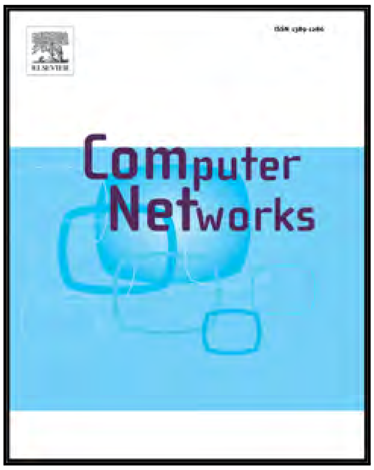
Accepted Manuscript



Vehicular Cloud Computing: Architectures, Applications, and Mobility

Azzedine Boukerche, Robson E. De Grande

PII: S1389-1286(18)30005-7
DOI: 10.1016/j.comnet.2018.01.004
Reference: COMPNW 6352

To appear in: *Computer Networks*

Received date: 28 July 2017
Revised date: 4 January 2018
Accepted date: 8 January 2018

Please cite this article as: Azzedine Boukerche, Robson E. De Grande, Vehicular Cloud Computing: Architectures, Applications, and Mobility, *Computer Networks* (2018), doi: 10.1016/j.comnet.2018.01.004

Vehicular Cloud Computing: Architectures, Applications, and Mobility

Azzedine Boukerche

School of Electrical Engineering and Computer Science
University of Ottawa
Canada

Robson E. De Grande

Department of Computer Science
Brock University
Canada

Abstract

Intelligent transportation systems are designed to provide innovative applications and services relating to traffic management, as well as to facilitate the access to information for other systems and users. The compelling motivation for employing underutilized onboard resources for transportation systems and the advancements in management technology for Cloud computing resources has promoted the concept of Vehicular Clouds. This work gathers and describes the most recent approaches and solutions for Vehicular Clouds, featuring applications, services, and traffic models that can enable Vehicular Cloud in a more dynamic environment. We have considered a large number of applications and services that showed relevance in the scope of the transportation system, benefiting its management, drivers, passengers, and pedestrians. Nevertheless, the high traffic mobility imposes as a significant challenge in implementing a Vehicular Cloud on continually changing physical resources. The dynamics of the environment bring fundamental issues and increase the complexity of building this new type of Cloud. By analyzing the existing traffic models, we found that Vehicular Cloud computing is technologically feasible not just in the static environment, like a parking lot or garage where vehicles are stationary, but also the dynamic scenarios, such as highways or streets where vehicles move.

Keywords: Cloud Computing, Mobile Cloud Computing, Vehicular Cloud Computing, Vehicular Networks, Traffic Models

1. Introduction

The automotive industry has been showing revolutionary changes lately, investing in the incorporation of more technological features in vehicles, as well as allowing drivers to access sophisticated *smart vehicles* for daily use. Any typical current vehicle is considered a *computer-on-wheels* because it comes equipped with a powerful onboard computer, large-capacity storage device, sensitive radio transceivers, collision radars, and a GPS device. At the same time, Cloud computing [1] has presented fast-paced advancements that have allowed it to support built dynamic Clouds in mobile environments.

The flexibility that Cloud has introduced for the on-demand provision of resources and services over the Internet has allowed it to be recognized as a public utility [2]. For instance, Amazon Elastic Compute Cloud (Amazon EC2) has become the largest provider of dynamic compute capacity in the Cloud. This growth is justified by the increasing interest of businesses into elastically and scalably renting Cloud services, platforms, or software instead of building and maintaining their data centers [3]. Consequently, these allied interests further advance the development of Cloud computing, which introduces a pay-as-you-go model, scalability, resources on demand, virtualization technology, and Quality of Service (QoS) as its key features. The accelerated interests in its versatility have made Cloud computing the major technological trend in IT, with massive investments and corporate efforts to migrate their business into this new paradigm.

The benefits and challenges in the field have motivated several germinal works, promoting the introduction of a Vehicular Cloud framework [4]. This initial work abstracts the current issues and classifies the solutions as a matter of leveraging on the underutilized vehicular resources, such as network connectivity, computational power, storage, and sensing capability, which can be shared with vehicle owners. Furthermore, this framework is expected to enable business models to aggregate resources and rent them to potential consumers, similarly to the traditional Cloud infrastructure.

In this context, a vehicular Cloudlet has been presented, which consists of a group of vehicles that can share resources through vehicle-to-vehicle (V2V) communication, vehicle-to-infrastructure (V2I) communication, or even vehicle-to-everything (V2X) communication. As depicted in Figure 1, a vehicular Cloudlet can be established in the road environment (mobile Cloudlet), which consists of moving vehicles, or on a parking lot (static Cloudlet), which is relatively stationary and formed by parked vehicles [5, 6]. Consequently, each kind of Cloudlet suits a range of different Vehicular Cloud services, which might be dependent on the degree of the mobility and dynamic alterations on the disposition of Cloud elements. As examples of suitability, the highway scenario introduces very high mobility and promotes a mobile vehicular Cloudlet ideal for data dissemination services. On the other hand, vehicles waiting in still urban traffic might form a Vehicular Cloud that can well serve the execution of traffic management applications, which may identify solutions to alleviate traffic congestion. In an even more static situation, the Cloudlet composed of static parked vehicles proves to be relatively stable environment for computing tasks and storing data, providing resources for traditional Cloud services.

Evolved from Vehicular Ad-Hoc Network (VANET), Vehicular Cloud Computing (VCC) has been receiving increasing attention. VCC is a very appealing technology due to its features and capabilities in supporting a series of novel, relevant, or sensitive applications. Additionally, VCCs are designed to initiate objectives that directly match everyday transportation needs, such as enabling computational services at low cost to authorized users, reducing traffic congestion, and implementing services to improve road safety. This work is intended as a reference to provide an extensive study the most recent VCC works, aiming to introduce a better understanding of the principles of Vehicular Cloud Computing mechanisms.

As depicted in Figure 2, the existing techniques are classified according to

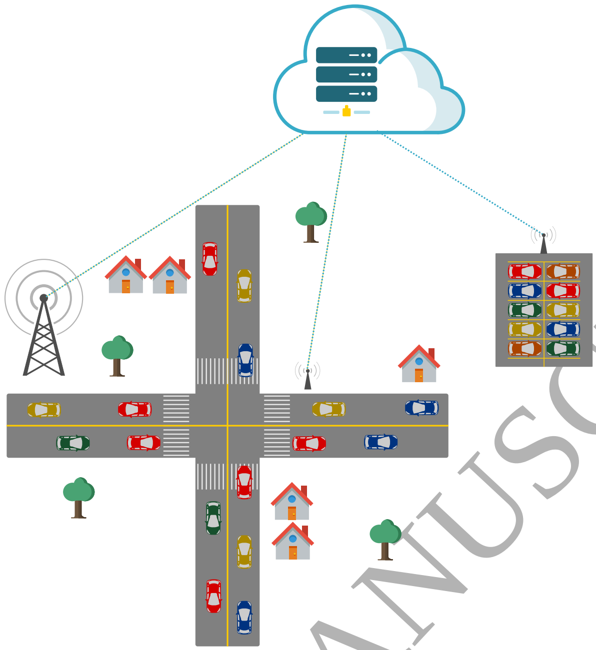


Figure 1: Vehicular Cloudlets in Static and Dynamic Scenarios

their contributions for enabling the construction of Vehicular Cloud Computing. These techniques are identified as an aggregation of two paradigms: Vehicular Networks and Cloud Computing. Vehicular Ad-hoc Networks mostly develop from Mobile Ad-hoc Networks (MANETs), in which the real communication among networking nodes is a single hop or multi-hop based. Besides that, VCC has emerged from Mobile Cloud Computing (MCC), where the primary concern consists of reaching traditional Cloud services that supplied through an infrastructure, similar to the access to telecommunications and other data-based services.

The remainder of the paper is organized as follows. Section 2 summarizes the architecture and applications of VANETs. Section 3 provides a brief introduction of traditional Cloud Computing and its key services. Section 4 offers an overview of Mobile Cloud Computing, which has similar features to the Vehicular Cloud. Section 5 discusses the taxonomy, architecture, services, and

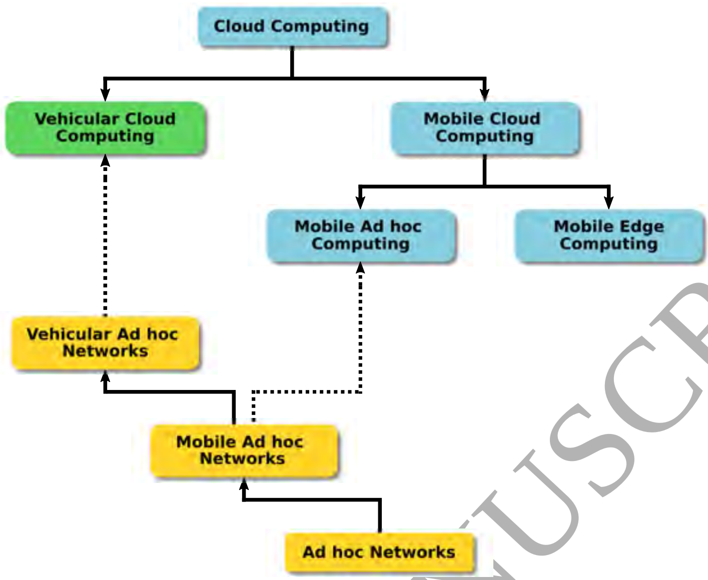


Figure 2: Paradigm Evolution towards Vehicular Cloud Computing.

applications of the Vehicular Cloud. Section 6 focuses on presenting the existing traffic models, which estimate the number of computing resources in the dynamic vehicular environment. Finally, Section 8 briefly summarizes the work and presents remarks for future research directions.

2. Vehicular Networks

On a daily basis, modern big cities face chronic transportation problems including slow traffic flow, traffic congestion, and emergencies caused by accidents. An increase in the number of vehicles has caused a growth of complexity in allowing proper traffic management and tackling transportation issues. The steep rise in the traffic volume has led cities to unsafe scenarios, unpleasant driving experiences, and unsustainable urban conditions. Usually, traffic services do not issue notifications of congestion status with enough time for preventive actions, which is worsened by the current transportation infrastructure lacking in traffic safety and efficiency. Transportation management systems have improved with the introduction of additional lanes in highways and a reduction in the number of traffic signs to enable a more fluid flow of vehicles. Nevertheless, such approaches provide a temporary solution and are ineffective for targeting the core urban issues.

On the other hand, several recent technological advancements have empowered the implementation of various traffic management services and driving applications, such as accurate control of traffic flow, cooperative monitoring of traffic, and detection of traffic hazards. All services and applications depend on, or can benefit from, the concept of connected vehicles, which cooperatively propagate and disseminate information. VANETs' design has been inspired by Mobile Ad-hoc Networks [7]; investments in this vehicular communication paradigm is a reflection of the constant growth in popularity with advancements in technologies, solutions, and support to a wide range of applications. Vehicular Networks primarily incorporate Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I), as well as Vehicle-to-Everything (V2X) communication strategies [8, 9, 10, 6].

Several initiatives have dedicated significant efforts in to employ Vehicular Networks to tackle unfortunate and unpredictable traffic hazards and counter unpleasant driving events [11]. These issues pertinent to transportation have implications for solving problems in adequately disseminating and collecting real-time data [12]. Since these new networks have enabled a unique communication environment for Intelligent Transportation Systems (ITS), Vehicular Networks currently extend this context to primarily involve safety, privacy and security interests.

2.1. Architecture of Vehicular Networks

Figure 3 presents a general architecture for VANETs and illustrates possible communication scenarios. V2V and V2I communication strategies are depicted in the figure, demonstrating the use of Wireless Access in Vehicular Environment (WAVE). The common communication strategies in Vehicular Networks are listed as follows:

- *Vehicle-to-Everything.* V2X comprises the transfer of data between a vehicle and any other element provided with networking capabilities, including any element involved in this communication [13]. This vehicular communication method is general and includes other specific approaches, such as V2I, V2V, Vehicle-to-pedestrian (V2P), Vehicle-to-device (V2D), and Vehicle-to-grid (V2G).

- *Vehicle-to-Vehicle.* V2V consists of communications among vehicles in ad-hoc mode [14]. The communication is made possible through multi-hop transmissions of vehicles that are in range, and it supports the valuable transmission of traffic-related messages, such as road status, driving conditions, and traffic accidents.

- *Vehicle-to-Infrastructure.* V2I represents the communications between vehicles and fixed network infrastructures for the exchange of information [14]. These communications, in most cases, involve the access of external networks, such as the Internet or any traditional Cloud service, through RSUs and gateways. The communication links provided in V2I introduce better security, but the bandwidth they demand is higher than for V2V links.

The spectra of the 5.9 GHz frequency band have been allocated for the Dedicated Short-Range Communication (DSRC) protocol to foster the implementation and allow the operation of V2V communications [15]. Onboard units (OBU), roadside units (RSU), and certification authorities (CA) are the principal components that constitute the environment of Vehicular Networks, considered core parts of their architecture.

- *Onboard Unit.* OBU is present in every single vehicle; it is responsible for collecting provided data and can later be used by applications and services. Embedded with a set of sensors, as well as other pieces of equipment, OBU aggregates information about the position, speed, and acceleration/deceleration of a vehicle, and it communicates the data to other vehicles or RSUs. This component also receives messages, being capable of verifying them, processing tasks, and avoiding security attacks.

- *Road-side Unit.* RSU is an essential communication component that allows communication with network infrastructure. This unit is in a fixed location and is in charge of collecting and disseminating data that might be related to traffic status, such as accidents ahead, nearest parking spots, gas stations, and length of a traffic queue. To be capable of executing its functions, an RSU presents at least one networking interface that connects it to the *Internet* and to short-range wireless communication. In this way, the component works a gateway that allows vehicles to access network infrastructure, enabling Internet connections and access to services. Also, an RSU can serve as a monitoring endpoint that collects information from roads and vehicles [5], so it is usually mounted along the road and intersections, assisting with traffic coordination.
- *Certification Authority.* CA is responsible for preventing security issues in Vehicular Networks [16]. This component is aware of the nearby vehicles, and it provides entity validation and assessment on information sources when receiving messages. A CA is a fully trusted party in the environment and is crucial for Vehicular Networks. In the architecture, commonly trusted entities serve as CAs, such as the municipal transportation department.

2.2. Comparison of Vehicular Networks and MANET

Vehicular Networks consist of an extension from MANETs, representing a subclass that focuses on the vehicular environment [17]. In the broad context of MANETs, RSUs and OBUs communicate in an ad-hoc domain where the RSUs are static, fixed nodes and OBUs are mobile nodes. With communications enabled through the IEEE 802.11p protocol, some particularities distinguish the vehicular context:

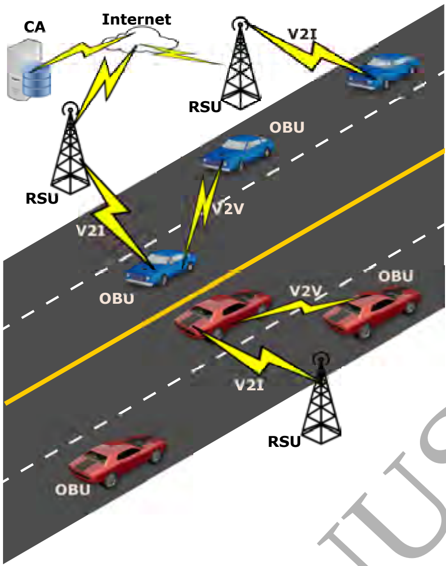


Figure 3: General Architecture of Vehicular Networks.

- *Every packet transmission in the ad-hoc network is limited by coverage distance.* A wireless node can only transmit data to and through its neighboring nodes, which forward packets beyond its coverage. Consequently, nodes in MANETs mostly need ad-hoc routing protocols for communicating, such as on-demand routing, table-driven routing, and hybrid routing. However, MANET Routing protocols are not suitable for Vehicular Networks since they cannot cope with the fast-changing ad-hoc network topology.

- *The number of nodes composing a Vehicular Network is considerably larger than in MANET.* Urban scenarios are composed of many vehicles, considering that every vehicle represents a single mobile node. Usually, as a matter of simplification, the Vehicular Network area is divided into smaller regions, allowing the design of protocols, services, and applications that better suits the local singularities. Stationary RSUs, which serve as a communication gateways, interconnect these regions.

- *The topology of the Vehicular Networks is defined according to the layout of the road network and traffic conditions.* In contrast to descriptions from MANETs, vehicles do not move arbitrarily in a region; existing road segments and traffic regulations, such as stops, traffic lights, and road directions, delimit vehicle routes. This urban traffic topology increases the predictability of mobile nodes in Vehicular Networks when compared with the MANET nodes.

- *Functional lifespan of MANET nodes is restrictive.* Power consumption is a significant concern for the design of effective solutions. In most cases of applications and services, throughput is compromised due to the limitations of energy capacity, creating bottlenecks in the system and requiring communication alternatives. On the other hand, Vehicular Networks contain nodes known as OBUs that are not prone to power limitations, as well as storage, since vehicles include batteries that supply enough power to support all needed computational and communication tasks.

- *OBU is capable of performing sophisticated and more complex computation tasks.* Without power restrictions, an OBU can receive relatively computation-intensive processes, introducing opportunities for implementing advanced services and applications.

- *Vehicular Networks are mostly ephemeral* [18]. Nodes in Vehicular Networks present high mobility because vehicles move fast on roads; this creates a network topology that changes constantly, and connectivity might benefit from the opportunism of peers and communication ranges [19]. Mobility is the most significant characteristic of Vehicular Networks and is its notable main difference against MANETs. Traffic of vehicles predominately dictates the distribution density of nodes. In this case, mobility analyses must account for behavioral factors; for instance, density is higher during rush hour and lower during night time. Also, speed and direction, major factors that constitute the movement of vehicles, are by some means considered unpredictable, causing very short *connection* time among vehicles. Thus, the design of algorithms, protocols, and systems for such networks must consider mobility as a vital factor.

2.3. Vehicular Network Applications

Massive interest and efforts to explore Vehicular Networks in traffic environments have generated numerous applications for a vast set of purposes. Such applications fit into three main classes [20, 16, 21]:

- *Applications for road safety.* The context of these applications focuses on making use of the communication capabilities that Vehicular Networking introduces as an enabler to provide critical messages. These messages can support hazard notifications, collision avoidance, and warnings, which can provide information about harsh weather conditions for the sake of road safety and traffic accident reduction.

- *Applications for traffic efficiency.* The design of these applications aims to aid drivers and the transportation management system. They provide means for better working with the road, assisting decision-making during driving, and complementing decision processes with additional information. For instance, applications can assist in warning about the possibility of traffic jams, detecting vehicles that are queuing up, or overtaking other vehicles.

- *Applications for passenger comfort.* These value-added applications prioritize the comfort of driver and passengers. They are designed to make vehicular trips more pleasant by providing onboard services, such as Internet access, messaging, and infotainment.

3. Cloud Computing

In past years, exceptional advancements have been made in Cloud Computing (CC) thanks to technological advancements in computing and networking [22, 23, 24]. Cloud Computing presented a paradigm shift in the global information technology (IT) industry. Based on [25], CC is defined as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources, such as networks, servers, storage, applications, and services, which can be rapidly provisioned and released with minimal management effort or service provider interaction. CC has introduced several novel characteristics for distributed computing solutions when compared to traditional local systems:

- CC presents on-demand provisioning of computational resources, storage, and IT services. It has infinite computing resources available to users; consequently, users are not necessarily required to plan for the supply of physical resources;

- Users have the opportunity to rent services and resources according to their requirements at a given moment. CC can offer flexible alternatives for users to "purchase" extra hardware resources only when there is an increase in their needs. This feature is helpful for small- to medium-size businesses in which significant financial commitments can be avoided, skipping the building of an IT infrastructure.

- CC further provides users with the capability to rent computational resources for a required period. As a result, users have the benefit of leasing resources based on the length of a project, and these same resources can be released after the project ends. This temporary lease facilitates budget management in which the progress of a project does not demand the purchase of physical servers.

With all the advantages that introduce flexibility for the management of resources on project changing requirements, users and enterprises have shown an increasing migration of their IT services and data to CC servers. An excellent example of this trend is Amazon Elastic Computing Cloud (EC2), which corresponds to one the largest Cloud service platforms and providers with millions of globally spread users.

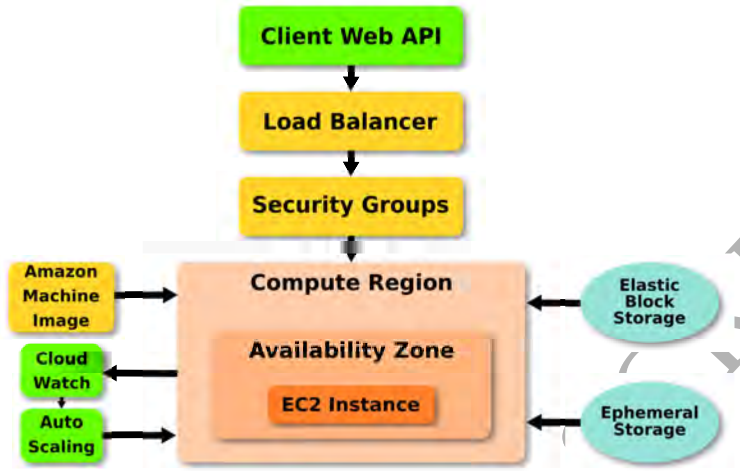


Figure 4: Architecture of Amazon EC2.

Figure 4 illustrates the general architecture of Amazon EC2 [26]. EC2 provides a web API where authorized users can access and reserve Cloud services. The platform enables access to the computing resources, EC2 instances and storage, Elastic Block Storage, and Ephemeral Storage, in an elastic manner. To reach for efficiency, a component known as Cloud Watch monitors the utilization of resources, and it reports results to the Auto Scaling system. This system is responsible for decision-making in scaling services, according to the needs of users. Cloud services have reached a certain level of maturity and are offered in three major classes, namely Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [22, 27].

3.1. IaaS: Infrastructure as a Service

A Cloud service provider makes elastic resources, such as computing capabilities, network connections, and storage capacity, available to customers through virtualization technology [28]. Amazon EC2 is a strong example of this type of Cloud service. OpenStack, on the other hand, is an open-source project that has gained considerable popularity among other software platforms for Cloud Computing, grounded on infrastructure-as-a-service provisioning aspects [29]. This kind of service contains a series of inter-related components that build their resource provision platform. In the context of OpenStack, Nova coordinates processing hardware pools, Neutron controls networking resources, Cinder is in charge of black storage, and Swift manages object storage. All these components together are indirectly accessible through web-based dashboards: Horizon, command-line tools, and RESTful APIs.

3.2. PaaS: Platform as a Service

PaaS offers the means for users to develop, run, and manage applications, providing a platform where users do not need to be concerned about the lower-level details of the environment. Consequently, the services in this category allow users to configure settings and control the deployment of software. The service provider is responsible for guaranteeing the availability of servers, networks, storage, and other support services needed to host the application for customers. Microsoft Azure [30] and Google AppEngine [31] are regular platforms that serve as representative examples of this category of services.

3.3. SaaS: Software as a Service

A Cloud service provider in the context of SaaS model offers access to a suite of software that resides in its datacenter [32]. The provider is responsible for maintaining software, licensing applications, and making software available to customers as a service on demand. Usually, the suite consists of office and messaging, Data Management System (DBMS), computer-aided design, customer relationship management (CRM), and antivirus. Working as a subscription-based service, SaaS facilitates the rental of software for a period, as an alternative to its purchase. Since the whole suite of software is made available by the same service provider, this class of services makes it possible to avoid incompatibility issues between software, which users typically experience; consequently, concerns of matching the operating system of users with software are nonexistent due to the entire service being available through Web browsers. The IBM SmartCloud [33] platform is a significant example of the SaaS model.

4. Mobile Cloud Computing

Mobile devices have grown in quantity, capabilities, accessibility, and popularity; this growth has enabled the delivery of high services and applications through them, which has promoted a new research area named Mobile Cloud Computing (MCC) [34, 35]. The designs in this new field devise mobile devices as input and output endpoints and assume that remote Cloud servers process intensive computation tasks.

The MCC forum [36] defines MCC, in its simplest terms, as an infrastructure in which mobile devices do not process data or store data within them; these tasks are performed remotely by other entities. Therefore, Mobile Cloud applications transfer the stress on data storage and burden on computing power from mobile devices to the Cloud. This feature allows applications and mobile computing to be available not only to smartphone users, but also to a broad range of mobile service users.

MCC is expected to offer significant potential for IT applications and presents unique characteristics, such as mobility, portability, and communication. Several features are unique in MCC [35]:

- *Extension of battery lifetime.* Even with several technological advancements, battery lifetime still consists of significant concerns for mobile devices. Limited, insufficient battery life conditions users to charge their devices with a high frequency and restricts the use of light-weight applications. This reduced autonomy represents an adverse characteristic of these electronic devices. Several methods have been proposed that counter this negative aspect and reduce power consumption through efficient CPU performance and smart screen and disk management. This limitation involves modifications in the structure and architecture of smart devices, which might not be even feasible. Also, rich services and applications still implicate on execution constraints in the devices, including products that have already been in use, although such techniques are capable of increasing battery lifetime. In this particular case, MCC substantially serves to offload the processing load and storage burden from mobile devices, migrating computation-intensive tasks that involve complex processing to remote Cloud servers. Concerns regarding battery lifetime have motivated the design of several offloading strategies [37, 38] mapped in intricate simulation scenarios [39].

- *Improvement of processing capability and storage capacity.* MCC is intended to increase the execution performance of applications and services and extend storage capacity. Through better performance, MCC contributes to a decrease in execution time and energy consumption. Even though the processing capabilities of devices have advanced significantly, they still present limited physical resources and are not suitable for hosting rich, complex applications. With offloading models, MCC can support the execution of such applications as offloading strategies can migrate partitioned tasks and seamlessly store large data in the Cloud through wireless networks.

- *Enhancement of reliability.* Mobile devices have a considerable benefit on improving reliability through MCC. Because of the partial or complete remote execution of applications and services, data and results are stored and replicated on remote Cloud servers. Offloading tasks considerably lowers the risk of data loss; compared to mobile devices, servers are more stable and resilient to failures. In addition to the increased security that may result from the use of the Cloud, several levels of data security models in MCC introduce more robust data protection for users and service providers. For instance, Cloud-based mobile digital rights management (DRM) strategies help to preserve the integrity of massive unstructured digital content, such as video and audio, from being inappropriately accessed and distributed in an unauthorized manner [40].

Due to its direct evolution from Cloud Computing and mobile computing, MCC have inherited several advantages [25, 35]:

- *Dynamic Provisioning.* MCC establishes resource provisioning to users under authorized access and on-demand fine-grained, self-service grounds. This feature facilitates the supply of resources and services without prior planning, flexibly promoting allocation according to the current needs of consumers. The apriori planning consequently allows service providers to implement applications in a constrained amount of resources, which increases as the system scales up to accommodate the growth of applications.

- *Ease of Access.* The online interface in MCC boosts accessibility considerably; users reach out for resources at any time and to depend on Internet access through wireless networks.

- *Ease of Integration.* Service providers are capable of distributing resources and sharing costs. This feature enables vast support to a myriad of applications in their context, meeting compatibility requirements.

The offloading of tasks and storage from mobile devices must not necessarily be conducted to Cloud servers. In the context of Mobile Edge Computing (MEC), the resources positioned at the edge of the mobile network shorten the communication path and distance between mobile devices and the Cloud [41]. This strategy allows for greater accessibility of highly demanded services and applications in mobile devices, considering the restrictive time constraints and delay requirements. Consequently, MEC allows users to avoid the excessive extra load imposed on both radio and backhaul of mobile networks, decreasing communication latency considerably through caching strategies [42]. From an architectural perspective, MCC presents an entirely centralized resource utilization approach while MEC enables a fully distributed access to resources in many locations. A three-tier architecture defines the general structure of MEC, in which Cloudlets are intermediate elements between mobile devices Cloud *centralized* servers. Decentralization permits mobile devices to connect to Cloudlets through the local mobile network, just within a hotspot area.

In more volatile and flexible scenarios where the connectivity of mobile devices do not allow continuous access to any communication infrastructure, Mobile Ad-Hoc Cloud raises as a potential solution for MCC offloading. Mobile Ad-Hoc Cloud (MAC) consists of a more particular context of MCC, where a device utilizes the available resources of neighboring mobile devices [43]. In this approach, the capabilities of a device are extended, but only through access to closer resources, avoiding delays. MAC allows a solution that does not depend on full connectivity with the infrastructure to promote MCC benefits. However, the rapidly changing mobile environment creates a lack of connectivity and justifies MAC, while also imposing several challenges in establishing a Cloud of mobile devices. As derived from MCC, these difficulties involve task offloading and scheduling, resource allocation and management, discovery, and formation of the MAC [44]. The limitation of MAC is that it requires the collaboration of nearby mobile devices to share their resources, needing incentives to motivate participation. Another crucial constraint in the MAC model consists of costly decision-making in task allocation, as well as Cloud formation and maintenance, against the benefits of offloading computing and storage to other capability-impaired devices.

5. Vehicular Cloud Computing

The automotive industry has been building considerably intelligent vehicles, which already contain robust computing, communication, storage, and energy resources. These features differentiate modern vehicles from mobile devices, which are assumed to present constrained resources. VCC, in this case, works in the *opposite direction* of the MCC paradigm. MCC attempts to leverage the capabilities of devices while VCC allows underutilized or available vehicle resources to be harvested; this extends to MEC, in which vehicles may, in a hypothetical scenario, serve as a Cloudlet to the devices of passengers. When it comes to MAC, VCC shares closer characteristics to this subset of MCC paradigm since VCC attempts to dynamically form a Cloud with the nearby vehicles as for a targeted task, application, or service. However, VCC involves a very particular set of characteristics that set it aside from MAC. VCC deals with the extreme volatility of the environment and resources, the distinct movement of vehicles, demand of applications, and sporadic availability, as well as the business model it encompasses, which allows resources to be harvested from vehicles to other elements, such as mobile devices.

Recent works [45, 4, 46] have coined the term, adequately defined the concept of, and delimited the first designs of Vehicular Cloud Computing. These works presented VCC with the principal goal that consists of gathering, allocating, and utilizing available resources on board of vehicles. These resources comprehend capabilities ranging from processing, communication, storage, and sensor, which may be dynamically *harvested* in groups of vehicles under the authorization and agreement of owners and drivers. The Cloud, drivers, passengers, and users benefit from efficiently aggregating all surrounding resources and making them effectively available as Cloud services to the public. Such VCC services are convoluted and relevant, complementing regular Cloud Computing services. As defined in its first scheme [46], VCC concerns a *a group of vehicles which are broadly autonomous and contain computing, communication, sensing, and physical resources that might be coordinated and allocated dynamically assigned to authorized users.* Table 1 introduces a brief comparison of the major characteristics of CC, MCC, and VCC.

Regarding computation capability and storage capacity, it is noticeable that the conventional Cloud servers present the most powerful local computing resources, and a modern vehicle typically includes an onboard computer, which has much greater computing capabilities than that in a mobile device. Besides, mobility is a valuable and an inherent part of the physical resources of MCC and VCC while the resources of a traditional CC are commonly located in the fixed data center. CC is also not restricted by energy consumption due to the constant power supply in the data center; similarly, VCC is not constrained by the power since a vehicle contains a large-capacity battery, and the running engine continually recharges it. However, the battery life of mobile devices confines

Table 1: Major characteristics of Cloud Computing, Mobile Cloud Computing, and Vehicular Cloud Computing

Feature	**CC**	**MCC**	**VCC**
Support to Mobile Resources	No	Yes	Yes
Computational Capability	Highest	Lowest	Medium sized
Battery Limitation	No	Yes	No
Storage Capacity	Highest	Lowest	Medium sized
Autonomous Composition	No	No	Yes
Resource Flexibility	Static	Static	Highly Dynamic
Network Architecture	Client-Server	Client-Server	Peer to Peer or Client-Server
Physical Resources	Local or Remote Servers	Local Mobile Devices or Remote Server	Local Vehicles or Remote Servers

MCC. Finally, VCC is the only type of Cloud that builds on highly dynamic physical resources, and it can be formed autonomously, depending on the traffic situation.

Modern vehicles contain computational capabilities that are relatively robust for allowing access to demanding rich services and applications. Cloud Computing is capable of expanding such capabilities through the provisioning of *limitless* resources, thus enhancing access to services. Furthermore, vehicles' built-in resources tend to be underutilized for long periods of time, such as when vehicles are parked or motionless in traffic jams. Such situations enable promising opportunities for exploring underused computing resources to assemble Clouds autonomously. By the concept of connected vehicles, surrounding elements can participate in a VCC, comprising a critical strategy that contributes to tackling complex issues in real-time and in loco; many works, such as in [46], recognize this approach as a paradigm shift in transportation systems. Vehicles' mobility and the scale of the environment require sophisticated resource management strategies to coordinate resources efficiently [47, 48, 49, 50].

Both paradigms are essential elements for the design of solutions in Intelligent Transportation Systems, including the support to an extensive range of applications and services, such as the estimation and prediction of traffic conditions in a simulation environment [51]. However, due to the high mobility of the elements that compose a VCC, the challenges that it endures with the dynamic interconnection of vehicles are very similar to the issues experienced in Vehicular Networks [52]. Even though extensive efforts have been applied to find solutions that tackle the highly dynamic changes of communication topology in Vehicular Networks, the challenges persist and are actively present in VCC, which relies entirely on communication connections among vehicles.

5.1. Taxonomy of Works on Vehicular Cloud Computing

Upon a comprehensive overview of the most relevant works related to VCC, Figure 5 summarizes the approaches in the general taxonomy of Vehicular Cloud Computing derived from previous efforts [53]. Approaches covered in this survey focus on applications and services for different categories of vehicular Cloudlets.

Due to the practical interests in the already-existing transportation infrastructure and Cloud services, VCC introduced opportunities for a vast number of potential applications [46]. Among these applications, a proposal to compose and build a data center in parking lots was notable because of its low level of complexity and implementation convenience. The applicability of this approach resides on the existence of a large number of vehicles in garages of companies or parking lots of shopping malls that rest still for a reasonable amount of time. Together with this condition, the onboard storage of these vehicles is aggregated as a fundamental element in assembling a data center.

Applications involved with the dynamic traffic management system consist of another type of VCC application [4], which directly deals with common transit issues targeting mutual benefits. Traffic congestion includes a crucial concern in metropolitan areas, and it is steadily growing because of the rising number of vehicles on the roads. The condition of being in congested traffic leads to a position where drivers might agree to share their vehicular computing resources with the traffic management system. The vehicles, in this case, can receive the partitioned load of processing calculations and running simulations; such tasks might be part of a procedure for determining alternative solutions to alleviate

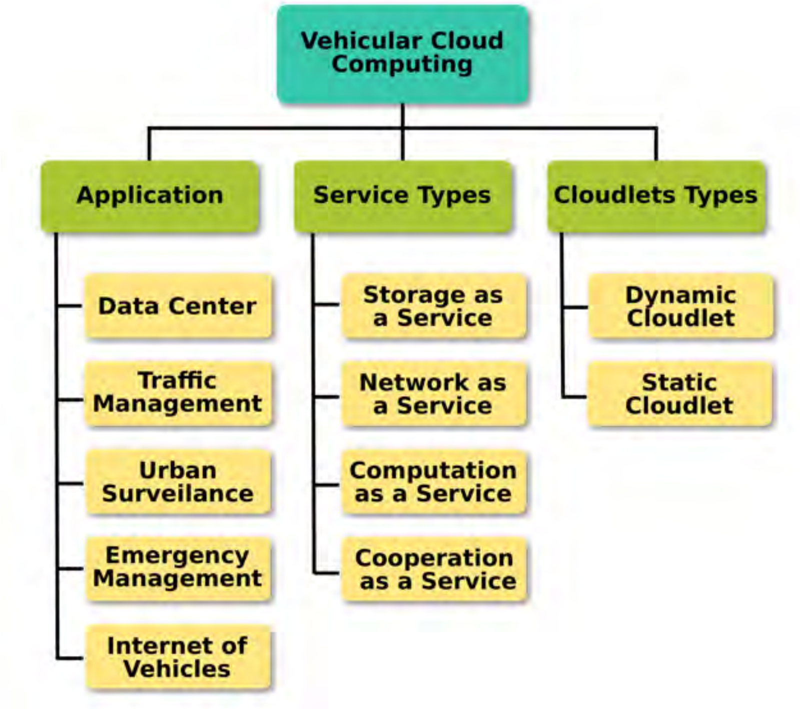


Figure 5: Classification of the Works on Vehicular Cloud Computing

congestion through traffic light control. Moreover, several works have devised other potential Vehicular Cloud applications, such as dynamic traffic signal optimization, projected evacuation management, and dynamic assignment of high-occupancy vehicle lanes [46].

Applications nevertheless require a set of innovative VCC services so that they can be built and transparently or seamlessly integrated into the current urban system. These novel services, such as NaaS, STaaS, CompaaS, and CaaS, consequently demand an infrastructure and models so that they can be openly available for public access [54].

The mobility of the environment allows VCC to be classified into two major groups [55]. One group represents a static Vehicular Cloudlet, assembled through the aggregation of resources of vehicles that show no movement; these vehicles are still and resting in a parking lot, for example. In this particular case, the VCC resembles a traditional Cloud because vehicles show a long-stay behavior. The other VCC group contains dynamic Vehicular Cloudlets, so it deals with and relies on highly dynamic traffic flow. Applications that depend on rapid dynamic changes of the urban environment can profit from these dynamic Cloudlets; as previously mentioned, these applications may assist the transportation system with traffic-related issues or emergency situations.

For instance, the work presented in [56] devises a static Vehicular Cloud as a germinal work towards the implementation of a VC. The work densely investigated the expected number of vehicles in a long-term parking lot at an international airport. By using a stochastic process, the system could be modeled according to the time-varying and departure rates for estimating the number of available resources for assembling a data center. The expected number of vehicles, as well as its variance, was deduced in a closed form in this work, conforming the behavior of incoming and outgoing vehicles in the parking lot. The work provided a time-dependent probability distribution function that represented the occupancy of the parking lot, and it determined that the limiting nature of the parameters fade away as initial conditions of the system. The obtained values for those parameters can support the sure estimation of the number of vehicles in a parking lot, which can be decidedly estimated at any given time.

On the other hand, works with dynamic Vehicular Clouds deal with rather more intricate and unstable environments. The works described in [57, 58, 59] observed and modeled the average quantity of computing resources embedded within vehicles in a road segment. The work represented the free-flow of vehicles, as well as traffic jams, through stochastic traffic models.

5.2. Architecture of Vehicular Clouds

Several works have proposed architectures to define Vehicular Clouds [53]. Out of these efforts, three significant communication layers support the general architecture of Vehicular Cloud Computing. As illustrated in Figure 6, these layers are the Onboard Layer, represented by Onboard Units, Communication Layer, characterized by Communication Components, and Cloud Computing Layer, which consists of Cloud Computing Resources, Virtualization Layer, Vehicular Cloud Services, and API. The elements of the Cloud Computing layer present internal structural dependencies, leading to its division in distinct sublayers. In the Onboard layer, components allow vehicles to detect the condition of the environment, the status of the road, and the behavior of drivers by using a series of onboard sensors, such as environmental sensors, smartphone sensors, sensors internal to vehicles, and driver behavior recognition [60, 61]. Distributed storage is essential for maintaining the information collected by sensors so that real-time applications can efficiently access it. Each vehicle also presents computing and storage units, complementing its sophisticated suite of embedded sensors. These elements comprehend the basic blocks that form Cloud computing resources.

The second layer focuses on vehicular connections, enabling vehicle-to-infrastructure and vehicle-to-vehicle communication strategies. Because vehicles contain IEEE 802.11p transceivers, they are capable of exchanging data with the VC through both V2V and V2I methods. Several protocols thus enable communication, such as Dedicated Short Range Communication (DSRC) [62, 63], Wireless Access in Vehicular Environment (WAVE) [64], and 3G or 4G cellular communications.

The V2V architecture allows vehicles to communicate with each other in an ad-hoc fashion. The networking features are enabled as long as vehicles are within a valid communication range. In this sense, a VCC can be assembled dynamically, autonomously, and opportunistically through vehicles via their V2V connections. This communication architecture allows the implementation of several flexible, fluid applications. For instance, an emergency warning system might generate messages according to the driving behavior or the road conditions. Abnormal aspects cause vehicles to send requests to the VC storage pool. Consequently, all vehicles participating in this VC receive emergency messages, which indicate the precise geographical location of the identified atypical event [65].

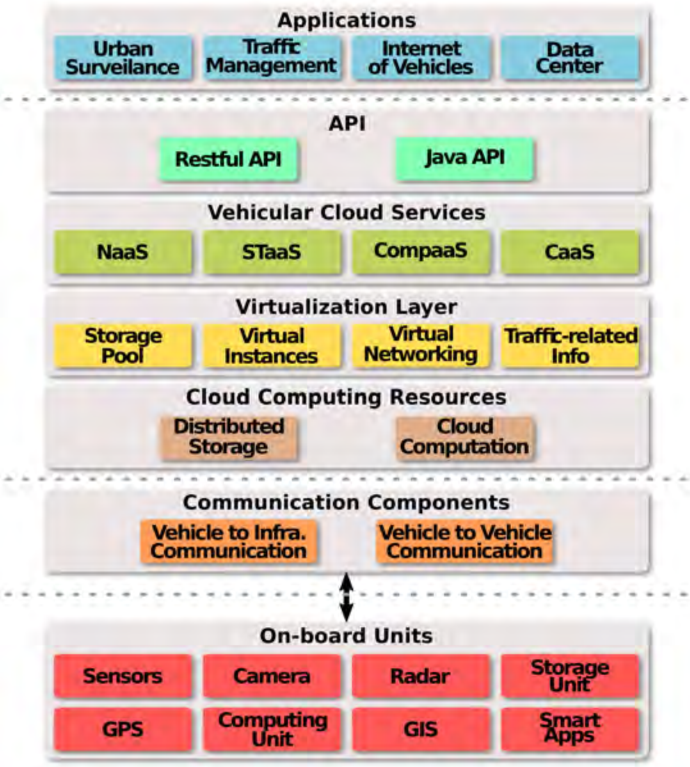


Figure 6: General Vehicular Cloud Computing Architecture

V2I comprises the second component of the communication layer. This element complements the V2V communication architecture. Transportation services and applications may opt for the more stable V2I when transmitting control information between a vehicle and other vehicles, the infrastructure, or the Cloud. This infrastructure-based approach also allows multiple autonomous Vehicular Clouds, based on the support of RSUs that function as gateways to an external network. This communication strategy can enable a data sharing platform over the Internet, for instance. Therefore, drivers are capable of accessing a broader view of the traffic condition, which can enhance the level of safety for drivers and consequently decrease the number of automotive accidents.

As depicted in the architecture, the five internal sub-layers are essential for the Cloud Computing layer: Cloud computing resources, virtualization layer, Vehicular Cloud services, API, and application layer. The computational and storage capacity of vehicles participating the in VC combine as Cloud Computing resources. The aggregation of computing resources allows authorized individuals to utilize them according to predefined quotas; the virtualization technology flexibly supports access to computing power, pooled storage, and traffic-related information. Computation as a Service (CompaaS), Storage as a Service (STaaS), Network as a Service (NaaS), and Cooperation as a Service (CaaS) comprise the the primary services, implemented in the Vehicular Cloud. Such Cloud primary services are made available through an API, which allows the development of a wide range of applications. According to the presented VCC architecture, the software applications are represented by the Application layer. Authorized end users, such as the municipal traffic management system or drivers, can reach these applications remotely through the Internet.

5.3. Vehicular Cloud Services

As previously discussed, with the due infrastructure to support it, Vehicular Cloud can enable an extensive variety of services. This broad range of services is grouped in several service type categories, the most significant of which are Computation as a Service, Network as a Service (NaaS), Storage as a Service (STaaS), and Cooperation as a Service (CaaS) [66].

5.3.1. Computation as a Service

Computation-as-a-Service shares similar objectives as STaaS; it addresses the aggregation of available, unused computation resources of vehicles, making them accessible through a service to authorized users. Vehicular Cloud Computing consists of a very recent concept that deals with a very sensitive and complex context: the exploration of computational resources in highly mobile environments. The study conducted in [56] can serve to estimate possible computing capabilities within a parking lot, where vehicles rest still for a predictable amount of time. The challenge in this situation is how to define a proper model that allows effective migration of tasks in and from parked vehicles, redistributing tasks when vehicles leave or arrive in the parking lot [67, 68]. In this case, the model needs to consider the migration burden, which involves suspending tasks, saving their execution status, identifying new hosts, and transferring jobs; all these factors are vital in determining the feasibility of task offloading in such dynamic scenarios [69].

5.3.2. Network as a Service

Providing connectivity consists of a major and complex challenge in urban environments. Vehicular Networks mitigate the issue by the use of fixed roadside access points (APs) and mobile vehicular users. Recent initiatives towards NaaS have focused on employing V2V and V2I communication to enable connectivity and transfer data to and from the Cloud. A study has been developed in this context to investigate the potential performance of downloading content from the Vehicular Cloud. The work formulated and solved the problem through linear programming, exploring the condition and availability of each communication medium. The different data transfer paradigms analyzed in the formulation consisted of direct transfers, connected forwarding, and carry-and-forward. The direct communication access a peer has with the Cloud characterizes direct transfers, meaning that it can directly download content from an AP. On the other hand, connected forwarding comprehends the data transfer through multi-hop paths established among vehicles. Carry-and-forward is comprised of vehicles storing data until it can be forwarded to the destination when the path between the AP and the downloader shows intermittent connections.

5.3.3. Storage as a Service

Each modern vehicle is assumed to be equipped with large storage capacity as technology has allowed storage media to achieve portable sizes and reduced costs. Such advancements have motivated work [56] determining the parameters that condition the design and implementation of a data center in the parking lot of an international airport. The data center exploits the availability of onboard resources of parked vehicles. This work assumes that vehicles are connected to power sources so that there is no restriction in power consumption and battery autonomy. The work also considers a business model for encouraging air travelers to contribute through benefits on free parking or automotive services; in this case, this approach assumes that vehicle owners are motivated to share unused resources for building up the Cloud.

In this same sense, another work focused on designing a two-tier architecture for a vehicular data center [70]. The devised architecture centers on allowing the proper control of excessive storage in a parking lot. The proposed approach concerns the constant incoming and outgoing vehicular resources, which follows the model of a random variable; this design differs from the traditional data center due to dynamic changes in the presence of resources. The work described in [71] introduced a replication-based fault-tolerant storage to resolve this variable presence of resources. This work complements the data center design by including a copy principle, where the system enables clients to keep N copies of the original media in each of the N available storage servers. The use of replicas allows successful data retrieval as long as one of the servers is still available, or reachable, in the data center and contains an intact replica.

5.3.4. Cooperation as a Service

Coordinated collaboration among vehicles introduces a wide range of benefits to system and applications. The collaboration promotes data collection and task processing in an on-demand, regulated form. For instance, a Navigator Assisted Vehicular route Optimizer (NAVOPT), developed in [72], focused on composing an onboard navigation server. The server relies on the traditional Cloud and the cooperation of vehicles that participate in a Vehicular Cloud. Each vehicle in the proposed system is responsible for detecting its geographical position through its GPS device and reporting this information to the navigation server through wireless communication. The server aggregates the data from several vehicles and constructs a traffic load map for a given area; it also determines optimal routes for each vehicle, returning this optimized path to its respective

Table 2: Vehicular Cloud Applications for Data Centers.

Application	**Mobility**	**Major Feature**	**Trad. Cloud Depend.**
Airport Datacenter [56]	None	Resource Aggregation	No
Pics-on-wheels [73]	None	Storage resources	Yes
Parked Vehicles [74]	Low	Spatio-temporal Network	No
VM Migration [75]	High	Minimum Network Cost	No
Data Prefetching [76]	High	Data Dissemination	No

vehicle.

5.4. Vehicular Cloud Applications

Tables 2, 3, 4, 5, and 6 summarize some representative vehicular services and application prototypes or proposals from the year 2012 to the year 2016. We can observe that the common characteristic of the diverse list of Vehicular Cloud services and applications consists of being independent of the traditional Cloud; applications, with the support of VCC, are capable of autonomously executing without the need of computing resources from servers. This self-sufficiency aspect exploits the availability of onboard physical resources, such as computing power, storage, sensing devices, GPS, GIS, collision radars, cameras, and transceivers. This list of physical resources just tends to grow as the automotive industry increasingly invests in smart and intelligent devices embedded in vehicles. These advancements enable novel applications that complement or expand the features of Vehicular Clouds.

5.4.1. Data Centers

The applications in this category are characterized by building and maintaining Vehicular Clouds as data centers or being part of a data center. Most of the existing works in this class of applications deal with a little mobility of vehicles, which are parked and considered available for a relatively extended period. Table 2 briefly describes a summary of such works.

The first work in this context effectively illustrates the vehicular Cloud for data center applications [56]. As previously described, the work combined vehicular computing resources in the long-term parking lot of a typical international airport. The approach employs a designed model as a stochastic process method to identify the number of vehicles resting in the parking lot of a given airport. Therefore, the model provides analytical results concerning the availability of computational resources in a rather stable environment.

In the same parking lot scenario, another application work took advantage of the excessive available storage resources of vehicles in parking lots [73]. The auxiliary vehicular data center (VDC) assembled from such resources can adequately help mitigate the load pressure on traditional data centers. This work considered a relatively static scenario, and it characterized the dynamic behavior of a garage containing a limited number of parking spaces. Besides this characterization, the work considered communication costs in the context of a two-tier data center, essential for correctly defining each resource management policy.

Another VDC application continued in the same parking scenario with more dynamic parameters [74]. In the context of this work, parked vehicles were used to build a spatiotemporal network enabling storage infrastructure. As a consequence, vehicles in a road segment are capable of replacing the role of RSUs. The work made use of the Virtual Cord Protocol to enable access to moving vehicles in the assembled Vehicular Cloud. The communication of vehicles in the nearby urban traffic network introduces challenges to management communication in topology that changes dynamically. The feasibility of the protocol, as well as the Vehicular Cloud, is demonstrated through simulations. This work also introduces an evaluation which is conducted to determine efficiency in the storage and retrieval of data in Vehicular Clouds formed by parked vehicles.

An approach has employed linear programming [75] for optimizing the use of communication resources, observing roadside units as potential Cloudlets to serve the needs of vehicles. The work focused on minimizing the overall network cost when performing VM migrations and conducting regular data traffic. In this work, roadside Cloudlets work as an auxiliary Cloud resource that might receive a virtual machine (VM), respective to a vehicle, noting the benefits of the new Fog Computing paradigm. The optimization model formulates the VM migration problem as a mixed-integer quadratic programming problem to minimize the current network costs in VCC. A polynomial time two-phase heuristic algorithm is proposed to tackle the computation complexity due to a large number of vehicles in an urban traffic network. The solution provides the means to determine if a VM migration is beneficial in the context of performance gain, communication and migration overhead.

Another work introduced a vehicle route-based data prefetching scheme, which was devised to maximize data dissemination success probability in VDC [76]. In this particular case, wireless connectivity was stochastically unknown, and the capacity of local storage was restricted. A deterministic, greedy data dissemination algorithm was employed when the stochastic features of the network connectivity success rate were predefined. On the other hand, stochastic MAB-based online learning framework is applied to learn stochastic characteristics of network connectivity success rate when its distribution is unknown.

5.4.2. Traffic Management

Traffic management applications aim to assist the traffic flow of vehicles by preventing congestion or even avoiding congested areas. The development of such applications envisions real-time constraints on the input data; the restrictions reside on dynamically changing environmental conditions and proper responses to accommodate updates in reasonable time frames. Appropriate recommendations to traffic management systems or drivers directly influences the effectiveness in improving the flow of vehicles. Table 3 summarily lists the existing known VCC applications for traffic management.

User-Driven Cloud Transportation System made use of real-time data, which the user (driver) fed into the application [77]. In this work, a Cloud transportation system was devised to allow predictions of traffic congestion. The system also enables the construction of traffic maps, aiming to support *smart driving*, which recommends a path to the driver according to a defined model. The application employs a scheme of user-driven crowd-sourcing to acquire driver data for

Table 3: Vehicular Cloud Applications for Traffic Management.

Application	Mobility	Major Feature	Trad. Cloud Depend.
User-Driven System [77]	High	Path recommendation	Yes
Intellectual Road Infrastructure [78]	High	Cost minimization	No
Context-Aware Parking [79]	High	Probability Analysis	No

predicting traffic jams and building a real-time traffic model. As for a measure of performance improvement on the calculations, the application also engages in a Map-Reduce computing model and algorithm for traffic data processing. The final interface is developed and made available on an Android-based prototype application.

Another similar traffic management application consists of the Intellectual Road Infrastructure [78], which also collected and aggregated data in real-time to readily determine traffic conditions. Therefore, this work proposed a road infrastructure to monitor and control traffic in a real-time manner. This real-time coordination is envisioned to minimize overall transportation costs and improve safety. The proposed system accesses devices and vehicles supplied with onboard radio frequency identification (RFID), GPS, and sensors to gather information about traffic conditions. After data aggregation finishes, the system processes the information in the Cloud and presents valuable results via an online service.

The third traffic management application differed from the previous two applications because it focused on a particular context, which involved identifying parking spots in real-time. This recommendation system, Context-Aware Dynamic Parking Service [79], extended the traditional parking services in garages by dynamically updating the available parking spots for incoming vehicles. In this case, the Vehicular Cloud can aid the parking of vehicles on road segments. The system employed a method of probability analysis to enable relevant traffic authorities to determine dynamically whether the road can be authorized to

Table 4: Vehicular Cloud Applications for Internet of Vehicles.

Application	Mobility	Major Feature	Trad. Cloud Depend.
Pics-on-wheels [73]	Low	Internet of Things	No
Autonomous Driving [80]	High	Driving decision-making	No
Multimedia Services [81]	High	Rebroadcasting	No

provide context-aware parking services. As a result, the context information must represent the expected parking duration of vehicles on the roadside.

5.4.3. Internet of Vehicles

This class of VC applications enables vehicles to provide and spread a network access service, such as mobile devices using the Internet of Things concept. This class of applications exploits connectivity among vehicles, as well as physical devices and other elements embedded with sensors, actuators, software, or electronics to empower data exchange and other applications. Consequently, vehicles can autonomously supply resources or sensitive data collected through sensors, which are essential for implementing a series of keen ITS, as well as ITS-related, services. Table 4 briefly lists the existing most relevant applications for an Internet of vehicles.

Pics-on-wheels consists of a surveillance application where the vehicles can serve as a vital element in providing appropriate images, which can support monitoring or inspection of events in a given region [73]. Vehicles, mobile service nodes, participate in the surveillance application, introducing dynamicity in the video capture and extending the surveillance coverage beyond the capacity of stationary cameras. In the particular case of this application, a Vehicular Cloud server estimates the number of qualified vehicle candidates in the Zone of Interest (ZoI). The server also either accepts or rejects picture requests from individuals that inquire after a ZoI. Upon the selected candidates, the Vehicular Cloud server invites authorized vehicles in the ZoI to provide the pictures.

Another Internet of Vehicles application requires more dynamic decision-making to provide or enrich Autonomous Driving [80]. In this application, a Vehicular Cloud was able to be established by the aggregate resources from neighboring vehicles, RSUs, and their potential interconnections. The Autonomous Driving application makes use of images of the next three road segments. These images are provided through the Vehicular Cloud and are essential for evaluating the traffic conditions and enabling preventive decision making. The members of the Vehicular Cloud are responsible for feeding the repository with images in real-time so that the whole network can access the published content.

Another application made use of the Internet of Vehicles to improve the delivery of multimedia services to drivers [81]. The onboard advanced, embedded devices enhance vehicles so that they are capable of performing processing and collection of multimedia in VCs; such vehicles can provide processing power, as well as storage of content. This application thus introduces an enhanced adaptive probabilistic rebroadcasting method; the method is designed to counter the saturation of the communication through large (multimedia) packet sizes in high-density networks. Linear programming is used to determine a target bitrate for the videos recorded in the vehicles and control the volume of acquired data.

5.4.4. Urban Surveillance

As in the description of the first Internet of Vehicles application, there is high interest in surveillance, identifying suspicious behavior, monitoring sensitive areas, or auditing earlier events. Due to the dynamicity of an urban area, the high mobility of vehicles matches the constant changes. The collective work of elements joined through a Vehicular Cloud can facilitate and promote the provision of data to support surveillance in a region of interest. Table 5 summarily presents the list of existing applications.

The first surveillance application, called Smart Traffic Cloud, made use of data collected by mobile devices to conduct analysis and identify patterns [82]. The application was consequently able to provide a real-time traffic condition map, which was developed at runtime using data collected from the mobile phones of commuters. This proposed system enables a more active map con-

Table 5: Vehicular Cloud Applications for Data Centers.

Application	Mobility	Major Feature	Trad. Cloud Depend.
Smart Traffic Cloud [82]	High	Map-Reduce / Ontologies	No
Video Reporting [83]	High	Video recording	Yes

struction. The proposed software infrastructure allows traffic data collection, manages the current condition of traffic, analyzes the input of the information, and presents the results in flexible manner. Smart Traffic Cloud employs a Map-Reduce framework and an ontology database to handle the distributed data and parallel analysis.

The other application in the context of surveillance focused on the collection of video from specific regions of interest [83]. The Cloud-assisted video reporting service allows vehicles to participate in the VC to report videos instantly; the videos are used for supporting the investigation of traffic accidents and reports to official entities or ambulance vehicles, guaranteeing a timely response from them. In a situation demanding such videos, the respective vehicles send the media to the VC, preferably through a communication route; the application may resort to the 5G network when V2V routes are nonexistent. An authentication process is achieved to provide privacy and security for the video transmission by using a digital signature that associates an encrypted accident video with a pseudonym.

5.4.5. Security and Infotainment

Vehicular Cloud applications can expand to assist other systems, such as healthcare, or enhance already existing services developed through Vehicular Networks, such as infotainment. The most relevant works in these varied contexts comprise the improvement of security in healthcare and the efficient delivery of content and entertainment to driver and passengers. Table 6 briefly lists the two applications.

Due to the importance of securing the communication channel to highly sensitive applications in healthcare, authentication is crucial for enabling the

Table 6: Vehicular Cloud Applications.

Application	Scope	Mobility	Major Feature	Trad. Cloud Depend.
RFID Authentication [84]	Health Care	High	Petri Net model	No
InCloud [85]	Infotainment	Low	Infotainment	Yes

access to only accredited parts. An intelligent RFID-enabled Authentication scheme ensured security for healthcare applications in VCC environment [84]. The model of the scheme assumes that vehicles and RSUs contain low- and high-frequency RFID readers and that patients wear RFID-enabled wristbands. The scheme relies on a PetriNet-based authentication model to guarantee a secure communication between RSUs and the central Cloud.

A framework based on the service-oriented architecture concept, called InCloud, was designed to facilitate the design of infotainment applications [85]. This Cloud-based middleware framework works in the scope of assisting media delivery for vehicular applications. Thus, the application design principles include data fusion, context-awareness, reusability, and loose-coupling. The middleware makes use of preprocessing Internet data, filtering relevant content, and fusion of content from different sources to enhance performance. Based on the proposed framework, the design of three infotainment applications served as a proof of concept.

6. Mathematical Modeling of Traffic Flow for Vehicular Clouds

Vehicular Cloud Computing completely relies on the availability of vehicular resources, and forming and maintaining a VCC depends on the presence of nearby, reachable vehicles. In the transportation context, vehicles might either be parked or in movement. The former case introduces a static scenario in which departure and arrival rates can be inferred from a history, consisting of a simple condition for determining the number of vehicles and their probable time of availability. The latter case establishes a dynamic, intricate scenario where the reachability of vehicles rapidly changes. Consequently, defining models that can represent the behavior of vehicles is crucial for generating estimations and supporting the management of resources within the VCC.

A model consists of a simple representation of reality; it enables simulations, considering a finite number of assumed factors. Through the use of simplification and relevant suppositions, a model is capable of representing real scenarios, which contain a countless number of elements. The context of traffic flows includes many factors that influence traffic load, such as road geometry, traffic signs, road conditions, weather, and driver behavior. Besides that, a precise mathematical model of the traffic is highly complex, as well as unfeasible, due to plenty of random events that occur.

Therefore, traffic flow models also need assumptions. Several works have incorporated traffic models for supporting solutions of Vehicular Networks and Clouds. These models represent realistic traffic behavior in highways and urban centers, serving as a mechanism for analyzing the performance of novel approaches, such as data dissemination schemes, routing protocols, and prediction of available onboard resources. Three distinct categories classify such models, providing different levels of abstraction and detail in their representation.

6.1. Car-Following Models

Car-following models comprehend the class of microscopic models [86]. Consequently, they individually characterize the behavior of every vehicle in the traffic flow. Several models have been proposed to represent the driving behavior in a road segment. Such models introduce a more adaptive approach to the behavior modeling of vehicles instead of optimization perspectives on the traffic flow, in general. Table 7 summarizes the notations used in models described in this section.

ID model consists of a representation of an intelligent driver model [87, 88], requiring a minimum number of relevant parameters. The model is intended to simplify calibration through a uniform description of the phase transformation in the traffic flow, which would occasionally happen from a free-flow to an entirely queued-up, congested flow. Formulas 1 and 2 summarize the model.

Table 7: Notation of the traffic flow models

Variable	**Description**
$a_n(t)$	acceleration of vehicle n at time t
$a(n)$	maximum acceleration of vehicle n
δ	the acceleration index
b	the reasonable deceleration
T	reaction delay time
$\Delta v_n(t)$	speed between the current and following vehicles $(v_{n+1}(t) - v_n(t))$
v_0	the ideal speed
r	coefficient that determines the different of velocity
V(n,t)	speed of vehicle n at time t
$V^*(n)$	aimed speed of vehicle n
$\Delta x_n(t)$	distance between current and following vehicles $(x_{n+1}(t) - x_n(t))$
s_a	the expected headway
s_0	the still safe distance between vehicles
X_i	$Y = h(X_1, ..., X_n)$ - abstract model parameters
$D_{KL_{X_i^c}}(g(y\|\bar{x}_i^c)\|\|f(y))$	$K - L$ distance as the total sensitivity index of X_i
$g(y\|\bar{x}_i^c)$	yielded PDF of Y on the complementary set of X_i

$$a_n(t) = a_n(0)\big[1 - (v_n(t)/v_o)^{\delta} - ((S_n^*(v_n(t), -\Delta v_n(t)))/s_a)^2\big] \tag{1}$$

where

$$S_n^*(V_n(t), -\Delta v_n(t)) = s_0 + v_n(t)T' - \frac{v_n(t)\Delta v_n(t)}{2\sqrt{a_n(0)b}} \tag{2}$$

A full velocity difference model was designed to accommodate realistic behavior of acceleration and deceleration of vehicles [89]. The model, as described in Formula 3, is based on a general force model and defines the flow starting velocity that reacts faster to the velocity wave.

$$a_n(t) = a\big[V(\Delta x_n(t)) - v_n(t)\big] + r\Delta v_n(t) \tag{3}$$

The design of feasible, realistic car-following models must follow several basic assumptions [90]:

1. Drivers act only upon the status or condition of the vehicles ahead; they do not consider any of the vehicles following them.

2. A vehicle moves trailing the vehicles in front of it and in the same lane; it does not overtake the vehicles ahead.

3. Road conditions are optimal; vehicles show same moving performance; drivers always behave commonly and with the same predictable driving conducts.

Such assumptions regulate the existing models and have be used to produce a relative entropy-based probabilistic sensitivity analysis [90]. This analysis is intended to better estimate the values of relevant microscopic traffic model parameters. Formula 4 summarily describes the analysis, which employs K-L distance.

$$D_{KL_{X_i^c}}(g(y|\bar{x}_i^c)||f(y)) = \int_{-\infty}^{\infty} g(y|\bar{x}_i^c) \cdot \ln \frac{g(y|\bar{x}_i^c)}{f(y)} dy \tag{4}$$

Car-following models are employed to represent the dynamics of vehicular traffic flows. With much less restrictive assumptions, these models allow a substantial number of parameters to be considered. The parameters, to some degree, can simulate real finite aspects, such as weather situation, the reaction time of drivers, road conditions, and vehicles' technical specifications. All these details result in a significant degree of accuracy at the cost of complexity. The speed of vehicles consists of an essential element in these models, being a determining factor in the four modes of vehicle movement [91]:

- Free-driving mode is most suitable for free-flows of vehicles. Vehicles are sparsely distributed in road segments and do not encounter obstacles in their paths. Consequently, vehicles move at any velocity, respecting the speed limit of the road.

- Approaching mode represents the situation in which a vehicle moves faster than the vehicle that is ahead, obeying a defined safe distance between them.

- Following pattern consists of a vehicle accelerating or decelerating according to the vehicle ahead to keep a specified distance between them.

- Braking mode is employed for a vehicle to enlarge the distance from the vehicle ahead in case the gap falls below a defined safety condition.

The listed movement patterns are variations of a more simplistic car-following model, which concerns the current speed of a vehicle and its acceleration to achieve a desired average speed. As a result, it is worth noticing that this model might be interpreted as an extension of the Gipps model [92], as described in Formula 5.

$$V_a(n, t+T) = V(n,t) + 2.5a(n)T \cdot (1 - \frac{V(n,t)}{V^*(n)})\sqrt{0.025 + \frac{V(n,t)}{V^*(n)}} \qquad (5)$$

Car-following models, as well as microscopic models, are usually more realistic with traffic behavior. However, they are highly computation-intensive, especially in scenarios where urban areas may contain a large number of vehicles. Moreover, the analytical car-following framework imposed high complexity for obtaining closed-form results.

6.2. Traffic Stream Models

Unlike car-following models, traffic stream models do not focus on the behavior of an individual vehicle. These models instead represent vehicles collectively, as in vehicular streams. As a consequence, the stream models are classified as macroscopic models because they interpret a vehicular traffic flow as a hydrodynamic stream. Traffic flow rate, vehicular density, and speed of vehicles consist of the main macroscopic parameters for representing the vehicular traffic flow. This simplified approach shows that these models contain a smaller number of variables, which makes traffic streams less complex, facilitating their implementation. This lower complexity causes stream models to be more popular than microscopic models for the design of data communication strategies in Vehicular Ad Hoc Networks. Nevertheless, the macroscopic traffic models used in open research works contain assumptions that are not consistent, concerning case-specific aspects. Table 8 lists the main notation elements employed in the traffic stream models described in this section.

Table 8: Notation of the traffic flow models

Variable	Description
$f'(\psi)$	tangent to the curve $f(\cdot)$ at ψ
$\psi 1$ and ψn	the curves where arc-length is calculated
$f(\psi)$	principal curve where it does not intersect itself, has finite length in a bounded subset, and is self-consistent
γ	mean of the Gaussian random variable
μ	mean vehicle flow rate ($vehicles/minute$)
σ	standard deviation of the Gaussian random variable
ϕ	rate of the exponential random variable
m	shift of the exponential random variable
w_G	weight of the normal distribution
w_E	weight of the exponential distribution
Pn	probability of number of vehicles within $[SD]$
P_n	$P_n(t)$ for $t \to \infty$
$\bar{R}$	average residence time
$\widetilde{R}$	approximate average resident time
$E(T)$	expected residence time in $[SD]$
n	number of vehicles within road segment $[AB]$ in a given instant
N	number of vehicles in $[SD]$
$\widetilde{N}$	approximate average number of vehicles in $[AB]$
$f_R^{Cox}(r)$	Coxian approximation of $f_R(r)$
$m1$	$1 + (\mu 1/2c_v^2(\mu 1 - \mu 2))$
$\mu 1$	$2\mu_R$
$\mu 2$	$(\mu 1/c_v^2)$
μ_v	mean maximum vehicle flow rate in $[AB]$
t	time
L_{SD}	length of the road segment
M	normalization factor
σ_V	standard deviation of average speed
τ	time range
W	waiting time in the queue
λ	Poissonian arrival rate
σ_τ^2	standard deviation of vehicular leaving time
ρ	traffic density
w	waiting time in the queue
$\delta(w)$	PDF of the leaving time

Analytical experiments have been conducted over highways in real, large urban environments: Madrid, Spain [93]; and Beijing, China [94]. These studies enabled the collection of a significant amount of data through sensors placed along highways. These works have applied well-known data processing tools, like principal component analysis (PCA) and expectation-maximization based algorithm. These tools allowed them to draw the general distribution of the traffic for a period. Nonetheless, the work is limited by the applicability of the models since they are tightly attached to the scenarios where the studies have been conducted, conditioning the variability of the data.

In work described in [94], a model was proposed using principal curves (PC) as a paradigm extension from PCA to accommodate the variations in the input data curves. An algorithm is used to adjust the curves following an initialization that leads to an iterative execution that defines projections, expectations (recalculations), and a stop condition. The model is mainly driven by the arc-length of the curve towards the traffic stream data, as represented in Formula 6, while the model applies the PC smoothing on the curve when $f(\psi)$ is consistent, as described in Formula 7.

$$l = \int_{\psi 1}^{\psi n} ||f'(z)|| = \sum_{i=1}^{n} ||f(\psi_{i+1}) - f(\psi_i)|| \tag{6}$$

$$f(\psi) = E(X|\psi_f(X) = \psi), \forall \psi \in R^1 \tag{7}$$

The work described in [93] explored the inter-arrival times of vehicles in a road segment to define the general distribution of traffic and define a free-flow model. The work considered the dependence between following vehicles through the use of correlation analysis to identify their movement speed. The work also observed the burst of vehicles, where the short distance between them and their almost-constant relative speed directly delimited it. Thus, a model was described in this work to represent the time between traveling vehicles that followed a Gaussian-exponential model to identify the inter-arrival times, as outlined in the definition of the random variable in Formula 8.

$$f_A(t) = w_G \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(t-\gamma)^2}{2\sigma^2}} + w_E \phi e^{-\phi(t-m)} \tag{8}$$

The traffic model in [95] considered vehicles moving in one direction on a straight road segment. Two random variables, V and T, are used to characterize the movement of a vehicle. V consists of the speed level of a vehicle. The speed level contains two probable values: high speed (V_H) and low speed (V_L). Parameter T conditions the switch of the speed level into these two values; these parameters comprehended an exponentially distributed random variable with parameter ϕ. The model introduced in this work assumes that a vehicle shows a possibility of maintaining a speed level V_H (V_L) for a certain amount of time T before switching to V_L (V_H).

Another macroscopic free-flow traffic model is described in [96], which defined the following assumptions:

- Density of vehicles on a road segment might be low or medium.
- An independent and identically distributed Poisson process represents the arrival of incoming vehicles.
- Traffic lights, stop signs, and other obstacles do not disrupt the movement of vehicles on a road segment.
- Speed shows variation that is normally distributed in the interval $[V_m in V_m ax]$, maintaining the same velocity along the road segment.

These assumptions are essential in defining their proposed model, which provides useful estimations on the residence time of a vehicle and the number of vehicles in a road segment. Formula 9 describes the density of vehicles while Formulas 10 and 11 define the approximate average residence time and the number of vehicles, respectively.

$$P_n = \frac{(\mu_v \bar{R})^n e^{-\mu_v \bar{R}}}{n!} \tag{9}$$

$$\widetilde{R} = \int_0^\infty r \cdot f_R^{Cox}(r) dr = \frac{m1}{\mu 1} + \frac{1 - m1}{\mu 2} \tag{10}$$

$$\widetilde{N} = \sum_{n=0}^{\infty} n \cdot \widetilde{P_n} = \mu_v \widetilde{R} \tag{11}$$

The work described in [57, 97, 98] proposed a split of the stream model into two well-defined scenarios so that each part of the model can provide a more realistic representation of the behavior of the traffic. Free-flow traffic and queueing-up traffic models comprehend the two typical traffic situations modeled in this approach. Even though both of them attempt to represent the movement of vehicles in a stream abstraction, they make use of stochastic modeling to define the number of vehicles and their *residence* time in a road segment for a more precise estimation of available resources in a Vehicular Cloud.

In the Free-flow model, the vehicle arrival is modeled as a Poisson process. This model considers that there is a free-flow of vehicles: there are no obstacles or conditional stops while the nodes are traversing the road segment, and the vehicles can move at any speed within the allowed range. Consequently, the stream of vehicles presents a low to medium density, as well as residence time in the road segment. Through this model, Formula 12 describes the average or estimated number of vehicles inside a road segment $[SD]$ while Formula 13 provides the probability mass function of the number vehicles inside the same road segment.

$$E(N) = \mu E(T) = \mu \int_0^\infty t f_T(t)\, dt \quad = \mu \int_0^\infty \frac{M \cdot L_{SD}}{t\sigma_V \sqrt{2\pi}} e^{-\left(\frac{\frac{L_{SD}}{t} - \bar{V}}{\sigma_V \sqrt{2}}\right)^2} dt \qquad (12)$$

$$P_n = \frac{[\mu E(T))]^n e^{-\mu E(T)}}{n!} \qquad (13)$$

The queueing-up model attempts to introduce a more realistic view of the traffic scenario of large urban areas, where road segments become congested with the increase in the number of vehicles traversing roads, reaching their maximum capacity. In this model, obstacles are the cause of a bottleneck that narrows the stream out of the road segment, conditioning a vehicle departure rate smaller than the arrival rate on the road. Formula 14 gives the estimated number of vehicles in the queue while Formula 15 represents the probability density function of the waiting time in the queue, where $w = 0$ means that a vehicle has zero waiting time.

$$E(N) = \lambda E(W + \tau) = \rho + \frac{\lambda^2\sigma_\tau^2 + \rho^2}{2(1-\rho)} \tag{14}$$

$$f_W(w) = (1-\rho)\delta(w) + \rho(\mu - \lambda)e^{-(\mu-\lambda)w} \quad w > 0 \tag{15}$$

Similar work has been developed to model the free-flow movement of vehicles in a highway [99] to support Vehicular Networks. The study considers a multi-lane scenario where there is a tracking of the mobility of vehicles. A joint Poisson distribution and a conditional probability of vehicle residence in a lane are employed to represent the flow. Thus, the model considers vehicles departing from and arriving in lanes along the highway. Under the same lane mobility scenario, other works [59, 99] have explored the occurrence of lane changes taking as base the free-flow traffic model in [98], adding the concept of speed varying interval-split modeling.

6.3. Stochastic Traffic Models

Stochastic traffic models are usually employed on works to represent the overall aspects of mobility through a simpler form, which condition them to adopt restrictive assumptions and make them deviate from reality. Random Direction Mobility and Random Walk models consist of the most popular stochastic mobility models [100, 101]; the major standard feature in both involves the ability of mobile nodes to move freely in any arbitrary direction, according to the road topology of a region. The models described in this section adopt different representations for the random mobility of nodes in traffic, and Table 9 summarizes their notation. This category of models also contains other features, which are briefly listed as follows:

- The organization of the roads follows a grid topology, and the vehicles follow a random movement inside the scenario;
- Destinations are determined randomly by vehicles, which move toward them in constant velocity [102, 101]. Consequently, the grid topology constricts the movement directions to vertical and horizontal;

Table 9: Notation of the stochastic traffic models

Variable	Description
$Crowd_a(u)$	acceleration of a *crowd* of vehicles
u, w	starting positions
u', w'	starting vectors of mobile nodes for the interval $[0, \Delta]$
Δ	time variation
c	constant
$f_1(s)$	stationary density of S
v_0 and v_1	node speeds
P_{pause}	probability of pausing
s_n	new speed mobile node
α	tuning parameter
$\bar{s}$	average speed
$s_{x_{n-1}}$	random variable from a Gaussian distribution
d_n	new direction mobile node
$\bar{d}$	average direction
$d_{x_{n-1}}$	random variable from a Gaussian distribution
$b(t)$	next position at time t
τ	random Gaussian variable with variance σ
P_{i+1}	new position
P_i	old position
a	acceleration
v	random vector

- There are no interactions among vehicles; the vehicular density, speed, and flow rate, which consist of the inter-correlated parameters commonly present in macroscopic models and traffic stream models, are not considered.

In the work described in [100], two mobility models were defined to consider different situations. One model, velocity bounded model, is used to represent the mobility of pedestrians in a vehicular scenario while another model, acceleration bounded model, is used to model the behavior of vehicles' movement, as described in Formula 16, limiting the acceleration to a maximum value. Through the analysis, the model attempts to define persistent routes, which holds when edges uphold for a given Δ time span. As a result of the analysis, the work derives from an equation that determines the transmission range of *links* for a time interval in the context of Vehicular Networks, as shown in Formula 17. Moreover, a mobility spanner is also defined as a graph to represent the paths that exist between nodes $u, w \in S$ for some constant c, as described in Formula 18, showing the possible connection among nodes in a given moment.

$$Crowd_a(u) = \{w \in S \setminus \{u\} : |u - w|_2 \leq \frac{1}{2} a_{max} \Delta^2 \text{ and } |u' - w'|_2 \leq \frac{1}{2} a_{max} \Delta\} \tag{16}$$

$$|(u, u'), (w, w')|_a = max\{|u - w|_2, |u - w + (u' - w')\Delta|_2 + a_{max}\Delta^2\} \tag{17}$$

$$\sum_{i=1}^{k} |p_i - 1, p_i| \leq c \cdot |u, w|_\alpha \tag{18}$$

A model based on a stationary distribution was added to the random waypoint mobility model [103]. In this work, the authors explored the possible variations according to the stationary distributions for both speed and location. They defined a pausing parameter to study the stationary situations, so the movement of nodes, or vehicles, is represented based on the amount of pause they presented in a given movement, having a stop time equal to zero. This model adopts a 2D perspective of the environment, in which x and y coordinates of a given mobile node characterizes the movement of nodes. By computing the moving and paused states, the random waypoint mobility model is derived, providing a uniform distribution between v_0 and v_1 node speeds. Formula 19 specifies the node density at a given time.

$$\begin{aligned} f_1(s) &= f_1(s|Paused)P_{pause} + f_1(s|Notpaused)(1 - P_{pause}) \\ &= \begin{cases} 0 & s < 0 \\ P_{pause} & s < 0 \\ 0 & 0 < s < v_0 \\ \frac{1-P_{pause}}{slog(v_a/v_0)} & v_0 \leq s \geq v_1 \\ 0 & s > v_1 \end{cases} \end{aligned} \tag{19}$$

A study [104] conducted a series of analyses to evaluate a set of mobility models towards their particular characteristics. The work classified the models into entity and group mobility categories and then realized a performance analysis to demonstrate which characteristics of a model make it better suited to a determined situation, in the context of ad-hoc networks. As described in Formulas 20 and 21, GaussMarkov mobility model served as the principal element for the simulation of personal communication systems and Ad Hoc Networks, and it is mainly intended for adjusting to different levels of randomness through modifications on its parameters. In the context of group mobility models, Exponential Correlated random mobility model creates the movement of nodes and vehicles, defining the next position of a given group of nodes but being limited to the motion patterns of the nodes, as represented in Formula 22. The Pursue mobility model improves on the limitation of movement patterns by defining a target to which the nodes track and follow. This model further enhances the previous group model by adding an update for the new position of a mobile node, as shown in Formula 23, in addition to Formula 22.

$$s_n = \alpha s_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha^2) s_{x_{n-1}}} \tag{20}$$

$$d_n = \alpha d_{n-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha^2) d_{x_{n-1}}} \tag{21}$$

$$b(t+1) = b(t)e^{-\frac{1}{\tau}} + \left(\sigma \sqrt{1 - \left(e^{-\frac{1}{\tau}}\right)^2}\right) r \tag{22}$$

$$P_{i+1} = P_i + a(t - P_i) + v \tag{23}$$

6.4. Summary

Table 10 presents a succinct summary of the described traffic models. The car-following model offers a high level of detail and introduces aspects of adaptability in the model. This model is suitable for large-scale simulated environments where each entity, node, or vehicle, acts independently. There is a direct relation between these models and microscopic agent-based simulations, where each agent can produce the behavior of a vehicle following a predefined behavior

Table 10: Strengths and Weaknesses of Existing Traffic Models.

Traffic Model	**Strengths**	**Weaknesses**
Car-Following Model [86, 90]	•Large number of realistic params •Interaction among vehicles •Relatively high degree of accuracy	•Computationally expensive •High complexity •Impractical analytical results
Traffic Stream Model [95, 96]	•Collective vehicle flows •Analytical results easily obtainable •High-level traffic behavior	•Unrealistic assumptions
Stochastic Traffic Model [100, 101]	•Based on stochastic theory •Simplistic	•Restrictive assumptions •Non-realistic •Overlook on the fundamental features

and instantiated parameters.

Nevertheless, the other model types focus on representing optimal mobility scenarios. They attempt to ease the task to identify min-max situations for determining or estimating parameters that can condition the scheduling of tasks and allocation of resources for a Vehicular Cloud, such as capacities, load, availability, residence, and density, in a given moment. For instance, prediction of traffic load oscillations is essential for estimating the amount of time vehicles are available to access a service or build a Cloud, using the same vehicles to compute a solution for a possible traffic jam due to current load conditions.

7. Discussion and Research Directions

The three-layered architecture of Vehicular Cloud Computing allows a pragmatic resolution for the supporting subsystems, physical resources, connectivity, and Cloud management. Since vehicles can serve as in-loco *Cloud resources*, they can support services and applications as Cloudlets, which then implement a three-tier structure in the context of offloading tasks in an MCC approach where there are mobile devices, Cloudlets, and the Cloud. However, the status of vehicles, which can be parked or moving, conditions Cloudlets as either static or dynamic. In addition, VCC may closely resemble a MAC when it is dynamically formed through the ad-hoc connections of the vehicles on a road segment for enabling more computation power to conduct tasks. However, the characteristics of the environment and the vehicles condition the resource provision model and offloading strategies, which allows more opportunities for harvesting resources at the cost of dealing with higher mobility of computing nodes.

The VCC services described in this study extend from the traditional Cloud services, exploring the needs of applications existing in the context of intelligent transportation systems. Computation as a Service enables vehicles struggling to run complex applications to extend their computation capabilities by offloading their load to nearby vehicles. In addition to vehicles, mobile devices and the Cloud can take advantage of harvesting resources. Network as a Service works to extend the communication capabilities of vehicles by bridging the communication to the Internet through the virtualization of the network and access to the interface that enables connectivity. The management of virtualized communication resources in VCC can hypothetically simplify the connectivity of vehicles, implementing pay-as-you-go service provisioning models. Building large-scale data centers from available onboard vehicle storage enables Storage as a Service. The use of VCC storage shares the same benefits as Cloudlets in the MCC paradigm, which permits a reduction of communication delays and bandwidth consumption for the transfer of data over the Internet. Cooperation as a Service enables vehicles to collaborate among themselves for supporting complex applications or implementing inherently distributed tasks, such as coordinated data collection over the urban environment for tracking traffic load.

From the potential VCC services, several applications have been devised and implemented, covering diversified purposes, such as building data centers, enabling traffic management, supporting an Internet of Vehicles, conducting urban surveillance, allowing more efficient infotainment, and promoting health care services. From the description of these existing applications, most of them depend on static or semi-static scenarios, where the vehicles are still in parking lots. This static situation has been extensively explored in works devising VCC-supported data centers, where the frequency with which vehicles join or leave the VCC is relatively low. Other applications make use of dynamic VCC scenarios, but they use cooperation between vehicles to collect and exchange information, such as in urban surveillance. There is not yet any design that takes advantage of the VCC computational resources through scheduling strategies on workflows for offloading tasks in the VCC vehicles. Static scenarios would allow the offloading strategies to resemble the ones in MCC. The dynamic scenarios, on the other hand, would incur sophisticated scheduling strategies that account for the mobility of vehicles, as well as the availability of resources.

Some fundamental aspects are just assumed when designing VCC applications: participation of vehicles and security. The voluntary collaboration of vehicles and drivers enables the construction of VCC. Consequently, the efficiency of applications must account for different degrees of participation and availability of vehicles. Also, the development of business models directly promotes collaboration and increases the number of participating vehicles, making use of incentives and bidding systems. Security becomes a major concern when data is temporarily exchanged to remote resources, requiring security policies and trust models to guarantee that the integrity of data and privacy of users are kept and avoid the malicious tampering of VCC resources.

Mobility models then appear as crucial for the management of VCC. The models can support prediction and enhance efficient utilization of resources. Microscopic and macroscopic models have been extensively employed in traffic management systems to determine and predict traffic load, so adapted and extended versions of these models can allow the availability of vehicular resources in road segments to be determined. The availability consists basically of residence time of vehicles in the road and the number vehicles on the same road. These metrics are critical for enabling more precise scheduling of tasks on VCC. From the three classes of traffic flow models, the free-flow model type presents the best cost-benefit in bringing up quick estimations. Due to the high mobility, these models will be incorporated more deeply throughout VCC with the purpose of estimating overall movement tendency of vehicles or the individual behavior of a single vehicle for tracking resources.

8. Conclusion

Vehicular Cloud has become a significant area of research due to the powerful onboard computing and sensing resources of the modern vehicles. These underutilized resources can be collected from all vehicles in the vicinity to build a Vehicular Cloud autonomously. The VCC is considered complementary to general Cloud computing and has more traffic-related services and applications such as public surveillance, traffic management, and environmental monitoring.

In this work, we primarily presented the technical revolution of Vehicular Cloud computing based on an extensive review of the literature. Several works have been developed in the past, assisting the advancements from Vehicular Networks to Vehicular Clouds. This work also classified, listed, and discussed a series of services and applications which have been, or shared the features to be, implement in Vehicular Clouds. In the end, we provided a broad description and classification of the existing traffic models, which are essential for the VC implementation in the dynamic environment.

At this stage, however, some areas related to VCC remain unexplored for researchers such as the high mobility, unstable communication links, task migration and the security issues in the implementation of VCC. From the existing research results, we could tell that VCC is still in its initial stage towards technological maturity. However, we believe that VCC is expected to be a feasible solution for the highly intelligent traffic system.

Acknowledgments

This work is partially supported by Canada Research Chairs Program, NSERC, and DIVA Strategic Networks.

References

References

[1] National Institute of Standards and Technology (NIST), Draft cloud computing synopsis and recommendations definition of cloud computing, [Online] http://csrc.nist.gov/publications/drafts/800-146/Draft-NIST-SP800-146.pdf (2011).

[2] B. Hayes, Cloud computing, ACM Communications 51 (7) (2008) 9–11.

[3] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia, A view of cloud computing, ACM Communications 53 (4) (2010) 50–58.

[4] S. Olariu, I. Khalil, M. Abuelela, Taking vanet to the clouds, International Journal of Pervasive Computing and Communications 7 (1) (2011) 7–21.

[5] D. Baby, R. D. Sabareesh, R. A. K. Saravanaguru, A. Thangavelu, Vcr: Vehicular cloud for road side scenarios, in: Proceedings of the Second International Conference on Advances in Computing and Information Technology, 2013, pp. 541–552.

[6] S. Abdelhamid, R. Benkoczi, H. S. Hassanein, Vehicular Clouds: Ubiquitous Computing on Wheels, 2017, pp. 435–452.

[7] Y. Ma, Y. Guo, X. Tian, M. Ghanem, Distributed clustering-based aggregation algorithm for spatial correlated sensor networks, IEEE Sensors Journal 11 (3) (2011) 641–648.

[8] P. Papadimitratos, A. La Fortelle, K. Evenssen, R. Brignolo, S. Cosenza, Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation, IEEE Communications Magazine 47 (11) (2009) 84–95.

[9] M. Amadeo, C. Campolo, A. Molinaro, Information-centric networking for connected vehicles: a survey and future perspectives, IEEE Communications Magazine 54 (2) (2016) 98–104.

[10] F. A. Silva, A. Boukerche, T. R. M. B. Silva, L. B. Ruiz, E. Cerqueira, A. A. F. Loureiro, Vehicular networks: A new challenge for content-delivery-based applications, ACM Computer Surveys 49 (1) (2016) 11:1–11:29.

[11] C. Toh, Ad Hoc Wireless Networks: Protocols and Systems, 1st Edition, 2001.

[12] M. Watfa, Advances in Vehicular Ad-Hoc Networks: Developments and Challenges, 1st Edition, 2010.

[13] S. Müller, T. van Roermund, M. Steigemann, Communication Beyond Vehicles—Road to Automated Driving, 2016, pp. 55–64.

[14] S. S. Kaushik, Review of different approaches for privacy scheme in vanets, International Journal of Advanced Engineering Technology 5 (2013) 2231–1963.

[15] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, H. Zedan, A comprehensive survey on vehicular ad hoc network, Journal of Network and Computer Applications 37 (2014) 380 – 392.

[16] M. N. Mejri, J. Ben-Othman, M. Hamdi, Survey on VANET security challenges and possible cryptographic solutions, Vehicular Communications 1 (2) (2014) 53 – 66.

[17] P. Ranjan, K. K. Ahirwar, Comparative study of vanet and manet routing protocols, in: Proceedings of the International Conference on Advanced computing and communication Technologies, 2011, pp. 517–523.

[18] M. Raya, P. Papadimitratos, V. Gligor, J.-P. Hubaux, On data-centric trust establishment in ephemeral ad hoc networks, in: Proceedings of the IEEE 27th Conference on Computer Communications, 2008.

ACCEPTED MANUSCRIPT

[19] A. Darehshoorzadeh, R. E. De Grande, A. Boukerche, Toward a comprehensive model for performance analysis of opportunistic routing in wireless mesh networks, IEEE Transactions on Vehicular Technology 65 (7) (2016) 5424–5438.

[20] L. Miao, K. Djouani, B. J. van Wyk, Y. Hamam, Evaluation and enhancement of ieee 802.11 p standard: A survey, Mobile Computing 1 (1) (2012) 15–30.

[21] H. Vahdat-Nejad, A. Ramazani, T. Mohammadi, W. Mansoor, A survey on context-aware vehicular network applications, Vehicular Communications 3 (2016) 43 – 57.

[22] W. Kim, Cloud computing: Today and tomorrow, Journal of object technology 8 (1) (2009) 65–72.

[23] L. Badger, T. Grance, R. Patt-Corner, J. Voas, Draft cloud computing synopsis and recommendations, NIST special publication 800 (2011) 146.

[24] G. Galante, L. C. E. d. Bona, A survey on cloud computing elasticity, in: Proceedings of the IEEE Fifth International Conference on Utility and Cloud Computing, 2012, pp. 263–270.

[25] P. M. Mell, T. Grance, Sp 800-145. the nist definition of cloud computing, Tech. rep., National Institute of Standards & Technology (2011).

[26] V. A. Gandhi, C. Kumbharana, Comparative study of amazon ec2 and microsoft azure cloud architecture, International Journal of Advanced Networking & Applications (2014) 117–123.

[27] G. Lu, W. H. Zeng, Cloud computing survey, Applied Mechanics and Materials 530 (2014) 650–661.

[28] R. Ghosh, F. Longo, V. K. Naik, K. S. Trivedi, Modeling and performance analysis of large scale iaas clouds, Future Generation Computer Systems 29 (5) (2013) 1216–1234.

[29] X. Wen, G. Gu, Q. Li, Y. Gao, X. Zhang, Comparison of open-source cloud management platforms: Openstack and opennebula, in: Proceedings of the 9th International Conference on Fuzzy Systems and Knowledge Discovery, IEEE, 2012, pp. 2457–2461.

[30] B. Wilder, Cloud architecture patterns: using microsoft azure, O'Reilly Media, Inc., 2012.

[31] A. Zahariev, Google app engine, Helsinki University of Technology.

[32] T. Mäkilä, A. Järvi, M. Rönkkö, J. Nissilä, How to define software-as-a-service – an empirical study of finnish saas providers, in: Software business, 2010, pp. 115–124.

[33] P. Iannucci, M. Gupta, et al., IBM SmartCloud: Building a Cloud Enabled Data Center, IBM Redbooks, 2013.

[34] Z. Sanaei, S. Abolfazli, A. Gani, R. Buyya, Heterogeneity in mobile cloud computing: Taxonomy and open challenges, IEEE Communications Surveys Tutorials 16 (1) (2014) 369–392.

[35] H. T. Dinh, C. Lee, D. Niyato, P. Wang, A survey of mobile cloud computing: architecture, applications, and approaches, Wireless Communications and Mobile Computing 13 (18) (2013) 1587–1611.

[36] MCC Forum, Discover the world of mobile cloud computing, [Online] http://http://www.mobilecloudcomputingforum.com/ (2011).

[37] S. Guan, R. E. De Grande, A. Boukerche, A novel energy efficient platform based model to enable mobile cloud applications, in: Proceedings of the 21st IEEE Symposium on Computers and Communication, 2016, pp. 914–919.

[38] S. Guan, R. E. De Grande, A. Boukerche, A cloudlet-based task-centric offloading to enable energy-efficient mobile applications, in: Proceedings

ACCEPTED MANUSCRIPT

of the IEEE Symposium on Computers and Communications, 2017, pp. 564–569.

[39] S. Guan, R. E. De Grande, A. Boukerche, An hla-based cloud simulator for mobile cloud environments, in: Proceedings of the 20th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications, 2016, pp. 128–135.

[40] P. Zou, C. Wang, Z. Liu, D. Bao, Phosphor: A cloud based drm scheme with sim card, in: Proceedings of the 12th International Asia-Pacific Web Conference, 2010, pp. 459–463.

[41] P. Mach, Z. Becvar, Mobile edge computing: A survey on architecture and computation offloading, IEEE Communications Surveys Tutorials 19 (3) (2017) 1628–1656.

[42] H. Zhijun, R. E. De Grande, A. Boukerche, Towards efficient data access in mobile cloud computing using pre-fetching and caching, in: Proceedings IEEE International Conference on Communications, 2017, pp. 1–6.

[43] A. Mtibaa, A. Fahim, K. A. Harras, M. H. Ammar, Towards resource sharing in mobile device clouds: Power balancing across mobile devices, SIGCOMM Computer Communication Review 43 (4) (2013) 51–56.

[44] I. Yaqoob, E. Ahmed, A. Gani, S. Mokhtar, M. Imran, S. Guizani, Mobile ad hoc cloud: A survey, Wireless Communications and Mobile Computing 16 (16) (2016) 2572–2589.

[45] M. Abuelela, S. Olariu, Taking vanet to the clouds, in: Proceedings of the ACM International Conference on Advances in Mobile Computing and Multimedia, 2010, pp. 6–13.

[46] S. Olariu, T. Hristov, G. Yan, The next paradigm shift: From vehicular networks to vehicular clouds, in: Mobile Ad Hoc Networking: Cutting Edge Directions, 2nd Edition, John Wiley & Sons, Inc., 2013, Ch. 19.

[47] R. I. Meneguette, A. Boukerche, R. E. De Grande, SMART: an efficient resource search and management scheme for vehicular cloud-connected system, in: Proceedings of the IEEE Global Communications Conference, 2016, pp. 1–6.

[48] R. I. Meneguette, A. Boukerche, D. L. Guidoni, R. E. De Grande, A. A. F. Loureiro, L. A. Villas, A flow mobility management architecture based on proxy mobile ipv6 for vehicular networks, in: Proceedings of the 21st IEEE Symposium on Computers and Communication, 2016, pp. 732–737.

[49] R. I. Meneguette, A. Boukerche, A cooperative and adaptive resource scheduling for vehicular cloud, in: Proceedings of the IEEE Symposium on Computers and Communications, 2017, pp. 398–403.

[50] A. Boukerche, R. I. Meneguette, Vehicular cloud network: A new challenge for resource management based systems, in: Proceedings of the 13th International Wireless Communications and Mobile Computing Conference, 2017, pp. 159–164.

[51] R. E. De Grande, A. Boukerche, S. Guan, N. Aljeri, A modular distributed simulation-based architecture for intelligent transportation systems, Concurrency and Computation: Practice and Experience 28 (12) (2016) 3409–3426.

[52] R. Hussain, J. Son, H. Eun, S. Kim, H. Oh, Rethinking vehicular communications: Merging vanet with cloud computing, in: Proceedings of the IEEE CloudCom, 2012, pp. 606–609.

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

[54] A. Ashok, P. Steenkiste, F. Bai, Enabling vehicular applications using cloud services through adaptive computation offloading, in: Proceedings of

the 6th International Workshop on Mobile Cloud Computing and Services, 2015, pp. 1–7.

[55] L. Gu, D. Zeng, S. Guo, Vehicular cloud computing: A survey, in: Proceedings of the IEEE Globecom Workshops, 2013, pp. 403–407.

[56] S. Arif, S. Olariu, J. Wang, G. Yan, W. Yang, I. Khalil, Datacenter at the airport: Reasoning about time-dependent parking lot occupancy, IEEE Transactions on Parallel and Distributed Systems 23 (11) (2012) 2067–2080.

[57] T. Zhang, R. E. De Grande, A. Boukerche, Vehicular cloud: Stochastic analysis of computing resources in a road segment, in: Proceedings of the 12th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks, 2015, pp. 9–16.

[58] F. Zhang, R. E. De Grande, A. Boukerche, Accuracy analysis of short-term traffic flow prediction models for vehicular clouds, in: Proceedings of the 13th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks, 2016, pp. 19–26.

[59] F. Zhang, R. E. De Grande, A. Boukerche, Macroscopic interval-split free-flow model for vehicular cloud computing, in: 21st IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications, 2017, pp. 107–114.

[60] G. Castignani, T. Derrmann, R. Frank, T. Engel, Driver behavior profiling using smartphones: A low-cost platform for driver monitoring, IEEE Intelligent Transportation Systems Magazine 7 (1) (2015) 91–102.

[61] D. Garrett, D. Mann, Assessment of electronic sensor data to remotely identify a motor vehicle and monitor driver behavior, uS Patent 20,150,371,465 (2015).

[62] Q. Xu, R. Segupta, D. Jiang, D. Chrysler, Design and analysis of highway safety communication protocol in 5.9 ghz dedicated short range communi-

cation spectrum, in: Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference, Vol. 4, 2003, pp. 2451–2455.

[63] J. B. Kenney, Dedicated short-range communications (dsrc) standards in the united states, Proceedings of the IEEE 99 (7) (2011) 1162–1182.

[64] D. Jiang, L. Delgrossi, Ieee 802.11 p: Towards an international standard for wireless access in vehicular environments, in: Proceedings of the IEEE Vehicular Technology Conference, IEEE, 2008, pp. 2036–2040.

[65] A. Kajackas, A. Vindašius, Š. Stanaitis, Inter-vehicle communication: Emergency message delay distributions, Elektronika ir Elektrotechnika 96 (8) (2015) 55–58.

[66] M. Whaiduzzaman, M. Sookhak, A. Gani, R. Buyya, A survey on vehicular cloud computing, Journal of Network and Computer Applications 40 (0) (2014) 325 – 344.

[67] R. Yu, Y. Zhang, H. Wu, P. Chatzimisios, S. Xie, Virtual machine live migration for pervasive services in cloud-assisted vehicular networks, in: Proceedings of the 8th International ICST Conference on Communications and Networking in China, 2013, pp. 540–545.

[68] T. Refaat, B. Kantarci, H. Mouftah, Dynamic virtual machine migration in a vehicular cloud, in: Proceedings of the IEEE Symposium on Computers and Communication, 2014, pp. 1–6.

[69] V. Soto, R. E. De Grande, A. Boukerche, REPRO: time-constrained data retrieval for edge offloading in vehicular clouds, in: Proceedings of the 14th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks, 2017, pp. 47–54.

[70] L. Gu, D. Zeng, S. Guo, B. Ye, Leverage parking cars in a two-tier data center, in: Proceedings of the IEEE Wireless Communications and Networking Conference, 2013, pp. 4665–4670.

[71] B. Chen, R. Curtmola, G. Ateniese, R. Burns, Remote data checking for network coding-based distributed storage systems, in: Proceedings of the ACM Workshop on Cloud Computing Security Workshop, 2010, pp. 31–42.

[72] W. Kim, M. Gerla, Navopt: Navigator assisted vehicular route optimizer, in: Proceedings of the International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, 2011, pp. 450–455.

[73] M. Gerla, J. T. Weng, G. Pau, Pics-on-wheels: Photo surveillance in the vehicular cloud, in: Proceedings of the International Conference on Computing, Networking and Communications, 2013, pp. 1123–1127.

[74] F. Dressler, P. Handle, C. Sommer, Towards a vehicular cloud - using parked vehicles as a temporary network and storage infrastructure, in: Proceedings of the ACM International Workshop on Wireless and Mobile Technologies for Smart Cities, 2014, pp. 11–18.

[75] H. Yao, C. Bai, D. Zeng, Q. Liang, Y. Fan, Migrate or not? exploring virtual machine migration in roadside cloudlet-based vehicular cloud, Concurrency and Computation: Practice and Experience 27 (18) (2015) 5780–5792, cpe.3642.

[76] R. Kim, H. Lim, B. Krishnamachari, Prefetching-based data dissemination in vehicular cloud systems, IEEE Transactions on Vehicular Technology 65 (1) (2016) 292–306.

[77] M. Ma, Y. Huang, C. H. Chu, P. Wang, User-driven cloud transportation system for smart driving, in: Proceedings of the IEEE 4th International Conference on Cloud Computing Technology and Science, 2012, pp. 658–665.

[78] V. Hahanov, W. Gharibi, B. A. A. Abbas, S. Chumachenko, O. Guz, E. Litvinova, Cloud traffic monitoring and control, in: Proceedings of the

IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems, Vol. 01, 2013, pp. 244–248.

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

[80] M. Gerla, E.-K. Lee, G. Pau, U. Lee, Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds, in: Proceedings of the IEEE World Forum on Internet of Things, 2014, pp. 241–246.

[81] M.-K. Jiau, S.-C. Huang, J.-N. Hwang, A. V. Vasilakos, Multimedia services in cloud-based vehicular networks, IEEE Intelligent Transportation Systems Magazine 7 (3) (2015) 62–79.

[82] W. Q. Wang, X. Zhang, J. Zhang, H. B. Lim, Smart traffic cloud: An infrastructure for traffic applications, in: Proceedings of the IEEE 18th International Conference on Parallel and Distributed Systems, 2012, pp. 822–827.

[83] M. H. Eiza, Q. Ni, Q. Shi, Secure and privacy-aware cloud-assisted video reporting service in 5g-enabled vehicular networks, IEEE Transactions on Vehicular Technology 65 (10) (2016) 7868–7881.

[84] N. Kumar, K. Kaur, S. C. Misra, R. Iqbal, An intelligent rfid-enabled authentication scheme for healthcare applications in vehicular mobile cloud, Peer-to-Peer Networking and Applications (2015) 1–17.

[85] M. Saini, K. M. Alam, H. Guo, A. Alelaiwi, A. El Saddik, Incloud: a cloud-based middleware for vehicular infotainment systems, Multimedia Tools and Applications (2016) 1–29.

[86] Z. You-zhi, Z. Ning, Review and new insights of the car-following model for road vehicle traffic flow, in: Proceedings of the 6th International Asia Conference on Industrial Engineering and Management Innovation, 2016, pp. 87–96.

[87] A. Kesting, M. Treiber, Calibrating car-following models by using trajectory data: Methodological study, Transportation Research Record: Journal of the Transportation Research Board (2088) (2008) 148–156.

[88] M. Treiber, A. Hennecke, D. Helbing, Congested traffic states in empirical observations and microscopic simulations, Physical Review E 62 (2000) 1805–1824.

[89] R. Jiang, Q. Wu, Z. Zhu, Full velocity difference model for a car-following theory, Physical Review E 64 (2001) 017101.

[90] R. Zhong, K. Fu, A. Sumalee, D. Ngoduy, W. Lam, A cross-entropy method and probabilistic sensitivity analysis framework for calibrating microscopic traffic models, Transportation Research Part C: Emerging Technologies 63 (2016) 147 – 169.

[91] S. Panwai, H. Dia, Comparative evaluation of microscopic car-following behavior, IEEE Transactions on Intelligent Transportation Systems 6 (3) (2005) 314–325.

[92] P. Gipps, A behavioural car-following model for computer simulation, Transportation Research Part B: Methodological 15 (2) (1981) 105 – 111.

[93] M. Gramaglia, P. Serrano, J. Hernandez, M. Calderon, C. J. Bernardos, New insights from the analysis of free flow vehicular traffic in highways, in: Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, 2011, pp. 1–9.

[94] D. Chen, J. Zhang, S. Tang, J. Wang, Freeway traffic stream modeling based on principal curves and its analysis, IEEE Transactions on Intelligent Transportation Systems 5 (4) (2004) 246–258.

[95] A. Abdrabou, W. Zhuang, Probabilistic delay control and road side unit placement for vehicular ad hoc networks with disrupted connectivity, IEEE Journal on Selected Areas in Communications 29 (1) (2011) 129–139.

[96] M. Khabbaz, W. Fawaz, C. Assi, A simple free-flow traffic model for vehicular intermittently connected networks, IEEE Transactions on Intelligent Transportation Systems 13 (3) (2012) 1312–1326.

[97] T. Zhang, R. E. De Grande, A. Boukerche, Urban traffic characterization for enabling vehicular clouds, in: Proceedings of the IEEE Wireless Communications and Networking Conference, 2016, pp. 1–6.

[98] T. Zhang, R. E. De Grande, A. Boukerche, Design and analysis of stochastic traffic flow models for vehicular clouds, Ad Hoc Networks 52 (2016) 39–49.

[99] M. Zarei, A. M. Rahmani, Analysis of vehicular mobility in a dynamic free-flow highway, Vehicular Communications 7 (2017) 51–57.

[100] C. Schindelhauer, T. Lukovszki, S. Rührup, K. Volbert, Worst case mobility in ad hoc networks, in: Proceedings of the Fifteenth Annual ACM Symposium on Parallel Algorithms and Architectures, 2003, pp. 230–239.

[101] A. A. Agashe, S. K. Bodhe, Performance evaluation of mobility models for wireless ad hoc networks, in: Proceedings of the First International Conference on Emerging Trends in Engineering and Technology, 2008, pp. 172–175.

[102] H. Ramzan, A. Iqbal, Throughput and connectivity using constraint based mobility model for mobile ad hoc networks, in: Proceedings of the IEEE International Multitopic Conference, 2008, pp. 249–254.

[103] W. Navidi, T. Camp, Stationary distributions for the random waypoint mobility model, IEEE Transactions on Mobile Computing 3 (1) (2004) 99–108.

[104] T. Camp, J. Boleng, V. Davies, A survey of mobility models for ad hoc network research, Wireless Communications and Mobile Computing 2 (5) (2002) 483–502.

Azzedine Boukerche (FIEEE, FEiC, FCAE, FAAAS) is a full professor and holds aCanada Research Chair position at the University of Ottawa(Ottawa). He is the founding director of the PARADIS-EResearch Laboratory, School of Electrical Engineering and Computer Science (EECS), Ottawa. Prior to this, he held afaculty position at the University of North Texas, and hewas a senior scientist at the Simulation Sciences Division,Metron Corp., San Diego. He was also employed as afaculty member in the School of Computer Science, McGillUniversity, and taught at the Polytechnic of Montreal.He spent an year at the JPL/NASA-California Institute ofTechnology, where he contributed to a project centered on specificationand verification of the software used to control interplanetary spacecraft operatedby JPL/NASA Laboratory. His current research interests include Wireless Ad Hocand sensor networks, wireless networks, mobile and pervasive computing, wirelessmultimedia, QoS service provisioning, performance evaluation and modeling oflarge-scale distributed systems, distributed computing, large-scale distributedinteractive simulation, and parallel discrete-event simulation. He has publishedseveral research papers in these areas. He served as a guest editor for theJournal of Parallel and Distributed Computing (special issue for routing for mobilead hoc, special issue for wireless communication and mobile computing, andspecial issue for mobile ad hoc networking and computing), ACM/Kluwer WirelessNetworks, ACM/Kluwer Mobile Networks Applications, and Journal of WirelessCommunication and Mobile Computing. He has been serving as an Associate Editor of

ACM Computing Surveys, IEEE Transactions on Parallel andDistributed systems, IEEE Transactions on Vehicular Technology, Elsevier AdHoc Networks, Wiley International Journal of Wireless Communication andMobile Computing, Wileys Security and Communication Network Journal,Elsevier Pervasive and Mobile Computing Journal, IEEE Wireless CommunicationMagazine, Elseviers Journal of Parallel and Distributed Computing,and SCS Transactions on Simulation. He was the recipient of the Best ResearchPaper Award at IEEE/ACM PADS 1997, ACM MobiWac 2006, ICC 2008, ICC 2009 andI-WCMC 2009, and the recipient of the Third National Award for TelecommunicationSoftware in 1999 for his work on a distributed security systems on mobile phoneoperations. He has been nominated for the Best Paper Award at the IEEE/ACMPADS 1999 and ACM MSWiM 2001. He is a recipient of an Ontario Early Research Excellence Award(previously known as Premier of Ontario Research Excellence Award), OntarioDistinguished Researcher Award, Glinski Research Excellence Award, IEEE CS Golden Core Award,IEEE Canada Gotlieb Medal Award, IEEE ComSoc Expectional Leadership Award, IEEE TCPP Excpetional Leadership Award.He is a cofounder of the QShine International Conference on Quality ofService for Wireless/Wired Heterogeneous Networks (QShine 2004). He served asthe general chair for the Eighth ACM/IEEE Symposium on Modeling, Analysis andSimulation of Wireless and Mobile Systems, and the Ninth ACM/IEEE Symposiumon Distributed Simulation and Real-Time Application (DSRT), the program chairfor the ACM Workshop on QoS and Security for Wireless and Mobile Networks,ACM/IFIPS Europar 2002 Conference, IEEE/SCS Annual Simulation Symposium(ANNS 2002), ACM WWW 2002, IEEE MWCN 2002, IEEE/ACM MASCOTS 2002, IEEEWireless Local Networks WLN 0304; IEEE WMAN 0405, and ACM MSWiM 9899,and a TPC member of numerous IEEE and ACM sponsored conferences. He served asthe vice general chair for the Third IEEE Distributed Computing for Sensor Networks(DCOSS) Conference in 2007, as the program cochair for GLOBECOM 20072008Symposium on Wireless Ad Hoc and Sensor Networks, and for the 14th IEEE ISCC2009 Symposium on Computer and Communication Symposium, and as the financechair for ACM Multimedia 2008. He also serves as a Steering Committee chairfor the ACM Modeling, Analysis and Simulation for Wireless and Mobile SystemsConference, the ACM Symposium on Performance Evaluation of Wireless Ad Hoc,Sensor, and Ubiquitous Networks, and IEEE/ACM DSRT.

Robson Eduardo De Grande is currently an Assistant Professor in the Department of Computer Science at Brock University. He was formerly the Network Manager of the NSERC DIVA Strategic Research Network and a Senior Research Associate at the School of Electrical Engineering and Computer Science, University of Ottawa, Canada. He received his Ph.D. degree at the University of Ottawa in 2012 and his M.Sc. and B.Sc. degrees in Computer Science from the Federal University of Sao Carlos, Brazil in 2006 and 2004, respectively. He served as TPC Chair of IEEE/ACM DS-RT 2014. His research interests include large-scale distributed and mobile systems, Cloud computing, high performance computing, performance modeling and simulation, computer networks, vehicular networks, and distributed simulation systems.