Is SDN the Solution for NDN-VANETs?

Eirini Kalogeiton, Zhongliang Zhao and Torsten Braun
Institute of Computer Science, University of Bern, Bern, Switzerland
Email: {kalogeiton, zhao, braun} @inf.unibe.ch

***Abstract*—Named Data Networking (NDN) is a future Internet architecture suitable for Vehicular Ad Hoc Networks (VANETs), since it provides solutions to frequent topology changes in VANETs. In NDN messages are exchanged according to their content and not to the location of the hosts. SDN provides centralized network control. This paper presents use cases for vehicular NDN scenarios, where SDN is a suitable solution to solve arising problems in VANETs. In particular, SDN assists for setting up rules that are used by RSUs to dictate to the vehicles how to perform forwarding in the VANET. Moreover, SDN could optimize the allocation of network resources. Finally, we present a network communication approach between different geographical areas using wired communication between SDN controllers or wireless communication in vehicles that are used as data mules in these areas. The areas are defined based on the communication range of RSUs.**

***Index Terms*—SDN, NDN, VANETs**

I. Introduction

Vehicular Ad Hoc Networks (VANETs) have been regarded as an ideal solution for future intelligent transportation systems (ITS). It is predicted that a quarter billion of vehicles will be connected by 2020 to form new in-vehicle services and automated driving capabilities [1]. Nevertheless, current VANET architectures can not meet many requirements of future ITS applications, such as low delay, high density, etc. To guarantee efficient communication and cooperation services for large-scale VANETs, an intelligent coordination scheme is required.

Software Defined Networking (SDN) has attracted attention from the research community in mobile wireless networks like VANETs. SDN is also considered as one of the promising solutions that can handle the dynamic nature and dense deployment scenarios of future VANET applications [2]. The decoupling of the network forwarding functions (data plane) and the network control (control plane) brings potentials to offer flexibility, programmability, and centralized control knowledge, which facilitates flexible network management and control for large scale VANETs.

Named Data Networking [3] has been proposed to solve issues that exist in today's TCP/IP communication architecture. Today communication problems do not only arise from the current TCP/IP architecture, i.e. establishing reliable stable connections via end to end communication, but also from the number of the interconnected devices that exist. The main issue of TCP/IP over the last years is facing the increasing number and the heterogeneity of interconnected devices. IPv4 is not suitable to deal with this problem. Hence, IPv6 has been proposed but not frequently used yet, since it requires the replacement or configuration of any communication device (routers, switches, etc).

In this paper we present related work in SDN and VANETs as well as NDN and VANETs. In addition, we present use cases that improve the performance of the overall VANET, by using NDN in vehicular nodes and SDN controllers that communicate with Road Side Units (RSUs). In particular, we propose the population of Forward Information Base tables in nodes via an SDN controller, the interface alternation dependent on the required content, and the exchange of messages between different regions that exist in a city area. In the proposed approaches, we utilize an SDN controller that is responsible for making many decisions for the VANET operations.

The rest of this paper is organized as follows: Section II gives an overview of SDN and how it applies to VANETs. Section III presents NDN and how it applies to VANETs. In Section IV we present a few use case scenarios for combining SDN and NDN in VANETs and discuss the advantages for this combination. Finally, we conclude this paper in Section V.

II. Software Defined Networking

Recent mobile cloud applications require guaranteed network performance more than conventional client-server applications. Such mobile cloud applications incur tight and real-time interactions with the cloud and sometimes offload network-intensive workloads to the cloud. The emergence of Software-Defined Networking (SDN) [4], whose key features include separation between the data plane and the control plane as well as logically centralized network control, has potential to completely reshape the field of computer networking. Going beyond conceptual models, the existing deployment of SDN in major data centers has demonstrated the substantial gain and flexibility of SDN. However, existing initiatives of SDN focus mostly on wired networks such as data center, service provider, and enterprise networks. Given the substantial potential of SDN, it is imperative that we develop a better understanding of the potential benefits and issues of applying SDN to wireless networks, including cellular, cyber-physical, Vehicular Ad-Hoc Networks, etc.

Recently, some efforts have been made to bring the concept of SDN into wireless networks. However, applying SDN to the wireless domain is challenging due to the fact that wireless networks feature many characteristics that often do not exist in wired networks. Due to the intrinsic nature of non-stationary wireless channels, wireless communication service quality is subject to wireless transmission power,

978-1-5386-2077-9/17/$31.00 ©2017 IEEE

interferences, etc. To tackle these problems, [5] focused on the problem of enhancing Wi-Fi networks with SDN benefits, and presented *AeroFlux*, a scalable software-defined wireless network that supports large enterprise and carrier-grade Wi-Fi deployments with low-latency programmatic control of fine-grained Wi-Fi-specific transmission settings. *SoftOffload* [6] provides intelligent mobile data offloading by collecting various traffic contexts from both end-users and network operators. It performs mobile offloading to increase user-side throughput and reduce network congestion. The Open Networking Foundation (ONF) [7] has formed a Wireless and Mobile Working Group to explore how to adapt OpenFlow for use in cellular access networks and wireless transport networks for telecommunications providers [8].

Some initial efforts have been made to integrate the SDN concept to VANETs [2], [9]. [10] proposed an SDN-based VANETs architecture to enhance the flexibility and programmability of VANETs and introduced new services and features to today's VANETs. [11] proposed a new VANET architecture called FSDN that combines SDN and Fog Computing as prospective solution. FSDN provides flexibility, scalability, programmability and global knowledge while Fog Computing offers delay-sensitive and location-aware services, which could satisfy the demands of future VANETs scenarios. In VANET communications, the dissemination of a safety message is of critical importance. The US Federal Communications Commission has allocated DSRC (dedicated short range communication), which is a 75 MHz licensed spectrum at a 5.9 GHz band solely for vehicle-to-vehicle and vehicle-to-infrastructure communication. The 75 MHz spectrum is divided into seven channels. One of them is the control channel (CCH), which is used exclusively for safety communications. [12] proposed to use an SDN controller, hosted at an RSU, to help improving the dissemination of DSRC short messages. However, [13] has proven that the performance of using an SDN controller at RSUs became limited when the number of vehicles connected with RSUs increases.

III. Named Data Networking

NDN [3] is an Internet architecture that does not rely on end-to-end communication to establish a connection between two hosts. Messages are now exchanged based on their content, and not according to the location of hosts. Unlike today's TCP/IP, where a user connects to an endpoint (server) that is located in a fixed location to send and receive messages, NDN users send a request with a unique name and receive the requested message as an answer to this. The advantages by this method are various: A user can retrieve information via a neighbor instead of requesting information to a server that is located further away to save network resources. In addition, users retrieve messages from many different sources, thus congestion and collisions are reduced.

Vehicles can carry, store, and forward messages [14]. Therefore, vehicles can also carry, store, and forward NDN messages. Every user is equipped with a storage unit that stores NDN messages. In addition, the user can forward these messages and also carry them to another location, if the user is mobile. NDN uses two different types of messages for information retrieval: Interests and Data messages. When a user requests information, it sends an Interest message with a unique name. That name is an identifier for the information requested. When a user responds with the requested information, it sends a Data message. NDN has its own data structure that consists of three tables: Pending Interest Table (PIT), Forward Information Base (FIB) and Content Store (CS). When a node receives an Interest message, it checks its PIT to identify if the node has forwarded the same Interest before and did not receive an answer yet. In this case, the node stores information of the incoming Interest to the PIT. When the Data message comes back, the node will check the PIT to find all nodes that requested this Data message and send the latter to the nodes. When a node receives a Data message, the node stores this Data message into the CS, such that this message becomes available for other nodes' requests. If a node does not identify a PIT entry or there is no matching entry in its CS, the node checks its FIB table. The FIB shows how the forwarding of an Interest message should be done, in order for this Interest to arrive at a node that holds the corresponding Data message.

In VANETs, NDN seems to be a suitable approach, since the network topology changes continuously. In VANETs an end to end communication approach is not optimal, not only because of the change of the topology, but also due to the message characteristics, i.e. lifetime of a message and the geographical area that the message may concern. For instance, in case of traffic in a small local road, this related congestion concerns only vehicles that will go through this road. With a TCP/IP approach, this information would be stored at a server that might be located further away from the area. This information will be deleted according to the server strategy. NDN allows the local dissemination of a message in a specific area, since vehicles are transmitting the messages. With an appropriate strategy, vehicles could stop the transmission of a message when they travel outside of a marked area. NDN in VANETs has been studied over the last years. Authors in [15] propose an opportunistic geo-inspired content based routing method to reduce the transmissions of Interests when nodes request content in VANETs. Authors in [16] present a Mobile Cloud architecture in Mobile Ad Hoc Networks based on SDN. They extend the OpenFlow protocol by adding a local SDN controller to each node that acts as a back-up mechanism in case the global SDN controller communication becomes unavailable, and use different wireless interfaces. Authors in [17] describe a data propagation mechanism in NDN-VANETs. They propose to use hop counts to reduce the number of Data packets that exist in the network. Authors in [18], [19] describe possible SDN architectures in V-NDNs, such as caching and forwarding techniques. They also propose the use of SDN to identify popular content and cache it to the most influential nodes in the VANET.

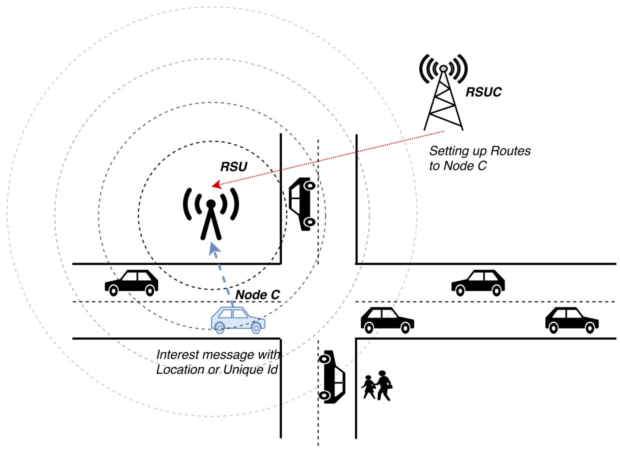

Fig. 1: Discovery of Content sources

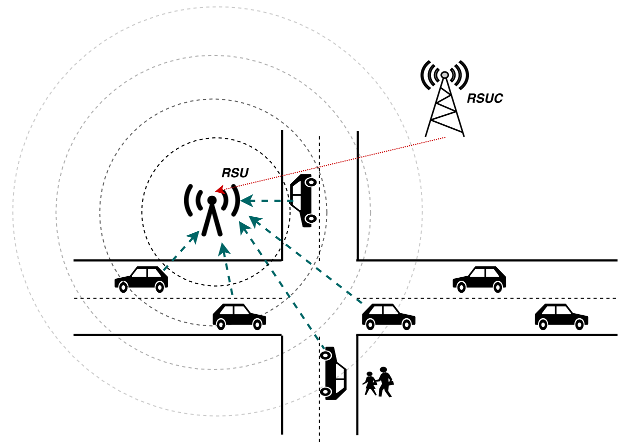

Fig. 2: Centralized SDN approach in a V-NDN

IV. S-VNDN, Case Scenarios

In this section we highlight the advantages that SDN could have in an NDN-VANET. Using SDN makes the network more efficient and optimized in terms of saving network resources (bandwidth) and computation power. An SDN controller could perform all the necessary calculations and install the resulting rules from these calculations to different network components. We propose use case scenarios, where SDN could be utilized in VANETs. We identify problems that exist in these cases and give the appropriate SDN solutions.

A. Communication Techniques

In order to exploit the capabilities of SDN in VANETs, we choose to include an SDN controller (we will refer to it also as controller) in the VANET topology, which communicates and sets up rules at the RSUs. In order to set up forwarding rules, an RSU, and consequently a controller, should know about the potential content sources that exist in the VANET. To discover content sources we propose that every content source (e.g. a vehicle) in the VANET sends an Interest message to the controller, when a network connection exists. This Interest message contains characteristics in order for the controller to set up routes between controller and content source. The characteristics that we propose are either the current location of the content source or a unique id that the content source has. In addition, in highways and in cases when vehicles travel with high speed, a content source could send its predicted location or its end point (final destination of a vehicle).

By combining the location and network traffic knowledge and by using mobility prediction techniques, an SDN controller could set up paths to a predicted future location of the content source. This idea is illustrated in Fig. 1. Node C, which holds a content object, sends an Interest message to the RSU, and the RSU communicates with the SDN controller. This Interest message contains node's C unique id, or its current location, or its final destination. Upon the reception of this Interest message the controller sets up routes at node C by populating the FIBs (as described in Section IV-B). These routes consist either of the unique id of node C or the current location of node C or the predicted location of node C.

As described in [10], there are three ways that an SDN controller could be utilized in a VANET. We focus our interest mainly on two and enhance them by using NDN. The first approach is a centralized approach, where all vehicles redirect their messages to an RSU. The second approach is a semi-centralized approach, where vehicles choose to forward a message to an RSU or to another vehicle. This choice is made according to the rules (forwarding strategy) that an SDN controller has set up on the RSUs, and consequently the vehicles.

1) Centralized: According to [10], every node in the network could direct its traffic through the controller. In this case the SDN controller is the network element that performs all decisions about the network. An NDN node will send every Interest message to the controller. The latter is responsible for identifying a suitable content source (as described in Section IV-A) and forwarding the Interest message to this source. In addition, the same method will be performed when a node sends a Data message. The Data message will be forwarded to the controller, which is responsible to forward it to the requester.

Although this approach is centralized and could achieve acceptable results on wired networks, it is not appropriate for VANETs to forward all traffic to a controller. In VANETs the expected data traffic can not be accurately predicted, since vehicles travel with high speeds. Even if data traffic can be estimated, the vehicular traffic could differ depending on unpredictable conditions. This issue leads to a possible bottleneck around the RSUs. Even when an RSU is equipped with many interfaces, redirecting all the data traffic to it may cause collisions and congestion. This leads to unsuccessful message delivery. An example of the above scenario is illustrated in Fig. 2. Every node in the VANET will send the Interest message to the RSU. The RSU will then forward the message to the content source according to the rules that the controller has set.

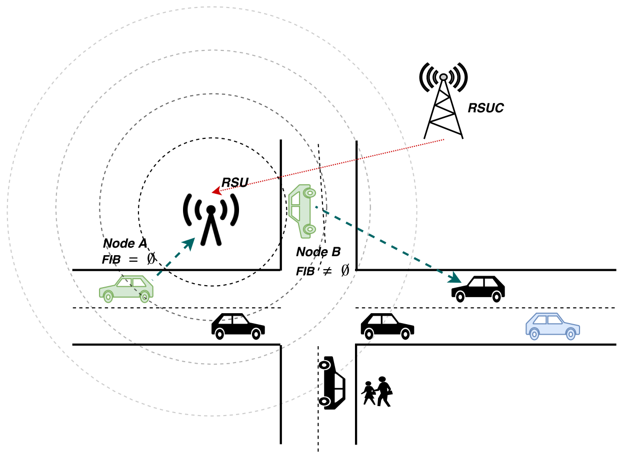

Fig. 3: Semi-Centralized SDN approach in a V-NDN

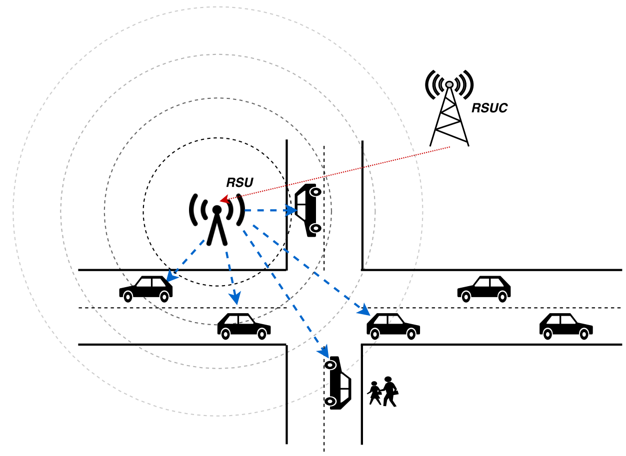

Fig. 4: FIB population

2) Semi-Centralized: The second approach is based on [10], and is a semi-centralized approach. Specifically, we propose that all NDN nodes in the network send a message to the RSU only, if the nodes do not have a FIB entry to satisfy their request. When a node sends an Interest message, the node checks its FIB entry to identify routes to potential content sources. If no FIB entry exists, the NDN node sends the Interest to the RSU, such that the RSU identifies and forwards the message. We propose to use the same techniques for identifying possible content sources as described in Section IV-A. An example is shown in Fig. 3. Node A and node B each send an Interest message. In order to send this message, they both check their FIB tables to identify potential routes to the content source. Node B's FIB is not empty, so node B sends the message to the route that exists in the FIB. On the other hand, node A's FIB is empty, so node A sends an Interest message to the RSU, such that the RSU performs the forwarding according to the rules that the controller has installed at the RSU.

B. FIB Population

Another advantage that an SDN controller provides in an NDN-VANET is the population of the FIB tables in the NDN nodes. Specifically, if the controller has knowledge of the network, the controller could populate the FIB tables in every node. To populate the FIBs, the controller should know the appropriate routes to the content sources. For identification of these routes, we propose the same techniques as described in Section IV-A. As shown in Fig. 4, the RSU sends information about the routes to the VANET nodes. Nodes will then select routes according to their information from their populated FIBs.

C. Adaptive Forwarding and Interface Allocation Mechanisms

In VANETs, IEEE 802.11p has been proposed as the preferred communication protocol, in order to reduce data packet collisions and at the same time to increase the distance range, since the transmission distance range is bigger than traditional Wi-Fi (up to 1km). In VANETs different applications have different requirements. For instance, a safety application requires minimum latency, but an infotainment application (e.g. a video) could be more delay tolerant. For these reasons, we propose to equip the RSUs and the NDN nodes with several different interfaces. Specifically, an SDN controller sets up rules to forward messages differently, according to the application. In particular, a rule could include broadcasting a safety message that affects the whole VANET. Moreover, it is likely that when a message is urgent and affects a large area, it needs to be forwarded to the whole area, with minimum delay and with the smallest number of retransmissions to avoid collisions and broadcast storms. Equipping the VANET nodes with multiple interfaces leads to sending and receiving messages simultaneously. In addition, we propose to use multiple communication technologies, such as LTE, together with Wi-Fi. In this case, an SDN controller could set up rules to send messages through different interfaces (LTE for safety messages) in order to guarantee fast data delivery and reduce collisions.

As illustrated in Fig. 5, let us assume that nodes A and C run two different applications. We also assume that all VANET nodes are equipped with multiple interfaces, Wi-Fi and LTE. We propose to use Wi-Fi for local and ad-hoc dissemination of messages between vehicles and between vehicles and RSUs and LTE for traffic offloading and sending important information to larger areas. Node A is running a video application, requesting video segments, by sending Interest messages to the RSU. The RSU will then forward the Interests to potential content sources via Wi-Fi, according to the rules that the SDN has installed in the the FIB. At the same time, node C runs a safety application and sends via Wi-Fi the Interest message to the RSU. The SDN controller has installed the necessary rules to the RSU, to transmit every safety message via LTE to all nodes in the VANET. Thus, the RSU sends the Interest from node C to all nodes via LTE.

D. Communication between SDN Areas

An RSU can only cover a limited geographical area. In a city there exist multiple RSUs, as shown in Fig. 6. An

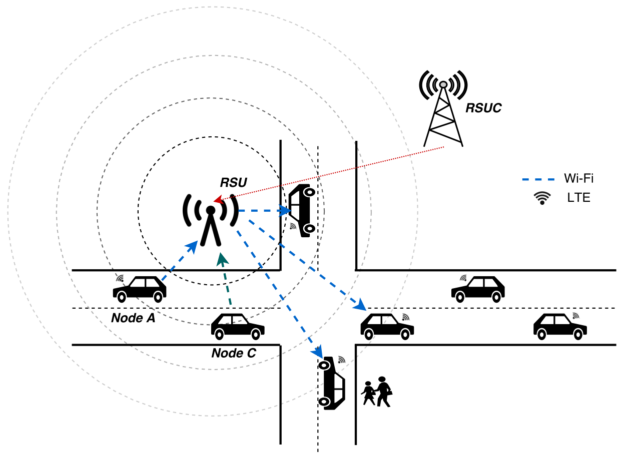

Fig. 5: Different forwarding decisions, according to application

SDN controller connects with them. Another SDN controller is responsible for setting up rules for the other RSUs that are located in different areas. Keeping that in mind, in a city there will be multiple areas, each of them will be covered by one controller. We propose to enable the communication between these areas via wired communication between the controllers. When an emergency occurs, or when information needs to be transfered between areas of a city, a wired communication presents stability and security. The wired communication techniques between SDN controllers are beyond the scope of this paper.

On the other hand, it is possible that some information is not urgent, or that wired communication will not be available (e.g. when defects occur and maintenance should be performed or in case of disasters). Hence, the exchange of messages via wired communication between the defined SDN areas is not possible. In this case, we can set up rules to connect these areas. We propose to use the store-carry-and-forward mechanism of vehicles [14], that travel from one area to another, and use the vehicles as data mules to transfer messages between areas.

For the above cases the controller plays a crucial role in the network. The rules that the controller installs at RSUs differ dependent on scenarios and applications. We propose to use a controller to either explicitly deploy content objects at certain vehicles or to configure parameters for retrieving content objects (e.g. lifetime of a content object). The main parameter of the content deployment is the time that a vehicle caches the message. We propose to configure the caching time parameter as follows: First, we consider the vehicular traffic jam that will exist between the two areas, among which the vehicle will travel. When a road is congested, a vehicle should store a message a longer time, since the travel for data muling takes longer. This is due to the fact that the vehicle will travel slower in a congested road than in a free road. In addition, we consider the distance between the source area and the destination area. A controller should set longer caching timers, as the distance between the areas is growing. Finally, the speed of the vehicle is also considered. The faster a car is traveling, the smaller the caching timer should be.

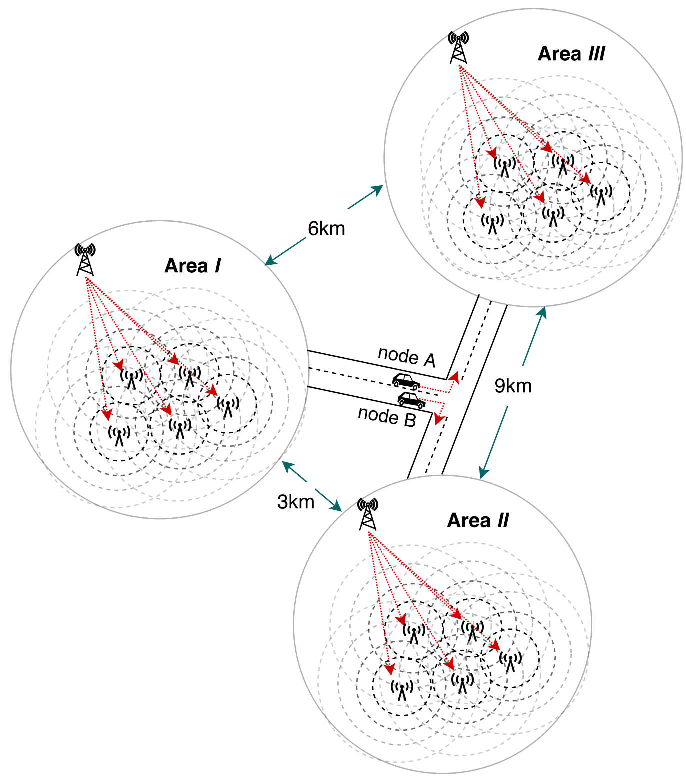

Fig. 6: Communication between different areas that are defined by RSUs range

Let us consider Fig. 6 where the network is divided into three clusters. A vehicle (node A) carrying a message is leaving area *I* and is heading towards area *III*. Node A carries a safety message, because there was an accident on a road of area *I*. According to Section IV-C, in safety applications the SDN controller sets up rules, e.g. the broadcast of a safety message. Consequently, node A will broadcast this safety message. This message should be forwarded to all vehicles traveling to area *I*. The distance between area *I* (node A's source point) and area *III* (node A's destination point) is 6 km (Fig. 6). The caching time that the SDN controller (located in area *I*) has set at RSUs and subsequently at nodes that leave area *I* is t_0. Node B also leaves at the same time from area *I* with destination area *II*. Node A will broadcast the safety message, thus node B will also cache the same safety message for t_0. The distance between area *I* and area *II* is 3km, as seen in Fig. 6. After time t_0, we assume that node B has already entered area *II*, still carrying the safety message (since the message has not expired yet). Consequently, node B will enter area *II* with a message that does not affect area *II* (since this safety message is about area *I*). Moreover, this safety message occupies space on its Content Store. Thus, node B could discard useful information, e.g. information for area *II*, if node B's buffer is full of messages from other areas.

The vehicular traffic between areas and the speed of vehicles are also variables that have to be considered when developing caching rules. For the above reasons, we propose to use the controller in each area to adapt the lifetime of messages that are stored in the cache of each vehicle. To conclude, the SDN controller should configure the messages' lifetime, based on:

1) Vehicles' destination. A controller should know the destination of a vehicle, in order to forward the appropriate message to it.
2) Distance between the traveling areas of a vehicle.
3) Traffic jam along the traveling path of the vehicle.
4) Speed of the traveling vehicle.

V. CONCLUSIONS

In this work we proposed SDN use cases for Vehicular Ad Hoc Networks that use the NDN architecture. In particular, we identify use cases that present some specific problems and we proposed the use of SDN in order to solve these problems. An SDN controller having global or local knowledge of the network and vehicular traffic information, e.g. density of the VANET, location, and direction of vehicles, number and location of content sources, can be exploited in order to install forwarding rules at the RSUs. The RSUs then can install these forwarding rules at the FIBs of the vehicles. In addition, we propose the use of SDN in order to route messages via different interfaces. A controller is responsible for identifying the appropriate interface to forward a message, based on the type of application that generated the message. For instance, for a safety application, LTE could be utilized, since it provides global coverage in the affected area. Finally, SDN could be responsible for the exchange of messages between different SDN areas, either by wired connections, or by using vehicles as data mules and install caching and forwarding rules to them.

ACKNOWLEDGMENTS

This work was undertaken under the CONTACT project, CORE/SWISS/15/IS/10487418, funded by the National Research Fund Luxembourg (FNR) and the Swiss National Science Foundation (SNSF).

REFERENCES

[1] "Gartner predicts by 2020, a quarter billion connected vehicles," http://www.gartner.com/newsroom/id/2970017, accessed: 2017-05-18.

[2] K. Zheng, L. Hou, H. Meng, Q. Zheng, N. Lu, and L. Lei, "Soft-defined heterogeneous vehicular network: architecture and challenges," *IEEE Network*, vol. 30, no. 4, pp. 72–80, July 2016.

[3] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang *et al.*, "Named data networking," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 66–73, 2014.

[4] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, Jan 2015.

[5] J. Schulz-Zander, N. Sarrar, and S. Schmid, "Towards a scalable and near-sighted control plane architecture for wifi sdns," in *Proceedings of the Third Workshop on Hot Topics in Software Defined Networking*, ser. HotSDN '14. New York, NY, USA: ACM, 2014, pp. 217–218. [Online]. Available: http://doi.acm.org/10.1145/2620728.2620772

[6] A. Y. Ding, Y. Liu, S. Tarkoma, H. Flinck, and J. Crowcroft, "Demo: An open-source software defined platform for collaborative and energy-aware wifi offloading," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '15. New York, NY, USA: ACM, 2015, pp. 182–184. [Online]. Available: http://doi.acm.org/10.1145/2789168.2789174

[7] "The open networking foundation," www.opennetworking.org/index.php, accessed: 2017-05-18.

[8] "Onf wireless mobile working group (wmwg)," https://goo.gl/jFosxc, accessed: 2017-05-18.

[9] C. Jiacheng, Z. Haibo, Z. Ning, Y. Peng, G. Lin, and S. Xuemin (Sherman), "Software defined internet of vehicles: architecture, challenges and solutions," *Journal of Communications and Information Networks*, vol. 1, no. 1, pp. 14–26, 2016. [Online]. Available: http://dx.doi.org/10.1007/BF03391543

[10] I. Ku, Y. Lu, M. Gerla, F. Ongaro, R. L. Gomes, and E. Cerqueira, "Towards software-defined vanet: Architecture and services," in *Ad Hoc Networking Workshop (MED-HOC-NET), 2014 13th Annual Mediterranean*. IEEE, 2014, pp. 103–110.

[11] N. B. Truong, G. M. Lee, and Y. Ghamri-Doudane, "Software defined networking-based vehicular adhoc network with fog computing," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, May 2015, pp. 1202–1207.

[12] A. Di Maio, M. R. Palattella, R. Soua, L. Lamorte, X. Vilajosana, J. Alonso-Zarate, and T. Engel, "Enabling sdn in vanets: What is the impact on security?" *Sensors*, vol. 16, no. 12, 2016.

[13] K. Liu, J. K. Y. Ng, V. C. S. Lee, S. H. Son, and I. Stojmenovic, "Cooperative data scheduling in hybrid vehicular ad hoc networks: Vanet as a software defined network," *IEEE/ACM Transactions on Networking*, vol. 24, no. 3, pp. 1759–1773, June 2016.

[14] Z. D. Chen, H. Kung, and D. Vlah, "Ad hoc relay wireless networks over moving vehicles on highways," in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*. ACM, 2001, pp. 247–250.

[15] Y.-T. Yu, Y. Li, X. Ma, W. Shang, M. Sanadidi, and M. Gerla, "Scalable opportunistic vanet content routing with encounter information," in *Network Protocols (ICNP), 2013 21st IEEE International Conference on*. IEEE, 2013, pp. 1–6.

[16] I. Ku, Y. Lu, and M. Gerla, "Software-defined mobile cloud: Architecture, services and use cases," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2014 International*. IEEE, 2014, pp. 1–6.

[17] S. H. Ahmed, S. H. Bouk, M. A. Yaqub, D. Kim, and M. Gerla, "Conet: Controlled data packets propagation in vehicular named data networks," in *Consumer Communications & Networking Conference (CCNC), 2016 13th IEEE Annual*. IEEE, 2016, pp. 620–625.

[18] R. Soua, E. Kalogeiton, G. Manzo, J. M. Duarte, M. R. Palattella, A. Di Maio, T. Braun, T. Engel, L. A. Villas, and G. A. Rizzo, "Sdn coordination for ccn and fc content dissemination in vanets," in *Ad Hoc Networks*. Springer, 2017, pp. 221–233.

[19] J. M. Duarte, E. Kalogeiton, R. Soua, G. Manzo, M. R. Palattella, A. Di Maio, T. Braun, T. Engel, L. A. Villas, and G. A. Rizzo, "A multi-pronged approach to adaptive and context aware content dissemination in vanets," *Mobile Networks and Applications*, pp. 1–13, 2017.