

Using a Protocol for the Administration, Discovery, and Management of Cloud Services in a Virtualized Computing Environment

B Mahesh Reddy¹, O V Sowmya², G B Veeresh³, K Srujana Kumari⁴

^{1,2,3,4} Asst. Professor, Department of CSE, K.S.R.M College of Engineering(A), Kadapa

Abstract

With the development of car ad hoc networks, improvements in driver comfort, security, and productivity are feasible. However, the linked vehicles' existing communication, storage, and processing capabilities are not being leveraged to their full potential to meet the service requirements of ITS. The efficiency of the cloud may be brought to vehicle networks, which is why vehicle Cloud Computing (VCC) has promise. In this research, we show how to allocate resources in a way that maximizes the VCC system's potential profit. Consideration of resource variability, in addition to the VCC system's revenues and expenses, may be used to determine the motivation for the system. An endless horizon is a useful metaphor for the optimization issue. Once the VCC system's state space, action space, reward model, and transition probability distribution have been established, a Semi-Markov Decision Process (SMDP) may be carried out. To find the best plan of action, an iterative process is performed. Describes the actions to do after a certain condition has been satisfied. The numerical findings show that the SMDP-based technique has a good chance of enhancing performance while maintaining a manageable level of complexity.

INTRODUCTION

In recent years, researchers and businesses alike have shown a great deal of interest in vehicle networks. Vehicles are equipped with a wide range of smart sensors and gadgets for data collection and processing [1] [2]. Meanwhile, the inter-vehicle connection may be provided by the use of different wireless communications Technologies. automotive-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication paradigms are the two most common kinds used in the automotive services industry. The manuscript was received on December 6, 2014, and after two rounds of revisions (March 11 and May 18), it was approved on June 13, 2015. 2015 IEEE. All rights reserved. You may use this information for non-commercial purposes. If you want to use this content for anything other than personal study, you must first contact the IEEE at pubs-permissions@ieee.org for approval.

The National Science Foundation of China (No.61331009), the National High-Tech R&D Program (863 Program 2015AA01A705), the National Key Technology R&D Program of China (No.2015ZX03002009-004), and the Fundamental Research Funds for the Central Universities (No.2014ZD03-02) have all contributed to this study.

Beijing University of Posts & Telecommunications, Beijing, China 100088, is home to Kan Zhen (email:kzheng@ieee.org) and Hamlin Men. Both men work at the Key Lab of Universal Wireless Communications at the Ministry of Education. Pericles Chatzimisios may be reached at P.O. Box 141, 57400 Sindos, Thessaloniki, Greece; he works in the Department of Informatics at Alexander TEI of Thessaloniki (ATEITHE). Lei Lei works at Beijing Jiao tong University in Beijing, China, 100044 at the State Key Laboratory of Rail Traffic Control & Safety. Xiamen (Sherman) Sheen is a professor at the University of Waterloo in Waterloo, Ontario, Canada, in the Department of Electrical and Computer Engineering. N2L 3G1. Message passing [3]. Through Dedicated Short Range Communication (DSRC) or cellular networks, for example, V2I communication allows cars to connect to the Internet through a roadside base station. [4] [5]. Vehicular networks, which collect and analyze information about vehicles, may significantly enhance transportation safety, reduce traffic congestion, and enhance the driving experience [7]. [8] [9]. Because of the rapid development of computers and communication

Developed [6], the argument is made for high-tech cars to be seen as service providers rather than passive recipients of such services. Therefore, the idea of Vehicular Cloud Computing (VCC) has been presented, which uses the computing, communication, and storage capabilities of Vehicle Equipments (VEs) [10] such as on-board computers and mobile phones. Services in the VCC system may be broken down into four categories based on the resources they make use of: "Network-as-a-Service (Naas)", "Storage-as-a-Service (Stash)", "Sensing-as-a-Service (Seas)", and "Computation-as-a-Service (Camas)". [10]. Only Camas is of importance and is further investigated in this work because the computer power of cars is fast being enhanced to allow them to function as suppliers of computing services. To ensure that the VEs get enough support, this article proposes deploying a layered-cloud computing architecture for the VCC system. Vehicular Clouds (VCs) may be thought of as one of the computing capacity suppliers in addition to the Remote Cloud (RC) in the proposed architecture. Since automobiles are manufactured by several companies,

have varying capacities for computing. The virtualization approach must be created to abstract and slice the heterogeneous physical resources into virtual resources, which are shared by numerous VEs in the VCC system, in order to address this problem. Each car in a virtual fleet is expected to have its own virtualized RUs in this article.

RELATED WORK

A few of works on the VCC have been carried out to enhance the services capabilities of VEs. VCC is very similar to a Mobile Cloud Computing (MCC) system but it brings in new characteristics. In [11], the VCC system is divided

into three architectural frameworks, namely Vehicular Clouds (VCs), Vehicles using Clouds (Vices), and Hybrid Clouds (HCs), respectively. Moreover, it has been pointed out that in order to form the VCs can effectively deal with services locally produced and improve the experience of VEs [12]. In [13], the Parked Vehicle Assistance (PVA) is proposed to overcome sparse/unbalanced traffic and greatly promote network connectivity by considering the parked vehicles as static cloud nodes. Also, the parked cars are utilized to sense vehicles that are not in line-of-sight in order to improve safety [14]. A two-tier data center architecture that leverages the excessive storage resources in parking lots has been studied in [15]. Furthermore, the main focus of works in [17] and [18] is the security of the VCC system. There are also certain works in the literatures on the resource allocation problem to improve the computing capability of the VEs in the VCC system. A game-theoretical approach is presented for effective resource management in roadside cloud set to provide services to several vehicles [16]. Similarly, a distributed and adaptive resource management is proposed for optimal exploitation of Cognitive Radio and soft-input/soft output data fusion in Vehicular Access Networks [19] [20], in which the energy and computing limited car smart phones are enhanced by offloading their traffic to the local or remote cloud. However, both of them have not considered that vehicles can share the resources between each other. Consequently, a scheduling model is presented, in which the unpredictable available computation resources in the VCC system are also considered [21]. The current paper attempts to deal with the limitations of the previous works and proposes a resource allocation scheme to better serve the VEs (especially MUEs) in the VCC system that is consisted of RC and VCs. Although computation resource allocation in a mobile cloud computing system was studied in [22], this scheme cannot be applied in the VCC system due to variability feature of the available resources in VCs. Moreover, different from the model in [21], the requests in this paper can be allocated with more than one RUs and processed in parallel. Furthermore, although node mobility is considered in traditional mobile cloud computing to achieve effective job scheduling, the total long-term expected reward of system still cannot be obtained in a satisfactory manner as Shown in [23] [24].

VEHICULAR CLOUD COMPUTING SYSTEM

A. System model

Fig. 1 shows a typical VCC system, in which vehicles in movement constitute a dynamic VC. In particular, VEs that act like smart phones can enjoy vast computing power by submitting the service requests to the VCC system in order to save the energy and enhance the processing speed. A vehicle is assumed to have one basic computation RU in the VCC system. When a service request arrives at the system, it has to make the decision of whether accepting it by the VC or transferring it to the RC. If the request is assigned to the VC, the decision of allocating how many RUs to it has to be made based on current available resources. Otherwise, a transfer decision is made instead, and then the service request may be submitted to the RC. For the sake of illustration, an example is also given in Fig. 1. Requests by VE A and VE B are accepted from the VC while the request by VE C is obliged to be transferred to the RC. After VE A and VE B are admitted, 3 RUs and 2 RUs are allocated to them, respectively. All the decisions are made to achieve the specified objective in the VCC system. The list of important notations of this paper is given in Table I.

Assume that there are M available RUs in the VC, which varies with the arrival and departure of vehicles. K is the maximal number of the vehicles that the VC can support, i.e., the number of RUs in the VC cannot exceed K . Each arrival service request can be allocated with I RUs, where

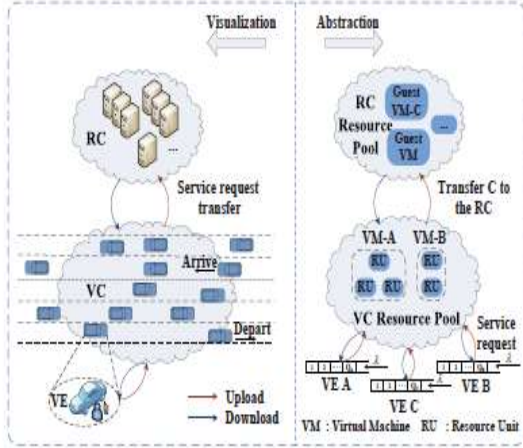


Fig. 1. Illustration of a typical VCC system.

$I \in \{1; 2; NR\}$; $NR \leq M$. The arrival rates of service requests and vehicles follow Poisson distribution with λ_p and λ_v , respectively. Let μ_p denote the computing service rate of the request in case of only one RU allocated. Then, the service time of a request is $1/\mu_p$ in case that I RUs are allocated. In addition, the departure rate of vehicles is denoted as μ_v . Considering the dynamic characters of the service requests and vehicle arrivals, the action of the current epoch may directly lead to considerable change of next state so as to have serious impacts on the system expected total reward. In other words, the action to maximize the reward of the current epoch may become unwise in the long run especially when the resources in the VC are relatively scarce. Therefore, our objective in this paper is to maximize the long-term expected total reward by properly allocating the resources in the VCC system.

B. System States

The system state s reflects the current requests with different number of RUs, the available resources in the VC and the event of requests and vehicles. Therefore, the state set can be denoted by S , i.e.,

$$S = \{s | s = (n_1, n_2, \dots, n_{NR}, M, e)\}. \quad (1)$$

Where n_i is the number of service requests that have been allocated with i RUs, and e represents an event in the set $E \in E = \{A; D1; D2; DNR; B1; B-1\}$. Here A denotes the arrival of the service request, D_i means the departure of a request assigned with $M - i$ RUs, $B1$ and $B-1$ describe the arrival and departure of a vehicle, respectively. Thus, the Number of occupied RUs in the VC is $\sum_{i=1}^{NR} i \cdot n_i$, which satisfies $\sum_{i=1}^{NR} i \cdot n_i \leq M$. Moreover, the number of system states can be denoted by N .

C. Actions

In this model, several possibilities of action a can be taken in the action set A , i.e.

$$\mathcal{A} = \{-1, 0, 1, 2, \dots, N_R\}. \quad (2)$$

TABLE I
LIST OF IMPORTANT NOTATIONS.

K	Maximal number of vehicles that the VC can support
M	Number of available RUs in the VC
N_R	Maximal number of RUs that the VC can allocate to a service request
N	Number of system states
λ_p	Arrival rate of new service requests
μ_p	Service rate of the requests
λ_v	Arrival rate of new vehicles
μ_v	Departure rate of vehicles
n_i	Number of served services allocated with i RUs
D_i	Departure of a service request allocated with i RUs
A	Arrival of a service request
B_1/B_{-1}	Arrival / departure of a vehicle
w_e	Energy income weight
w_d	Delay income weight
β_e	Price of per energy saving
β_d	Price of per delay saving
γ	Cost per transmit time
E_l	Energy consumed by executing the request at VE
D_l	Time consumed by executing the request at VE
P	Transmission power of VEs
δ_1	Transmission delay between the VE to the VC
δ_2	Transmission delay between the VC to the RC
I	Income obtained by the VCC system
ξ	Compensation to VE by the VCC system
α	Continuous-time discount factor

When an event occurs, the VCC system decides which action $a(s)$ needs to be taken from the action set \mathcal{A}_s based on the current state s , i.e.,

$$\mathcal{A}_s = \begin{cases} \{-1\}, & e \in \{D_1, D_2, \dots, D_{N_R}, B_1, B_{-1}\} \\ \{0, 1, 2, \dots, N_R\}, & e = A \end{cases}, \quad (3)$$

Completes and departs from the VCC system or a vehicle arrives at and leaves the VCC system, and no action is required except the information of the available RUs in the VCC system has to be updated. When receiving a request, one of two actions may be chosen either to accept with I RUs from the VC, i.e., either $a(s) = I$, or to transfer it to the RC, $a(s) = 0$.

D. Rewards

Given an action a , the system reward under the current state s is denoted by

$$r(s, a) = k(s, a) - g(s, a), \quad (4)$$

Where $k(s; a)$ is the instant revenue of the VCC system by taking action a under state s in case that event e occurs, which consists of both the income and cost of the VCC system. Since the main benefits of the VCC system are to save the power consumption and speed up the processing rate of VEs [26], the income has to include the effects of both of them [27].

Meanwhile, the cost of the system is the transfer expense to send and receive the request. $G(s; a)$ is the expected system cost before the next decision epoch. Furthermore, $k(s; a)$ of the VCC system can be described by

$$k(s, a) = \begin{cases} [w_e \beta_e (E_l - P \cdot \delta_1) + w_d \beta_d (D_l - 1/i \mu_p - \delta_1)] - \gamma \delta_1, & a = i, e = A \\ I - \gamma(\delta_2 + \delta_1), & a = 0, e = A \\ 0, & a = -1, e \in \{D_1, D_2, \dots, D_{N_R}, B_1\} \\ 0, & a = -1, e \in \{B_{-1}\}, \sum_{i=1}^{N_R} i \cdot n_i \neq M \\ -\xi, & a = -1, e \in \{B_{-1}\}, \sum_{i=1}^{N_R} i \cdot n_i = M \end{cases} \quad (5)$$

The details of the revenue function are explained as follows:

1) When a service request is admitted to the VC, the instant revenue $[w_e (E_l - P \cdot \delta_1) + w_d (D_l - 1/i \mu_p - \delta_1)]$ can be earned by the system. $(E_l - P \cdot \delta_1)$ and $(D_l - 1/i \mu_p - \delta_1)$ are the saved energy and time when processing the computing task in the VC, respectively. δ_e and δ_d are the price of per unit energy and time. Different weights, i.e., w_e and w_d , can be predefined according to different purposes, where $w_e + w_d = 1$. The transfer expense is denoted as δ_1 , which is the cost of the VCC system to receive the computing task from the VEs and send back the results. More specially, since the request has already been accepted by the VC, the VE can enjoy the service by transmitting the task to the VC and receiving the feedback from it, which consumes $P \cdot \delta_1$ energy and δ_1 time at this stage. For the purpose of analysis, the transmitted power and received power are assumed to be identical [28]. If the request is allocated with i RUs by the VCC system, the service time spent for finishing the task is $1/i \mu_p$.

No.	Actions	State Transition
1	$a = 0$	$s_{next1} = \{1, 1, 1, M, A\},$ $s_{next2} = \{1, 1, 1, M, D_i\}, i \in \{1, 2, 3\}$ $s_{next3} = \{1, 1, 1, M, B_1/B_{-1}\},$
2	$a = 1$	$s_{next1} = \{2, 1, 1, M, D_i\}, i \in \{1, 2, 3\}$ $s_{next2} = \{2, 1, 1, M, A\}$ $s_{next3} = \{2, 1, 1, M, B_1/B_{-1}\}$
3	$a = 2$	$s_{next1} = \{1, 2, 1, M, D_i\}, i \in \{1, 2, 3\}$ $s_{next2} = \{1, 2, 1, M, A\}$ $s_{next3} = \{1, 2, 1, M, B_1/B_{-1}\}$
4	$a = 3$	$s_{next1} = \{1, 1, 2, M, D_i\}, i \in \{1, 2, 3\}$ $s_{next2} = \{1, 1, 2, M, A\}$ $s_{next3} = \{1, 1, 2, M, B_1/B_{-1}\}$

TABLE II
TRANSFORMATION OF ACTIONS AND CORRESPONDING STATES.

Where $\tau(s; a)$ is the expected service time from the current state to the next state in case that action a is taken under State s , and $c(s; a)$ is the cost rate of $\tau(s; a)$ in case that action a is selected. Moreover, $c(s; a)$ can be characterized By the number of occupied RUs in the VC due to its limited computing capability, i.e.

$$c(s, a) = \sum_{i=1}^{N_R} i \cdot n_i. \quad (7)$$

SMDP-BASED SCHEME FOR VEHICULAR CLOUD COMPUTING

In our analysis, the state transition is determined by the action a under state s . Let us consider the system state $s = (1; 1; 1; M; A)$ as an example and the corresponding state transition under different actions is shown in Table II. Furthermore, the state transition probability under different actions plays an important role on the acquired optimal policy. Thus, in this section, we first derive the state transition probability matrix. Then, the reward function is revised since a discounted model is utilized. Finally, we provide the optimal policy that can be found by utilizing the value iteration algorithm.

A. Transition Probability

Under a given state s and an action a , the expected service time between two continuous decision epoch is denoted by $\tau(s; a)$. Thus, the mean event rate for specific s and a values is the sum of rates of all the events in the VCC system, which can be expressed by

$$\sigma(s, a) = \tau(s, a)^{-1} = \begin{cases} (M+1)\lambda_p + \lambda_v + \mu_v + \sum_{j=1}^{N_R} j n_j \mu_p, & e = B_1, a = -1 \\ (M-1)\lambda_p + \lambda_v + \mu_v + \sum_{j=1}^{N_R} j n_j \mu_p, & e = B_{-1}, a = -1 \\ M\lambda_p + \lambda_v + \mu_v + \sum_{j=1}^{N_R} j n_j \mu_p + i\mu_p, & e = A, a = i \\ & i \in \{0, 1, \dots, N_R\} \\ M\lambda_p + \lambda_v + \mu_v + \sum_{j=1}^{N_R} j n_j \mu_p - i\mu_p, & e = D_i, a = -1 \\ & i \in \{0, 1, \dots, N_R\} \end{cases} \quad (8)$$

Where μ_v is the departure rate of vehicles, and $(M\lambda_p + \lambda_v)$ is the total arrival rate of requests and vehicles. Since λ_p is the arrival rate for requests per vehicle, the arrival rate of requests of the VCC system can be denoted by $M\lambda_p$. The departure rate of requests is explained as follows. When a vehicle joins or leaves the VCC system, the total number of occupied RUs by the existing requests is not changed, which can be denoted by $N \sum_{j=1}^{N_R} j n_j$. Thus, the departure rate of vehicles can be computed as $N \sum_{j=1}^{N_R} j n_j \mu_p$. When a request arrives, the number of occupied RUs can be given by $(N \sum_{j=1}^{N_R} j n_j + i)$ no matter which action taken by the VCC system. Thus, the corresponding departure rate

Is computed as $(\sum_{j=1}^N j n_{j-p} + i_p)$. When a request is served and leaves the system, the number of occupied RUs becomes $(\sum_{j=1}^N j n_{j-i})$. The departure rate of request is $(\sum_{j=1}^N j n_{j-p-i_p})$. Next, $P(s'|s; a)$ is defined as the transition probability from state s to state s' under an action a , which can be calculated under different events, i.e.

- State $s = (n_1, \dots, n_{N_R}, M, A)$,

$$P(s'|s, a) = \begin{cases} \frac{M\lambda_p}{\sigma(s, a)}, & a = 0, s' = (n_1, \dots, n_{N_R}, M, A) \\ \frac{n_i i \mu_p}{\sigma(s, a)}, & a = 0, s' = (n_1, \dots, n_{N_R}, M, D_i) \\ \frac{\lambda_w}{\sigma(s, a)}, & a = 0, s' = (n_1, \dots, n_{N_R}, M, B_1) \\ \frac{\mu_v}{\sigma(s, a)}, & a = 0, s' = (n_1, \dots, n_{N_R}, M, B_{-1}) \\ \frac{(n_i+1) i \mu_p}{\sigma(s, a)}, & a = i, s' = (n_1, \dots, n_i+1, \dots, n_{N_R}, M, D_i) \\ \frac{n_m m \mu_p}{\sigma(s, a)}, & a = i, m \neq i, \\ & s' = (n_1, \dots, n_i+1, \dots, n_{N_R}, M, D_m) \\ \frac{M\lambda_p}{\sigma(s, a)}, & a = i, s' = (n_1, \dots, n_i+1, \dots, n_{N_R}, M, A) \\ \frac{\lambda_w}{\sigma(s, a)}, & a = i, s' = (n_1, \dots, n_i+1, \dots, n_{N_R}, M, B_1) \\ \frac{\mu_v}{\sigma(s, a)}, & a = i, s' = (n_1, \dots, n_i+1, \dots, n_{N_R}, M, B_{-1}) \end{cases} \quad (9)$$

- State $s = (n_1, \dots, n_{N_R}, M, D_i)$,

$$P(s'|s, a) = \begin{cases} \frac{M\lambda_p}{\sigma(s, a)}, & a = -1, s' = (n_1, \dots, n_i-1, \dots, n_{N_R}, M, A) \\ \frac{(n_i-1) i \mu_p}{\sigma(s, a)}, & a = -1, s' = (n_1, \dots, n_i-1, \dots, n_{N_R}, M, D_i) \\ \frac{n_m m \mu_p}{\sigma(s, a)}, & a = -1, m \neq i, \\ & s' = (n_1, \dots, n_i-1, \dots, n_{N_R}, M, D_m) \\ \frac{\lambda_w}{\sigma(s, a)}, & a = -1, s' = (n_1, \dots, n_i-1, \dots, n_{N_R}, M, B_1) \\ \frac{\mu_v}{\sigma(s, a)}, & a = -1, s' = (n_1, \dots, n_i-1, \dots, n_{N_R}, M, B_{-1}) \end{cases} \quad (10)$$

B. Discounted reward model

Assume that the time between two decision epochs is exponentially distributed, i.e.

$$F(t|s, a) = 1 - e^{-\sigma(s, a)t}, \quad \text{for } t > 0. \quad (13)$$

Since the system state does not change between decision epochs, the expected discounted reward is defined based on The discounted reward model found in [29] [30]

$$\begin{aligned}
 r(s, a) &= k(s, a) - c(s, a) E_s^a \left\{ \int_0^\tau e^{-\alpha t} dt \right\} \\
 &= k(s, a) - c(s, a) E_s^a \{ [1 - e^{-\alpha \tau}] / \alpha \} \\
 &= k(s, a) - c(s, a) / [\alpha + \sigma(s, a)],
 \end{aligned} \tag{14}$$

C. Solution

A discounted model is applied to obtain the maximum total long-term expected discounted reward [30]. With a stationary policy $\pi: S \rightarrow A$, the total long-term expected discounted reward can be given by

$$v_\alpha^\pi(s) = E_s^\pi \left[\sum_{n=0}^{\infty} e^{-\alpha \sigma_n} r(s_n, a_n) | s_0 = s \right]. \tag{15}$$

$$v_\alpha^*(s) = v_\alpha^{\pi^*}(s) = \max_{\pi} v_\alpha^\pi(s). \tag{16}$$

$$v(s) = \max_{a \in A_s} [r(s, a) + \lambda \sum_{s' \in S} p(s'|s, a) v(s')]. \tag{17}$$

TABLE III
SYSTEM PARAMETERS IN THE VCC SYSTEM.

Parameter	Value	Parameter	Value
N_R	3	K	3-13
λ_p	1.9	μ_p	8
λ_v	4.8	μ_v	8
w_e	0.5	w_d	0.5
β_e	2	β_d	2
γ	2	E_l	20
D_l	20	P_l	4

$$\tilde{r}(s, a) = r(s, a) \frac{\alpha + \sigma(s, a)}{\alpha + y}, \tag{18}$$

$$\tilde{\lambda} = y / (y + \alpha), \tag{19}$$

$$\tilde{p}(s'|s, a) = \begin{cases} 1 - \frac{[1 - p(s|s, a)] \sigma(s, a)}{y}, & s' = s \\ \frac{p(s'|s, a) \sigma(s, a)}{y}, & s' \neq s \end{cases} \tag{20}$$

Thus, after normalization, (17) can be rewritten as

$$\tilde{v}(s) = \max_{a \in A_s} [\tilde{r}(s, a) + \tilde{\lambda} \sum_{s' \in S} \tilde{p}(s'|s, a) \tilde{v}(s')] \tag{21}$$

Since the proposed model is the infinite SMDP with finite state and action spaces, the value iteration can be used to solve the optimization problem given by (21). A detailed description is provided in Algorithm 1.

Algorithm 1

Step 1: Set $\tilde{v}(s) = 0$ for each state s . Specify $\varepsilon > 0$, and set $k = 0$.

Step 2: For each state s , compute $\tilde{v}^{k+1}(s)$ by

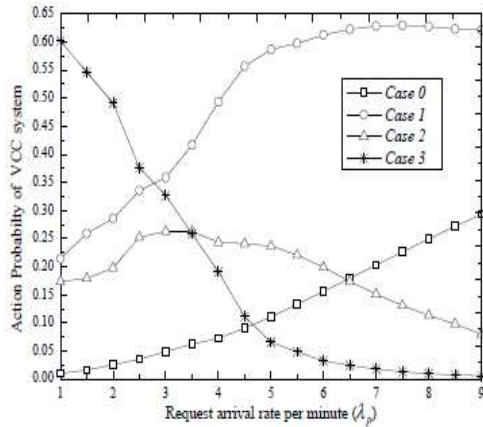
$$\tilde{v}^{k+1}(s) = \max_{a \in A_s} [\tilde{r}(s, a) + \bar{\lambda} \sum_{s' \in S} \tilde{p}(s'|s, a) \tilde{v}^k(s')].$$

Step 3: If $\|\tilde{v}^{k+1} - \tilde{v}^k\| < \varepsilon(1 - \bar{\lambda})/2\bar{\lambda}$, go to **Step 4**. Otherwise, increase k by 1 and go back to **Step 2**.

Step 4: For each $s \in S$, compute the stationary optimal policy and stop.

$$d_\varepsilon^*(s) \in \arg \max_{a \in A_s} [\tilde{r}(s, a) + \bar{\lambda} \sum_{s' \in S} \tilde{p}(s'|s, a) \tilde{v}^{k+1}(s')]$$

In our paper, the norm function is defined as $\|\tilde{v}\| = \max_s \|\tilde{v}(s)\|$ for $s \in S$. Since the operation in Step 2 corresponds to a contraction mapping, the convergence of the value iteration is ensured by Banach Fixed-Point Theorem [29]. Thus, the function $\tilde{v}^k(s)$ converges in norm to $v^*(s)$. Note that the convergence rate of the value iteration algorithm is linear with the rate $\bar{\lambda}$.



CONCLUSION AND FUTURE WORK

In this study, we provide a Semi-Markov decision process (SMDP) based approach to resource allocation in an infinitely scalable Vehicular Cloud Computing environment. An optimal decision-making method is refined via iteration with the goal of increasing the expected total reward of the VCC system. Experiments show that anticipate reward significantly outperforms other allocation strategies when λ_p is big or K is low. The SMDP-based system also has lower complexity than the SA scheme. By investigating the effect of parameter tolerance on the optimal

scheme in the VCC system, we may perhaps develop more reliable and applicable schemes in the future. Considering the ever-increasing scale of a VCC system, this is a daunting challenge....

REFERENCES

According to [1] "Power electronics and motor drives in electric, hybrid electric, and plug-in hybrid electric vehicles," Y. Lee and K. Rajashekara, IEEE Trans. Ind. Electron., vol. 55, no. 6, pp. 2237-2245, Jun.

For instance, see [2] K. Chan, T. Dillon, and E. Chang, "An intelligent particle swarm optimization for short-term traffic flow forecasting using on-road sensor systems," IEEE Trans. Ind. Electron., vol. 60, no. 10, pp. 4714-4725, August 2013.

"An Adaptive Anycasting Solution for Crowd Sensing in Vehicular Environments," IEEE Trans. Ind. Electron., vol. PP, no. 99, pp. 1-1, Jun. 2015; M. Baguena Albaladejo, C. Calafate, J. Cano, and P. Manzoni.

The following is a direct quote from: [4] J. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," Proc. IEEE, vol. 99, no. 7, pp. 1162-1182, Jul.

According to [5] "LTE for vehicular networking: a survey," published in IEEE Commun. Mag., vol. 51, no. 5, pp. 148-157, May. 2013, by G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro.

According to [6] "Industrial-QoS oriented remote wireless communication protocol for the Internet of construction vehicles," published in IEEE Trans. Ind. Electron., volume PP, number 99, pages 1-2 in June 2015, M. Jin, X. Zhou, E. Luo, and X. Qing.

According to [7] "Inter-vehicle communication: Quo vadis," published in IEEE Commun. Mag., volume 52, issue 6, pages 170-177, June 2014.

[8] J.A.F.F. Dias, J.J.P.C. Rodrigues, C. Mavromoustakis, F. xia, "A cooperative watchdog system to detect misbehavior nodes in vehicular delaytolerant networks," IEEE Trans. Ind. Electron., vol. PP, no. 99, pp. 1-1, 2016.

Apr. 2015.

The optimal location of two lanes for hybrid VANET-sensor networks, C.C. Lin and D.J. Deng, IEEE Transactions on Industrial Electronics, volume PP, number 99, pages 1-1, April 2015.

"Vehicular cloud computing: a survey," IEEE Globecom Workshops, 2013, pp. 403-407. [10] L. Gu, D. Zeng, and S. Guo.

"Rethinking vehicular communications: merging VANET with cloud computing," by R. Hussain, J. Son, H. Eun, S. Kim, and H. Oh, was published in Proc. IEEE CloudCom in 2012, on pages 606-609.

"Vehicular cloud networking: architecture and design principles," by E. Lee, M. Gerla, and S. Oh; published in IEEE Communications Magazine, volume 52, issue 2, pages 148–155; February 2014.

"PVA in VANETs: stopped cars are not silent," by N. Liu, M. Liu, W. Lou, G. Chen, and J. Cao; published in Proc. IEEE INFOCOM, 2011, pages 431-435.

"Cooperative awareness at low vehicle densities: How parked cars can help see through buildings," by D. Eckhoff, C. Sommer, R. German, et al., was published in Proc. IEEE GLOBECOM (2011), pages 1-6.

According to [15] "Leverage parking cars in a two-tier data center," by L. Gu, D. Zeng, S. Guo, et al., published in Proc. IEEE WCNC, 2013, pages 4665-4670.