SOFTWARE DEFINED VEHICULAR NETWORKS: ARCHITECTURES, ALGORITHMS, AND APPLICATIONS

Named Data Networking for Software Defined Vehicular Networks

Syed Hassan Ahmed, Safdar Hussain Bouk, Dongkyun Kim, Danda B. Rawat, and Houbing Song

The authors discuss both SDN and NDN enabled VNs, and present an architecture that combines SDN functionalities within VNs to retrieve the required content using NDN. They also discuss a number of current research challenges and provide a precise roadmap that can be considered to jointly address such challenges by the research community.

ABSTRACT

Named data networking and software defined networking share mutual courage in changing legacy networking architectures. In the case of NDN, IP-based communication has been tackled by naming the data or content itself, while SDN proposes to decouple the control and data planes to make various services manageable without physical interference with switches and routers. Both NDN and SDN also support communication via heterogeneous interfaces and have been recently investigated for vehicular networks. Naïve VNs are based on the IP-based legacy, which is prone to several issues due to the dynamic network topology among other factors. In this article, we first see both SDN and NDN enabled VNs from a bird's eye view, and for the very first time, we present an architecture that combines SDN functionalities within VNs to retrieve the required content using NDN. Moreover, we discuss a number of current research challenges and provide a precise roadmap that can be considered for the research community to jointly address such challenges.

INTRODUCTION

The rapid growth in Internet traffic has triggered a plethora of research and development projects in the wide domain of communications. Today, we prefer to use high bandwidth and expect a great quality of experience (QoE) in the communication technologies ranging from cellular, Wi-Fi, WiMAX, and Bluetooth to the Internet of Things (IoT) [1]. Similarly, the past two decades have brought tremendous advancements in the transportation and automation industries, where the assurance of safety and security have become the baseline of what we are perceiving today; for example, autonomous cars, safety/non-safety information dissemination between vehicles (V2V), infrastructure-based vehicle communications (V2I), and heterogeneous vehicular networks (VNs) [2].

The key applications for VNs include, but are not limited to, traffic conditions, accident warnings, pedestrian collision warning systems, smart parking, auto-braking systems, live video streaming, and live gaming. However, the main technical challenges in VNs are related to the high volatility and dynamicity of vehicles' mobility. Even though the Dedicated Short-Range Communication (DSRC) and Wireless Access Vehicular Environment (WAVE) protocol suites have been playing a sophisticated role in the initial stages of VN implementation, it is hard to ensure low latency, high quality, and secured content or data[1] retrieval in a robust manner. Moreover, the DSRC and WAVE protocols are based on the conventional TCP/IP originally designed for a single conversation between two end-to-end entities widely known as *client* and *host*.

Regardless of the applications' motivation (i.e., safety or non-safety), the main purpose of connecting vehicles is to share the content to fulfill the applications' requirements. However, dynamic mobility makes it difficult to have reliable communication of the content between connected vehicles. The main reason is that the current standards were originally proposed for static and quasi-static environments. Despite the fact that these standards tend to support mobility and fast content delivery in VNs, the applications still require a destination address to deliver the content. Hence, the communication is contingent on the vehicle's identity (IP and/or medium access control, MAC, address). Hence, the path establishment, maintenance, and identity assignment in VNs are challenging and generate much overhead. From a non-safety application's point of view, it requires content irrespective of the identity and location of the actual provider or producer.

Meanwhile, named data networking (NDN) [3] as an extension of content-centric networks (CCNs) [4] has been merged into VNs (VNDN) as a future networking architecture [5]. VNDN basically assigns a *name* to the content rather than the device (i.e., vehicles), and that name is used to retrieve the required content. In VNDN, we consider a simplified pull-based communication, where a content requesting vehicle (the *consumer*) sends an Interest message, and the infrastructure or vehicle with the required content (the *provider*) sends back the Data message. Interest contains the required content name and unique NONCE value to identify the Interest message and avoid its duplicate transmission. On the other hand, the Data message contains the same content name and the embedded security information (e.g., digital signature) within it. Therefore, instead of securing the connection between consumer-provider node pairs, the security is inherently augmented with the Data. Additionally, VNDN supports multiple interfaces for reliable and quick fetching of the required content. Every NDN enabled vehicle maintains the following basic data structures:

---

[1] In the context of this article, the terms *data* and *content* are interchangeable.

Digital Object Identifier: 10.1109/MCOM.2017.1601137

Syed Hassan Ahmed, Safdar Hussain Bouk, and Dongkyun Kim are with Kyungpook National University; Danda B. Rawat is with Howard University; Houbing Song is with Embry-Riddle Aeronautical University.

0163-6804/17/$25.00 © 2017 IEEE

- Content store (CS): This caches data or contents either generated or received by the vehicle.
- Forwarding information base (FIB): It stores the outgoing interface(s) associated with the name prefixes to forward the Interests.
- Pending Interest Table (PIT): This keeps track of the names or name prefixes, NONCEs, and incoming interfaces of the received Interest(s). The entries are kept for a certain period and removed when the Interests are satisfied or their lifetime in the PIT expires.
- NONCE List: It records the NONCEs of all the pending entries of the satisfied Interests from the PIT to prevent an Interest loop. All entries are timestamped and purged after a certain time period.

An Interest is uniquely identified by the NONCE plus content Name. A node receiving an Interest first checks the NONCE list, to check whether the Interest has been recently satisfied or not. If no entry is found in the NONCE list, a record of the received Interest is scanned in the PIT to verify whether the Interest is still pending or not. The entry in the PIT shows that the Interest has already been forwarded. On the contrary, the NONCE and Name are stored in the PIT along with the Interface from where the Interest was received (called *InFace*). The PIT entry is purged once the Interest is satisfied. If a node receives multiple copies of the pending Interest, the InFace(s) and other information are aggregated in the PIT record with the same Name. In a scenario where a node receives a Data message, it first checks the PIT record. Based on the PIT search result, the Data message is either forwarded, if there is an entry in the PIT, or dropped otherwise. The satisfied Interest's record is removed from the PIT, and NONCE(s) information is stored in the NONCE list. An Interest loop occurs when a node receives another copy of the satisfied Interest from the path with large delay and can be avoided by checking the Interest's record in the NONCE list. This operational mechanism of Interest and Data messages is summarized in Fig. 1.

Benefits of applying NDN in VNs are discussed thoroughly in our recent works [6–8]. To be precise, the VNDN separates the functions that assist in locating and supplying the required content from the underlying communication technologies used by the VNs. The preparation nature of the data retrieval process brings in the discussion of software defined networking (SDN) [9]. We have known SDN as an alternative and emerging way to look at the networking architecture [10]. Here it is worth mentioning that any networking system is based on the control plane, the data plane, and the management plane. The control and management plane serve the data plane, which bears the traffic that the network exists to carry. The management plane, which is responsible for administrative traffic, is considered a subset of the control plane. In the conventional networks, all three planes are implemented in the firmware of routers, switches, and related network elements. SDN decouples the data and control planes by removing the control plane from network hardware and implements it in software using OpenDayLight, which enables software-based access and, as a result, makes network administration much more flexible. Moving the control plane to software allows dynamic access and administration. Similarly, the administrator can change any network switch's rules when necessary, including prioritizing, de-prioritizing, or even blocking specific types of packets with a very granular level of control. Explicitly, both VNDN and SDN share the concept of core architectural changes from their own perspectives.

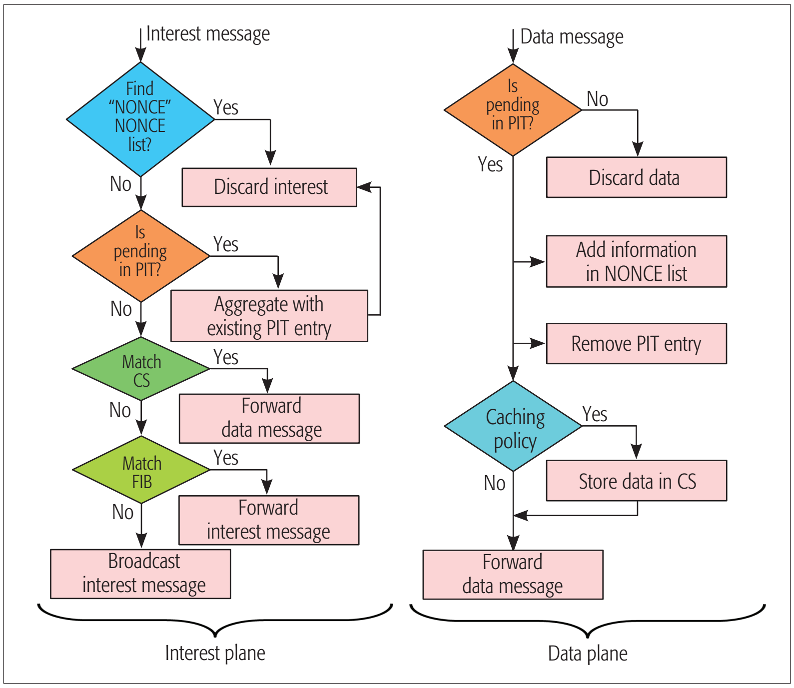

Figure 1. *Interest* and *data* message processing in VNDN.

This article discusses the possible cohesion of VNDN and SDN to support robustness in content retrieval. Although there are recent efforts being carried out for combining the future Internet architectures with SDN features, to the best of our knowledge, we will be leading the research community in bridging both technologies for VNs. In the following section, a basic overview of SDN-based VNs and NDN-enabled VNs is presented. Furthermore, a novel architecture for retrieving the named data in VNs using SDN features, followed by the open research issues, is presented.

BIRD'S EYE VIEW OF SOFTWARE DEFINED-NAMED DATA VEHICULAR NETWORKS

The cultivation of VNs through conventional technologies, such as the IEEE 802.11x family and third/fourth generation (3G/4G) cellular infrastructures, while utilizing the existing switches and routers has led us to several issues and challenges such as frequent disconnection due to fast mobility, short-range communication, and so on. In order to address these issues, several concepts such as intelligent transportation systems (ITS), V2V, and V2I have been proposed [11]. However, the tremendous increase in the number of vehicles and the different requirements of users cannot be addressed through conventional switching and routing techniques. SDN presents intelligent mechanisms and solutions to switch and route the vehicular data by incorporating several parameters including traffic density, multiple

The unification of the interfaces of various technologies can be made possible by decoupling the control and data planes in SDVN. This will allow us to make centralized decisions by designing flow table entries at the data plane side. However, the approach of decoupling planes is not straightforward and can lead to several issues and challenges.

network interface, and so on. In addition to communication in a heterogeneous wireless environment, a unified abstraction model is needed to link the communication of different technologies on a single platform.

SDVN: SDN TRANSITION TO VEHICULAR NETWORKS

The unification of the interfaces of various technologies can be made possible by decoupling the control and data planes in the software defined vehicular network (SDVN) [12]. This will allow us to make centralized decisions by designing flow table entries at the data plane side. However, the approach of decoupling planes is not straightforward and can lead to several issues and challenges.

In a typical SDVN, the communication of the Data messages is logically controlled by the centralized control plane. However, such centralization of the control plane mainly dependeds on the vehicles' trajectory predictions. In a naïve wired SDN, the communication between the data and control planes is carried out by high-speed optical fiber; however, in the case of SDVN, the communication is mainly performed via wireless links. Thus, the cost of communication and the dynamic topology have great impact on SDVN deployment and configuration. Nevertheless, intelligent techniques are needed to tackle the topology variations by using various techniques such as self-configuration and self-healing. Moreover, SDVN is highly dependent on fast and efficient virtual routing and switching of the Data messages; therefore, the modeling of single-hop and multihop communication can be revised and remodeled to meet the requirements of vehicular applications in a scalable manner. Likewise, the introduction of 5G will somehow solve the data forwarding capability by high-speed networking, but again the linking of SDVN with 5G will bring several challenges. To be precise, designing a communication model for SDVN becomes more challenging due to the limited coverage and short-range communication technologies such as Wi-Fi that cannot address the connectivity of vehicles. More or less, fetching important information from the cloud [13] and the rest of communication can be achieved through Wi-Fi technology.

SDN-NDN MUTUALITY

Nevertheless, there are similarities in the concept of SDN and NDN; for example, a tentative change to conventional switches and routers in the form of decoupled planes and retrieving the required content by naming the content instead of devices (end-to-end devices, NDN).

Similarly, the previously scanned literature considers NDN to exchange Interest/Data messages over a single wireless interface (i.e., 802.11 in the case of VNs). However, it is foreseen that vehicles will be equipped with a variety of wireless communication interfaces thanks to the NDN forwarding strategy, which is natively able to deal with multiple network interfaces [14]. Likewise, SDN also tends to support multiple interfaces by allowing the control plane to make decisions based on the packet's requirements. Moreover, NDN can decide, for each incoming Interest, which outgoing face to use (e.g., the best-performing face, one of a subset of possible faces, or any available face). On the other hand, in SDN, the decision depends on the implemented forwarding strategy, which may take as input the face cost, determined by the routing protocol, and the face rank, maintained by the flow tables. In the following section, we present an architecture for combining SDN-NDN for VNs and discuss the tentative working principles of our proposed architecture.

WHEN SDVN MEETS VNDN: AN ARCHITECTURE

The main objective of the proposed architecture is to provide SDN services and benefits to efficiently communicate named contents over VNs with better network resource utilization. SDN-based named data VNs consist of mobile vehicles, fixed infrastructure points — roadside units (RSUs), base stations (BSs), and so on — and the SDN controller. The proposed software defined vehicular named data networking (SDVNDN) architecture can be deployed as a clean slate or an overlay architecture on the current IP. For the control and data planes, any of the wireless interfaces in the network can be used by the VNs. All these VN elements run our proposed SDN architecture. The working principle of each component of the proposed architecture is discussed below.

COMPONENTS OF THE PROPOSED ARCHITECTURE

SDN Controller: The SDN controller is a network entity that has an overall network view to effectively orchestrate the network elements to efficiently perform the NDN operations including content caching, intelligent Interest and Data message forwarding, broadcast control, push-base services, and so on. Upon reception of an Interest with a specific content name prefix, it can easily delegate the forwarding, caching, and other strategies to the SDN-enabled nodes to avoid network congestion and provide quality of service (QoS)-based content communication between the provider and consumer nodes. The controller also forwards the prefix(es) (name or IP prefix depending upon the clean slate or overlay implementation, respectively,) to the network elements to receive Interests no forwarding rules of which are defined in the FIB table.

Caching: In NDN, per packet caching is performed by the nodes involved in the content forwarding process. The caching policies include cache all, probability-based caching, popularity-based caching, and no caching. Other than the caching policies, the cache location and caching pattern are also important in NDN. Cache location and caching pattern determine where the content should be cached and in what format (either whole or chunk-based content caching), respectively. Due to the dynamic nature of the network, the fixed elements at the edge of the VNs or potential nodes (maybe parked vehicles or vehicles with low mobility) may play a vital role in caching contents. It is worth investigating the network performance when RSUs, BSs, access points (APs), or potential nodes are selected as caching anchors. The SDN controller must select the feasible caching location(s) to avoid unnecessary copies of the content to balance the CS capacity and search overhead in the network. In the case of a large content, each Data message communicates a fragment or chunk of the content. Each content chunk in a separate Data message is forwarded through dynamic multiple paths, where each node along a path may be different when content chunks are forwarded to the consumer node. In this situation, the whole content may not be forwarded through the same set of vehicles. However, the question arises of whether it will be feasible to cache random fragments of the content and for how long those fragments should be cached. Because of the dynamic topology, the vehicles may receive unsolicited fragments of a content, and the network must describe the caching policies for unsolicited content fragments.

Content Naming: In NDN, either the whole content or a chunk of the content is identified and accessed using the name. The well-known NDN implementation of integrated computing networking (ICN) uses the hierarchical name structure consisting of several components in the name, each separated by /. In the VN scenario, each name component defines the relationship between content or content chunks and its spatial and/or temporal information where the content belongs. As the content searching and forwarding decisions depend on its name, the name must contain necessary components to precisely and efficiently receive the content from the network.

Intelligent Forwarding: The FIB table plays a pivotal role in content communication in the NDN. It contains name prefix, outgoing face ID(s), and some key parameters including the rank of the face(s). Every time an Interest is satisfied through a specific face, its rank is increased, which makes it more suitable for Interest forwarding in the future. Conversely, the successive data retrieval failures reduce the face's rank. This rank value is used to prioritize the face and its preference to forward Interests. If the rank value exceeds the threshold, it may be purged from the FIB. Therefore, the Interest satisfaction success probability depends on the updated FIB information.

If no entry is found in the FIB table, the Interest is forwarded to the controller, and based on the global network view and provider information, it defines the FIB entries plus their corresponding outgoing faces. These entries are delegated to the intermediate nodes between the consumer and provider nodes. The controller should include faces in FIB entries that satisfy the Interest's QoS requirements (e.g., minimum delay, low cost, high bandwidth). Once the Interest is successfully forwarded to the provider (either through single path or multiple path, or in a broadcast manner) and PIT entries are maintained at intermediate nodes, the content communication back to the consumer is a relatively straightforward operation.

Push-Based Forwarding: Push-based communication is one of the fundamental requirements of safety-based vehicular applications (accident or crash warning, emergency vehicle approaching, blind spot warning, etc.). However, there is no standard communication mechanism specified in the name data communication architectures to forward time-critical warning messages in a push-based manner. Most push-based communication requires location and heading formation of the vehicles. In an accident warning scenario, a warning message generated by the vehicle should be forwarded in time to all the following vehicles behind the warning issuing vehicle(s). Conversely, in the emergency vehicle approaching scenario, the warning message should be forwarded to the vehicles before the warning message generating vehicle. This module formulates the rules based on the warning type and spatio-temporal traffic information from the topology indicator. The rules include warning dissemination area, time-critical QoS requirements, and warning dissemination direction. Those rules are periodically advertised in the network to keep all network elements ready in advance to efficiently communicate warnings.

Intrinsic Data Security: In named data architecture, the content security is intrinsic in the content itself. Every content chunk and its corresponding name in the Data message are digitally signed to prove the binding between them. Signatures are mandatory in each Data message. The provider information along with the public key evidence verifies the provenance of the data. Keys along with their digital certificates are communicated as a Data message in the NDN. However, the consumer and producer vehicles must use agreed upon security policies (public key verification and signing policies) to imply the content or content chunk verification. Every vehicle shares its security and access control policies for accessing the cached contents from the CS with the controller through fixed infrastructure nodes. Having a global view of the network, the security block of the SDN controller disseminates the security and access control policies to the nodes involved in the Interest and Data message propagation.

Congestion Control: There may be a case in VNs where Interest-Data traffic may accumulate at any node in the network depending on the events or popularity of contents in a specific network region. Due to a large number of Interests and contents' flow through a node, the Interests may not be satisfied because there is specific out face link congestion, or the CS may be full or getting larger (depending on the caching policy), which may increase the cache miss ratio. To alleviate the congestion at any network point, the nodes keep traffic status of every face, and this information is shared with the controller. In view of the network traffic information, the controller selects different faces as well as the caching points in the network to evenly distribute the network traffic.

Topology Indicator: In the case of a mobile producer, the location, heading, speed, and content prefix can be used by the SDN controller to predict availability of the providers and the consumers. Every vehicle shares this information to the fixed infrastructure network elements and the information of its neighboring nodes. Information provided by the topology indicator, content prefix manager, and state information modules are used to devise and disseminate the forwarding rules to the intermediate nodes between consumer and producer.

Content Prefix Manager: Every network element publishes its CS prefix list along with the respective expiration time and vehicle identification information to the SDN controller. If a content is cached at multiple providers, a single prefix along with the multiple provider IDs are stored in the list. When the controller receives an Interest,

The proposed SDVNDN architecture can be deployed as a clean slate or an overlay architecture on the current IP. For the control and data planes, any of the wireless interfaces in the network can be used by the VNs. All these VN elements run our proposed SDN architecture.

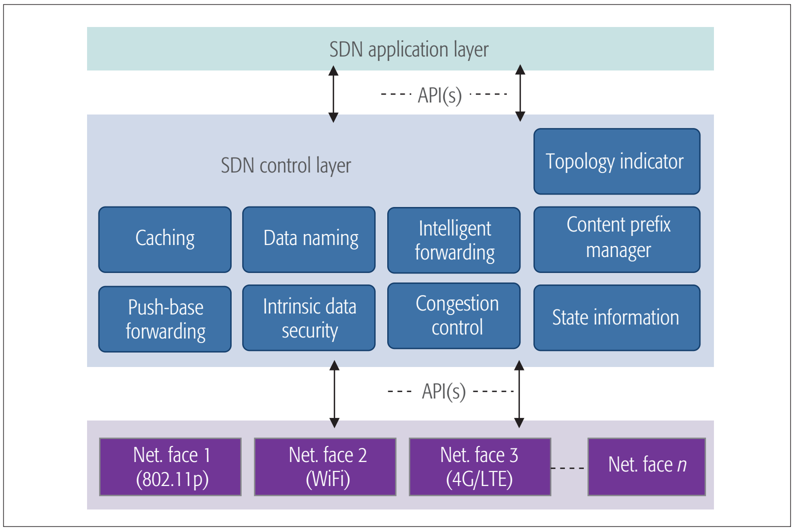

Figure 2. Proposed architecture for SDN-NDN-enabled VNs.

it performs prefix search in the content prefix list and finds the potential producer(s).

State Information: Every node monitors the Interest-Data traffic over each face (incoming and outgoing faces), per prefix traffic, and CS information (content replacement, hit or miss ratio, storage utilization, etc.). Every VN element shares its state information with the controller, and the controller manages this information in the state information table. This information helps the controller formulate better congestion control and caching policies.

Working Principles

In NDN architecture, the FIB serves an essential role to forward Interest messages in the network. Similarly, the availability of a content or chunk(s) of contents in CS depends on the popularity of the content, caching policy, and the timestamp until which that content is available in the CS. Once the timestamp expires, the entry in CS becomes stale and is purged accordingly. Additionally, the VN has a highly dynamic network topology due to high mobility of vehicles, and each vehicle may be equipped with more than one wireless interface. Therefore, the Interest messages may not be forwarded through the same set of interfaces and forwarding rules where they were satisfied previously.

Pull-Based Communication and Forwarding: Consider the example of a simple named data communication scenario in software defined VNs, as shown in Fig. 2. A consumer vehicle generates an Interest message including the content name and the unique Identifier (e.g., NONCE in NDN architecture). In a clean slate implementation, when an Interest message is received by the RSU or BS, it is simply forwarded to the controller through a predefined prefix in the FIB table. Intermediate network elements (switches, routers, etc.) between the controller and the RSU/BS forward the Interest using a vanilla NDN/CCN forwarding mechanism by maintaining the PIT entries. Once the controller receives an Interest, it searches the requested content's name prefix in the name prefix database to locate the potential content provider. After locating the content, the controller sets the forwarding, caching, and other policies based on the QoS requirements from the metadata provided in the Interest as well as the current network state. These policies are forwarded to all the VN elements involved in the Interest satisfaction process through the control plane. When network elements receive forwarding policies related to the Interest, which includes the name prefix and the outgoing face IDs of all the intermediate network elements between the consumer and provider nodes, they update their FIB entries. Subject to these policies, the Interest is forwarded to the provider node in a unicast or broadcast manner involving either the static network components or the vehicles. Refer to both cases in Fig. 3, where the Interest is forwarded in an ad hoc manner to provider A or through fixed infrastructure network elements to provider B.

In the case of an overlay architecture, the content name and related information from an Interest message is forwarded as an option in the IP packet to the controller through the controller advertised IP prefix and port information. Once the content provider is located by the controller, its address and forwarding rules are passed on to the network elements between the consumer and provider nodes. Upon reception of this information, the Interest is forwarded to the provider node. Due to the intermittent behavior of

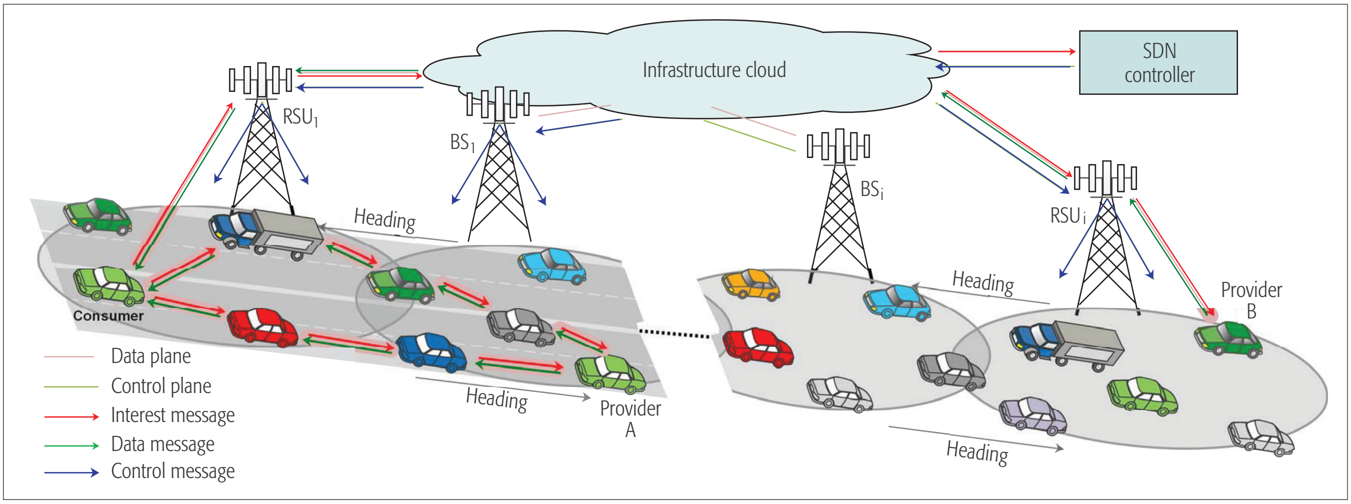

Figure 3. Pull-based named data retrieval in SDN-NDN-based VNs.

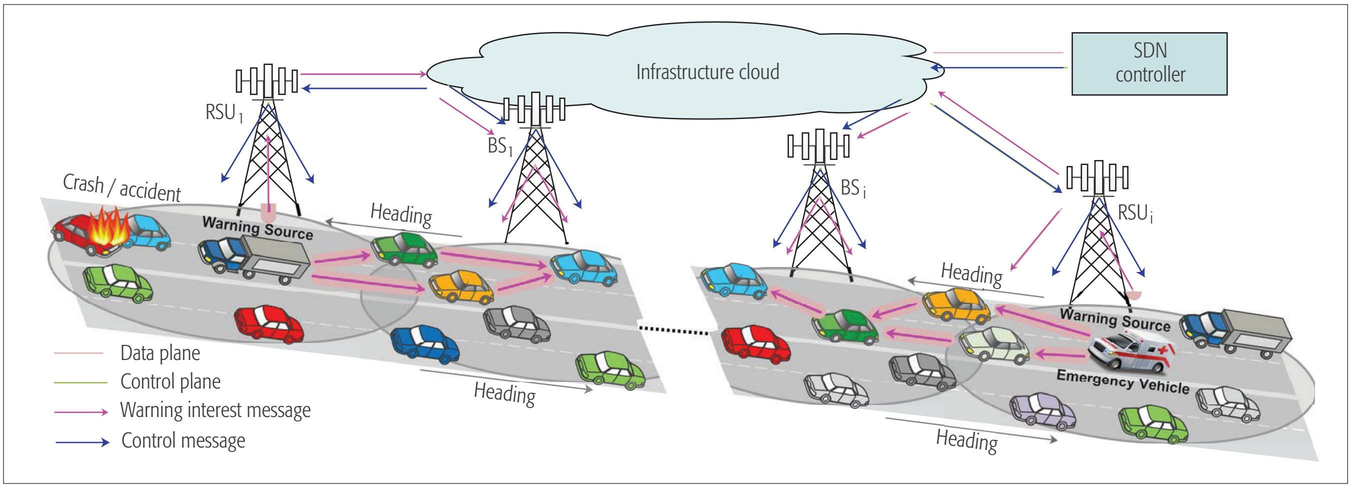

Figure 4. Push-based emergency message forwarding in SDN-NDN-based VNs.

VNs, the controller may define rules to forward Interests through multiple paths or in a broadcast manner. In VNs, there may be several consumers that generate Interests with similar name prefixes; therefore, the controller may define rules to combine Interests to collectively satisfy them.

Push-Based Communication and Forwarding: In our proposed architecture, the warning information is embedded as an extension in an Interest message with a flag to distinguish it from the pull-based Interest message. This special warning Interest message also includes the warning type, forwarding direction, PIT lifetime, and the warning generating vehicle's heading and location information. The warning Interest message is not forwarded to the controller; instead, the forwarding rules are periodically disseminated by the SDN controller in the network to avoid warning dissemination delay. Based on the heading direction of the warning message generating vehicle and the type of warning message, the VN's elements disseminate the warning in the specific region of the network accordingly. A push-based warning Interest for emergency vehicle approaching and crash scenarios are shown in Fig. 4. In case of a crash, the warning Interest is generated by a vehicle near the emergency scene. When RSU$_1$ and the trailing vehicles of the warning message source receive the message, they extract the warning information and forward to the application faces as well as the specified outgoing faces to inform the trailing vehicles and/or infrastructure elements. The trailing vehicles may receive the warning Interest from the RSUs, BSs, or the vehicles ahead of them. Warning Interests may be generated by the vehicles involved in the accident as well.

OPEN ISSUES AND RESEARCH ROADMAP

Our discussions above show that the cohesion of SDN and NDN in VNs can be beneficial in many ways. However, the proposed architecture is the first attempt and in its early stages, still far from the seamless transition. A few of the key open issues are:

- DN has been studied for wireless mobile networks to easily disseminate new services in the network. Along with the rapid increase in devices and services, more data is being generated, and thus we need to have sophisticated algorithms to search the required content. For such massive searching, NDN hierarchical naming of the content will be sufficient. Still, some real-time check using Hadoop and Dataset is missing from studies of SDN and NDN for wireless interfaces.
- In the joint venture of SDN, NDN, and VNs, forwarding tables play a vital role in making systems more efficient. Based on the current literature, it is hard to find research work focused on the active participation of forwarding rules tables. Mostly, we extract the content from tables and run algorithms on the upper layers; however, it could be more flexible to make intelligent tables' structures to enable them to take decisions.
- Similarly, communicating the content over a VN via SDN to cellular networks brings several issues and challenges [15]. For example, an abstract level of representation is needed to bridge SDN with VNs and cellular technologies. However, designing a generic platform for SDVN incorporating heterogeneous technologies can lead us to inappropriate network resources, network fragmentation, and interoperability between these networks.
- Finally, the existing trajectory prediction schemes need further modification to address the localization of vehicles over the SDN-NDN. Moreover, the decoupling of the data and control planes is not straightforward; therefore, high-speed Internet is required for intercommunication between both planes at two separate locations and devices.

CONCLUSION

We have foreseen the emergence of software defined networking and named data networking to retrieve the content in vehicular networks. The preliminary discussions highlighted the rationale behind SDN-based VNs, NDN-enabled VNs, and their similarities. Furthermore, the transition of future VNs has been discussed, and a contemporary detailed architecture of the SDN-NDN-based VNs has been proposed. Although SDN and NDN research is rapidly growing and approaching maturity, there is still the need for more attention from the research community to pave the foundation and guarantee effectiveness in the context of wireless and dynamic mobile networks such as VNs.

Although, SDN and NDN research is rapidly growing and approaching maturity, there is still the need for more attention from the research community to pave the foundation and guarantee effectiveness in the context of wireless and dynamic mobile networks such as VNs.

Acknowledgments

This study was supported by the BK21 Plus project (SW Human Resource Development Program for Supporting Smart Life) funded by the Ministry of Education, School of Computer Science and Engineering, Kyungpook National University, Korea (21A20131600005). Furthermore, the work of Danda B. Rawat is supported in part by the U.S. National Science Foundation (NSF) under Grants CNS-1658972 and CNS-1650831.

References

[1] J. Chen *et al.*, "Software Defined Internet of Vehicles: Architecture, Challenges and Solutions," *J. Commun. and Information Networks*, Vol. 1, no. 1, 2016, pp. 14–26.
[2] K. Zheng *et al.*, "Soft-Defined Heterogeneous Vehicular Network: Architecture and Challenges," *IEEE Network*, vol. 30, no. 4, July–Aug. 2016, pp. 72–80.
[3] L. Zhang *et al.*, "Named Data Networking," *SIGCOMM Comp. Commun. Review*, vol. 44, no. 3, July 2014, pp. 66–73.
[4] V. Jacobson *et al.*, "Networking Named Content," *Proc. 5th Int'l' Conf. Emerging Networking Experiments and Technologies*, Dec. 2009, pp. 1–12.
[5] S. H. Bouk *et al.*, "Named-Data-Networking-Based ITS for Smart Cities," *IEEE Commun. Mag.*, vol. 55, no. 1, Jan. 2017, pp. 105–11.
[6] S. H. Bouk *et al.*, "DPEL: Dynamic PIT Entry Lifetime in Vehicular Named Data Networks," *IEEE Commun. Letters*, vol. 20, no. 2, Feb. 2016, pp. 336–39.
[7] S. H. Ahmed, S. H. Bouk, and D. Kim, "RUFS: RobUst Forwarder Selection in Vehicular Content-Centric Networks," *IEEE Commun. Letters*, vol. 19, no. 9, Sept., 2015, pp. 1616–19.
[8] S. H. Ahmed *et al.*, "CODIE: Controlled Data and Interest Evaluation in Vehicular Named Data Networks," *IEEE Trans. Vehic. Tech.*, vol. 65, no. 6, June 2016, pp. 3954–63.
[9] R. Soua *et al.*, "SDN Coordination for CCN and FC Content Dissemination in VANETs," *Ad Hoc Networks*, Springer Int'l. Publishing, 2017, pp. 221–33.
[10] K. Xu *et al.*, "Toward Software Defined Smart Home," *IEEE Commun. Mag.*, vol. 54, no. 5, May 2016, pp. 116–22.
[11] A. Al-Fuqaha *et al.*, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Commun. Surveys & Tutorials*, vol. 17, no. 4, 4th qtr. 2015, pp. 2347–76.
[12] I. Ku *et al.*, "Towards Software-Defined VANET: Architecture and Services," *13th Annual Mediterranean Ad Hoc Networking Wksp.*, Piran, 2014, pp. 103–10.
[13] M. Shojafar, N. Cordeschi, and E. Baccarelli, "Energy-Efficient Adaptive Resource Management for Real-Time Vehicular Cloud Services," *IEEE Trans. Cloud Computing*, no. 99, 2016, pp. 1–1.
[14] M. Amadeo, C. Campolo, and A. Molinaro, "Priority-Based Content Delivery in the Internet of Vehicles through Named Data Networking," *J. Sensor and Actuator Networks*, vol. 5, no. 4, 2016, p. 17.
[15] F. Modesto and A. Boukerche, "A Novel Service-Oriented Architecture for Information-Centric Vehicular Networks," *19th ACM Int'l. Conf. Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2016, pp. 136–39.

Biographies

Syed Hassan Ahmed [S'13, M'17] (s.h.ahmed@ieee.org) received his B.S. in computer science from Kohat University of Science and Technology, Pakistan. Later, he completed his combined Master's and Ph.D. in computer engineering from the School of Computer Science and Engineering, Kyungpook National University (KNU), Korea, in 2017. In 2015, he was a visiting researcher at the Georgia Institute of Technology, Atlanta. So far, he has published over 70 international journal and conference articles in addition to two Springer brief books. From 2014 to 2016, he won the Best Research Contributor award in the Workshop on Future Researches of Computer Science and Engineering, KNU. In 2016, he also won the Qualcomm Innovation Award at KNU. He is also an ACM member while serving several well reputed conferences and journals as a TPC member and reviewer, respectively. His research interests include sensor and ad hoc networks, cyber-physical systems, vehicular communications, and future Internet.

Safdar Hussain Bouk [SM'16] (bouk@knu.ac.kr) received his B.E. degree in computer systems from Mehran University of Engineering and Technology, Jamshoro, Pakistan, in 2001. He received his M.S. and Ph.D. in engineering from the Department of Information and Computer Science, Keio University, Yokohama, Japan, in 2007 and 2010, respectively. He worked as an assistant professor in the Department of Electrical Engineering, COMSATS Institute of Information Technology, Islamabad, Pakistan, from 2010 to 2014. Currently, he is working as a research professor at KNU. His research interests include wireless ad hoc, sensor networks, underwater sensor networks, vehicular networks, and information-centric networks.

Dongkyun Kim [M'16] (dongkyun@knu.ac.kr) received his BS degree from the Department of Computer Engineering, KNU. He also received his M.S. and Ph.D. degrees from the School of Computer Science and Engineering, Seoul National University, Korea. He was a visiting researcher at the Georgia Institute of Technology, Atlanta, in 1999. He also performed a postdoctoral program in the Computer Engineering Department, University of California at Santa Cruz, in 2002. He has been on the Organizing Committees and Technical Program Committees of many IEEE or ACM conferences. He received the Best Paper Award from the Korean Federation of Science and Technology Societies, 2002. He has been involved in many editorial activities in several well-reputed international journals. Currently, he is a professor in the School of Computer Science and Engineering, KNU. His current research interests include connected cars, vehicular ad hoc networks, Internet of Things (M2M/D2D), Wi-Fi networks (including Wi-Fi Direct), wireless mesh networks, wireless sensor networks, and future Internet. He is the corresponding author for this article.

Danda B. Rawat [M'10, SM'13] (db.rawat@ieee.org) is an associate professor in the Department of Electrical Engineering & Computer Science and Founding Director of CWiNs Research Lab at Howard University, Washington, DC. His research focuses on wireless networks, cybersecurity, cyber-physical systems, Internet of Things, big data analytics, wireless virtualization, software-defined networks, and vehicular/wireless ad hoc networks. He is a recipient of an NSF Faculty Early Career Development (CAREER) Award. He has served as an Editor/Guest Editor for over 15 international journals, and has served on the Organizing Committees of several international conferences including IEEE INFOCOM 2015, 2016, 2017, and 2018, IEEE CCNC 2016, 2017, and 2018, and IEEE AINA 2015.

Houbing Song [M'12, SM'14] (Houbing.Song@mail.wvu.edu) received his Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, in August 2012. In 2017, he joined the Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL, where he is currently an assistant professor and the founding director of the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us). In 2017 he began serving as an Associate Technical Editor for *IEEE Communications Magazine*. He was the very first recipient of the Golden Bear Scholar Award, the highest faculty research award at West Virginia University Institute of Technology in 2016.