Mobile Edge Computing versus Fog Computing in Internet of Vehicles

Eugen Borcoci, Marius Vochin, Serban Obreja
University POLITEHNICA of Bucharest - UPB
Bucharest, Romania
Emails: eugen.borcoci@elcom.pub.ro, marius.vochin@upb.ro, serban@radio.pub.ro

***Abstract* —Vehicular networks and the recent Internet of Vehicles (IoV) are continuously developing, aiming to solve the current and novel challenging needs in the domain of transportation systems. Edge computing offers a natural support for Internet of Vehicles, supporting fast response, context awareness, and minimization of the data transfer to the centralized data centers - all these being allowed by the edge computing availability close to mobile vehicles. Multi-access (Mobile) Edge Computing, fog computing, cloudlets, etc., are such candidates to support IoV; their architectures and technologies have overlapping characteristics but also differences in approach. A full convergence between them has not yet been achieved. Also, it is still not completely clarified which solution could be the best trade-off to be adopted in the Internet of vehicles context and for which use cases. This paper is not a complete survey, but attempts a preliminary evaluation of some of the currently proposed Mobile Edge Computing and fog computing solutions for vehicular networks.**

Keywords — Internet of Vehicles; Vehicular Networks; Fog computing; Edge computing; Software Defined Networking; Network Function Virtualization.

I. INTRODUCTION

Vehicular communications, networks and associated services constitute a significant area of research, development and implementation in the framework of the *Intelligent Transport System* (ITS) [1] and, more recently, *Internet of Vehicles.* Supporting networking technologies have been developed, e.g., *Dedicated Short-Range Communications* (DSRC) and also higher functional layers such as the *Wireless Access in Vehicular Environments* (WAVE) [2]. The IEEE 802.11a/p and respectively IEEE 1609 represent a mature set of standards for DSRC/WAVE networks. For wide area, 4G, *Long Term Evolution* (Advanced) (LTE-A) are used and, in the future, 5G is a strong candidate.

Vehicular Ad Hoc Networks (VANET) [3] (a larger class will be denoted as VNET) generally VNETs have been defined to support basic communications types: vehicle to vehicle (V2V), - to road (V2R), or - to Infrastructure (V2I) in uni- or bi-directional mode (note that, some authors include V2R into V2I type). The basic VANET functional components are the *On-Board-Unit* (OBU), inside the vehicles and *Road-Side–Unit* (RSU) placed on the roads. The RSUs communicate with vehicles through wireless access and among them via external networks. The main use cases of VANET have been oriented to safety and traffic management.

Novel Internet of Vehicles architectures have been recently proposed to extend VANETs and aiming a global span of a vehicle network [4-7]. IoV can be also considered as a special case of *Internet of Things* (IoT) [8], where the "things" are either vehicles or their subsystems. The IoV connects the vehicles and RSUs through different *Wireless/Radio Access Technologies* (WAT/RAT), while traditional Internet and other heterogeneous networks cover the wide area. IoV aims to serve a large range of applications and use cases, including those coming from ITS, V(A)NETs and other novel ones:

- Safety and management oriented

Safety: emergency call, warnings (wrong-way, lane change, overtaking), automatic breaking, automatic speed control; *Traffic and navigation management*: real time traffic information, parked car and parking space locating, parking space offers and booking, speeding evidence, navigation area extension, multi-modal transportation, traffic signaling, localizing events, logging, etc.; *Remote telematics*: car surveillance, fuel usage optimization, remote locking/unlocking, stolen vehicle recovery, driving behavior analysis, diagnostic actions, etc.

- Business oriented

Infotainment: Wi-Fi in vehicle, content downloading, online radio and streaming, SMS, advertisements, calendar and address book, Facebook/WhatsApp, location sharing/tracking family/friends, connected drive; *Insurance*: group/family/usage/season/region-based; *Car Sharing*: booking for family/group car/group-parking, car pooling and sharing; *Other services*: cloud/fog/edge various services, mobile toll payment, driving behavior analysis, etc.

To develop the above applications, IoV can take benefit from centralized *Cloud Computing* (CC) combined with *Edge Computing (EC)* [9][10].

EC moves the cloud computing capabilities (computation, storage) at the network edge, thus offering for IoV more appropriate features than CC: faster time response, more flexibility in functional distribution, context awareness, resource usage optimization and reduction in the amount of data exchanged between a cloud data center and a vehicle.

We adopt here a general view, that EC can represent any set of computing and network resources distributed along the path between data sources and cloud data centers.

However, there is not yet a unique vision on “edge” semantics, except the common attribute of proximity to the data sources. Note that some studies see the EC as restricted to edge devices only (some edge nodes are defined, which can be composed of smart sensors, smart phones, and smart vehicles, even a special edge servers). Currently, significant overlap exists between particular EC architectures and convergence is predicted in the near future. However, several sets of EC specifications have been elaborated by independent organizations. The most relevant ones, are *Multi-access (Mobile) Edge Computing* (MEC) [11][12] and *Fog computing* (FC) [13]-[15]. Their deployment is strongly supported by industry and operators.

Virtualization technologies constitute an important “tool” in developing edge computing. In the architectural management and control planes, *Software-defined networking* (SDN) [16] and *Network Function Virtualization* (NFV) [17] are seen as a strong EC and IoV support, given their features like flexibility, programmability, abstraction via virtualization, dynamicity, etc. SDN decouples the data/user plane from control plane and logically centralizes the control. NFV moves into software many network functions that traditionally have been implemented by dedicated hardware and software. SDN and NFV can be applied independently, but their cooperation (they could be seen as “orthogonal”) is considered as a powerful approach.

MEC, FC, cloudlets, etc., are attractive for V(A)NET and IoV. They have overlapping characteristics but also expose differences in approach. The best trade-off to be adopted in a specific IoV context is still not clear. This paper attempts a preliminary comparison of some MEC and FC solutions for IoV.

Note that, given the limited dimension of this paper, it is not claimed to be a complete survey on the topics. The analysis is high level only; the aim is not to detail certain functions or services, but to evaluate several variants of solutions and some guidelines for selection of an approach appropriate for IoV specific use cases. While they are important in vehicular environment, some topics are not discussed in this study, e.g., security and privacy issues.

The paper is organized as follows. Section II is a very short overview of MEC/Fog related work. Section III is a summary overview of IoV layered architectures. Section IV selects samples of solutions to realize VANETs/IoV in MEC or FC approaches, including some SDN/NFV based. Section V tries to identify some pros and cons for MEC and FC variants, while emphasizing the points of convergence. Section VI contains conclusions and possible future work.

II. RELATED WORK: EDGE COMPUTING ARCHITECTURES

This section presents a very short summary of the MEC and FC architectures; both of them can be considered as strong candidates to support IoV. More comprehensive descriptions can be found in references [10]-[15].

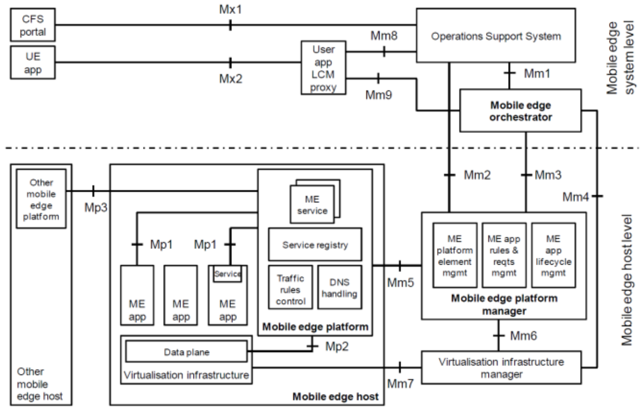

Figure 1. MEC reference architecture (ETSI)[11][12]

A. Multi-access(Mobile) Edge Computing

MEC architecture promoted mainly by ETSI [11][12], offers low latency/response time, high bandwidth, location and context awareness, reduction in amount of data transferred from/to a terminal device to a centralized cloud data center, etc.

ETSI has established in 2014 the MEC Industry Specification Group which provided first specifications. In 2017, the MEC name (and scope) has been extended to *Multi-access Edge Computing* [12] - to include non-cellular and fixed access cases. MEC supports multi-services and multi-tenancy; third-parties may also make use of the MEC storage and processing capabilities.

The MEC resources are placed at the network edge (e.g., in *Radio Access Network* – RAN, i.e., Base Stations, or in aggregation points, etc.). The key element is MEC *mobile edge host* (MEH) playing the role of an application server. It is integrated in RAN and provides computing resources, storage capacity, connectivity, and access to user traffic, radio and network information.

The MEC reference architecture is presented in Figure 1 (details, in [11][12]). The mobile edge host level is the main MEC sub-system, composed of: the mobile edge host (MEH) and its management. The MEH includes a virtualization infrastructure (based on *Network Function Virtualisation Infrastructure* – NFVI - coming from ETSI NFV framework) and the *mobile edge platform* (MEP), supporting the execution of mobile edge applications.

The MEC server can be installed in various places at the network edge: at the 4G/LTE macro base station (eNB); at the multi-technology (3G/LTE) cell aggregation site; at the Radio Network Controller (RNC) site, for 3G. MEC is seen as an efficient technology to support V(A)NET/IoV [4]-[7]. Vehicles connected to the distributed edges may send/receive information to/from other vehicles or through the network, almost in real-time.

B. Fog Computing

Fog computing (FC) [13]-[15] is another recent EC technology complementary to CC (FC will not replace the CC, but cooperation cloud/fog is envisaged). The FC distributed platform brings computation and storage close to its data sources, to reduce the latency and cost of delivering data to a remote cloud. FC has been proposed originally to support the IoT, introduced by Cisco [13].

An important FC-related document is taken by the *OpenFog Consortium* (2015) [15]. That is why this section dedicatees more space to it. OpenFog consortium defines FC as a system-level horizontal architecture that distributes resources and services of computing, storage, control and networking *anywhere along the continuum* from a cloud data center down to things. On the other side, *MEC*, originally targets only the very edge part of the network (e.g., RAN). That is why, some authors consider MEC as a special case of FC.

FC can support multiple industry verticals and application domains delivering intelligence and services to users and business. FC capability is spanning across multiple protocol layers and is not dependent on specific access systems. FC focuses the processing efforts outside the cloud data center i.e., in the fog area. Data are gathered, processed, and stored within the network, by way of an IoT gateway (GW) or an FC node (FN). Information is transmitted to this GW from various sources and it is processed in FN; then pertinent data (plus additional command - if necessary), are transmitted back, towards the devices. A FN can process data received from multiple end points and send information exactly where it is needed.

OpenFog Consortium has defined a flexible deployment hierarchical model for FC, IoT oriented, as presented in Figure 2 [15]. Several use cases can be accommodated in this model.

The case 1 shows a FC-based only system (the CC cannot be used for some reasons like response time, special requirements, special transportation systems environments, unavailability, etc.). In the case 2, the operation-centric information processing is done by FNs located close to the infrastructure/process being managed while cloud processing is performed only for event-to-action time window ranging from hours to months. The case 3 shows the local fog infrastructure used for time-sensitive computation, while the cloud is used for the balance of operational and business-related information processing (commercial device monitoring, mobile network acceleration, content delivery networks – CDNs).

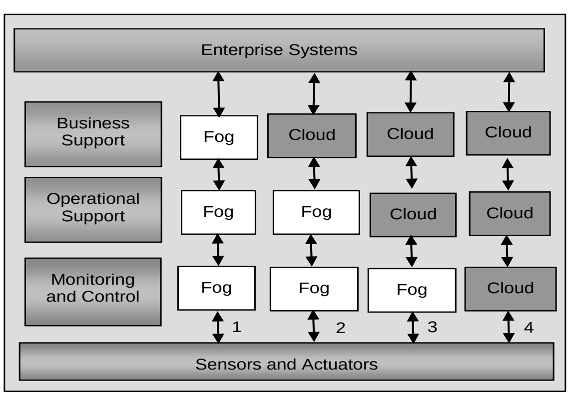

Figure 2. IoT System deployment models - variants [adapted from [14]]

The case 4 supports use cases like agriculture, connected cars, and remote weather stations. The cloud is used for the entire stack due to the constrained environments in which the deployment of fog infrastructure may not be feasible or economical. However, FNs at the device layer may get some of the monitoring and control functions for safety related control. The enterprise systems integrate with cloud for business operations.

Recently the work [18] extended the FC scope, by defining *Fog of Everything (FoE)* to serve *Internet of Everything (IoE).* The FNs are usually virtualized networked data centers, which run on top of (typically, wireless) *Access Points* (APs), at the edge of the access network, resulting in a three-tier IoE-Fog-Cloud hierarchy. In this context, a “thing” (fixed, nomadic or mobile) is a resource-limited user device that needs resource augmentation in order to execute its workload. The work [18] proposes a hierarchical general architecture for a FoE virtualized platform, integrating the building blocks:

- *IoE layer*, where a number of (possibly, heterogeneous) things operate over multiple spatial clusters;
- *wireless access* network (fixed/mobile), to support *Fog-to-Thing* (F2T) and *Thing-to-Fog* (T2F) communication through TCP/IP connections running atop, e.g., *IEEE802.11/15* single-hop links;
- a set of *inter-connected FNs*, that act as virtualized cluster headers;
- *inter-Fog backbone* (wireline/wireless) providing inter-Fog connectivity and making feasible inter-Fog resource pooling;
- *virtualization layer*, allowing things to augment their limited resources by exploiting the computing capability of a corresponding virtual clone. This last one runs atop a physical server of the FN that currently serves the cloned thing;
- the resulting *overlay inter-clone virtual network*, that allows P2P inter-clone communication by relying on TCP/IP E2E connections.

The corresponding protocol stack [17] comprises four layers:

IoE layer provides services like: *(a)* T2F access through a reservation-based collision free access protocol for the things served by a same FN; *(b)* F2T broadcast services.

Fog layer performs: *(a)* energy-efficient management of the networking and computing physical resources equipping each FN, and *(b)* energy-efficient management of the inter-Fog traffic conveyed by the wireless backbone.

Overlay layer supports the overlay inter-clone P2P network by: (a) inter-Fog clone migration; it can be supported by the implementation of the so-called *Follow-Me-Cloud* framework (e.g., Taleb et al., [18]), to solve “live” inter-Fog clone migration, in response to the thing mobility; (b) dynamic management of the required migration bandwidth, to minimize the energy consumed by clone migrations.

Cloud layer orchestrates the overall Cloud-Fog-IoE platform based on the specific features and requirements of the running applications. The solutions must be tailored on the expected attributes of the supported applications.

III. RELEVANT IOV LAYERED ARCHITECTURES

This sub-section shortly presents some recent IoV architectures, given that edge computing functionalities should be integrated in such IoV architectures. Some relevant ones have been selected.

Among the first proposals, there is Bonomi et al. [8] four-layered architecture, for connected vehicles and transportation. A layer includes groups of functions, which could be mapped on one or more classical TCP/IP layers. Also, the four layers correspond to different geo-locations of the subsystems (vehicles, networking infrastructure, cloud data centers, etc.). The bottom layer (L1-end points) represents the vehicles, and their communication protocols (basically for V2V communication, using the IEEE 802.11a/p). The L2 (infrastructure), represents communication technologies to interconnect the IoV actors (via WiFi, 802.11p, 3G/4G, etc.). The L3 (operation) performs management; it verifies and ensures compliance with all applicable policies, to regulate the information management and flow. The L4 (cloud- public, private or enterprise) is based on a defined profile coupled with the possibility of receiving services (voice, enterprise video and data) on demand. This architectural view is still defined at a high-level view and does not detail the mapping of the functions sets to different levels.

Kayvartya et al. [4] have proposed a comprehensive IoV five-layer architecture, to support an enriched set of vehicular communications, in addition to traditional V2V, V2R/V2I, i.e., *Vehicle-to-Personal* devices (V2P) and *Vehicle-to-Sensors* (V2S). Each particular IoV communication type can be enabled using a different WAT, e.g., IEEE WAVE for V2V and V2R, Wi-Fi and 4G/LTE for V2I, CarPlay/NFC (*Near Field Communications*) for V2P and WiFi for V2S. The system includes vehicles and *Road Side Units* (RSU), but also other communication devices. Embedding such a large range of devices makes the IoV more complex, (compared to VANET), but more powerful and market oriented. Three architectural planes are defined: *management*, *operation* and *security*. This allows mapping of various existing protocols and functions (e.g., taken from ITS) to architectural layers. The network model is composed of three functional entities: *client, connection* and *cloud.*

The five layers in [4] are: *perception, coordination, artificial intelligence, application* and *business.* The perception layer (PL) includes the traditional physical layer functions and some additional for sensing and actuating actions. The coordination layer (CL) represents a virtual universal network coordination entity for heterogeneous network technologies (WAVE, Wi-Fi, 4G/LTE, satellites, etc.). The artificial intelligence layer (AIL) is represented by a generic virtual cloud infrastructure, working as an information processing and management centre. It stores, processes and analyzes the information received from the lower layer and then takes decisions. Its major components are: *Vehicular Cloud Computing (VCC), Big Data Analysis (BDA)* and *Expert System.* The AIL should meet the requirement of applications and services of the AL. The application layer (AL) contains smart applications (e.g., for traffic safety and efficiency, multimedia-based infotainment and web-based utility). The *business* layer (BL) includes IoV operational management functions, basically related to business aspects.

The above 5-layer architecture does not discuss how to distribute computation intelligence between a central cloud and fog/edge units in combined cloud-fog/MEC solution. Neither SDN-like control nor NFV implementation possibilities are discussed.

F. Yang et al. [5] suggest an IoV architecture, by considering the driver-vehicle-environment coordination. IoV is defined as an open converged network system (controllable, manageable, operational, and trustable) based on multi-human, multi-machine, multi-vehicle, and environment coordination. It senses, recognizes, transmits, and computes the large-scale complex static/dynamic information of human, vehicle, network communication and road traffic infrastructure, using advanced ICT technology.

Four layers are defined: the *environment sensing and control* layer, *network access* and *transport* layer, *coordinative computing control* layer, and *application* layer. The work also summarizes the core technologies of each layer. The coordinative computing control layer has a special role to coordinate among human-vehicle-environment. The application layer provides various types of services and is open (i.e., it can support novel services and business operating modes). The types of services can be: closed (related to the specific industry applications) or open (i.e., various existing open applications, such as real-time traffic services provided by Internet service providers or by third party providers). Neither MEC/FC nor SDN/NFV approaches are discussed in [5]. The homogeneity of sub-layers is rather low in terms of their components. No architectural split in planes is proposed; so, it is rather difficult to see how to map different already developed functions and protocols (coming from ITS, WAVE, etc.) to the layers of this architecture; this seems to be still an open issue.

A seven-layer (6+1) IoV architecture is proposed by Contreras-Castillo et al. in [6], to support collaboration between multi-users, multi-vehicles, multi-devices (sensors, actuators, mobile devices, access points), multi-communication models (point to point, multi-point, broadcast, geo-cast) and multi-networks (wireless or wire networks with various technologies like Wi-Fi, Bluetooth, WiMAX, 3G, 4G/LTE, etc.). The layers are (bottom-up list): User interaction (lowest layer), Data acquisition, Data filtering and pre-processing, Communication, Control and management, Business (highest layer). A macro-layer is defined and named Security; it is rather a cross layer entity. The cloud services are located at business level (as vehicular cloud computing) while we believe that a more natural placement is below to application layer. Some mixture of "layers" and "plane" notions is apparent; there is a lack of orthogonality of different "layers". The architecture does not emphasize any MEC/Fog solutions or integration of SDN/NFV approach.

Here, it is considered that Kayvartya et al. [4] architecture is a good IoV model, enough flexible to accommodate computing technologies like MEC and FC.

IV. MEC AND FOG SOLUTIONS INTEGRATED IN IOV

This section presents IoV relevant systems which include MEC and Fog approaches to identify some pros and cons of each solution in the IoV context.

A MEC-based model of a vehicular network is developed by K. Zhang, et al., in [20]. The architectural levels are: *Virtual Computation Resource Pool*-incorporating the network and cloud resources outside the MEC; *MEC level* – implemented as MEC servers placed in the RAN; *RSUs units* placed on the roads; mobile units (vehicles). A special focus is on the computation off-loading process, to preserve the service continuity in a mobile environment. Vehicles in transit may pass through several RSUs and MEC servers during the task-off-loading process, and they can off-load their computation task to any MEC servers that they can access. Two methods are possible: selection of the target MEC servers or selecting (for a while) of a new path from the mobile vehicle to the same MEC server (keeping as much as possible the same serving MEC server in order to avoid too frequent moving of virtual machines).

J. Liu et al., [21] propose an SDN-enabled network architecture assisted by MEC, while integrating different types of access technologies. The architectural components are (top-down hierarchical list): Remote Data Center; Backbone network, Regions (each one contains MEC servers collocated with an SDN controller, BS and mobiles organized in VANETs). The MEC servers can inter-communicate via a mesh of fixed network link. This architecture has an SDN-like control, comprising three planes (Data, Control and Application) each including typical functions. The Data Plane (DPl) includes SDN-"switches" (VANET, BS, Ethernet); lower layer technologies (IEEE 802.11p, LTE/5G, Wire NIC, etc.). The Control Plane (CPl) has two sub-layers: lower sub-layer with functions such as Position/Channel sensing, Flow table management, Forwarding strategy; upper sublayer: Trajectory prediction, Interface sensing, Radio Resource control, Traffic redirection. The Application Plane (APl) (in the SDN semantics) includes Topology management, Resource Management, Traffic Offload, SDN controller.

The interface between CPl and DPl is based on extended OpenFlow or other similar protocol. The details of the layer mapping on SDN/NFV and fog/edge approach are not discussed. The MEC server is considered as belonging to the infrastructure and it is transparent to the client (vehicle). The client can request services from or deliver packets to a remote cloud server. If the requested service is deployed on an MEC server, then the BS redirects it to the MEC server. MEC usually stores recent traffic data and responds to real-time events. RSUs collect the real-time road conditions and deliver them to the MEC server (via BS). The traffic data should be pre-processed by the MEC server before they are delivered to the remote cloud server by means of data synchronization. The remote cloud server stores traffic data permanently and makes a traffic prediction based on real-time and historical data.

K. Zheng et al. [22] propose an IoV architecture called software-defined heterogeneous vehicular network (SERVICE), based on Cloud-RAN technology suitable for the dynamic nature of heterogeneous VANET functions and various applications. A multi-layer Cloud-RAN multi-domain is introduced, where resources can be exploited as needed for vehicle users. The system is hierarchically organized (three levels of clouds are defined: remote, local and micro clouds) and virtualization techniques (offering flexibility) are considered for implementation. However, this work does not map the architecture on specific MEC or fog solutions. The high-level design of the soft-defined HetVNET is presented. The SDN control is organized on two levels (one primary controller and several secondary controllers; each one of the latter controls a given service area). A complete layered functional IoV architecture is not in the paper scope.

A Fog-SDN architecture called FSDN is proposed for advanced VANET by Truong et al. [23], for V2V, V2I and Vehicle-to-Base Station communications. The Fog computing brings more capabilities for delay-sensitive and location-aware services. The SDN components (hierarchically top-down listed) are: SDN Controller (it controls the overall network behavior via OpenFlow – interfaces; it also performs Orchestration and Resource Management activities for the Fog nodes); SDN RSU Controller (RSUC) (controlled by the central SDN controller; each RSUC controls a cluster of RSUs connected to it through broadband connections. The RSUC can forward data, and store local road system information or perform emergency services. From Fog perspective RSUC are fog devices); SDN RSU (it is also a Fog device); SDN Wireless Nodes (vehicles acting as end-users and forwarding elements, equipped with OBU); The system also contains Cellular Base Station (BS) performing traditional functions (they are SDN-controlled via OpenFlow protocol and can also offer Fog services). This study does not map the functions on a full layered architecture.

Kai et al. [23] work presents an overview of Fog–SDN solution for VANET and discuss several scenarios and issues. It is shown that a mixed architecture Fog-SDN can be powerful and flexible enough, to serve future needs of IoV.

OpenFog Consortium presents in [15] a complex system, cloud-fog-based, for Transportation Scenario (Smart Cars and Traffic Control). They took into account the high amount of data generated (multiple terabytes of data every day from the combinations of light detection and ranging (LIDAR), global positioning systems (GPS), cameras, etc.). So, a combined cloud-fog computing approach is required; the system can be supported by OpenFog Reference Architecture. Figure 3 shows an overview of an intelligent highway application of the OpenFog RA.

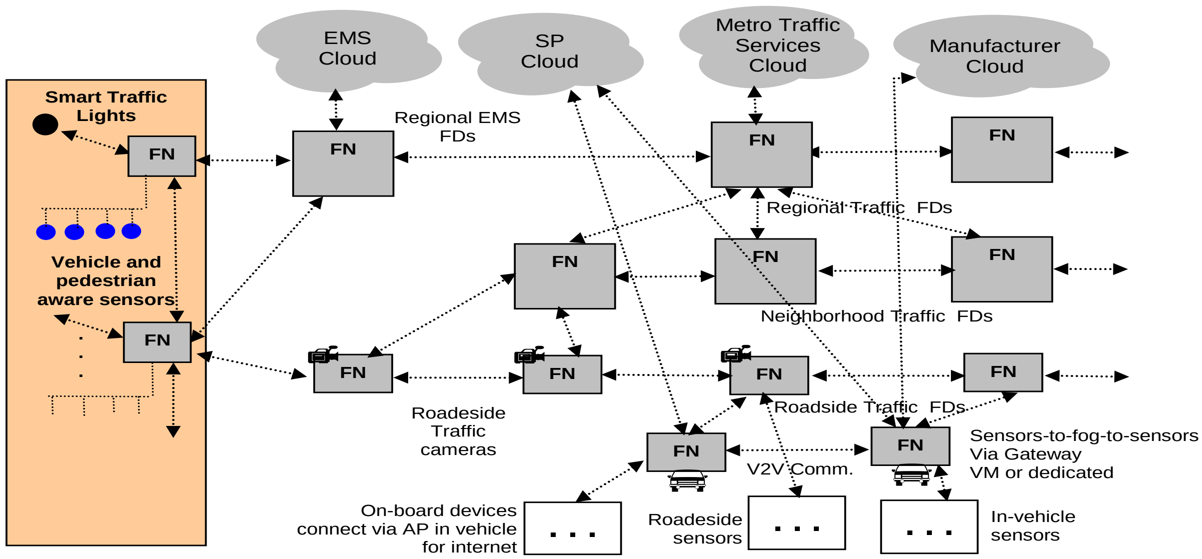

Figure 3. General cloud-fog-based example system for transportation scenario (smart cars and traffic control) [adapted from [14]]
EMS- Element Management System; SP- Service Provider; FD- Fog Device; FN- Fog Node

The system is based on a fog environment containing a rich set of interactions among multiple fog and multiple cloud domains. The fog architecture is hierarchical and distributed which is an important advantage. Some important capabilities of fog technology while applied in IoV domain are illustrated: a rich set of interactions among multiple fog and cloud domains, including Element Management Systems (EMS), service provider (SP), metro traffic services, and system manufacturer clouds; mobile fog nodes supporting V2V, V2I and V2X interactions; multiple fog networks owned and operated by different authorities providing similar (and different) functionality; multi-tenancy across fog nodes allows to consolidate multiple fog networks; both private and public fog and cloud networks used by a single end point device. The system includes several types of sensors (and actuators) referred to as “things.” Things include roadside and on-vehicle entities, to provide data, so that the various systems (lights, cars, etc.) can carry out their functions (e.g., vehicle driving autonomously). Smart transportation systems also manage the actuators that control parts of the infrastructure, such as traffic signals, gates, and digital signs. The vehicles connect to the cloud and a hierarchy of fog nodes that service the autonomous vehicle or traffic control systems.

The vehicle itself can be a mobile FN, communicating with other FNs as they become available. The mobile fog node can perform all required in-vehicle operations autonomously (if it cannot connect to other FNs or to the cloud). In-vehicle fog nodes provide services including infotainment, Advanced Driver Assistance Systems (ADAS), autonomous driving, collision avoidance, navigation, etc. The Transportation Fog Network has a three-level hierarchy of FNs; the first is the infrastructure (roadside) fog nodes. The roadside fog sensors collect data from other devices such as roadside cameras. The FNs perform some local analysis for local actions. Data from the first level is aggregated and sent up to the second and third levels of the hierarchy—neighborhood and regional fog nodes—for further analysis and distribution.

The above use case demonstrates the goal of the OpenFog RA for smart cars and traffic control, i.e., to ensure an open, secure, distributed, and scalable architecture that optimizes real time capabilities within a multi-supplier ecosystem. The transportation example shows a complex system of autonomous things and infrastructure generating massive amounts of data. This use case highlights the need for fog computing to enable safe and effective operations in IoT, 5G, AI and other advanced scenarios.

V. MEC VERSUS FOG IN IOV

The previous section selected some relevant examples of architectures and systems to illustrate the MEC/FC usage in the context of IoV. Note that also other proposals are published in the literature, both for MEC and Fog. While not all aspects of the architectural could be discussed in few examples, some distinctive features can be emphasized.

To a question like “selection of Mobile Edge computing, versus Fog computing for IoV system” one cannot have a unique general answer, given the facts that - both architectures and technologies are edge-oriented; they have as a main idea to move the cloud computing-like capabilities to the networks edges in order to obtain advantages mentioned in section II. The second reason is that a realistic selection could depend significantly on the IoV services needed - out of a large set described in Introduction section.

MEC/FC have quite a lot of common characteristics like: low latency; support for real time interactions, location awareness and mobility and large number of server nodes; geographical distribution proximity to the end devices (single network hop or few hops); service location at the edge of the local network; various working environment outdoor (streets, base stations, etc.) or indoor (houses, cafes, etc.); wireless communication access: WLAN, WiFi, 3G, 4G, ZigBee, etc., or wired communication (part of the IP networks); weak dependence on the quality of core network; low bandwidth costs and energy consumption. However, both have weak computation and storage capabilities, which raises a need for them to cooperate with CC.

Both MEC and FC can benefit from technologies like SDN and NFV in different architectures. Both MEC and Fog can be compliant with the layered architectures described in Section III.

TABLE I. MEC VERSUS FOG DIFFERENCES

Criterion	MEC	Fog computing
Placement of node devices	Servers running in Base Stations Network Controller/Macro Base Station	Anywhere - between end devices and cloud: Routers, Switches, Access Points, Gateways
Compute Distribution and Load Balancing	Employ a strategy of placing servers, apps or small clouds at the edge	Broader architecture and tools for distributing, orchestrating, and securing resources and services across networks
Software Architecture	Mobile Orchestrator based (strongly specified)	Fog abstraction layer based (only partially specified)
Standardization/ specifications	ETSI/	/OpenFog Consortium
Context awareness	High	Medium
Proximity	One hop	One or multiple hops
Access Mechanisms	Mobile networks: 3G/4G/5G	Wi-Fi, Mobile networks, etc.
Virtualization and management mechanisms	Strongly specified by ETSI (NFV framework)	Larger view of virtualization. In progress at OpenFog Consortium
Hierarchical structure of the overall system	Possible	Yes: multiple levels of cooperating nodes, supporting distributed applications
Horizontal scalability	Medium	High
Internode Communication	Possible - between Mobile Edge Hosts	Native support for communication between Fog nodes
Communication with Cloud	Possible	Fog-cloud is usually considered necessary
Modular architecture with multiple access modes	Edge deployments are typically based on gateways with fixed functionality. However they can be made more flexible and dynamic by using NFV.	Highly modular architecture; every Fog node has exactly the resources its applications need; it can be dynamically configured.
Topology of server nodes	Less flexible (limited by RAN spread)	More general and very flexible
5G compliant specifications	Full compliancy	Work in progress

There are also differences between FC and MEC from several points of view (see also [25][26]), as summarised in Table I. The presented criteria can serve in order to make a selection of MEC/FC in a specific IoV use case.

The MEC/FC paradigm can offer, in the context of IoV, support for a large variety of applications, use-case scenarios, and heterogeneous end devices. On the other side, different use cases and applications might have their own set of requirements and trade-offs which can determine which solution between MEC or FC is the appropriate choice.

Note that, for a given set of use cases to be provided by an IoV system, the problem of selecting MEC or FC approach is multi-criteria one. Among the parameters for selection there are those presented in Table I, where appropriate weights should be assigned to them.

Last but not least, one has to consider the strong effort for cooperation between different organizations, towards a convergence of vision in the domain of edge computing (including MEC, Fog, Cloudlets, etc.)

VI. CONCLUSIONS AND FUTURE WORK

This paper presented a preliminary comparative view of Mobile Edge Computing and Fog computing, used as support technologies in Internet of Vehicles, from architectural and technological point of view. A comparison of the technologies has been performed in Section V, identifying the common MEC/FC characteristics and also differences which could be considered when selection of MEC/FC has to be done (depending on the target use cases) in order to implement a given IoV system.

The conclusion of this study is that given the large variety of target IoV systems and use case envisaged, there is no winner MEC versus FC technology, but a selection should be done for each specific case in a *multi-criteria mode*. Different priorities can be assigned to criteria, depending on specific needs of use and business case.

However, some general guidelines can be expressed. MEC approach is more restricted than FC in terms of network dimension and vertical hierarchy, but the IoV development based on MEC can benefit form: detailed elaborated specifications coming from ETSI for MEC; powerful virtualization support defined by NFV technology which is fully compliant with MEC; SDN/NFV approach can be naturally applied in MEC implementation; resource management, mobility and task offloading are aspects better defined in terms of solutions in MEC framework than in fog computing.

Fog computing solutions for IoV have the advantage of being more general in terms of hierarchization, flexibility, geographical span, extension on the core network of FC capabilities. However, if selecting a fog computing solution for IoV then additional challenges should be considered [14], in comparison with traditional fog computing: the edge nodes can be highly mobile causing possible intermittent loss of connection to the remote cloud servers; the computation can be based on vehicular control engines, and therefore accuracy and safety criticality must be ensured; access control is important for vehicular fog computing environments and should be enforced sometimes in real-

ISBN: 978-1-61208-662-0

time mode to prevent delays of some critical decision related to traffic; in a vehicular environment, failure or sporadic behaviors of a few sensor nodes may affect the control decisions taken over a fog (ensuring correctness of the local computation needs to be ensured for intelligent or autonomous vehicles).

Future work should be done to detail some more specific and also quantitative problems for both MEC and Fog approaches in IoV, like resource management and scheduling in the virtualization context, the computation tasks off-loading (in a mobile context) problems to assure the service continuity, MEC/FC in network slicing context, security and privacy aspects, multi-tenancy capabilities, etc.

References

[1] ETSI EN 302 665 V1.1.0 European Standard Telecommunications series, "Intelligent Transport Systems (ITS);Communications Architecture" (2010-07).

[2] Y. Li, "An Overview of the DSRC/WAVE Technology", https://www.researchgate.net/publication/288643779, 2012, [Retrieved: May, 2017].

[3] S. Sultan, M. Moath Al-Doori, A.H. Al-Bayatti, and H.Zedan "A comprehensive survey on vehicular Ad Hoc Network", J.of Network and Computer Applications, Jan. 2014, https://www.researchgate.net/publication/259520963, [Retrieved: August, 2018].

[4] O. Kaiwartya, A.H. Abdullah, Y. Cao, A. Altameem, and M. Prasad, "Internet of Vehicles: Motivation, Layered Architecture, Network Model, Challenges, and Future Aspects", IEEE Access, Special Section on Future Networks, Architectures, Protocols and Applications, Vol. 4, pp.5536-5372, September 2016.

[5] F. Yang, J. Li, T. Lei, and S. Wang, "Architecture and key technologies for Internet of Vehicles: a survey", Journal of Communications and Information Networks, Vol.2, No.2, June 2017, DOI: 10.1007/s41650-017-0018-6.

[6] J. C. Contreras-Castillo et al., "A seven-layered model architecture for Internet of Vehicles", Journal of Information and Telecommunications , Vol. 1, No. 1, pp. 4–22, 2017.

[7] Y. Fangchun, W.Shangguang, L. Jinglin, L. Zhihan, and S.Qibo, "An overview of Internet of Vehicles", China Commun., vol. 11, no. 10, pp. 115, October 2014.

[8] F. Bonomi, "The smart and connected vehicle and the Internet of Things", San José, CA: WSTS, 2013, https://tf.nist.gov/seminars/WSTS/PDFs/1-0_Cisco_FBonomi_ConnectedVehicles.pdf, [August, 2018].

[9] K. Tocze and S. Nadjm-Tehrani, "A Taxonomy for Management and Optimization of Multiple Resources in Edge Computing", arXiv: 1801.05610v1 [cs.DC] 17 Jan 2018, [Retrieved: May, 2018].

[10] N. Mohan and J. Kangasharju, "Edge-Fog Cloud: A Distributed Cloud for Internet of Things Computations", arXiv:1702.06335v2 [cs.DC] 7 Mar 2017, [Retrieved: March, 2018].

[11] "Mobile edge computing (MEC); Framework and reference architecture", ETSI, Sophia Antipolis, France, Mar. 2016. Available: http://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf [Retrieved: January, 2018].

[12] T. Taleb, K.Samdanis, B.Mada, H. Flinck, S.Dutta, and D.Sabella, "On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration", IEEE Comm. Surveys &Tutorials, Vol19, No.3, pp.1657-1681, 2017.

[13] F. Bonomi, R. Milito, J. Zhu, and Sateesh Addepalli, "Fog Computing and Its Role in the Internet of Things", August 2012, http://conferences.sigcomm.org/sigcomm/2012/paper/mcc/p13.pdf, [Retrieved: August, 2018].

[14] S.B.Nath, H.Gupta, S.Chakraborty and S.K Ghosh, "A Survey of Fog Computing and Communication: Current Researches and Future Directions", https://arxiv.org/pdf/1804.04365.pdf, [Retrieved: June, 2018].

[15] OpenFog Consortium (2015), OpenFog Reference Architecture for Fog Computing,

[16] http://www.openfogconsortium.org/resources/#definition-of-fog-computing, [Retrieved: August, 2018].

[17] B. N. Astuto, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks", Communications Surveys and Tutorials, IEEE Communications Society, (IEEE), 16 (3), pp. 1617 – 1634, 2014.

[18] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network Function Virtualisation: Challenges and Opportunities for Innovations". IEEE Communications Magazine, pp. 90-97, February 2015.

[19] E. Bacarelli, P. G. Vinueza Naranjo, M. Scarpinti, M. Shojafar, and J. H. Abawaji, "Fog of Everything: Energy-Efficient Networked Computing Architectures, Research Challenges, and a Case Study", IEEE Access, Vol.5, 2017, pp. 9882-9910.

[20] T. Taleb and A. Ksentini, "Follow me cloud: Interworking federated clouds and distributed mobile networks" *IEEE Network*, vol. 27, no. 5, pp. 12_19, Sep./Oct. 2013.

[21] K. Zhang, Y. Mao, S. Leng,Y. He, and Y. Zhang, "Mobile-Edge Computing for Vehicular Networks- A Promising Network Paradigm with Predictive Off-Loading", IEEE vehicular technology magazine, pp. 36-44, June 2017.

[22] J. Liu, J. Wan, B. Zeng, Q. Wang, H. Song, and M. Qiu, "A Scalable and Quick-Response Software Defined Vehicular Network Assisted by Mobile Edge Computing", IEEE Communications Magazine, pp. 94-100, July 2017.

[23] K. Zheng, L. Hou, H. Meng, Q. Zheng, and N. Lu, "Soft-Defined Heterogeneous Vehicular Network: Architecture and Challenges", IEEE Network, vol. 30, pp. 72-79, July/August 2016.

[24] K. Kai, W. Cong, and L. Tao, "Fog computing for vehicular Ad-hoc networks: paradigms, scenarios, and issues", The Journal of China Universities of Posts and Telecommunications, www.sciencedirect.com/science/journal/10058885, http://jcupt.bupt.edu.cn, 23(2), pp. 56–65, April 2016, [Retrieved: August, 2018].

[25] N. N. Truong, G. M. Lee, and Y. Ghamri-Doudane, "Software defined networking-based vehicular ad hoc network with fog Computing", Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM'15), May 2015, Ottawa, Canada. Piscataway, NJ, USA: IEEE, pp. 1202−1207, 2015.

[26] K. Dolui and S. K. Datta, "Comparison of Edge Computing Implementations: Fog Computing, Cloudlet and Mobile Edge Computing", Global Internet of Things Summit (GIoTS), 2017, https://ieeexplore.ieee.org/document/8016213/, [Retrieved: January, 2018].

[27] OpenFog Consortium, "10 ways fog computing extends the edge", August 21, 2017, https://www.rtinsights.com/10-ways-fog-computing-extends-the-edge/, [Retrieved: August, 2018].