ISSN: 2454-9940

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940www.ijsem.org

Vol 9, Issuse.4 Nov 2021

Distributing Virtualized Computing Environment Cloud Resources with Service Management and Discovery Protocol

DR SATYA RANJAN PATTANAIK

Abstract

***Increases in driver convenience, safety, and efficiency are all possible with the advent of vehicle ad hoc networks. However, current communication, storage, and processing capabilities of the connected automobiles are not being fully used to fulfill the service demands of Intelligent Transportation Systems (ITS). Vehicular Cloud Computing (VCC) has potential since it brings the efficiencies of the cloud to vehicular networks. In this study, we provide a strategy for dividing up computing power that increases the VCC system's expected reward. The incentive for the system may be computed by taking into account not just the revenues and costs of the VCC system, but also the resource variability upon which these numbers are based. The optimization problem may be modeled as an infinite horizon Semi-Markov Decision Process after the state space, action space, reward model, and transition probability distribution for the VCC system have all been created (SMDP). An iterative procedure is used to determine the optimal strategy. Specifies the steps to do when a certain condition is met. Numerical results suggest that the SMDP-based strategy may significantly impro**ve performance while keeping complexity under control.*

INTRODUCTION

RECENTLY vehicular networks have gained extensive attention from both academia and industry. A variety of smart sensors and devices are installed on vehicles targeting at data acquisition and processing [1] [2]. Meanwhile, various wireless communications Technologies can be applied to provide the inter-vehicle connectivity. There are usually two types of communication paradigms for vehicle services, i.e., Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) Manuscript received December 6, 2014; revised March 11, 2015 and May 18, 2015; accepted June 13, 2015.

This work is funded in part by National Science Foundation of China (No.61331009), the National High-Tech R&D Program (863 Program 2015AA01A705), National Key Technology R&D Program of China (No.2015ZX03002009-004) and Fundamental Research Funds for the Central Universities (No.2014ZD03-02).

PROFESSOR, Mtech,Ph.D
Department of CSE
Gandhi Institute for Technology,Bhubaneswar.

Kan Zhen (email:kzheng@ieee.org) and Hamlin Men are with Key Lab of Universal Wireless Communications, Ministry of Education, Beijing University of Posts & Telecommunications, Beijing, China, 100088. Pericles Chatzimisios is with Department of Informatics, Alexander TEI of Thessaloniki (ATEITHE), P.O. Box. 141, 57400 Sindos, Thessaloniki, Greece. Lei Lei is with State Key Laboratory of Rail Traffic Control & Safety, Beijing Jiao tong University, Beijing, China, 100044. Xiamen (Sherman) Sheen is with Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. Communications [3]. V2I communication enables vehicles to connect to Internet via a roadside base station, e.g., either by Dedicated Short Range Communication (DSRC) or cellular networks [4] [5]. With supporting the vehicle-related data gathering and processing, vehicular networks can notably improve the transport safety, relieve the traffic congestion and increase the quality of driving experience [7] [8] [9]. As computing and communication technologies have been rapidly

Developed [6], the vehicles with powerful computing abilities are advocated to be regarded as service providers rather than being only service consumers. As a result, the concept of Vehicular Cloud Computing (VCC) has been proposed, that jointly makes use of computation, communication and storage resources in Vehicle Equipments (VEs) [10], e.g., on-board computer/communication devices or Mobile User Equipments (MUEs) carried by passengers. In general, services in the VCC system can be divided into four types according to the function of the resources, i.e., “Network-as-a-Service (Naas)”, “Storage-as-a-Service (Stash)”, “Sensing-as-a-Service (Seas)”, and “Computation-as a- Service (Camas)” [10]. Nowadays, since the computing ability of vehicles is rapidly increased in order to enable them to act as providers of computing services, only Camas is the interest of this paper and is further studied. In the current paper, we propose the deployment of a layered-cloud computing architecture for the VCC system in order to provide satisfactory services to the VEs. The proposed architecture includes not only a Remote Cloud (RC) such as a traditional centralized cloud but also Vehicular Clouds (VCs) that can be regarded as one of computing capability providers besides the RC. Vehicles are produced by different vendors and thus

Have inherently different computation resources. In order to deal with this issue, the virtualization technique has to be developed to abstract and slice the heterogeneous physical resources into virtual resources, which are shared by multiple VEs in the VCC system. In this paper, each vehicle in a VC is assumed to have virtualized Resource Units (RUs).

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

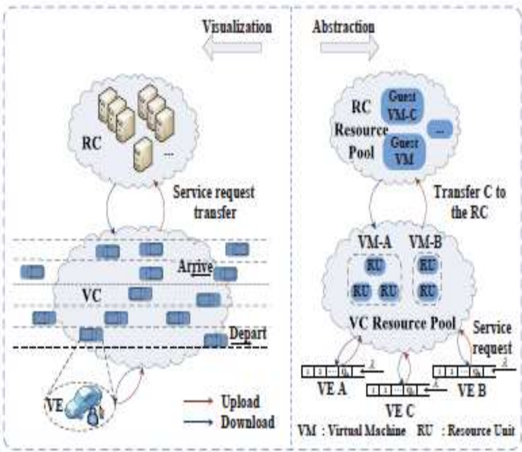


Fig. 1. Illustration of a typical VCC system.

I ∊ {1; 2; *NR*}; *NR* 6 *M*. The arrival rates of service requests and vehicles follow Poisson distribution with *_p* and *_v*, respectively. Let *_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=*imp* in case that *I* RUs are allocated. In addition, the departure rate of vehicles is denoted as *_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 has 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 scare. 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.,

$$\mathcal{S} = \{s | s = (n_1, n_2, ..., n_{N_R}, M, e)\}. \qquad (1)$$

Where *Ni* is the number of service requests that have been allocated with *i* RUs, and *e* represents an event in the set

E ∊ *E* = *{A; D1; D2; DNR; B1; B*−1*}*. Here *A* denotes the arrival of the service request, *Di* means the departure Of a request assigned with *Me* RUs, *B*1 and *B*−1 describe the arrival and departure of a vehicle, respectively. Thus, the Number of occupied RUs in the VC is *N*Σ*R I*=1 *I* · *in*, which satisfies *N*Σ*R I*=1 *I* · *in* 6 *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.

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

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 *As* based on the current state *s*, i.e.,

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

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* understate *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

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

The details of the revenue function are explained as follows:

1) When a service request is admitted to the VC, the instant revenue [*wee* (*El* − *P* · _1) + *waded* (*Dl* − 1=*imp* − _1)] can be earned by the system. (*El* −*P* · _1) and (*Dl* −1=*i_p* −_1) are the saved energy and time when processing the computing task in the VC, respectively. _*e* and _*d* are the price of per unit energy and time. Different weights, i.e., *we* and *wd*, can be predefined according to different purposes, where *we* + *wd* = 1. The transfer expense is denoted as _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 $P \cdot$ _1 energy and _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 $1=i\_p$.

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 _ $(s; a)$ 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 _ $(s; a)$ 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.

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

SMDP-BASED SCHEME FOR VEHICULAR CLOUD COMPUTING

In our analysis, the state transition is determined by the action a understate 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 _ $(s; a)$. Thus, the mean event rate for specific s and *values* is the sum of rates of all the events in the VCC system, which can be expressed by

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

Where _v is the departure rate of vehicles, and $(M\_p +\_v)$ is the total arrival rate of requests and vehicles. Since _p is the arrival rate for requests per vehicle, the arrival rate of requests of the VCC system can be denoted byM_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 $N\Sigma R$ $j=1$ jnj . Thus, the departure rate of vehicles can be computed as $N\Sigma R$ $j=1$ jnj_p. When a request arrives, the number of occupied RUs can be given by (NΣR $j=1$ $jnj+i$) no matter which action taken by the VCC system. Thus, the corresponding departure rate Is computed as (NΣR j=1 jnj_p + i_p). When a request is served and leaves the system, the number of occupied RUs becomes (NΣR j=1 $jnj-i$). The departure rate of request is (NΣR j=1 jnj_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 $s = (n_1, ..., n_{N_R}, M, A)$,

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

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

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

B. Discounted reward model

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

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

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]

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

C. Solution

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

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

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

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

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

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

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

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

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

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

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

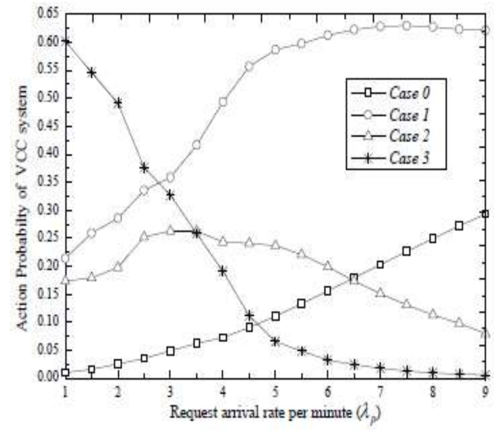
$$\tilde{v}^{k+1}(s) = \max_{a \in A_s}[\tilde{r}(s,a) + \tilde{\lambda}\sum_{s' \in S}\tilde{p}(s'|s,a)\tilde{v}^{k}(s')].$$

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

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

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

In our paper, the norm function is defined as ||~**v**|| = max |~*v*(*s*)| for *s* ∈ *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 ~*vk*(*s*) converges in norm to *v*~_ ∗(*s*). Note that the convergence rate of the value iteration algorithm is linear with the rate ~ _.



CONCLUSION AND FUTURE WORK

In this research, we present a Semi-Markov decision process-based method for allocating computing resources in a Vehicular Cloud Computing system with an unlimited time horizon (SMDP). To maximize the VCC system's projected total reward over the long run, an optimum decision making strategy is developed through the iteration process. Experiments demonstrate that when _p is large or K is low, anticipate reward outperforms alternative allocation schemes by a substantial margin. In addition, the complexity of the SMDP-based system is less than that of the SA scheme. We hope that further study will allow us to build more robust and practical schemes by examining the impact of parameter tolerance on the best scheme in the VCC system. Keeping in mind that a VCC system's size is growing fast makes this an even more difficult task...

REFERENCES

[1] Y. Lee, and K. Rajashekara, “Power electronics and motor drives in electric, hybrid electric, and plug-in hybrid electric vehicles,” *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2237-2245, Jun. 2008.

[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, Aug. 2013.

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

[4] J. Kenney, “Dedicated short-range communications (DSRC) standards in the united states,” *Proc. IEEE*, vol. 99, no. 7, pp. 1162-1182, Jul. 2011.

[5] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro, “LTE for vehicular networking: a survey,” *IEEE Commun. Mag.*, vol. 51, no. 5, pp. 148-157, May. 2013.

[6] M. Jin, X. Zhou, E. Luo, X. Qing, ”Industrial-QoS oriented remote wireless communication protocol for the Internet of construction vehicles,” *IEEE Trans. Ind. Electron.*, vol. PP, no. 99, pp. 1-1, Jun. 2015.

[7] F. Dressler, H. Hartenstein, O. Altintas, and O. Tonguz, “Inter-vehicle communication: Quo vadis,” *IEEE Commun. Mag.*, vol. 52, no. 6, pp. 170-177, Jun. 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,
Apr. 2015.

[9] C.C. Lin and D.J. Deng, Optimal two-Lane placement for hybrid VANET-sensor networks, *IEEE Trans. Ind. Electron.*, vol. PP, no.99, pp.1-1, Apr. 2015.

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

[11] R. Hussain, J. Son, H. Eun, S. Kim, and H. Oh, “Rethinking vehicular communications: merging VANET with cloud computing,” in *Proc. IEEE CloudCom*, 2012, pp. 606-609.

[12] E. Lee, M. Gerla, and S. Oh, “Vehicular cloud networking: architecture and design principles,” *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 148-155, Feb. 2014.

[13] N. Liu, M. Liu, W. Lou, G. Chen, and J. Cao, “PVA in VANETs: stopped cars are not silent,” in *Proc. IEEE INFOCOM*, 2011, pp. 431-435.

[14] D. Eckhoff, C. Sommer, R. German, *et al.*, “Cooperative awareness at low vehicle densities: How parked cars can help see through buildings,” in *Proc. IEEE GLOBECOM*, 2011, pp. 1-6.

[15] L. Gu, D. Zeng, S. Guo, *et al.*, “Leverage parking cars in a two-tier data center,” in *Proc. IEEE WCNC*, 2013, pp. 4665-4670.