departures of vehicles of the type that provides one RU.

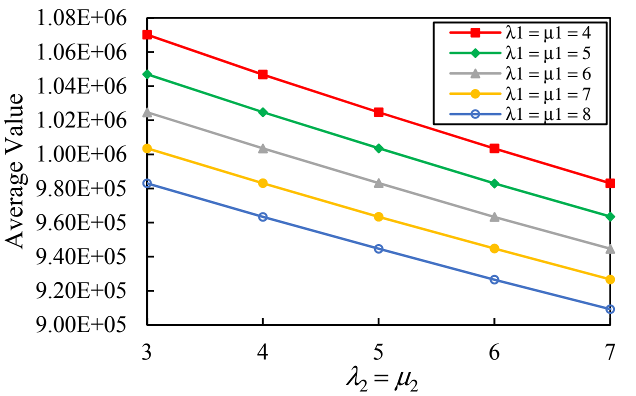


Fig. 4. Plots of the average value of running 20 times of the SMDP versus the value of $\lambda_2 = \mu_2$ under various combinations of $\lambda_1$ and $\mu_1$.

## VI. Conclusion

This work has proposed an SMDP model for resource allocation in the VCC system that considers heterogeneous vehicles and integrates V2V and V2I, to meet the practice. Introducing heterogeneity of vehicles and RSUs makes the model becomes increasingly complex, in which much more system states of the VCC system need to be considered, and the transition among states requires more computation and restrictions. Simulation shows that the SMDP for the system provides a promising approach for allocating resources in the system. A line of the future work is to further consider heterogeneity of service requests, and their influence to the resource allocation. In addition, it would be of interest to investigate fairness and priority of resource allocation to heterogeneous vehicles and service requests in the system.

## References

[1] Q. Ding, B. Sun, and X Zhang, “A Traffic-light-aware Routing Protocol based on street connectivity for urban vehicular ad hoc Networks,” *IEEE Communications Letters*, vol. 20, no. 8, pp. 1635-1638, 2016.

[2] M. Rebai, L. Khoukhi, H. Snoussi, and F. Hnaien, “Optimal placement in hybrid VANETs-sensors networks,” in *Proceedings of Wireless Advanced (WiAd)*, IEEE Press, pp. 54-57, 2012.

[3] C.-C. Lin and D.-J. Deng, “Optimal two-lane placement for hybrid VANET-sensor networks,” *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7883-7891, 2015.

[4] M. Gerla, “Vehicular cloud computing,” in *Proceedings of 11th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, IEEE Press, pp. 152-155, 2012.

[5] S. Bitam, A. Mellouk, and S. Zeadally, “VANET-cloud: A generic cloud computing model for vehicular ad hoc networks,” *IEEE Wireless Communications*, vol. 22, no. 1, pp. 96-102, 2015.

[6] L. Gu, D. Zeng, and S. Guo, “Vehicular cloud computing: A survey,” in *Proceedings of IEEE Globecom Workshops*, pp. 403-407, 2013.

[7] R. Hussain, J. Son, H. Eun, S. Kim, and H. Oh, “Rethinking vehicular communications: Merging VANET with cloud computing,” in *Proceedings of 4th International Conference on Cloud Computing Technology and Science (CloudCom)*, IEEE Press, pp. 606-609, 2012.

[8] E. Lee, E. K. Lee, M. Gerla, and S. Y. Oh, “Vehicular cloud networking: architecture and design principles,” *IEEE Communications Magazine*, vol. 52, no. 2, pp. 148-155, 2014.

[9] S. C. Ng, W. Zhang, Y. Zhang, Y. Yang, and G. Mao, “Analysis of access and connectivity probabilities in vehicular relay networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 1, pp. 140-150, 2011.

[10] M. Báguena, C. T. Calafate, J. C. Cano, and P. Manzoni, “An adaptive anycasting solution for crowd sensing in vehicular environments," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7911-7919, 2015.

[11] R. N. R. Woo and D. S. Han, “Performance of vehicular-to-infrastructure communication systems based on IEEE 802.11a,” in *Proceedings of International Conference on Consumer Electronics (ICCE)*, IEEE Press, pp. 454-454, 2010.

[12] H. Wu, R. Fujimoto, and G. Riley, “Analytical models for information propagation in vehicle-to-vehicle networks,” in *Proceedings of 2004 IEEE 60th Vehicular Technology Conference (VTC2004-Fall)*, vol. 6, pp. 4548-4552, 2004.

[13] F. Dressler, H. Hartenstein, O. Altintas, and O. K. Tonguz, “Inter-vehicle communication: Quo vadis,” *IEEE Communications Magazine*, vol. 52, no. 6, pp. 170-177, 2014.

[14] J. A. Dias, J. J. Rodrigues, F. Xia, and C. X. Mavromoustakis, “A cooperative watchdog system to detect misbehavior nodes in vehicular delay-tolerant networks,” *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7929-7937, 2015.

[15] R. Yu, Y. Zhang, S. Gjessing, W. Xia, and K. Yang, “Toward cloud-based vehicular networks with efficient resource management,” *IEEE Network*, 27, no. 5, pp. 48-55, 2013.

[16] K. Zheng, H. Meng, P. Chatzimisios, L. Lei, and X. Shen, “An SMDP-based resource allocation in vehicular cloud computing systems,” *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7920-7928, 2015.

[17] C.-C. Lin, P.-C. Chen, and L.-W. Chang, “On different-dimensional deployment problems of hybrid VANET-sensor networks with QoS considerations," *Mobile Networks and Applications*, in press.

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2690961, IEEE Internet of Things Journal



[18] Y. Cui, Z. Lai, and N. Dai, "A first look at mobile cloud storage services: Architecture, experimentation and challenge," *IEEE Network*, vol. 30, no. 4, pp. 16-21, 2016.

[19] Y. Liu, M. Lee, and Y. Zheng, "Adaptive multi-resource allocation for cloudlet-based mobile cloud computing system," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2398-2410, 2015.

[20] D. Kwak, R. Liu, D. Kim, B. Nath, and L. Iftode, "Seeing is believing: sharing real-time visual traffic information via vehicular clouds," *IEEE Access*, vol. 4, pp. 3617-3631, 2016.

[21] G. Yan, D. Wen, S. Olariu, and M. C. Weigle, "Security challenges in vehicular cloud computing, " *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 284-294, 2013.

[22] C. Wang, Y. Li, D. Jin, and S. Chen, "On the serviceability of mobile vehicular cloudlets in a large-scale urban environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2960-2970, 2016.

[23] P. Ghazizadeh, *Resource Allocation in Vehicular Cloud Computing*. Ph.D. Dissertation, Old Dominion University, Norfolk, VA, 2014.

[24] H. He, H. Shan, A. Huang, and L. Sun, "Resource allocation for video streaming in heterogeneous cognitive vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 7917-7930, 2016.

[25] X. Xiong, L. Hou, K. Zheng, W. Xiang, M. Hossain, and S. M. M. Rahman, "SMDP-based radio resource allocation scheme in software-defined internet of things networks," *IEEE Sensors Journal*, vol. 16, no. 20, pp. 7304-7314, 2016.

[26] R. Srinivasan and A. K. Parlikad, "Semi-Markov decision process with partial information for maintenance decisions," *IEEE Transactions on Reliability*, vol. 63, pp. 891-898, 2014.

[27] M. El Helou, M. Ibrahim, S. Lahoud, K. Khawam, D. Mezher, and B. Cousin, "A network-assisted approach for RAT selection in heterogeneous cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, pp. 1055-1067, 2015.

[28] H. Meng, K. Zheng, P. Chatzimisios, H. Zhao, and L. Ma, "A utility-based resource allocation scheme in cloud-assisted vehicular network architecture," in *Proceedings of IEEE International Conference on Communication Workshop (ICCW)*, pp. 1833-1838, 2015.

[29] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons Press, 2014.

2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2690961, IEEE Internet of Things Journal

## Author's comments with respect to Reviewer #1:

This paper has been revised closely following the suggestions of Reviewer #1. What follows are specific comments with respect to the points raised by Reviewer #1.

- ('- It is quite hard to follow formulas in the paper.')

  More explanations are added as suggested. See page 5, column 1, paragraph 6 of the revised paper.

- ('- The background of SMDP should be briefly explained before going to the proposed SMDP.')

  Added as suggested. See page 2, column 1, the two paragraphs above Section II of the revised paper.

- ('- Related works are not compared sufficiently therefore the novelty of the paper is not really convinced.')

  More related works are reviewed as suggested. See page 2, column 2, paragraphs 1 – 3 of the revised paper.



2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2690961, IEEE Internet of Things Journal

## Author's comments with respect to Reviewer #2:

Reviewer #2 did not request any further revision in the comments. The authors thank a lot for Reviewer #2's recommendation.



2327-4662 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.