Classification, Challenges and Critical Comparison of Proposed Solutions for Vehicular Clouds

Hassan Mistareehi and D. Manivannan
University of Kentucky, Lexington, Kentucky, USA.

Modern vehicles are equipped with devices that have computation, storage, communication and sensing capabilities. Vehicles can share the information collected with other vehicles and also can share this information with an external cloud. Vehicles can also form a cloud among themselves and use their underutilized resources to process and share the information collected by various vehicles. In this paper, we present a critical comparison and classification of vehicular cloud architectures proposed in the literature. We also explore the challenges, proposed solutions for meeting the challenges and their drawbacks in implementing vehicular cloud and identify some open issues that need to be addressed for the successful implementation of vehicular clouds.

Keywords: Vehicular cloud, VANET, Cloud computing

1. INTRODUCTION

Vehicular Ad hoc NETworks (VANETs) are a type of Mobile Ad hoc Networks (MANETs) that allow vehicles on roads to form a self-organized network. VANETs can help to collect information and share them with other vehicles. For example, it can warn drivers of road conditions that could cause an accident. In case of an accident, the velocity information exchanged between vehicles before a collision can help law-enforcement agency reconstruct the accident. Information about the accidents could be instantly sent to the law enforcement agencies which can help them reach the scene faster. When VANETs are in widespread use, information about traffic and road hazards could be acquired in real-time and fed into vehicle navigation systems to provide alternate driving routes. VANETs are likely to provide support for cooperative driving applications, which would allow vehicles to navigate without driver intervention (Bernsen and Manivannan, 2008; Lim and Manivannan, 2016).

In the last decade, cloud computing emerged as an economical solution for customers to rent IT infrastructures, platforms, or software, instead of investing money to own and maintain such services. The service providers give such flexible services to customers when they need them, and then they charge them based on their usage. Services provided by the cloud can be broadly divided into three types: IasS (Infrastructure as a Service), PasS (Platform as a Service), and SaaS (Software as a Service). While the primary features of the cloud are cost saving, on-demand service, resource pooling, scalability and ease of resource accessing, security and privacy concerns are the major obstacles for the widespread adoption of the cloud (Lim, Abumuhfouz, and Manivannan, 2015).

Vehicular networks can also benefit from cloud computing. Vehicles are equipped with computing, communication and storage resources, but they are often underutilized. In order to fully utilize the resources of vehicles in VANETs, Olariu et al. (Olariu, Khalil, and Abuelela, 2011) proposed vehicular cloud architecture, which combines VANET and cloud, called Vehicular Cloud (VC). A vehicular cloud is a collection of autonomous vehicles in VANET that contribute their un-

Authors' addresses: Hassan Mistareehi, Department of Computer Science, University of Kentucky, Lexington, KY 40506. Email: hassan.mistareehi@uky.edu
D. Manivannan, Department of Computer Science, University of Kentucky, Lexington, KY 40506. Email: mani@cs.uky.edu.

derutilized computing, sensing, and communication devices to the cloud. Vehicle's resources and the information shared by the vehicles with the cloud can be used in decision making (Eltoweissy, Olariu, and Younis, 2011).

In VANET, multiple can observe the same phenomena and propagate it to other vehicles which can result in redundant propagation of data and waste the vehicle's resources. A vehicular cloud allows vehicles to exchange their collected data with the cloud where it can be analyzed, verified, organized, aggregated and then propagated to the relevant vehicles/customers. Various applications can also benefit from using a VC. Some of these applications include accident alerts, parking management, road conditions alert, cooperative driving, traffic management. Planned evacuation system is another application that could benefit from using the VC. When disasters like hurricanes occur, VCs can contribute in organized evacuations. Vehicles can also receive software updates from cloud when vehicle manufacturers upload a new version of software (Lim et al., 2015; Yan, Wen, S.Olariu, and M.Weigle, 2013).

Paper Objectives

The objective of this paper is to classify, compare and characterize the various vehicular cloud architectures proposed in the literature. We classify vehicular cloud architectures into three categories: temporary, permanent, and hybrid and discuss the merits and demerits of each of them. Then, we discuss the various challenges in implementing vehicular cloud and some proposed solutions to meet these challenges. The merits and demerits of the solutions presented in the literature and open issues are also discussed.

The rest of the paper is organized as follows. In Section 2, we present a classification of vehicular cloud architectures. In Section 3, (i) we identify and classify challenges in implementing vehicular cloud, (ii) discuss some proposed solutions to overcome these challenges and (iiii) identify the drawbacks of the proposed solutions and also some open issues. Finally, Section 4 concludes the paper.

2. VEHICULAR CLOUD ARCHITECTURES: A CLASSIFICATION

In this section, we classify the vehicular cloud architectures into three categories: temporary, permanent, and hybrid and discuss their merits and demerits.

2.1 Temporary cloud

A temporary cloud consists of vehicles that together form a cloud temporarily which allows them to share their resources (e.g., computing, networking, and storage) for collecting, processing and disseminating information to other vehicles and other relevant customers as needed or carry out a requested task from other vehicles/entities.

Lee et al. (Lee, Lee, Gerla, and Oh, 2014) proposed Vehicular cloud networking (VCN). The goal of VCN is to form a temporary vehicular cloud through collaboration of vehicles in order to provide the needed services. Under VCN, one of the vehicles in the cloud is elected as a cloud leader based on some selected metrics (e.g., connectivity to other vehicles) and the rest of vehicles cooperate in the cloud formation process. The cloud leader broadcasts a resource request (RREQ) message to vehicles within its range. Vehicles willing to share their resources (e.g., storage, phenomena observed by sensors and computing resources) send a resource reply (RREP) message back to the cloud leader with information on their resource capabilities. After receiving RREP messages, the cloud leader selects cloud members and constructs a cloud. The cloud leader assigns tasks to cloud members taking into consideration the available resources of the respective members. The cloud members return the results back to the cloud leader after completing their tasks. After collecting the results from the cloud members, the cloud leader processes them and then publishes the final results. Vehicles may leave and join the cloud at any time; the cloud leader is responsible for managing the cloud. For example, when a vehicle leaves the cloud, the cloud leader selects another member in the cloud that has the necessary resources to complete the tasks assigned to the leaving member and assigns those tasks to that member. When the cloud leader no longer uses the cloud or moves out of the cloud, it sends a cloud release message to all the members, so they can join other clouds and contribute their resources.

A cluster-based vehicular cloud architecture has been proposed by Arkin et al. (Arkian, Atani, Diyanat, and Pourkhalili, 2015). Their architecture uses a clustering technique to solve the resource allocation problem (e.g., some applications need more storage and computation resources) by grouping the vehicles according to the vehicle's location and velocity and allowing vehicles in the same group to provide resources. In this scheme, vehicles form clusters and cluster heads (CH) are selected using fuzzy logic. They model a cluster head selection algorithm that allows to select a set of optimal CHs. CHs are determined based on their FitFactor, defined in the paper. A CH is responsible for the creation, maintenance, and deletion of vehicles in the cluster. All vehicles in the cluster register their resources with the CH. If a vehicle needs some resources of the vehicular cloud, it asks the CH. The CH is responsible for allocating all resources in the vehicular cloud. This approach is different from VCN (Lee et al., 2014) from the way in which CHs are selected. Fig. 1 shows the proposed cluster-based vehicular cloud architecture and Table I summarizes the merits and demerits of temporary cloud architectures discussed above.

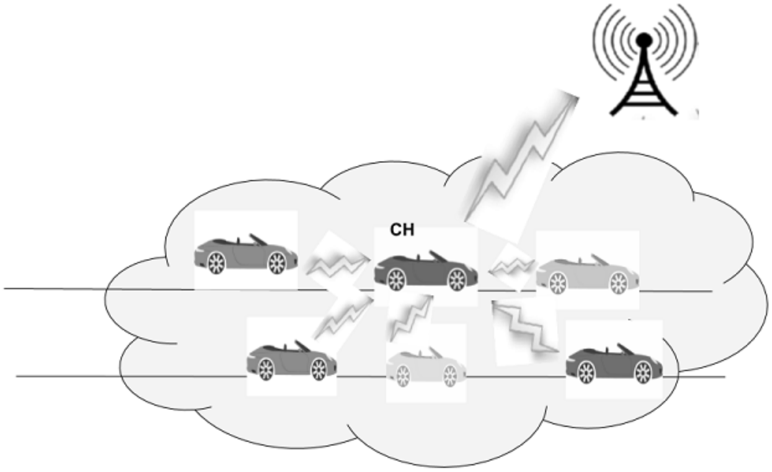

Fig. 1. Temporary cloud that uses cluster based vehicular cloud architecture (Arkian et al., 2015).

Temporary cloud architectures	**Merits**	**Demerits**
VCN (Lee et al., 2014)	- Provides services to members of VCN. - The cloud leader distributes the tasks to cloud members, taking into account the availability of their resources. - Allows processing of information by using resources of all vehicles in the cloud.	- Computing/ storage resources are limited compared to the conventional cloud. - The selection criteria of the cloud members and the format of the exchanged messages are not specified. - The cloud leader is a possible bottleneck because it is responsible for assigning tasks to cloud members, collecting results from cloud members and managing the cloud. - If a cloud leader fails, data aggregated maybe lost, in this case the cloud will become useless.
Cluster-based vehicular cloud (Arkian et al., 2015)	- Uses a clustering technique to solve the resource allocation problem and to improve vehicular networks performance. - A predefined criteria is used for selecting Cluster Head.	- If the CH fails, data aggregated may be lost.

Table I: Merits and demerits of temporary cloud architectures.

2.2 Permanent cloud

A permanent cloud supports increased computing processing capacity. In a permanent cloud, vehicles send the phenomena collected as well as contribute their hardware resources to the cloud. Then the permanent cloud processes the data gathered from the vehicles and provides the needed services to vehicles/drivers (e.g., parking information, road conditions, accidents, traffic information, etc).

Hussain et al. (Hussain and Oh, 2014) proposed Cooperation-Aware VANET Cloud. In their approach, vehicles and cloud infrastructure cooperate with each other to provide drivers with services such as traffic information and warning messages. Vehicles share observed phenomena with the cloud and the cloud processes the data gathered from the vehicles and shares the information with the vehicles in the cloud. This model consists of two architectures: The first architecture is VANET, which consists of vehicles serving both as producers (they send information to the cloud) and consumers (they get information from the cloud). The second architecture is the permanent cloud, which consists of Authenticator, Cloud Collecting Point(CCP), Cloud Knowledge Base(CKB), and Cloud Decision Module(CDM). The authenticator is responsible for handling contributions from vehicles and authenticating them. The data are collected at CCP and sent to CKB for processing, and then the processed information is shared with the VANET users. Virtualization layer that works as mediator between VANET and the cloud, takes inputs from VANET and passes them to the cloud and disseminates the output from the cloud to vehicles in the VANET.

Wan et al. (Wan, Zhang, Zhao, Yang, and Lloret, 2014) proposed a context-aware architecture with mobile cloud support for vehicular cyber-physical systems (VCPS). Context-awareness allows the adaptation of services according to many factors such as occurred changes in the environment, user preferences, user location, and capabilities of mobile devices. The applications and services in VCPS are divided into three different computational layers: location computational layer, vehicle computational layer and cloud computational layer. In the location computational layer, RSUs deployed at strategic locations can exchange information with OBUs installed on vehicles. When a vehicle with an OBU passes by an RSU, it can receive the updated traffic information from the RSU and share its own information (e.g., destination and vehicle route data) with the RSU. Vehicles that do not have an RSU within their transmission range can connect to RSUs through neighboring vehicles. In the cloud computational layer, there are multiple systems working with each other to share resources and provide a number services such as vehicle multimedia contents and traffic information.

Salahuddin et al. (M. Salahuddin and Guizani, 2015) proposed a vehicular cloud architecture, called RSU cloud, which uses RSUs and datacenters. The RSU cloud provides services that meet changing demands from the vehicles. The RSU cloud architecture exploits the flexibility and deep programmability offered in software-defined networking (SDN). In SDN, there are two communication planes, the physical data plane and an abstracted control plane. This decoupling of control and data planes enables the flexibility and programmability of the SDN. Drivers register with the RSU-cloud and the cloud keeps track of the driver's status like stability (alcoholic or not) and current location. Users can request service from the RSU cloud. Based on the user's request, RSU cloud will respond to users. In the RSU cloud, virtualization via virtual machines (VMs) and SDN is employed to dynamically instantiate, migrate, replicate services and reconfigure data forwarding rules in the network to meet the frequently changing service demands.

The cloud-based system proposed by Wang et al. (Xu, Wang, Amin, Martin, and Izard, 2015) consists of cloud, several types of radio access networks (RANs), and a set of vehicles. Vehicles are assumed to be equipped with GPS and OBUs. In the cloud, there are various servers, which could be either real physical machines or virtual machines. Vehicles from time to time report their status such as current location and speed through RSU to the cloud. RSUs send updates of their status such as the number of active vehicles and traffic load to the cloud. The cloud then processes the collected data and disseminates the information to vehicles that are in need.

Lim et al. (Lim et al., 2015) proposed a secure incentive based architecture for vehicular cloud to encourage vehicles to participate in the cloud. Tokens are given to the vehicles as a reward to participate in the cloud and vehicles can use the token to get services from the cloud. The proposed scheme has three phases. In phase 1, the service provider manager (SPM) sends a message asking vehicles for sharing their resources. When an interested vehicle receives the message and wants to share its resources with the cloud, it sends a message to the SPM through the RSUs. Then, the SPM authenticates the vehicle with the help of a trusted authority (TA). Once the vehicle is authenticated, the SPM signs a contract between the service provider and the vehicle and sends it to the vehicle, so the vehicle can start allowing its resources to be used by the cloud. Every vehicle uses its pseudo ID in all communications to protect its privacy. In phase 2, a vehicle sends a message with the proof of the work done to the SPM. The SPM verifies the proof of the work done, and sends a reward token request to the reward token system (RTS) so it can send tokens to the vehicle. In phase 3, the reward token earned for participating in the cloud are used as payment for the cloud services obtained. Vehicles can check their token balance with the OBUs to buy services from the cloud. Fig. 2 illustrates the secure architecture for vehicular cloud that encourage vehicles for participating in the cloud (Lim et al., 2015) and Table II summarizes the merits and demerits for permanent cloud architectures discussed above.

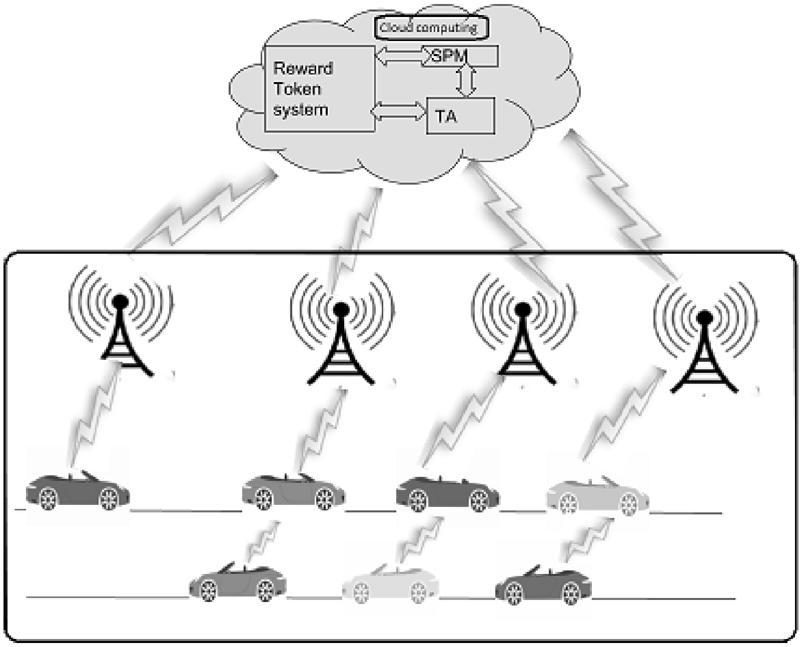

Fig. 2. Permanent cloud that uses incentive-based architecture for vehicular cloud (Lim et al., 2015).

Permanent cloud architectures	**Merits**	**Demerits**
Cooperation-Aware VANET Cloud (Hussain and Oh, 2014)	- This scheme provides the vehicles with traffic information and warning messages.	- This architecture does not encourage vehicles to participate in the cloud.
Context-aware architecture (Wan et al., 2014)	- They create a context-aware pervasive system for mobile vehicles, drivers, passengers, and relevant traffic authorities by designing a multi-layered architecture with cloud capability. - Each layer provides multiple context aware services.	- Security issues are not addressed.
RSU cloud (M. Salahuddin and Guizani, 2015)	- The RSU cloud hosts services to meet the demand from the OBUs in the vehicles. - The RSU cloud architecture explores the benefits from the flexibility and deep programmability offered in SDN. - In the RSU cloud, virtual machines (VMs) and SDN are used to dynamically instantiate, migrate, and/or replicate services and dynamically reconfigure data forwarding rules in the network to meet the frequently changing service demands.	- Despite the benefits of the programmability of RSU clouds, service instantiation, migrations, replication, and network reconfiguration will result in large overhead.

Cloud-based system (Xu et al., 2015)	- This approach helps in overcoming the limitations of vehicles for enabling advanced services. - It supports a larger network of up to eight times in size, compared with the one using the conventional cloud.	- Large communication latency and packet losses caused by connectivity discontinuity will make information provided by the cloud unusable.
A secure incentive based architecture for VC (Lim et al., 2015)	- This architecture encourages vehicles to contribute their underutilized resources to the cloud by issuing tokens which can be used by the vehicles to get services from the cloud. - Token transaction is secure and robust against attacks. - Integrity and authenticity of the messages exchanged between entities are ensured. - Privacy of vehicles is protected.	- If the RSU fails, data aggregated may be lost and will not be delivered in time.

Table II: Merits and demerits of temporary cloud architectures.

2.3 Hybrid cloud

A hybrid cloud is a combination of temporary cloud and permanent cloud. Vehicles can access permanent cloud as well as temporary cloud formed by vehicles to accomplish a specific task. The permanent cloud (stationary cloud) provides support for various software applications, computing and processing capabilities (e.g., storage devices, processors, servers, etc.) to vehicles. The temporary cloud using vehicular resources such as OBUs provides sensing information as a service, and in addition provides services such as support for communication infrastructure. So, a hybrid cloud benefits from both temporary and permanent clouds and provides the users better service.

Bitam et al. (Bitam, Mellouk, and Zeadally, 2015) proposed the VANET-CLOUD model to improve traffic safety and provide services to drivers. Their proposal uses both permanent and temporary clouds. Permanent cloud, which consists of stationary nodes (e.g., servers, workstations, etc.) offers cloud services such as software as a service (SaaS), infrastructure as a service (IaaS), and platform as a service (PaaS) to vehicles. In the VANET-CLOUD, the temporary cloud consists of vehicles that have computing resources (e.g., OBUs) installed on it and these vehicles together form a cloud. Their VANET-Cloud model consists of three layers. The client layer, formed by end users (an end user might be a general customer) uses communication and computing devices such as smartphone, laptop, OBU, and GPS and the end user can initiate his/her service request through a service access point (SAP). The second layer is a communication layer that connects client layer with cloud layer. This layer consists of several communication devices and networks such as VANETs, 3G/4G networks, cellular base stations, RSUs, and so on. The third layer is the cloud layer, which consists of stationary cloud and temporary cloud. The interconnection between permanent and temporary VANET-Clouds is enabled by a network consisting of all data centers of both VANET-Clouds. Therefore, the provider is responsible for managing and controlling the merged network using different networking techniques and protocols. For example, each vehicle in the temporary VANET-Cloud can access the permanent VANET-Cloud, and each node in the permanent VANET-Cloud can establish a connection with temporary VANET-Cloud nodes leading to a global network controlled by the service provider. As a result, a large and flexible vehicular cloud can be formed to serve many types of end users.

The three layer architecture (V-Cloud) proposed by Abid et al. (Abid, Phuong, Wang, Lee, and Qaisar, 2011) combines in-car vehicular cyber-physical systems, vehicle-to-vehicle network (V2V) and vehicle-to-infrastructure network (V2I) layers to help improve the safety and comfort of the drivers. The in-car layer consists of two types of sensors, which are the vehicles internal physical sensors and smart-phone embedded sensors. Smartphones monitor health and mood conditions of the driver through embedded sensors and send the collected information to cloud. The cloud will store this information, which can help predict the mood of drivers early. Vehicles in V2V network organize themselves into clusters. Each cluster has a cluster head to send all the information to other vehicles in the cluster as well as to neighboring cluster heads. Each cluster head will identify whether it is near any access point or not in order to transmit their cluster needed information to cloud computing environment. In V2I, vehicles connect with cloud through RSUs.

Chaqfeh et al. (Chaqfeh, Mohamed, Jawhar, and Wu, 2016) presented a model for vehicular cloud data collection for ITSs to provide route guidance and navigation alternatives based on the information about road conditions. This model consists of three phases: In phase1, a requesting vehicle broadcasts a route request (RREQ) to its desired region of interest (ROI) through one hop neighbors. In phase2, when the vehicles at the desired ROI receive RREQ, they cooperate to collect the desired data like environment condition data and sensed vehicular data (e.g., speed or distance). Vehicles at ROI form a vehicular cloud, and every vehicle competes to be a broker and the roadside unit (RSU) manages the process of broker election based on the connectivity criteria. When a broker is elected, the broker will collect the desired data from the members and then the broker sends it to a server in the internet cloud if further processing is required. In the case of simple request, VC resources may be sufficient to send a response. In complex cases, the broker communicates with the Internet cloud to allocate the required computing resources. In phase3, a route reply message (RREP) is created by the broker and then sent to the requesting vehicle. In case of simple requests like traffic conditions, the request would be processed in vehicular cloud. But in complex cases like finding alternate routes to avoid traffic congestion, the server from the Internet cloud would send a response to the broker. Fig 3 presents an architecture of Hybrid Cloud. Table III summarizes the merits and demerits for hybrid cloud architectures discussed in this section.

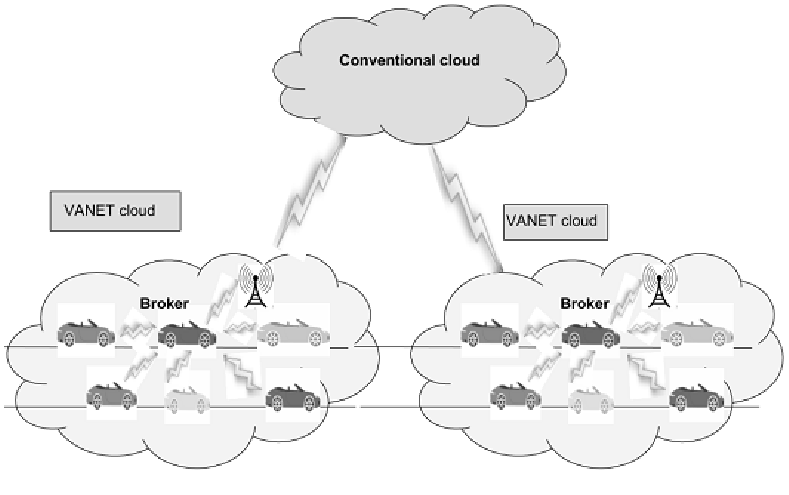

Fig 3. Hybrid cloud that combines VANET cloud and conventional cloud (Hussain, Son, Eun, Kim, and Oh, 2012)

Hybrid cloud architectures	**Merits**	**Demerits**
VANET-CLOUD (Bitam et al., 2015)	- This model benefits from the computing capabilities of vehicles that support processing, storage, as well as sensing to extend traditional cloud computing capabilities.	- Mobility of vehicles in temporary cloud can affect the performance of VANET-Cloud applications. - They did not address the security and privacy issues.
V-cloud (Abid et al., 2011)	- Combines the concept of VANET, CPS and Cloud Computing to provide safety and comfort for driver.	- Security and privacy issues are not addressed.
Merging VANET with cloud computing (Hussain et al., 2012)	- Authors suggest a concrete VANET cloud architecture to use underutilized on-board computing and communication units of vehicles.	- They did not address authentication, security and privacy issues.
VC data collection for ITSs (Chaqfeh et al., 2016)	- Provides route guidance and navigation alternatives based on the information about road conditions.	- If the broker fails, data aggregated maybe lost, and the cloud will not be able to do the intended functionalities.

Table III: Merits and demerits of hybrid cloud architectures.

3. CHALLENGES IN IMPLEMENTING VEHICULAR CLOUD AND SOME PROPOSED SOLUTIONS TO MEET THESE CHALLENGES

In this section, we highlight some of the challenges in implementing VC. These challenges include authenticity and integrity of messages, preserving the privacy of vehicles/drivers, handling selfish nodes, routing and data processing, virtualization, etc. Next, we describe these challenges and present a critical comparison of the solutions proposed in the literature addressing these challenges.

3.1 Authentication, integrity and privacy preservation

Authentication is one of the important requirements in vehicular cloud. A receiver should be able to verify that a transmitted message has been sent by an authentic member. For example, a single vehicle can claim to be one of hundred vehicles in order to give wrong information about congested road. Sharma et al. (Sharma, Bali, and Kaur, 2015) proposed dynamic key based authentication scheme for vehicular cloud computing for mutual authentication of senders and receivers.

Message integrity ensures that an intruder is not able to modify a message. Some of the driver information that needs to be protected from the intruders is driver identity, trip path and speed (Samara, Al-Salihy, and Sures, 2010). So when messages are disseminated, the authenticity of the vehicles that are disseminating messages should be verified, the integrity of the messages should be guaranteed and privacy of vehicles should be preserved. Raya et al. (Raya, Papadimitratos, and Hubaux, 2006) used a set of anonymous keys to ensure privacy; these keys are changed frequently and each key can be used once only and expires after its usage. These keys are stored in the vehicles tamper proof device (TPD). The TPD is responsible for all the operations related to key management and usage. Each key is certified by the issuing Certificate Authority (CA) and has a short lifetime.

Lin et al. (Lin, Sun, Ho, and Shen, 2007) proposed a secure and privacy preserving protocol for VANETs. This scheme provides privacy to ensure safety of the drivers. Group signatures are used to secure communication between vehicles and ID based signatures are used to secure communication with roadside units (RSUs). There are two types of mangers, the membership managers which provide security and system parameters to RSUs and send private and group keys to these units and the traffic managers which are responsible for collecting information when identities of vehicles need to be revealed. When a vehicle is determined as compromised, that vehicle will be excluded from the system and new group and private keys are generated for the remaining safe vehicles and are sent out.

Lim et al. (Lim et al., 2015) proposed secure incentive-based architecture for vehicular cloud. They used hash-based digital signature along with public key encryption to ensure the integrity and the authenticity of messages. When a message is sent, the sender attaches the digital signature to the message. The digital signature is made by encrypting the hash of the message using senders private key. The service provider manager (SPM) is connected to the trusted authority (TA) in the cloud. The TA helps the SPM in authenticating the sender. If the hash in the digital signature matches with the hash of the message calculated by the receiver, then the receiver is able to verify the authenticity and integrity of the message, but if the hash of the message does not match, the message is discarded upon arrival, so integrity is guaranteed. Also, the privacy of vehicles are protected by assigning unique pseudo IDs for each vehicle. These pseudo IDs are used to protect all private information and real identities of vehicles. If a malicious node is detected, the real ID of the malicious vehicle is revealed by the TA to the authorities for legal investigation.

3.2 Selfish nodes problem

In a VC, some vehicles, called selfish nodes, may not contribute their resources or collected phenomena to the cloud but only would like to exploit the resources of other vehicles; some schemes were designed to encourage selfish nodes to join the VC and contribute their resources. Lim et al. (Lim et al., 2015) proposed a secure architecture for vehicular cloud to encourage vehicles to participate in the cloud by contributing their underutilized resources to the cloud. In this method, tokens are given to the vehicles on reward to participate in the cloud and they can use it to get services from the cloud.

Incentive framework for vehicular cloud on the road was proposed by Kong et al. (Kong, Lu, Zhu, Alamer, and Lin, 2016) to encourage vehicles to contribute their under-utilized on-board resources to VC. It consists of the following types of entities: task server, RSU, leader vehicle, and vehicles. The task server selects a leader for vehicles and the leader works as the controller of VC. The task server is responsible for searching for on-board resources of the intelligent vehicles, manages the registration of vehicles, distributes the keys to the vehicles, and also maintains accounts for the registered vehicles. RSUs serve as gateway between the task server and vehicles and collect and transmit the related information generated by the vehicles to the task server and also send the messages from the task server to vehicles within their transmission range. The leader vehicle selects vehicles to collaborate for completing tasks and is responsible for organizing the on-board resources of the vehicles, publishing task to the vehicles, assigning task to each vehicle, calculating payments for the participating vehicles and collecting results after vehicles complete the tasks allocated to them. After vehicles complete the tasks, the leader vehicle sends the vehicles rewards to the task server through its closest RSU. Then the task server updates the rewards for each vehicle in its account.

3.3 Vehicular cloud management

Change in the number of resource providers (vehicles) over time affects cloud management. For example, some vehicles are parked in parking lot for several days, and if their owners agree to rent their resources, these vehicles need to be plugged into a power outlet to share their resources. The available resources can change depending on the arrival and departure of vehicles. A vehicle can leave the parking lot while its resources are being used by some applications. So, the issue is how to take into account the unpredictable nature of the vehicles arriving and departing parking lots to schedule resources and assign computational tasks to the various vehicles in the vehicular cloud. Other issues that may affect stability of the cloud are vehicle's velocity, broken V2V and V2I communications due to interferences and obstacles (Mekki, Jabri, Rachedi, and ben Jemaa, 2017).

In (Arkian et al., 2015; Bravo-Torres, Ordonez-Morales, Lopez-Nores, Blanco-Fernandez, and Pazos-Arias, 2014), the authors propose several clustering methods for electing a cloud controller or a set of leaders to address the above challenges in VANET clouds. Authors in (Arkian et al., 2015) have proposed a clustering technique to solve the resource limitation problem. Their method groups vehicles and vehicles in each group cooperate with each other to contribute their resources. Since some applications require large amounts of data to upload, download, and store, these applications need more storage and computation resources. Furthermore, they assume that all vehicles are equipped with a positioning system like GPS to get information about its location. In this method, vehicles form clusters and the ones that are more appropriate become cluster heads (CH). The CH is responsible for the creation, maintenance, and deletion of a vehicular cloud. All vehicles will register their resources with the CH and cloud resources are scheduled by CH. If a vehicle needs some resources of the vehicular cloud, it asks CH and CH selects the best cluster member (vehicle with sufficient unutilized resources) from the cluster to complete the requested service.

3.4 Routing and data processing in cloud

Route selection is an essential factor to avoid congested routes and transfer the data to the destination in a reasonable amount of time. In vehicular cloud, the selection of paths is challenging due to high mobility of vehicles. Selection of an entity for data processing is also challenging. Processing data can be done inside vehicles, infrastructure equipment (e.g., RSUs) or conventional cloud.

Kumar et al. (Kumar, Gollakota, and Katabi, 2012) proposed a cloud-assisted design for autonomous driving, which allows the cloud to access sensor data from autonomous vehicles as well as RSUs to assist autonomous cars in planning their routes. Moreover, it assists vehicles with determining efficient routes avoiding obstacles such as road work, accidents, traffic jam, etc. The cloud records the current route of all the vehicles and collects information about all obstacles. Then, the cloud sends alternate routes avoiding the obstacles. Thus, the cloud can assist vehicles in determining efficient routes. Authors in (Qin, Huang, and Zhang, 2012) proposed VehiCloud architecture to provide routing service for vehicular networks. Vehicles in VehiCloud monitor certain conditions in certain areas as well as predict their future locations. This information is sent to the Cloud decision module through terminals (e.g., RSUs), which is responsible for making the routing decision. As a result, the cloud can predict future traffic information by collecting the trajectory information of vehicles.

Wang et al. (Wang, Cho, Lee, and Ma, 2011) proposed a body area sensor network (BASN) formed by the vehicular devices and sensors attached to the drivers for measuring bio-medical information. It consists of a set of layers that process the collected data before making a suggestion to the driver. In the repository layer, data can be classified into low level context (e.g., temperature, blood-pressure) and high level context (e.g., gesture, activity) based on pre-processing techniques. In the knowledge processing layer, low level and high level context will be processed using techniques such as data mining, reasoning, k-means clustering, etc. Finally, some outputs will be delivered through the context-aware middle layer to the upper actuator layer and some actions will be taken like stop the car, turn slow and send alarm signals to the driver.

3.5 Virtualization in vehicular cloud

A vehicular cloud is a dynamic environment due to the mobility of vehicles; moreover, vehicular clouds have limited computing and storage capacity compared to traditional cloud. Thus, virtual machine (VM) management seems to be a challenge in vehicular clouds. For example, when the vehicles in a parking lot are used as data center, it is possible to store the data of the customers in a vehicle temporarily. This data needs to be moved from the devices before the vehicle leaves the parking lot. Therefore, the VC requires a virtual machine to manage the physical devices used to process or store data (Whaiduzzaman, Sookhak, Gani, and Buyya, 2014).

Yu et al. (Yu, Zhang, Gjessing, Xia, and Yang, 2013) studied cloud resource allocation and VM migration for effective resource management in cloud-based vehicular networks. They presented different scenarios of VM migration due to vehicles movement. Fig. 4 illustrates different VM migration scenarios. In the first case, when vehicle A moves from the coverage area of RSU-1 to RSU-2, a VM migration is needed. Since RSU-1 and RSU-2 connect to different cloudlets, guest VM-A should be transferred from roadside cloudlet-1 to roadside cloudlet-2. After that, A will access cloudlet-2 via RSU-2 to resume its service. In second case, when vehicle A moves from the coverage area of RSU-1 to RSU-2. Since these two RSUs connect to the same roadside cloudlet, there is no need for VM migration. However, radio handoff from RSU-1 to RSU-2 may still take a short period. In the third case, vehicle A moves from the coverage area of RSU-2 to RSU-1. Before As movement, nodes A, C and D have connection in an ad hoc manner. Vehicle C access the roadside cloud through vehicle A. The movement of A will cause the disconnection of C from the roadside cloud. In this case, guest VM-C will be transferred from the roadside cloud to the vehicle cloud in D. Then, vehicle C can continue its service through D. The last case is similar to that of case three, except that there is no direct link between vehicles C and D. In this case, guest VM-C has to be migrated from the roadside cloud to the central cloud. After that, C will access the central cloud to resume its service using long distance communications such as 3G/4G cellular networks.

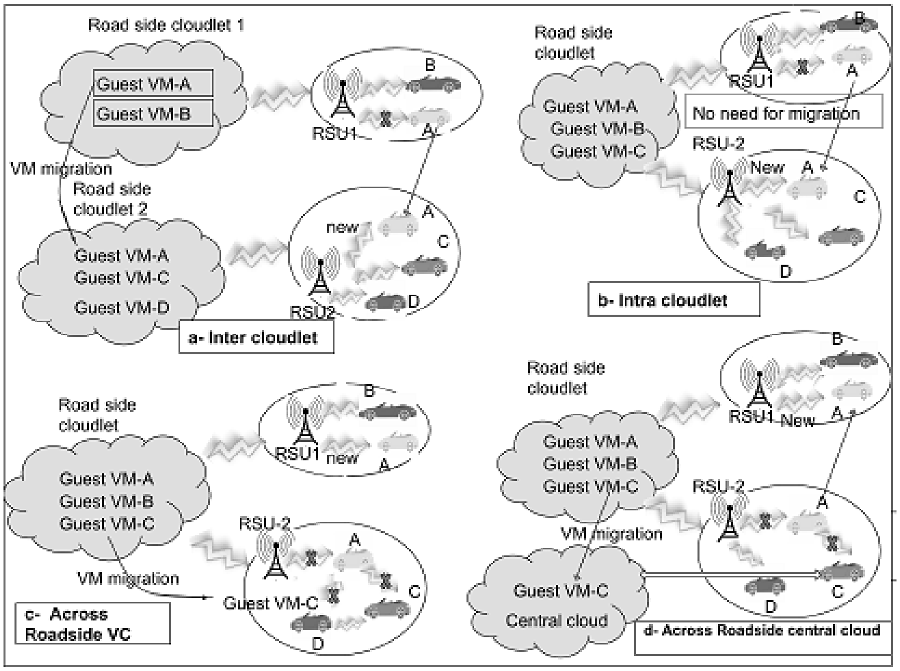

Fig. 4. Virtual machine migration scenarios
(Yu et al., 2013).

The VM migration scheme proposed by Reffat et al. (T. Refaat and Mouftah, 2014) for vehicular cloud works as follows: The source node chooses a destination node depending on the search criteria. If the destination node does not have enough resources to host a VM or if the VM cannot be migrated to the destination node in a pre-defined time window, migration is retried by excluding that destination. Otherwise, the VM is migrated to the destination node. After a certain number of migration attempts fail, migration is marked as unsuccessful. If migration is unsuccessful, the VM is directed to the RSU. The authors proposed two methods against random selection of the destination node. The first one is the vehicular virtual machine migration with least workload (VVMM-LW) and the second one is vehicular virtual machine migration with mobility-awareness (VVMM-MA). The first approach selects the vehicle with the lightest workload among the vehicles that are predicted to remain in the network as the destination node to migrate the virtual machine. The second approach uses the vehicle's routes and approximations of the future locations of all vehicles in the vehicular cloud and excludes the vehicles that are predicted to go off the network.

3.6 Context awareness in vehicular cloud

Context-aware information can provide more convenience and safety for drivers and passengers. For example, a context-aware service could be a live video of a planned route for the driver or real-time traffic update (Toutain, Bouabdallah, and Zemek, 2011). Sultan et al. (Al-Sultan, Al-Bayatti, and Zedan, 2013) proposed a context-aware driver behavior detection system to detect irregular behavior of drivers and notify other drivers on the road to prevent accidents from happening. This architecture is divided into three phases, namely, sensing, reasoning, and acting. In the sensing phase, the system collects information about the driver, the vehicle's state, and environmental changes. The reasoning phase involves reasoning about uncertain contextual information to get the behavior of the driver. They designed a Dynamic Bayesian Network (DBN) model to perform probabilistic reasoning to infer the behavior of the driver. This model combines information collected from different sensors capturing the driver's behavior and uses probabilistic inference to get the driver's current driving style. The driver and other vehicles are then alerted by triggering an in-vehicle alarm and by sending warning messages containing corrective actions to other vehicles in the VANET.

The multi-layer context-aware vehicular cloud architecture proposed by Wan et al. (Wan et al., 2014) has the following three layers: Vehicular computational layer, location computational layer, and cloud computational layer. In vehicular computational layer, context-aware driver behavior detection system is implemented. This system communicates with other vehicles to share context-aware road and safety information. The location computational layer uses the RSUs deployed at specific locations on the road to exchange information with OBUs. The cloud computational layer provides context-aware cloud services through interconnected clouds of automotive multimedia content cloud, traffic authority cloud, location based service cloud, automotive manufacturer cloud, and other application clouds. Authors describe cloud-assisted parking services that address traditional parking garage scenario for drivers. The context information of each parking space detected by sensors is forwarded to the traffic cloud through wireless sensor network (WSNs), 3G communication networks and the Internet. The collected data are processed in the cloud and then selectively transmitted to the drivers. This is helpful for providing more convenience services and evaluating the utilization levels of the parking garage. Table IV summarizes the merits and demerits of the solutions discussed above.

Proposed solutions	Issues addressed	**Merits**	**Demerits**
GSIS (Lin et al., 2007) Secure incentive-based architecture (Lim et al., 2015)	Authentication, integrity and privacy preservation Authentication, integrity and privacy preservation	- This scheme ensures privacy of the vehicles - Their scheme ensures source authentication, message integrity and privacy preservation.	- Cryptographic operations can cause large overhead. - This scheme assumes service providers are trustworthy.
Secure incentive-based architecture (Lim et al., 2015)	Selfish nodes problem	- Encourages vehicles to participate in the cloud by giving them tokens as reward which they can use to get services from the cloud.	- If the RSU fails, the received and collected data by the RSU can be lost.
A secure and privacy-preserving incentive framework for VC (Kong et al., 2016)	Selfish nodes problem	- Vehicles can earn payments for participating in and completing the accepted tasks.	- This proposed method didn't take into account the security issues related to guaranteeing the availability of the incentive mechanism. - The leader vehicle is bottleneck because it is responsible for all tasks. - If the leader vehicle fails, data aggregated could be lost.
A cluster-based VC with learning-based resource management (Arkian et al., 2015)	Vehicular cloud management	- This scheme uses clustering technique to provide resources cooperatively. - CH chooses a vehicle that has sufficient resources to complete a requested service.	- The resource allocation algorithm may cause service delay. - CH is bottleneck because it is responsible for all tasks. - If the CH fails, data aggregated is lost.
A cloud-assisted design for autonomous driving (Kumar et al., 2012)	Routing and data processing	- This scheme enables autonomous cars to plan safer and more efficient paths by sharing their sensor information with the cloud.	- Autonomous vehicles often require accurate localization. This is not addressed.
Cloud computing facilitated routing in vehicular networks (Qin et al., 2012) BASN (Wang et al., 2011)	Routing and data processing Routing and data processing	- Provide routing service for vehicles in the network. - Provide some real time services based on cloud computing techniques. - BASN, with context-aware reasoning and knowledge processing techniques, can improve drivers' safety and comfort.	- Security issues are not addressed. - Security issues are not addressed.

Toward cloud-based vehicular networks with efficient resource management (Yu et al., 2013)	Virtualization in VC	- They studied different scenarios to migrate VMs to provide services to vehicles.	- Central cloud has sufficient cloud resources but end-to-end communications delay can be large.
Dynamic VM migration in a VC (T. Refaat and Mouftah, 2014)	Virtualization in VC	- Aims to handle frequent changes in VC topology efficiently. - Increased fairness in vehicle capacity utilization across VC.	- Does not migrate the workload to multiple destinations simultaneously to maximize chances of success.
Context-aware driver behavior detection system (Al-Sultan et al., 2013)	Context awareness in VC	- Supports improving road safety. - Helps detecting irregular behaviors of drivers and notify other drivers on the road to prevent accidents from happening.	- Does not suggest appropriate corrective actions for other vehicles on the road.
Context-aware vehicular cyber-physical systems (Wan et al., 2014)	Context awareness in VC	- Helps in improving road safety and traffic management.	- Security issues are not addressed.

Table IV: Merits and Demerits of proposed solutions

3.7 Drawbacks of the proposed solutions and open issues

In this section, we present the drawbacks of the proposed solutions and open issues that need further investigation in vehicular cloud environment.

Temporary cloud architectures suggest clustering techniques to solve the resource limitation problem by encouraging vehicles to share resources; but this technique has some drawbacks. For example, a designated vehicle, called broker, is responsible for managing the cloud as well as collaborating with other cloud members for data processing. Computing resource capacity in temporary cloud is limited because it is a collection of mobile resources. Another problem is that the broker is a possible bottleneck because it is responsible for all tasks, so if it fails, data aggregated is lost and will not be delivered to the respective destinations. The selection criteria of the cloud members need to be addressed as well.

Mobility of vehicles can result in large communication latency and/or packet loss which will make information provided by the cloud unhelpful; this could affect autonomous vehicles which often require accurate localization. These issues need to be addressed in Vehicular clouds as well. Moreover, many of the solutions proposed and discussed in this paper do not address the security and privacy issues. These issues are important to avoid unauthorized access to the vehicular resources and to ensure message integrity as well as privacy preservation.

Vehicular clouds that interact with other clouds such as sensor clouds could provide new services to users. For example, a sensor cloud can detect earthquakes, volcano eruptions, fires, etc. and share this information with Vehicular clouds so vehicles can make informed decision.

4. CONCLUSION

In this paper, we classified the vehicular cloud architectures into three categories: temporary cloud which consists of vehicles that form a cloud together temporarily by sharing their resources, permanent cloud which processes the information collected from the vehicles and provides services to drivers like conventional cloud, and hybrid cloud which is a combination of temporary cloud and permanent cloud. We discussed the advantages and disadvantages of the different architectures. Then, we discussed the various issues related to implementing vehicular cloud, namely, authentication, integrity, privacy preservation, dealing with selfish nodes, routing, data processing, virtualization, and context aware information exchange. In addition, some proposed solutions that meet these challenges have been presented. Finally, we discussed drawbacks of these proposed solutions and open issue related to vehicular cloud.

References

Abid, H., Phuong, L. T., Wang, J., Lee, S., and Qaisar, S. 2011. V-cloud: vehicular cyber-physical systems and cloud computing. In *Proceedings of the 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies.* ACM, ACM, 1–5.

Al-Sultan, S., Al-Bayatti, A., and Zedan, H. 2013. Context-aware driver behavior detection system in intelligent transportation systems. *IEEE Transactions on Vehicular Technology 62,* 9 (November), 4264–4275.

Arkian, H., Atani, R., Diyanat, A., and Pourkhalili, A. 2015. A cluster-based vehicular cloud architecture with learning-based resource management. *The Journal of Supercomputing 71,* 4 (April), 1401–1426.

Bernsen, J. and Manivannan, D. 2008. Greedy routing protocols for vehicular ad hoc networks. In *Proceedings of Wireless Communications and Mobile Computing Conference.*

Bitam, S., Mellouk, A., and Zeadally, S. 2015. VANET-CLOUD: A generic cloud computing model for vehicular ad hoc networks. *IEEE Wireless Communications 22,* 1 (February), 96–102.

Bravo-Torres, J., Ordonez-Morales, E., Lopez-Nores, M., Blanco-Fernandez, Y., and Pazos-Arias, J. 2014. Virtualization in VANETs to support the vehicular cloud experiments with the network as a service model. In *IEEE Proceedings of Third International Conference on Future Generation Communication Technology.* IEEE, 1–6.

Chaqfeh, M., Mohamed, N., Jawhar, I., and Wu, J. 2016. Vehicular cloud data collection for intelligent transportation systems. In *Proceedings of IEEE Smart Cloud Networks and Systems.* IEEE, 1–6.

Eltoweissy, M., Olariu, S., and Younis, M. 2011. Towards autonomous vehicular clouds. *EAI Endorsed Transactions on Mobile Communications and Applications 1,* 1 (September), 1–11.

Hussain, R. and Oh, H. 2014. Cooperation-aware vanet clouds: Providing secure cloud services to vehicular ad hoc networks. *Journal of Information Processing Systems 10,* 1, 103–118.

Hussain, R., Son, J., Eun, H., Kim, S., and Oh, H. 2012. Rethinking vehicular communications: merging VANET with cloud computing. In *Proceedings of the 4th International Conference on Cloud Computing Technology and Science.* IEEE, IEEE, 606–609.

Kong, Q., Lu, R., Zhu, H., Alamer, A., and Lin, X. 2016. A secure and privacy-preserving incentive framework for vehicular cloud on the road. In *Proceedings of Global Communications Conference (GLOBECOM).* IEEE, 1–6.

Kumar, S., Gollakota, S., and Katabi, D. 2012. A cloud-assisted design for autonomous driving. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing.* ACM, 41–46.

Lee, E., Lee, E.-K., Gerla, M., and Oh, S. Y. 2014. Vehicular cloud networking: architecture and design principles. *IEEE Communications Magazine 52,* 2 (February), 148–155.

Lim, K., Abumuhfouz, I., and Manivannan, D. 2015. Secure incentive-based architecture for vehicular cloud. *Springer Lecture Notes in Computer Science 9143*, 361–374.

Lim, K. and Manivannan, D. 2016. An efficient protocol for authenticated and secure message delivery in vehicular ad hoc networks. *Vehicular Communications 4*, 30–37.

Lin, X., Sun, X., Ho, P.-H., and Shen, X. 2007. GSIS: A secure and privacy-preserving protocol for vehicular communications. *IEEE Transactions on Vehicular Technology 56,* 6 (November), 3442–3456.

M. Salahuddin, A. A.-F. and Guizani, M. 2015. Software-defined networking for RSU clouds in support of the internet of vehicles. *IEEE Internet of Things Journal 2,* 2 (April), 133–144.

Mekki, T., Jabri, I., Rachedi, A., and ben Jemaa, M. 2017. Vehicular cloud networks: Challenges, architectures, and future directions. *Vehicular Communications 9*, 268–280.

Olariu, S., Khalil, I., and Abuelela, M. 2011. Taking VANET to the clouds. *International Journal of Pervasive Computing and Communications 7,* 1 (February), 7–21.

Qin, Y., Huang, D., and Zhang, X. 2012. Vehicloud: Cloud computing facilitating routing in vehicular networks. In *Proceedings of IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 1438–1445.

Raya, M., Papadimitratos, P., and Hubaux, J. 2006. Securing vehicular communications. *IEEE Wireless Communications 13,* 5 (October), 8–15.

Samara, G., Al-Salihy, W., and Sures, R. 2010. Security analysis of vehicular ad hoc networks. In *Proceedings of Network Applications Protocols and Services*. IEEE, 55–60.

Sharma, M., Bali, R., and Kaur, A. 2015. Dynamic key based authentication scheme for Vehicular Cloud Computing. In *Proceedings of 2015 International Conference on Green Computing and Internet of Things*. IEEE, 1059–1064.

T. Refaat, B. K. and Mouftah, H. 2014. Dynamic virtual machine migration in a vehicular cloud. In *Proceedings of IEEE Symposium on Computers and Communication (ISCC).*

Toutain, F., Bouabdallah, A., and Zemek, R. 2011. Interpersonal context-aware communication services. *IEEE Communications Magazine 49,* 1 (January), 68–74.

Wan, J., Zhang, D., Zhao, S., Yang, L., and Lloret, J. 2014. Context-aware vehicular cyber-physical systems with cloud support: architecture, challenges, and solutions. *IEEE Communications Magazine 52,* 8 (August), 106–113.

Wang, J., Cho, J., Lee, S., and Ma, T. 2011. Real time services for future cloud computing enabled vehicle networks. In *Proceedings of International Conference on Wireless Communications and Signal Processing (WCSP)*. IEEE, 1–5.

Whaiduzzaman, M., Sookhak, M., Gani, A., and Buyya, R. 2014. A survey on vehicular cloud computing. *Journal of Network and Computer Applications 40*, 325–344.

Xu, K., Wang, K., Amin, R., Martin, J., and Izard, R. 2015. A fast cloud-based network selection scheme using coalition formation games in vehicular networks. *IEEE Transactions on Vehicular Technology 64,* 11 (November), 5327–5339.

Yan, G., Wen, D., S.Olariu, and M.Weigle. 2013. Security challenges in vehicular cloud computing. *IEEE Transactions on Intelligent Transportation Systems 14,* 1 (March), 284–294.

Yu, R., Zhang, Y., Gjessing, S., Xia, W., and Yang, K. 2013. Toward cloud-based vehicular networks with efficient resource management. *IEEE Network 27,* 5 (October), 48–55.

Mr. Hassan Mistareehi is a PhD student in the Department of Computer Science in the University of Kentucky. He received his B.S and M.S. degrees in Computer Science from Jordan University of Science and Technology. His research interests are Vehicular Networks, Ad Hoc Networks, Sensor Networks, and Network Security.

Dr. D. Manivannan is currently an associate professor of Computer Science at University of Kentucky, Lexington, Kentucky, USA. He received an M.Sc degree in mathematics from University of Madras, Madras, India. He received M.S and PhD degrees in computer and information science from The Ohio State University, Ohio, in 1993 and 1997 respectively. He published his research work in the following areas: fault-tolerance and synchronization in distributed systems, routing in wormhole networks, routing in ad hoc networks and vehicular ad hoc networks, channel allocation in cellular networks, wireless personal area networks and sensor networks. Dr. Manivannan has published more than 60 articles in refereed International Journals (a vast majority of which were published by IEEE, ACM, Elsevier, and Springer) and Proceedings of International Conferences.
He served as an Associate Editor of IEEE Transactions on Parallel and Distributed Systems and IEEE Communications Magazine. He served as Program co-chair of three International Conferences in the areas of reliable distributed systems and wireless networks and served as program committee member for over 40 International Conferences. He is on the Editorial Board of Information Sciences journal and Wireless Personal Communications journal. He served as reviewer for more than 30 International Journals published by ACM, IEEE, Elsevier, Springer, Oxford University Press, Taylor and Francis and others. He also served on several proposal review panels of US National Science Foundation and as external tenure reviewer for other Universities. Dr. Manivannan's research has been funded by grants from the US National Science Foundation and the US Department of Treasury.
Dr. Manivannan is a recipient of the Faculty Early Career Development award (CAREER award) from US National Science Foundation. He is a senior member of the IEEE and ACM.