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.ijasem.org

Vol 8, Issuse.3 Sep 2020

Vehicle Cloud Computing Resource Allocation Using SMDP

E.Muralidhar Reddy[1],.B.Venateswarlu[2],Dr.T.Sreenivasulu[3],Erugu Krishna[4],

Abstract—***Autonomous vehicle networks are expected to improve traffic flow and safety while also enhancing the driving experience for drivers. As a result, Intelligent Transportation Systems (ITS) cannot fully take use of the existing communication, storage, and computing capabilities of linked vehicles (ITS). Through Vehicular Cloud Computing, cloud computing's advantages may be used to vehicle networks (VCC). We propose an efficient allocation of computing resources to maximise the long-term anticipated reward of the VCC system. When determining the incentive for the VCC system, both income and expenses, as well as fluctuations in resources, are taken into account. An infinite-horizon Semi-Markov Decision Process (SMDP) is utilised to solve the optimization problem, using the provided state space, action space, reward model and distribution of transition probabilities of the VCC system as inputs. The best way to describe what has to be done is to utilise a state-space iteration technique. Numerically, the dramatic improvement in performance may be shown by***

Index Terms—in Vehicular Cloud Computing, Semi Markov Decision Process (SMDP) and resource allocation

INTRODUCTION

Recent attention has been given to vehicle networks by both academics and industry. In order to collect and analyse data, cars are equipped with a wide range of smart sensors and gadgets [1, 2]. There are a variety of wireless technologies available for inter-vehicle networking, as well. V2V and V2I communication paradigms are the two most common forms of vehicle service communication paradigms (V2I) Revisions were made in March and May; the manuscript was approved on June 13, 2015. IEEE is the copyright holder of this work. It is okay to use this content for your own personal purposes. This content may only be used for educational purposes, and permission must be requested by emailing pubs-permission@ieee.org. China's National Key Technology R&D Program, China's National Science Foundation, and the Fundamental Research Funds for Central Universities are among the sources of funding for this research project (No.2014ZD03-02). Beijing University of Posts & Telecommunications, Beijing, China, 100088, is home to the Key Lab of Universal Wireless Communications, which includes Kan Zheng and Hanlin Meng. P.O. Box 141, 57400 Sindos, Thessaloniki, Greece, Alexander TEI of Thessaloniki (ATEITHE) Department of Informatics. Lei Lei works at Beijing Jiaotong University's State Key Laboratory of Rail Traffic Control & Safety, Beijing, China, 100044. At the University of Waterloo in Waterloo, Ontario, Canada's Department of Electrical and Computer Engineering, Xuemin (Sherman) Shen works as a researcher. The 3G1 network of N2L companies [3]. A roadside base station, such as a DSRC or a cellular network, may be used to link automobiles to the Internet through V2I communication [4] [5]. Vehicle networks can significantly improve transportation security, alleviate traffic congestion, and enhance the driving experience by allowing the collection and processing of vehicle-related data [7]. [9] [8] Vehicles equipped with significant processing capabilities should be seen as service providers rather than service consumers, according to the authors [6].

Professor[1,2,3,4], Assistant Professor[1,2,3,4], ,Associate Professor[1,2,3,4],
Department of CSE Engineering,
Pallavi Engineering College,
Mail ID:krishna81.reddy@gmail.com, Mail ID:bvenkat1109@gmail.com,
Kuntloor(V),Hayathnagar(M),Hyderabad,R.R.Dist.-501505.

Consequently, a concept called Vehicular Cloud Computing (VCC) was presented, which combines computing, communication and storage resources in VEs (e.g., on-board computer/communication devices or MUEs) carried by passengers. Network-as-a-Service (NaaS), storage-as-a-service (StaaS), sensing and computation all fall under the umbrella term "Service as a Service" in the VCC system, which encompasses all four kinds of services mentioned above. [10]. CaaS is the focus of this article since cars' computer power is fast increasing in order to allow them to serve as suppliers of computing services. A layered-cloud computing architecture is proposed for the VCC system in this study in order to deliver appropriate services for the VEs. There is a Remote Cloud (RC) and a Vehicular Cloud (VC) in the proposed architecture, which may be seen as a computing capacity supplier in addition to the RC. It is possible for the VC to be either mobile or static, depending on the mobility of its vehicles. For example, a mobile VC is made up of moving vehicles, while a static VC is made up of stationary vehicles. Its unique properties set it apart from other types of cloud computing. One of them is the wide range of computing resources accessible in VCs. VC resources are time-varying because of the randomness of vehicle behaviour, such as cars joining and leaving VCs. A VCC system is assumed to have the following characteristics, i.e.: 1) service requests per vehicle arrive and depart in a random Poisson distribution; 2) both the arrivals and departures of vehicles in a VC follow the same distribution; and, finally, 3) the number of available resources in the VCC is dynamic and time-varying. This assumption is made for the sake of analysis. Because they are made by various companies, automobiles have a wide range of varying computational resources. The virtualization approach must be developed to abstract and slice diverse physical resources into virtual resources shared by numerous VEs in the VCC system in order to cope with this problem. Virtualized Resource Units are assumed in this article for each vehicle in a VC (RUs). VCC system resource allocation is the primary emphasis of this study, which examines how to optimise the long-term projected benefit of the system. At some point, the VCC system must decide whether or not to execute a service request received from a vehicle locally in a VC or to send it on to the RC. To add insult to injury, we must also address the problem of assigning resources for this service request if it is allocated to a virtual machine (VC). Using the VCC system, it is hoped that the user would get a reward depending on their actions. Both power consumption and processing time are taken into account when calculating the reward, which is a combination of revenue and costs. An infinite horizon Semi-Markov Decision Process (SMDP) is used to solve the resource allocation issue (SMDP). An analysis of the VCC system's state-action-reward model-transition probability distribution is performed to identify the best strategy for a given state, which dictates the action to be conducted. The SMDP-based scheme, or the best allocation policy, may be obtained by iteration. The SMDP-based allocation scheme outperforms the other two allocation schemes, namely the SA and GA schemes, in terms of numerical outcomes. The remainder of the paper is laid out in this way: An overview of the relevant literature is provided in Section II. Section III goes into detail on the system model for Vehicular Cloud Computing. SMDP concept, suggested model, and solution are detailed in Section IV of this document. Section V summarises the findings in terms of numbers and performance metrics. Ending thoughts and ideas for further research are included in Section VI.

II. RELATED WORK

The VCC has undergone a few upgrades to improve the capability of VEs. There are many similarities between the VCC and MCC systems, but it also has some unique features. Vehicle Cloud Computing (VCC) is broken down in [11] into three distinct architectural models: the VC, the VuC, and the Hybrid Cloud (HC). The formation of VCs capable of efficiently dealing with locally generated services and enhancing the VEs' experience has also been emphasised. [12]. In [13], the Parked Vehicle Assistance (PVA) is suggested to overcome sparse/unbalanced traffic and considerably increase network connection by using the parked automobiles as static cloud nodes. It is also used to detect vehicles that are not directly in the driver's line-of-sight. [15] has investigated a two-tier data centre design that makes use of the surplus storage space in parking lots. Furthermore, the VCC system's security is the primary focus of efforts in [17] and [18]. The VCC has undergone a few upgrades to improve the capability of VEs. A Mobile Cloud Computing (MCC) system is quite similar to VCC, although VCC has additional features. Vehicles utilising Clouds (VuCs) and Hybrid Clouds (HCs) are the three architectural structures that make up the VCC system as described in [11]. Furthermore, it has been noted that in order to build the VCs, they must be able to successfully deal with local services and enhance the experience of VEs [12]. It is suggested in

[13] to use Parked Vehicle Assistance (PVA) as a static cloud node to alleviate sparse/unbalanced traffic and boost network connection. It is also used to detect vehicles that are not directly in the driver's line-of-sight. In [15], a two-tier data centre design that utilises parking lot storage has been investigated. Furthermore, the VCC system's security is the primary focus of efforts in [17] and [18].

VEHICULAR CLOUD COMPUTING SYSTEM

The system's architecture model VCC systems, like the one seen in Fig. 1, often have a dynamic VC. When making service requests to the VCC, VEs that function like smartphones may take use of the tremendous computational capacity available. The VCC system assumes that a vehicle has just one fundamental computation RU. There are two options available when a new request comes in: either the VC (the system's primary service provider) accepts or rejects it. The VC must decide how many RUs to allot to the request depending on the existing availability of resources. Alternatively, the RC may be contacted and the service request may be transferred to the RC's attention. In Fig. 1, an example is also included for illustration's purpose. The VC accepts requests from VE A and VE B, but VE C's request must be sent to the RC. After VE A and VE B have been accepted, they are each given three RUs, with VE A receiving three and VE B receiving two. In the VCC system, all choices are taken in order to attain the stated goal. It's included in Table I, which includes all the most relevant points in this work. There are M available RUs in the VC, which changes as cars arrive and leave. Vehicle capacity is limited by K, which is defined as how many cars the VC can accommodate. I RUs may be assigned to each arrival service request.

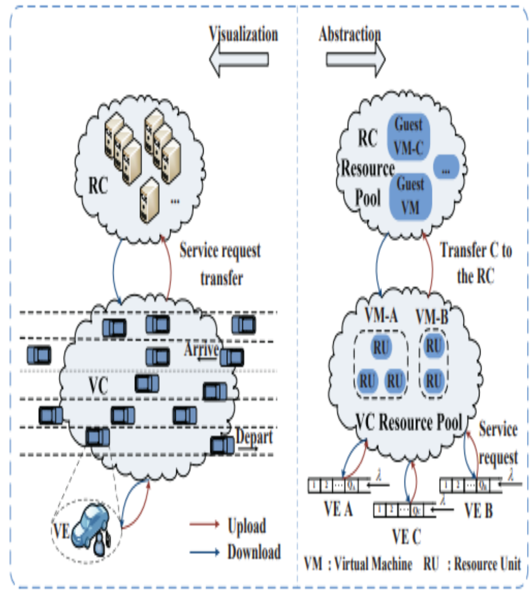


Fig. 1 shows a typical VCC system in action.

For example, NR 6 M. It follows the Poisson distribution with p and v for service requests and vehicles, respectively. When just one RU is granted, the computing service rate is referred to as p. If I RUs are assigned, then the service time for a request is 1/ip. Vehicle departure rates are sometimes known as v. To put it another way, the present epoch may directly affect future state, which in turn has a significant effect on predicted total reward, due to the dynamic nature of service requests and vehicle arrivals. For example, when resources are few, it may be imprudent to pursue a short-term goal of maximising the present epoch's reward. As a result, the goal of this article is to effectively allocate resources within the VCC system in order to maximise long-term predicted total reward.

System States

The present condition of the system indicates the number of RUs requested, the resources in the VC that are available, and the occurrence of requests and vehicles. This means that S may be indicated by S, which is the number of service requests that have been assigned with ni RUs, and e denotes an event in the collection e E = [A, D1, D2,..., DNR]. When a vehicle arrives or departs, we write B1 and B1. When a service request arrives, we write A. When a request assigned with I RUs leaves, we write Di. Consequently, the total number of available RUs in the VC is N, which is sufficient to meet the requirement of N = 1 + 1 + ni = 6 M. A further option is to express the number of system states (N) with the letter N. Actions The action set A in this model has several options for action a, for example,

TABLE I
LIST OF IMPORTANT NOTATIONS.

K	Maximal number of vehicles that the VC can su
M	Number of available RUs in the VC
N_R	Maximal number of RUs that the VC can alloca 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 I
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 V
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

$$k(s,a)=\begin{cases}[w_e\beta_e(E_l-P\cdot\delta_1) \\ +w_d\beta_d(D_l-1/i\mu_p-\delta_1))- \\ I-\gamma(\delta_2+\delta_1), \\ 0, & a=-1, e\in\{D_1 \\ 0, & a=-1, e\in\{B_- \\ -\xi, & a=-1, e\in\{B_-\end{cases}$$

The following are the specifics on how the revenue function works: 1) The system may earn the immediate income [wee(El P 1) + wdd(Dl 1/ip 1)] when a service request is accepted by the VC. Energy and time are saved when the computational work is performed in the VC by (El – P) and (Dl–1/i–p) accordingly. There are two prices for energy and time: e and d. Various weights, i.e., we and wd, may be pre-defined based on various reasons, where we + wd = 1. As an example, 1 denotes the VCC system's cost of receiving a computing job from VEs and transmitting the results back. This is especially true because the VC has already agreed to take on this assignment, and as a result, it just costs P 1 energy and time for the VE to send the task and get the reply from it. Transmission and reception power are considered to be equal in this study [28]. The service time required to complete the job is 1/ip if the request is assigned I RUs by the VCC system. In the event that there are not enough resources in the VC, the service request may be forwarded to the RC. As a result, the VCC system earns I income at the expense of transfer expenses such as 2 and 1. Here, 2 represents the cost of transmitting and receiving data from the RC. Revenue I may be estimated using [wee(ElP1)+wdd(Dl–1–2)] without taking into account the processing time, as the RC is considered to have significant computational capabilities. Furthermore, because of the long end-to-end communication latency, the VCC system should not transmit the requests to the RC if the resources in the VC are adequate. When a service departs or a vehicle enters the VCC system, there is no income. No income is generated in states where the VCC system has spare RUs to distribute when a vehicle departs the system. In the event that all the RUs have been used up, the VCC system must reimburse the request occupying this RU with a fee equal to the RU's value. It's because there are no extra RUs in the system that can be used to ensure that the request whose RU is departing has enough RUs to complete the task The next step is to estimate the total cost of the system.

TABLE II
TRANSFORMATION OF ACTIONS AND CORRESPONDING STATES.

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	$a=3$	$s_{next1}=\{1,1,2,M,D_i\}$ $,i\in\{1,2,3\}$ $s_{next2}=\{1,1,2,M,A\}$ $s_{next3}=\{1,1,2,M,B_1/B_{-1}\}$

where c(s, a) is the cost rate of (s, a) if action an is chosen under state s, and (s, a) is the projected service time from the current state to the next state. A further way to identify c(s, a) is to look at the number of RUs used in the VC, which is constrained in terms of processing power:

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

SMDP-BASED SCHEME FOR VEHICULAR CLOUD COMPUTING

The action an under state s determines the state transition in our study. As an example, take the system state (s = (1, 1, 1, M, A)), and the corresponding state transition under various actions is provided in Table II (see Figure 1). In addition, the likelihood of a state change under various actions has a significant impact on the optimum strategy. Since we'll be focusing on state transition probabilities in this part, we'll begin by calculating them. Once the discounted model has been implemented, the reward function must be reevaluated. Here we provide the optimum policy that may be discovered using the value iteration approach.

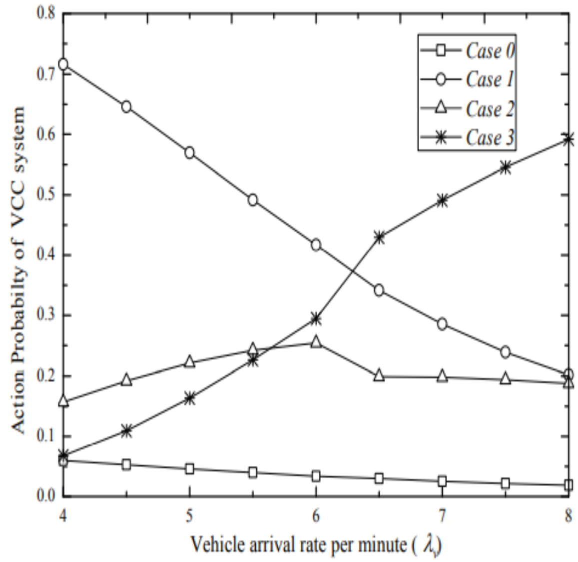
Transition Probability

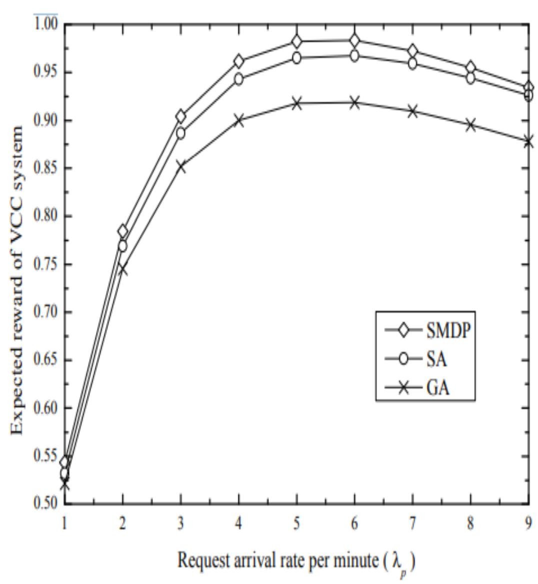
Using a given state s and an action a, the service time between two continuous decision epochs is referred to as (s, a). Since the total number of events in the VCC system can be stated as, the mean event rate for certain s and a values is

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

V. NUMERAL RESULTS AND ANALYSIS:

The suggested computation resource allocation technique should be evaluated. As a benchmark, we compare the suggested allocation scheme's performance to that of the following two allocation systems: • GA: Greedy Allocation At each epoch of decision-making, the VCC system will always assign the most RUs possible in order to maximise the system reward. plan to simulataneously anneal The SA method is often employed to identify near-optimal solutions to optimization issues since it is a typical heuristic approach [31]. It is, however, difficult to implement owing to the high computational cost of obtaining each new policy's objective function value, particularly in the case of a large number of system states. The SMDP-based technique has a polynomial complexity of O(N2) in order to get the best outcomes for resource allocation in the VCC system. But the GA and SA schemes have complexity of O(N) while the complexity of the first two is of order O(N3). Table III lists the parameters we employed in our study. It is possible to assign a service request 1, 2, or 3 RUs, depending on the VC's available resources. The maximum number of RUs that may be given to a single service request is NR = 3. The arrival rate of service requests and cars, as well as the maximum number of vehicles K that the VCC system can accommodate, may be changed for assessment purposes.. The VCC system assigns the request to the VC and provides it with 1, 2, and 3 RUs, correspondingly, in cases 1, 2, and 3. Case 0 signifies that the VCC system passes the request to the RC, whereas the rest of the cases are handled by the VCC system itself. There are three distinct p values shown in Figures 2, 3 and 4. When the number of requests per car is modest, as illustrated in Figure 2, the VCC system has enough of resources in the





CONCLUSION AND FUTURE WORK

An infinite horizon Semi-Markov Decision Process (SMDP) has been presented for the allocation of compute resources in a Vehicular Cloud Computing system in this research (SMDP). Decisions are made in an iterative manner to optimise the long-term total benefit of the VCC system using the iteration algorithm. For example, if you compare it to the Greedy Allocation (GA) scheme, the expected reward performance gains are about 7% when either p or K is large. In addition, the SMDP-based plan is less complicated than the SA scheme. More robust and practical methods may be developed as a result of our future research into the impacts of parameter tolerance on the optimum VCC system scheme. Taking into mind that a VCC's system size is continually increasing, this becomes a more difficult task. In addition, the VCC system prefers to distribute as many RUs as feasible to the VC when it allocates a single request to the system. Because of this, Case 3 has the greatest likelihood whereas Case 1 and Case 2 have lesser probabilities, and that of Case 0 is the lowest. Situations like this tend to shift as more requests arrive per car. Decisions are made more cautiously by the VCC system since the incentive of allowing a new request with only one or two RUs is more appealing. Because the VC is now obligated to approve the new request, it is no longer prudent to accept a request that has three RUs in it. Consequently, the likelihood of Case 3 decreases while the likelihood of Cases 1 and 2 increases. As more requests arrive, the VCC system starts to reduce the likelihood of Case 2 by allocating just one RU per request. In Fig. 3, the odds of Cases 2 and 3 grow while those of Cases 0 and 1 drop when the arrival rate of cars increases. Due to the fact that the VC's resources have a tendency to become adequate as vehicle arrival rates rise. Due to the abundance of resources, the chance of Case 2 diminishes when the arrival rate is large. Figure 4 shows that when the maximum number of cars supported by the VCC system increases, the chance of Case 3 increasing increases, which is also due to more RUs being allotted to the request. In the next section, we compare the performance of several VCC systems, such as the SA, GA, or SMDP-based systems. As can be seen in the graphs in Figures 5, 6, and 7, the total predicted reward changes over time. To put it another way, when more requests are approved and handled by VCs, the predicted total reward of the VCC system rises as a result. However, since the likelihood of a transfer to the RC increases, the predicted total reward decreases when the pace of requests is high. It

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. BaguenaAlbaladejo, 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. [16] R. Yu, Y. Zhang, W. Xia and K. Yang, “Toward cloud-based vehicular networks with efficient resource management,” IEEE Network Mag., vol. 27, no. 5, pp. 49-55, Sep. 2013.

[17] G. Yan, et al., “Towards secure vehicular clouds,” in Proc. CISIS, 2012, pp. 370-375. [18] G. Yan, et al., “Security challenges in vehicular cloud computing,” IEEE Trans. Intell. Transp. Syst., pp. 284-294, Mar. 2013.

[19] N. Cordeschi, D. Amendola, and E. Baccarelli, “Reliable adaptive resource management for cognitive cloud vehicular networks,” IEEE Trans. Veh. Technol., vol. PP, no. 99, pp. 1-10, Aug. 2014.

[20] N. Cordeschi, et al., “Distributed and adaptive resource management in cloud-assisted cognitive radio vehicular networks with hard reliability guarantees,” Veh. Commun., vol. 2, no. 1, pp. 1-12, Aug. 2014.