

Software Defined Networking-based Vehicular Adhoc Network with Fog Computing

Nguyen B. Truong

Dasan Networks Corp.,
Bundang-gu, Seongnam-si,
Gyeonggi-do, 463-400 South Korea
nguyentb@postech.ac.kr

Gyu Myoung Lee

School of Computing and
Mathematical Sciences,
Liverpool John Moores University
Liverpool, L3 3AF, UK
g.m.lee@ljmu.ac.uk

Yacine Ghamri-Doudane

L3i Lab, Science&Technologies Pole
University of La Rochelle
Avenue Michel Crépeau, 17042 La
Rochelle CEDEX 1, France
yacine.ghamri@univ-lr.fr

Abstract—Vehicular Adhoc Networks (VANETs) have been attracted a lot of research recent years. Although VANETs are deployed in reality offering several services, the current architecture has been facing many difficulties in deployment and management because of poor connectivity, less scalability, less flexibility and less intelligence. We propose a new VANET architecture called FSDN which combines two emergent computing and network paradigm Software Defined Networking (SDN) and Fog Computing as a prospective solution. SDN-based architecture provides flexibility, scalability, programmability and global knowledge while Fog Computing offers delay-sensitive and location-awareness services which could be satisfy the demands of future VANETs scenarios. We figure out all the SDN-based VANET components as well as their functionality in the system. We also consider the system basic operations in which Fog Computing are leveraged to support surveillance services by taking into account resource manager and Fog orchestration models. The proposed architecture could resolve the main challenges in VANETs by augmenting Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), Vehicle-to-Base Station communications and SDN centralized control while optimizing resources utility and reducing latency by integrating Fog Computing. Two use-cases for non-safety service (data streaming) and safety service (Lane-change assistance) are also presented to illustrate the benefits of our proposed architecture.

Index Terms—Software Defined Network, SDN, Fog Computing, Edge Computing, VANET, OpenFlow, Road-side-unit, RSU, Road-side-unit Controller, RSUC, Base Station, Lane Change, Data Streaming.

I. INTRODUCTION

Vehicular Adhoc Network (VANET) is a special case of Mobile Adhoc Network (MANET) in which mobile nodes are vehicles. The distinctive characteristics of VANET are the nodes are high mobility and tend to follow organized routes instead of moving at random. Furthermore, vehicles not only communicate with each other but also connect to Road-Side-Units (RSUs) and Base Station.

VANETs are expected to support various kinds of services such as route planning, traffic alert dissemination, context-aware infotainments, and mobile vehicular cloud services [1, 2]. Although some route planning and surveillance services are already deployed in reality; many challenges still exist in VANET systems because of difficulties in both deployment

and management. Some of the challenges seem to be unsolvable with the current architecture such as exponentially increasing in the number of connected devices, efficient resources utilization, unbalanced flow traffic, geo-graphical addressing, delay constraint due to high mobility and unreliable connectivity; and Quality of Service (QoS) [3]. Although self-organizing, distributed network fashion like MANET could be applied to “fill the gaps” in these scenarios, the lack of a dedicated mechanism for resources and connectivity management makes some services infeasible. Proposing novel network architectures are needed to deal with these issues.

In order to address the challenges, we consider two emerging networking paradigms; Software Defined Networking (SDN) and Edge Computing (Fog Computing). Recently, SDN has been considered as a new revolutionary thinking in networking, an alternative to traditional distributed approaches which controls the network in a centralized, systematic, programmable manner by decoupling the forwarding functions (data plane) and network controls (control plane). Until now, most of SDN applications are limited to wired environment, especially for data centers such as Google, Facebook and Cisco [4]. However, the potential of flexibility, programmability and centralized knowledge in SDN makes it an attractive approach to mobile wireless networks like VANETs. The separation of data plane and control plane simplifies network management which is extremely complicated when the number of nodes dramatically increases [5, 6, 7, 8] while the global, centralized intelligence will enable and improve VANET services. For supporting some safety services and traffic management which require local information and real-time processing, we integrate Fog Computing to our architecture. Fog is considered as a cloud server operating at the edge of the network; offering special services which require network context information, location awareness, and ultra-low latency; which could be satisfy the demands of future VANETs scenarios [9, 10, 11].

This proposed architecture could resolve the main challenges in VANETs by augmenting an SDN centralized control which takes into account various heterogeneous characteristics such as physical medium, mobility, topology, and capability; Vehicle-to-Vehicle (V2V), Vehicle-to-Base Station and Vehicle-to-Infrastructure (V2I) communications.

II. BACKGROUND AND RELATED WORK

In this section, we briefly introduce basic ideas and architectures of the SDN and Fog Computing. Several researches related to SDN architecture in wireless environment are also mentioned to steer the research line.

A. Software Defined Networking and OpenFlow

The core idea of SDN is the separation between network control (control plane) and forwarding functions (data plane). In SDN, network devices such as switches and routers just simply forwards packets according to policies (rules) located in each device. These rules are created or modified at a controller before sending to the network devices. The controller is the control logic dictating the overall network behavior. Figure 1 illustrates the reference architecture of the SDN [12].

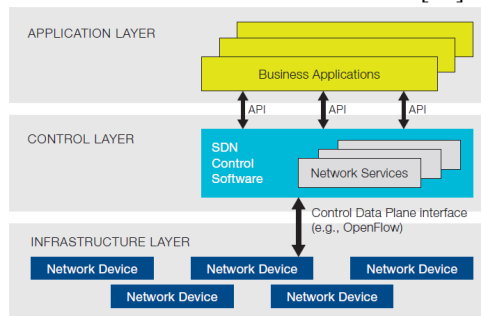


Fig. 1. The logical view of a SDN architecture

This novel networking paradigm offers many benefits compared to the traditional distributed approaches. Firstly, it simplifies networking in both development and deployment of new protocols and applications. With software-based controller, network operators are much easier to program, modify, manipulate and configure of a protocol in a centralized way without independently accessing and configuring individual network hardware devices scattering across the whole network. Secondly, SDN-based architectures provide centralized controller to the network with global knowledge of the network state which are capable of controlling network infrastructure in a vendor-independent manner. These network devices just simply accept policies from the controller without understanding and implementing various network protocols standards, resulting in directly control, program, orchestrate, and manage network resources at the SDN controller. This feature, thus, saves a lot of workforce and resources.

To deploy SDN architecture, the key requirement is a protocol for communicating between Data Plane and Control Plane (South-bound Interface SBI). This protocol should be standardized and vendor-agnostic so as networking companies should follow to develop their products. The SBI protocol facilitates heterogeneous networking switches and routes in the same way, thus, simplifies the operations of the network system. OpenFlow is the typical example of such kind of protocol [13, 14]. OpenFlow-enabled networking switches and routers maintain a forwarding flow-table containing three entries called rules, actions associated with each rule defining how packets should be processed, and statistic counting the number of packets and bytes for the flow. An OpenFlow-

enabled switch also operates a Secure Channel for the communications to SDN Controller, allowing data packets and flow-rule packet to be remotely sent from the controller using OpenFlow protocol [15]. The SDN Controller is in charge of making decision on adding, deleting and modifying the flow rules and actions on the forwarding flow-table via OpenFlow, thus, enabling the programmability to automatically configure the network. SDN applications and services directly communicate their requirements and network actions via the North-bound Interface (NBI) APIs to the SDN Controller.

B. Fog Computing

Fog Computing, embraced by Cisco in 2012, recently becomes an active research cloud-related area. It is the paradigm extending Cloud computing in which data, processing and services are concentrated at the edge of the network instead of entirely in the cloud. Fog Computing places the traditional cloud services closer to the end-users offering various benefits such as low latency, location awareness and mobility support. It integrates Core Cloud services, turns conventional data center into distributed, heterogeneous platforms. Fog Computing, thus, supports Internet of Thing (IoT) applications in vehicles networks, sensor/actors networks and industrial automation which require sensitive-delay and context-aware processing [10, 11]. To deploy Fog Computing, an orchestration is required to manage resources and services across edge devices. Fog services are provisioned, managed and tracked among such devices.

C. Vehicular Adhoc Networks

Vehicles in novel VANET systems such as Intelligent Transportation System (ITS) are equipped with On-Board Units (OBUs) with capability of processing units, sensors, localization system like Global Positioning System (GPS), and radio transceiver for Wireless Access in Vehicular Environment (WAVE) [16]. OBUs are also equipped with wide-range radio transceiver such as WiMax/3G/4G LTE for communication with cellular base station. RSUs are installed along road systems. They are stationary infrastructure such as CCTVs, Network Cameras, Lane-Checking Camera, Traffic Light, Sensors, and Access Points. Each cluster of RSUs is connected to a RSU control station (RSUC) for data storage and processing before sending to the Internet for services. V2V and V2I are standardized by IEEE WAVE. In intelligent VANET systems like ITS, vehicles not only play as the role of sender, receiver but also the router to transmit information to other vehicles or road infrastructure for traffic services.

The advances in OBUs and communications integrated in vehicles, RSUs and RSUCs change the traditional VANETs into Intelligent Vehicle Grid. Moreover, the rapid development of cloud computing soon paves the way to a further step of the Vehicle Cloud in which provides all kinds of cloud service to the transportation [31].

D. Related Work

There are some researches on applying SDN into wireless environment are proposed. M. Mendonca *et al.* utilized SDN into heterogeneous network environment by proposing a simple

topology in which Alice connects to the Internet through Bob's connection. Leveraging SDN results in capability of automatically reconfigure for better Bob's communication [7]. I. Ku, Y. Lu *et al.* proposed a SDN-based VANET architecture by introducing a SDN controller to the system [8]. They also considered some operational modes and fallback mechanism to adapt SDN to VANET scenarios. They simulated and compared some routing protocols in the proposed architecture with traditional MANET/VANET ones, showing the benefits in using SDN. A further step was made by M.A. Salahuddin *et al.* by applying SDN to RSU Cloud. They introduced a new component to VANET architecture called RSU micro-datacenter which supports virtualization and SDN Controller capability [17]. The traditional RSUs and specialized RSUs are forming RSUs cloud. The cloud controller, SDN controller and resource manager models are also proposed; implemented in one or more RSU micro-datacenter for cloud services. They also considered the reconfiguration overhead and proposed a heuristic algorithm to minimize the cost. Inspiring by their great work, we make use of their cloud resource management model to our architectural design considering Fog Computing.

III. SDN-BASED VANET ARCHITECTURE

This section describes the proposed SDN-based architecture, role and operation for each component. Fog Computing framework, which takes advantage of SDN concept, is also discussed to show the benefits in offering services.

A. FSDN VANET Architecture

In this part, we discuss the proposed SDN-based VANET architecture leveraging Fog Computing called FSDN VANET. The below SDN components are needed to incorporate for deploying the system:

- **SDN Controller:** the global intelligence which controls all the network behaviors of the entire SDN-based VANET system. It also plays as Fog Orchestration and Resource Management for the Fog.
- **SDN Wireless Nodes:** the vehicles act as the end-users as well as forwarding element, equipped with OBU and operating OpenFlow. They are Data Plane elements
- **SDN Road-Side-Unit:** RSU running OpenFlow and controlled by the SDN Controller. It is a Fog device.
- **SDN Road-Side-Unit Controller (RSUC):** A cluster of RSUs are connected to a RSUC through broadband connections before accessing to the SDN Controller. RSUC is OpenFlow-enabled and controlled by SDN Controller. Besides the responsibility of forwarding data, RSUCs also store local road system information and perform emergency services. RSUCs are fog devices, under the orchestration of SDN Controller.
- **Cellular Base Station (BS):** In our proposed architecture, BS is not simply carrying voice calls and conveying data, it is more sophisticated. BS is under the control of SDN controller, running OpenFlow, capable of delivering fog services. Similar to RSUC, BS is also local intelligence; a Fog device under the control of

SDN Controller. This architecture provides mobile operators opportunities to offer their clients valuable transportation services, motivates them participate in developing the proposed system.

FSDN VANET architecture is operated in heterogeneous network environment in which forwarding infrastructure use several wireless technologies to communicate. V2V and V2I could be wireless connection or WAVE. Vehicle to BS could be long-range wireless connection such as WiMax or 3G/4G LTE. RSU-RSUC, RSUC-RSUC, RSUC-SDN Controller, and BS to SDN Controller are broadband, high-speed connections [Fig.2]. The assumption is that each SDN wireless node is equipped with WiMax/3G/4G LTE interface mostly for Control Channel and WiFi/WAVE interface for Data Channel. The SDN wireless nodes are supposed to support fallback mechanism to turn back to conventional operations (i.e., AODV, DSDV and OLSR routing protocols) once SDN Controller connection is lost [8].

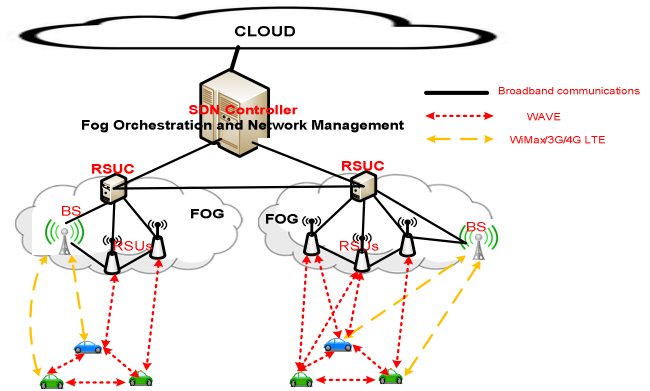


Fig. 2. SDN-based VANET architecture leveraging Fog Computing

B. Basic Operations

The radio communications of the proposed system are divided into two types: control plane communication and data plane communication. Control plane communication is for flow rules or policy rules while data plane communication is for data transfer. I. Ku *et al.* based on the control level of SDN Controller classified into three operation modes namely Central Control Mode, Distributed Control Mode and Hybrid Control Mode [7]. Because of enabling the Fog, our architecture should be operated in the Hybrid Control Mode in which SDN Controller does not take full control of the system, instead, shares the work with BSs and RSUCs [Fig. 3]. For instance, instead of sending specific flow rules, SDN controller will send an abstract policy called policy rules in which specific behavior will be decided by RSUCs or BSs depending on their own local knowledge. Data at RSUCs and BSs is then sent to data center through SDN Controller for global, long-term purposes [10].

To maintain the up-to-date network topology of the vehicles, a link layer mechanism in each vehicle could be used to periodically broadcast beacon messages for learning neighbor's information. This information is sent along with traffic data of the vehicle such as route map, position, speed, and sensor data to RSU or BS whenever possible.

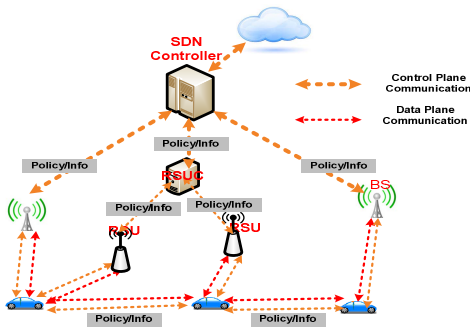


Fig. 3. Hybrid Control Mode in FSDN VANET

The SDN Controller not only gets vehicle information from BSs but also receives transportation data from RSUs. Therefore, it can build a global connectivity graph of the SDN wireless nodes and other necessary knowledge for various services. RSUC and BSs also receive information of vehicles and/or traffic data from a cluster of RSUs, thus, are able to process some surveillance services without entire knowledge as in SDN Controller. RSUC and BSs offer local, distributed intelligence with low latency and location awareness characteristics. These RSUCs and BSs are co-ordinated with each other by a Fog orchestration layer [10, 11] located at SDN Controller.

In Fog Computing concept, both central data centers and pervasive edge devices share their heterogeneous resources and support services. In our case RSUCs and BS, as the Fog devices, not only incorporate with the Cloud through SDN Controller but also share their resources for controlling vehicles. To enable Fog computing framework for the SDN-based VANET system, SDN Controller, RSUCs and BSs are not only equipped with SDN-capability but also offer virtualization for enabling Cloud (Fog) services. Therefore, a hypervisor, which is a low-level middleware, should be implemented at these physical devices for supporting abstraction of Virtual Machines (VMs) on them. Services are hosted at these VMs allowing service migration and replication. This technique is widely used in data centers for improving portability, resource utilization and fault tolerance [18]. The SDN Controller, as mentioned above, also plays as a Fog orchestration and a resource manager. Several software components are required to be implemented at SDN Controller as illustrated in Fig. 4.

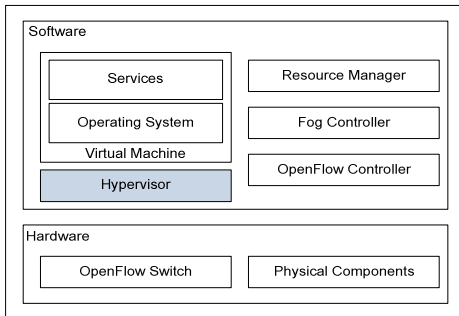


Fig. 4. SDN Controller hardware and software components

C. Resource Management

A Service-Oriented Resource Sharing architecture and mathematical model for SDN Controller, BSs and RSUCs based on the model proposed in [19] could be used as a prospective solution. Instead of task-oriented approach, the model bases on the key idea of service-oriented utility functions which efficiently optimizes utilities of the SDN Controller, BSs and RSUCs resources for services.

Suppose that SDN Controller, BSs and RSUCs host services through application installed in them. They request resources service by service. A service is composed of multiple tasks in which some tasks are processed using its resources; other tasks need to use resources from other nodes called outsourced tasks. RSUCs and BSs want to complete the tasks included in the service as soon as possible. To process tasks, a RSUC needs to use itself resources as well as other RSUCs resources [Fig. 5].

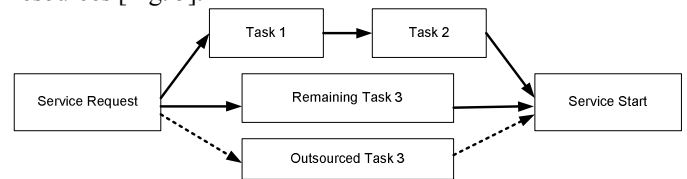


Fig. 5. A flow task of a service containing outsourced task

As a resource coordinator, SDN Controller orchestrates tasks and resources among BSs and RSUCs service-by-service to maximize the utilities of them for services. The benefit in applying SDN to the Fog is that SDN Controller has all knowledge of the BSs and RSUCs such as resources R and tasks T, thus, it can instructs to allocate their tasks to others for resource sharing optimization such as latency, performance and energy consumption by using convex optimization approaches.

D. Fog Controller

Services implemented at VMs in SDN Controller, BSs and RSUCs need an orchestration mechanism to disseminate information of data forwarding rules changes and service hosting. The orchestration is also in charge of service instantiation, replication and migration. A service operating along road system may require different number of BSs or RSUCs hosting the service, resulting in the need for physically migrating VMs in these RSUCs or BSs. This can be done by a Fog Controller incorporating with SDN Controller for automatically updating service hosting and data forwarding rules. That is the benefit of integrating SDN in the Fog architecture. However, the reconfiguration to service hosting, instantiation, migration, replication and data flow rules is costly, resulting in increasing latency and deteriorating Quality of Experience (QoE) [18, 20]. A good solution was proposed by Mohammad A. Salahuddin *et al.* in [17] considering optimizing the cost of reconfiguration in the RSUs cloud. They modeled the reconfiguration overhead in the RSUs cloud, proposed a delay model and a multi-object linear programming formulation and solve by using a heuristic algorithm.

IV. USE CASES

A. Data Streaming

VANETs applications are not only focused on developing safety services but also non-safety services such as online gaming, multimedia applications and video conferencing; and data streaming is the most important factor of the later. Streaming applications have been dramatically increased and contribute a significant amount of traffic in network. The key point is how to distribute data streaming traffic from data streaming server to some nodes in a vehicular network which is high mobility and topology changes; and low delay requirements. The basic scenario is demonstrated in Fig. 6.

There are two advantages that traditional VANET system cannot achieve compared to FSDN VANET. The first is SDN Controller has all knowledge of vehicle routes, RSUCs, and RSUs with resource manager and Fog orchestration, thus, is able to choose the optimal configuration for the service deployment. The second is the service provider makes aware of the red car, suppose that the red car is out of reach of RSUs. Therefore, the SDN controller can automatically offer service to the red car via the blue car and provision blue car's connection accordingly such as increasing the bandwidth for the connection, giving incentives for the blue cars and so on. FSDN VANET can dynamically adapt to topology changes and reconfigures accordingly to guarantee the QoE while be able to find the optimal reconfiguration.

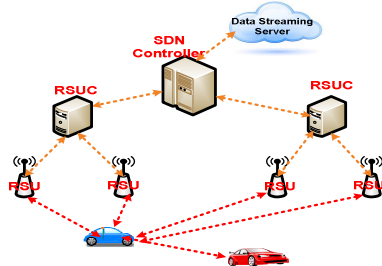


Fig. 6. Data Streaming reference scenario

B. Lane-Change Service

The proposed architecture is applied into another use-case called Lane Change Assistance [3, 21]. Conventional approaches for Lane-Change service face many challenges since it requires planning at microscopic traffic information such as road conditions and traffic flow variables. Current lane-change support applications such as Hella Warning System [22] and BlindSpot Detection System by MobiEye [23] just serve one individual vehicle. These systems just simply warn the driver of blind-spot and do not take other traffic variables into account. This is because an end-user vehicle cannot collect enough necessary and global, reliable knowledge to produce a good decision. Our hypothesis is that Lane-change decision should be coordinated and optimized in higher control level instead of individual car. Because SDN Controller has global, various kinds of information on the road system, it seems be more efficient to build an autonomous lane-changing system compared to conventional approach. In our proposed architecture, BSs, RSUs and RSUCs are Fog devices offering

some emergency services which are originally located in the application plane like Lane-Change Assistance.

To enable the service, a Lane-Change decision making system is needed to implement at both application plane and edge devices. Various lane-change models and safety criterion could be used for the system such as MOBIL [25, 26], IDM-LC [27], MANFIS [27] or several models in [24]. The models require trajectory data consisting of observations of vehicles at discrete points in time, provides useful information about some model parameters. The trajectory data are temporarily stored at RSUCs and BSs; and at data center for long-term uses. Additional explanatory variables required by the models, such as relative speeds, time and space headways and traffic map can also be obtained from data at RSUCs, BSs or data center.

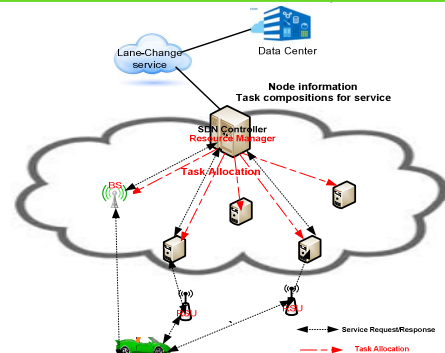


Fig. 7. Lane-change service in FSDN

When a vehicle is needed to change the lane (in either mandatory (MLC) or discretionary (DLC) mode [24]), a control plane request message is sent to an in-charge BS or RSU if possible. SDN Controller depending on the particular situation of the vehicle fully or partly incorporates with BSs and RSUCs service hosting to make decision. Note that because of using SDN architecture, the centralized controller grasp of global network knowledge, so it can quickly point out which BSs or RSUCs should incorporate in the tasks for the decision making process. The policy and actions (Policy/Info) are then sent to the participated vehicles through a BS (WiMax/3G/4G LTE) and in-charge RSUs (WAVE) [Fig. 7].

V. CONCLUSION AND FUTURE WORK

SDN has been the prospective solution to facilitate efficient management and deployment of network services. In this paper we have proposed a novel SDN-based architecture supporting Fog Computing for both safety and non-safety services. We have pointed out all the components and their roles participated in the system and showed how they can serve services. Resource management and Fog coordination models are also considered to build the Fog framework for the SDN-based VANET. Benefits of using our proposed architecture have also been illustrated by two use-cases called Data streaming and Lane-Change assistance services.

There are many directions needed to investigate to support the proposed architecture. The first could be a more appropriate resource management and Fog orchestration models that take SDN characteristics into account to support Fog computing. This model should consider particular properties of BS and

RSUC as well as cooperate with SDN Controller for network knowledge to optimize the resource utilization and service hosting, migration and replication. The second direction is designing protocols at SDN controller for optimizing data forwarding rules modification, improving the load-balancing technique and reducing services latency by utilizing the available resources in BSs, RSUCs, RSUs, and OBU integrated in vehicles. The third could be fallback/backup mechanisms in case of unreliable connections or failure between vehicles to the SDN controller to maintain services. In this case, the isolated nodes should be guaranteed to run some utmost important services. Last but not least, it is valuable to simulate the architecture, models, protocols and services using simulator such as ns-2, ns-3, Mininet, and SUMO; or conduct in a real test-bed. A real test-bed for SDN-based systems could be built using any kind of switch boards (i.e., net6501 board) installed openvswitch (OpenFlow-enabled) [29] as forwarding devices and several PCs installed Open SDN Controller (i.e., Floodlight) as the SDN Controller [30].

REFERENCES

- [1] A. Nandan, S. Das, G. Pau, and M. Gerla, "Cooperative downloading in vehicular ad hoc wireless networks," *IEEE Conf. WONS*, Switzerland, Jan. 2005, pp. 32–41.
- [2] O. Riva, T. Nadeem, C. Borcea, and L. Iftode, "Context-aware migratory services in ad hoc networks" *IEEE Trans. Mobile Comput.*, vol. 6, no. 12, pp. 1313–1328, Dec., 2007
- [3] G. Karagiannis, O. Altintas et al., "Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions", *IEEE Communications Surveys & Tutorials*, Vol.13, Nov., 2011.
- [4] S. Jain et al., "B4: experience with a globally deployed software defined WAN", *ACM SIGCOMM 2013*, New York, USA, 2013, pp. 3–14.
- [5] H. Kim, N. Feamster, "Improving Network Management with Software Defined Networking", *IEEE Communications Magazine*, Feb., 2013.
- [6] C.J Bernardos, A. de la Oliva, P. Serrano et al., "An architecture for software defined wireless networking", *IEEE Wireless Communications Magazine*, Jun., 2014.
- [7] M. Mendonca, K. Obraczka, T. Turletti, "The Case for Software-Defined Networking in Heterogeneous Networked Environments", *ACM conference on CoNEXT student workshop*, 2012.
- [8] I. Ku, Y. Lu, E. Cerqueira, R. Gomes, M. Gerla, "Towards Software-Defined VANET: Architectures and Services", *Mediterranean Ad Hoc Networking Workshop*, 2014.
- [9] F. Bonomi, "The smart and Connected Vehicle and the Internet of Things", *Invited Talk, Workshop on Synchronization in Telecommunication Systems*, 2013.
- [10] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, "Fog computing and its role in the internet of things", *MCC workshop on Mobile cloud computing*, Aug., 2012.
- [11] I. Stojmenovic, S. Wen, "The Fog Computing Paradigm: Scenarios and Security Issues", *Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2014.
- [12] Open Network Foundation (ONF), "Software-Defined Networking: The New Norm for Networks", 2012.
- [13] Open Networking Foundation (ONF), "Software-Defined Networking: The New Norm for Networks", White Paper, Apr., 2013.
- [14] I. Pepelnjak, "OpenFlow and SDN: hype, useful tools or panacea?", NIL Data Communications, [Online] http://tv.nil.si/openflow_and_sdn_hype_useful_tools_or_panacea/
- [15] N. McKeown, T. Anderson, et al., "Openflow: enabling innovation in campus networks," *SIGCOMM Computing Communication, Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008. [Online] <http://doi.acm.org/10.1145/1355734.1355746>
- [16] Intelligent Transportation System (ITS) Statewide Plan, Final Report, Sep., 2005, [Online] www.atacenter.org/programs/its/pdfs/StateITSPlan.pdf
- [17] M.A. Sulahuddin, A. Al-Fuqaha, M. Guizani, "Software Defined Networking for RSU Clouds in support of The Internet of Vehicle", *IEEE Internet of Things journal*, Nov., 2014.
- [18] S. Anja, W. Dargie, "Does Live Migration of Virtual Machines Cost Energy?," *IEEE 27th International Conference on Advanced Information Networking and Applications (AINA)*, 2013.
- [19] T. Nishio, R. Shinkuma, T. Takahashi, N. B. Mandayam, "Service oriented heterogeneous resource sharing for optimizing service latency in mobile cloud," *International Workshop on Mobile Cloud Computing and Networking, MobileCloud'13*, 2013.
- [20] M. Bari, A. Roy, S. Chowdhury, Q. Zhang, M. Zhani, R. Ahmed and R. Boutaba, "Dynamic Controller Provisioning in Software Defined Networks", *9th International Conference on Network and Service Management (CNSM)*, 2013.
- [21] Vehicle Safety Communications –Applications (VSC-A), Final Report, Sep., 2011.
- [22] "Hella provides lane-change warning for BMW", [Online] http://www.eetimes.com/document.asp?doc_id=1255594
- [23] Gat, Itay, Meny Benady, Amnon Shashua. "A monocular vision advance warning system for the automotive aftermarket." *SAE World Congress and Exhibition*, 2005.
- [24] T. Toledo, "Integrated Driving Behavior Modeling", PhD Dissertation, *Massachusetts Institute of Technology (MIT)*, Feb. 2003.
- [25] M. Treiber, A. Kesting, "Modeling Lane-Changing Decisions with MOBIL", *Traffic and Granular Flow '09*, pp 211-221, 2009.
- [26] K. Arne, T. Martin, H. Dirk, "General Lane-Changing Model MOBIL for Car-Following Models", *Traffic Flow Theory '07*, pp 86-94, 2007.
- [27] M. Chamieh, R. El-Kouatly, O. Abu-Amsha, "Impact of IDM Lane-Changing on the Performance of AODV on VANETs", *International Journal of Simulation Systems, Science & Technology (IJSSST)*, Vol. 13, No.6, Dec., 2012.
- [28] Ghaffari, Khodayari, Arvin, Alimardani, "Lane Change Trajectory Model Considering the Driver Effects Based on MANFIS", *International Journal of Automotive Engineering*, 2012.
- [29] Openvswitch, <http://openvswitch.org/>
- [30] Floodlight, <http://www.projectfloodlight.org/floodlight/>
- [31] M. Gerla, E.K. Lee, G. Pau and U. Lee, "Internet of Vehicles: From Intelligent Grid to Autonomous Cars and Vehicular Clouds", *IEEE World Forum on Internet of Things (WF-IoT)*, 2014.