Journal of Network and Computer Applications 138 (2019) 39–50

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

SDN-enabled recovery for Smart Grid teleprotection applications in post-disaster scenarios☆

Abdullah Aydeger*, Nico Saputro, Kemal Akkaya, Selcuk Uluagac

Department of Electrical & Computer Engineering, Florida International University, Miami, FL 33174, USA

ARTICLE INFO

Keywords:
SDN
Wireless networks
Smart grid
Teleprotection
GOOSE

ABSTRACT

Maintaining Smart Grid communications is crucial for providing power services. This requires a resilient communication architecture that can instantly self-repair any failures in the communication links or routes. Emerging Software Defined Networking (SDN) technology provides excellent flexibilities that can be applied to critical power grid applications. In this paper, we consider the problem of link failures in inter-substation communications and provide self-recovery by relying on wireless links that can be the only viable means for communication after disasters. Specifically, we propose an autonomous framework, which can not only detect link failures, but also establish either a WiFi or LTE-based link among substations through SDN capabilities. To be able to effectively evaluate the performance of this proposed SDN-enabled framework, we developed it in Mininet emulator. Since Mininet does not support LTE connections, we proposed several unique extensions to Mininet by integrating it with ns- 3 simulator that supports the LTE/WiFi protocol stacks. We conducted extensive experiments by considering a teleprotection application using GOOSE to assess the recovery performance of the proposed framework. The results show that SDN-based framework can meet the deadlines for teleprotection on wireless links during the times of link failures in a reliable fashion.

1. Introduction

The existing Power Grids worldwide are going through massive transformations to make it more reliable and connected with the ability to transfer data in two-ways, which is referred to as Smart Grid (Saputro et al., 2012). The data communication motivation necessitates upgrading the existing network infrastructure with different components. With these new transformations, Smart Grid systems will need to maintain a large-scale heterogeneous network that brings a number of challenges. One challenge is the ability of this networking infrastructure to self-heal itself during man-made or natural (e.g., hurricane, earthquake) disasters so that the damage of potential blackouts and temporary outages can be minimized (Goldman et al., 2012). Through continuous interactions between different components of the Smart Grid, the new energy infrastructure should reconfigure the control of the physical assets and network topology in an efficient manner and achieve resilient operations.

One envisioned application for Smart Grid that requires the interaction between different components is the ability of the power distribution substations to communicate with each other and with the control center. This is developed under IEC 61850 standard as IEC 61850-90-1 (IEC61850-90-1, 2010) and IEC 61850-90-2 (IEC61850-90-2, 2016) which will enable different applications such as *teleprotection* (Apostolov, 2012). Since the underlying communication medium for most of the current substation-to-substation links is based on wired communications technology (e.g., Ethernet, power-line communications (PLC)), they can be easily damaged during certain disasters (i.e., storms, floods, minor earthquakes that cause link failures but still keep the substations alive) that may eventually lead to the damage of the physical components of the power grid. Self-healing in such unfortunate situations requires a comprehensive coordination among the components of the Smart Grid and needs to rely on the availability of backup or redundant links/paths and devices. Assuming that most of the network infrastructure may be damaged and not functioning, we envision that a fast switch-over to wireless communications should be the priority for quick restoration of services. However, it is a challenge to quickly reconfigure the network infrastructure since one needs to first identify

☆ A preliminary version of this paper, titled: *Software Defined Networking for Resilient Communications in Smart Grid Active Distribution Networks*, has appeared in the Proceedings of IEEE International Conference on Communications (ICC 2016), Kuala Lumpur, Malaysia, May 2016.

* Corresponding author.
E-mail addresses: aayde001@fiu.edu (A. Aydeger), nsapu002@fiu.edu (N. Saputro), kakkaya@fiu.edu (K. Akkaya), suluagac@fiu.edu (S. Uluagac).

https://doi.org/10.1016/j.jnca.2019.04.011
Received 21 June 2018; Received in revised form 1 March 2019; Accepted 22 April 2019
Available online 2 May 2019
1084-8045/Published by Elsevier Ltd.

the failures and then manually fix such failures which may be slow or even infeasible in post-disaster scenarios.

These challenges can be perfectly addressed with the employment of the emerging software defined networking (SDN) paradigm that splits control of underlying communication infrastructure and data flow operations (Hu et al., 2014). One of the major goals in SDN is to be able to interact with the networking equipment (e.g., routers, switches) to create an open networking architecture. In this way, one can get a global view of the entire network and will be able to make global changes without having to access to each device's unique hardware. Eventually, various large-scale network architectures can be deployed and maintained easily while still featuring resiliency and robustness.

Considering such capabilities of SDN, we propose using SDN to enable interaction among the gateways in substations and/or control center of Smart Grid. Specifically, we propose to have resilient links at each substation through OpenFlow switches and utilize them in case a disaster occurs. These resilient links could be wireless, e.g., WiFi or LTE based. To the best of our knowledge (Rehmani et al., 2018a), our work is the first to leverage SDN-enabled devices and SDN controllers to do self-recovery through a *wireless* resilient communication infrastructure among the substations.

The SDN-based approach is also equipped with a link failure detection feature that enables real-time self-recovery. Such detection capability is a proactive approach based on the idea of OpenFlow fast-failover groups, which is supported starting from OpenFlow V1.1.0. With this method, our switches will be able to update their flow rules to use active ports with the highest priority while sending port-status messages to the SDN Controller so that it can find better alternative routes dynamically. Finally, we integrate cellular LTE links in the proposed approach that can work with SDN in an efficient manner. The LTE connection is provided by attaching a User Equipment (UE) to one of the interfaces of each SDN gateway switch in the substations.

With the above components, we layout the foundations of a comprehensive SDN-based framework for teleprotection applications. For a realistic testing of the proposed framework, we built a testing tool that can be used by researchers to implement LTE or WiFi-based connections with SDN switches. This is achieved by integrating the widely used Mininet Emulator (Lantz et al., 2010) with ns-3 that can support the LTE/IEEE 802.11 protocol stacks. Specifically, we designed and implemented a custom patch to have an LTE connection from one switch to another in Mininet by utilizing some of the ns-3 functions. To the best of our knowledge, this work is the first to provide such an integration for a more detailed network-oriented evaluation.

We would like to note that our contribution in this paper is mostly experimental that integrate multiple existing novel concepts and provides an evaluation framework to be able to assess the effectiveness of the proposed SDN-based approach. Our contributions can be divided to two items: First, we propose a novel SDN framework that can be used for the Smart Grid communications at the substation level. This framework includes different communication links among substations and SDN controller. We integrate a link failure detection feature to this framework that enables real-time self-recovery. Such detection capability is a proactive approach based on the idea of OpenFlow fast-failover groups, which is supported starting from OpenFlow V1.1.0. With this method, our switches will be able to update their flow rules to use active ports with the highest priority. We also integrate cellular LTE links in the framework that can work with SDN in an efficient manner. The LTE connection is provided by attaching a User Equipment (UE) to one of the interfaces of each SDN gateway switch in the substations. Second, for a realistic testing of the proposed framework, we built a testing tool that can be used by researchers to implement LTE or WiFi-based connections with SDN switches. This is achieved by integrating the widely used Mininet Emulator (Rehmani et al., 2018a) with ns-3 that can support the LTE/IEEE 802.11 protocol stacks. Specifically, we designed and implemented a custom patch to have an LTE connection from one switch to another in Mininet by utilizing some of the ns-3 functions. To the best of our knowledge, this work is the first to provide such an integration for a more detailed network-oriented evaluation.

We tested the effectiveness of our SDN-enabled framework by considering inter-substation communications that are exchanging Generic object oriented substation events (GOOSE) data among Intelligent Electronic Devices (IEDs) (Mackiewicz, 2006) on different teleprotection applications. Specifically, we analyzed the end-to-end packet transmission delay, packet loss, switching delay, and the number of packets missing the delay requirement of teleprotection as metrics which are caused by link failures. The results show that SDN-based recovery can meet the deadline of 4 ms for teleprotection applications, for both LTE and WiFi connections.

This paper is organized as follows. In Sections 2 and 3, we discuss the relevant work and provide some background on SDN and Smart Grid. In Section 4, we introduce the proposed SDN-enabled inter-substation network architecture. Detailed performance evaluation of the proposed work is given in Section 5. Finally, we conclude the paper in Section 6.

2. Related work

Power grid has a lot of domains (generation, transmission and distribution) and within each domain there are various applications with diverse requirements. Therefore, a solution applied to one domain or application will not be applicable to other application. Below, we summarize related work with applications in various domains that utilized SDN or dealt with recovery.

SDN-based Smart Grid resilience is studied for different Smart Grid applications (Gyllstrom et al., 2014; Ghosh et al., 2016; Ren et al., 2017). In Gyllstrom et al. (2014), the authors proposed SDN-based link failure recovery mechanism for PMU networks that handles link failure detection based on the rate of packet loss. They show the problem of Multicast Recycling is NP-hard and come up with their algorithm to find a backup tree. They consider multicast PMU application and tried to minimize the control plane signal overhead whenever find backup multicast trees. They also propose fast backup tree installation. The problem tackled in that paper is similar to ours but there are many differences that prevent us taking this solution to our application domain. First, they consider an IEEE bus system, which assumes a network topology that is connected with Ethernet links. In any failure, the backup is sought within the network by looking at alternative trees. And since they use multicasting, they strive to create a backup multicast tree using SDN. In our case, we do not have backup Ethernet or wired links. We rely on wireless links since the communication is mainly assumed between two substations. Switching to a wireless through fast failover property of OpenFlow and looking into data performance of wireless link is not studied in that paper. Moreover, the data transmission and latency requirements are also very much different than GOOSE in our application.

In Ghosh et al. (2016), the authors studied how the delay in Smart Grid communications due to the failure of the SDN controller (which represents a single point of failure) may impact the performance of the underlying critical physical system such as automatic gain control that regulates the grid frequency to a critical nominal value. In Ren et al. (2017), the authors proposed an SDN-based communications that incorporates network delay guarantee, automatic failover, and traffic prioritization functionalities for resilient microgrid operations. The delay guarantee is achieved by monitoring link latencies through the use of three types of special-purpose Ethernet frames. In these two works, the focus is on intra-substation communications while we focus on inter-substation communications which has longer distance between devices and has strict quality of service (QoS) requirements in terms of latency and reliability. Thus, a solution for an intra-substation communication would not fit into our case directly. What we investigate in this paper is the ability of SDN to meet the unique application requirements when used with the right components such as quick failure detection, proactive link/route selection etc. that has not been done in the previous works.

We also would like to point out that there has been a lot of work on general network topology recovery that do not directly apply to our case. We believe that network topology recovery problem is a different problem than ours since it is studied within the transmission domain and unique latency challenges are not enforced there. The approaches that can provide real-time communication such as (Kim et al., 2012) for demand response (Chai et al., 2015), for PMU data communications in distribution networks and (Johnston et al., 2006) for real-time monitoring can complement our approach if underlying GOOSE is not used for data communication. In our case, the unique challenge is to be able to perform switching within the time limits of GOOSE so that the substations will be protected.

Another recent work on SDN-based Smart Grid resilience that also attempts to thwart eavesdropping of traffic flows between the master station and a substation in SCADA network by dispersing traffic across multiple paths is presented in da Silva et al. (2015). For any given master station and a substation pair, *N*-shortest routes are calculated using Dijkstra's algorithm and an OpenFlow *Hard timeout* timer is used to initiate the route changing each time the timer expired. While we also follow the idea of redundancy in this work, we achieve it at the link level with wireless complementary links different from (da Silva et al., 2015). In Dorsch et al. (2014), the authors considered both resilience and QoS for Smart Grid applications when SDN is employed. Specifically, they considered the performance at the transmission and distribution networks focusing on Manufacturing Message Specification (MMS) traffic. The tests were carried out on an actual testbed of OpenFlow switches. They studied the recovery delay when one of the links was removed manually. The authors in Dorsch et al. (2016) proposed a hybrid approach of local and centralized solutions for failover case in Smart Grid networks. They used Bidirectional Forward Detection (BFD) as the local solution and implemented a module on SDN controller in order to react with the global knowledge of the network for the centralized solution. This hybrid approach can satisfy sub 50 ms requirement of carrier grade networks (Niven-Jenkins et al., 2009). In Zhang et al. (2016), Lee and Shin (2018) and Rehmani et al. (2018b), the authors propose resiliency in Smart Grid networks utilizing SDN capabilities by switching to an alternative path if the current one is down. The authors in Al-Rubaye et al. (2017) also propose network failover in Smart Grid with SDN switches, yet as an addition they use BFD packets in order to detect the failure quickly. In Kurtz et al. (2017), different failover strategies are observed and results are compared within an SDN environment. The same authors enhanced their work to enable hard service guarantees by applying Network Calculus in Dorsch et al. (2018). In all these works, the network mainly re-routes through another path. This is different from our case where we do not consider just re-routing, but employ redundant wireless links in substation to substation communications since we believe that in some cases there would not be any alternative way to do re-routing for some substations. Specifically our target communication infrastructure (inter-substation communication) is not so feasible for path protection considering the long distances and few number of substations in a specific area. We propose having alternative communication channel and use it for a short time during an emergency until the wired links are recovered. Even though we consider switching to wireless links at first, in SDN Controller we still consider checking if there is a better (faster) way to the destination afterwards. Thus, our framework covers their solution as well as improving it with resilient links in the network.

We also would like to note that this work is an extended work of our previous paper (Aydeger et al., 2016) which had some preliminary results by considering an MMS application under WiFi links within a substation communication. The extensions and difference from that paper are; (1) Consideration of LTE links for backup links and integrating LTE with Mininet. Previous paper focused only on WiFi; (2) Consideration of GOOSE under teleprotection and an encapsulation mechanism for GOOSE packets to be carried by LTE since LTE works with IP packets only; (3) Design and implementation of our link failure detection into the switches by utilizing SDN capabilities. There was not proactive detection in the previous work; (4) Extension of our framework to support multi-hop WiFi links along with the experiment results; (5) Implementation on the GENI testbed. Finally, this work is also very different than our work in Akkaya et al. (2015) where we surveyed possible SDN-based wireless technologies in Smart Grid environment. There is no resilience focus in that work.

3. Preliminaries

3.1. SDN advantages

SDN's main advantage and motivation is to move the control of the lookup tables inside the network devices to a separate location so that it can be controlled more easily centrally. Specifically, this can be described as separation of the packet forwarding from the way of how the forwarding tables are created and changed. These two processes are assumed to be on separate layers, which are referred to as *data plane* and *control plane* in SDN terminology.

One major problem with the traditional network settings involves updating the network elements after configuration, topology changes. By creating a programming interface to be able to update network elements centrally via SDN, such complexity in network management is eliminated.

3.2. Smart grid substation networking architectures

Smart Grid has three main components for generation, transmission, and distribution of the power. For transmission and distribution, substations at different geographical locations are used. For each of these components, transmission and distribution data networks are also used for communicating the collected field data.

A substation contains numerous IEDs, each generating various information about the status of some aspect of the substation. In new-generation substations, the IEC 61850 standard is used for substation automation, control, and wide area communications (Budka et al., 2010). SCADA systems are used for control centers to collect data from field devices such as PLCs, PMUs, and IEDs at a substation in real-time and perform control decisions at the control center in terms of reliability and quality of service (Budka et al., 2010). Inter-substation communications is also possible for protection data (e.g., for line distance protection) or control data (e.g., for interlocking functions) exchanges (IEC61850-90-1, 2010). In the next section, we explain in detail how these components can be interconnected with the emerging SDN technology utilizing numerous advantages it brings.

3.3. GOOSE background

GOOSE is a messaging scheme that provides fast and reliable communication between IEDs. As can be seen in Fig. 1, unlike MMS that utilizes TCP/IP protocols, GOOSE runs over Ethernet without any further networking layer protocol needed. It supports multicast communication and is based on publisher/subscriber model (Kriger et al., 2013). There are different timing requirements of IEC 61850 such as Type 1A, Type 1B and Type 2 according to (IEC61850-90-1, 2010). The requirements of GOOSE could be 1A, 1B depending on the application. It is used for a number of applications including but not limited to distributed protection scheme (Naik et al., 2011) and bus protection (Duong and Cueco, 2016).

GOOSE messages are consisted of following fields; (1) 6 bytes of source address and 6 bytes of multicast destination address. (2) The 802.1Q Virtual Local Area Network (VLAN) is 4 bytes and includes the TPID (Tag protocol identifier), TCI (Tag Control Information) and Ethertype. The TCI and Ethertype is total of 4 bytes and the value is

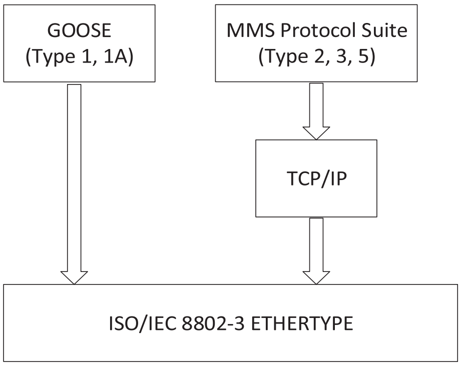


Fig. 1. The communication Layers of GOOSE vs. MMS.

0 × 88b8 for GOOSE. The length of Ethernet Protocol Data Unit (PDU) is the number of octets including the Ether type PDU header starting at APPID and the length of the Application Protocol Data Unit (APDU). Thus the length would be 8 + m, and m is less than 1492. The detailed analysis of GOOSE data structures can be found in Kriger et al. (2013).

4. SDN-based inter-substation network

4.1. Proposed model: overview

We propose an SDN-based communication infrastructure that can be deployed both within and among the substations as shown in Fig. 2. Specifically, each substation maintains an SDN gateway switch, which can be controlled by a global SDN controller that is located at the Utility Control Center. The global SDN controller can maintain the traffic among the substations (e.g., IEC 61850-90-1 GOOSE traffic) by inserting the flow rules in the table. The regular communication with the gateway can be achieved through a control line or the existing data network, typically wired. We consider having in-band communication between SDN switches and the SDN global controller since there are already established links between Utility Center and gateway switches, and SDN Controller runs on the same machine with Utility Center. The gateways at the substations can also be part of the local area network (LAN) within the substation and thus we also introduce a local SDN controller for controlling the traffic within this substation LAN. In this way, the flow table in the gateway can be accessible by multiple controllers, but the scope of these controllers are different. While the global SDN controller adjusts the inter-substation communications, the local SDN controller focuses on the interior traffic within a substation. We assume that the Global controller sits in a protected environment with backups and will not fail. Local ones can fail but they can be substituted with the global controller.

We considered emergency situations that cause the main wired link between two substations to go down and assumed that there would be backup link available. Note that wired backup link will not be reliable in case of a natural disaster since wired links are based on same features. To ensure a fast recovery after the disaster, a wireless link is used as a temporary backup link as shown in Fig. 2. The connection between two substations is based on the wireless redundant link that can be based on either a WiFi (802.11) or cellular (LTE) connection as will be detailed next.

4.2. Redundant back-up link using LTE public network

Fig. 3 shows our proposed SDN-based LTE redundant link model between substations. In our proposed model, a user equipment (UE) is connected to every local SDN switch. We assumed that a public LTE network is always available. In other words, we considered that the disaster does either not have any impact to the public LTE network or it is a partial impact which can be handled (e.g., UPS or drones are readily available to function the towers). Each UE has a Subscriber Identity Module (SIM) card from the provider of this public LTE network. When a wired link failure between two SDN switches is detected, the UE from each side is activated and they connect to LTE network. Then the switches changes their flow rule (by *fast-failover* capability as will be explained next) so that all traffic that previously passed through the failed wired link, is now redirected through these UEs. When this wired link is fixed, the SDN switch will change the flow table again to redirect traffic to the original wired link and UEs from both sides are disconnected from the LTE network.

The main issue in this model is related to the role of the UE. Typically, the UE is the end terminal in the LTE network and is assigned an IP address by the Evolved Packet Core (EPC) network (3GPP, 2011). Thus, the Packet Data Gateway (PGW) in the EPC network identifies the appropriate GPRS tunneling protocol (GTP) tunnel identity (TEID) for any downlink traffic based on the destination IP address in that traffic as depicted in Fig. 4. However, since in our case UE acts as a gateway to the substation network and the destination IP address in the down-

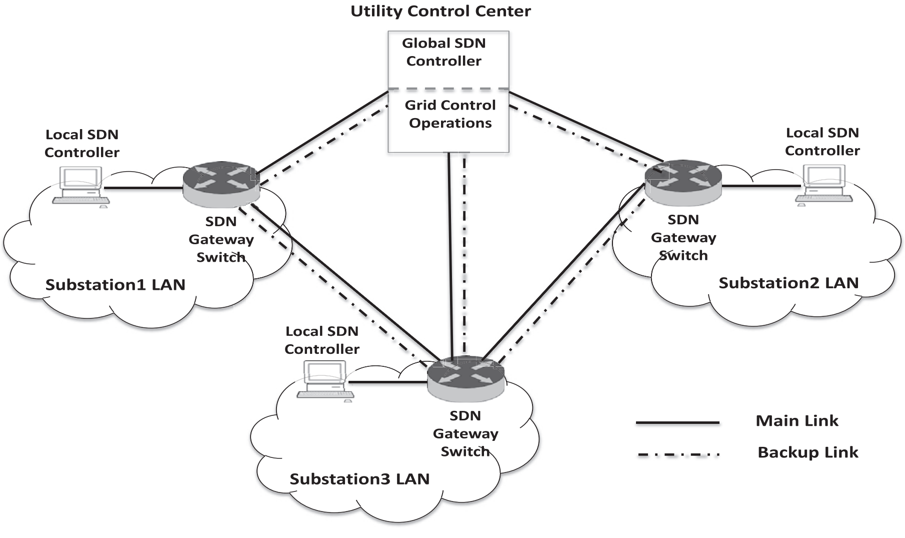


Fig. 2. Proposed SDN model for substation communications.

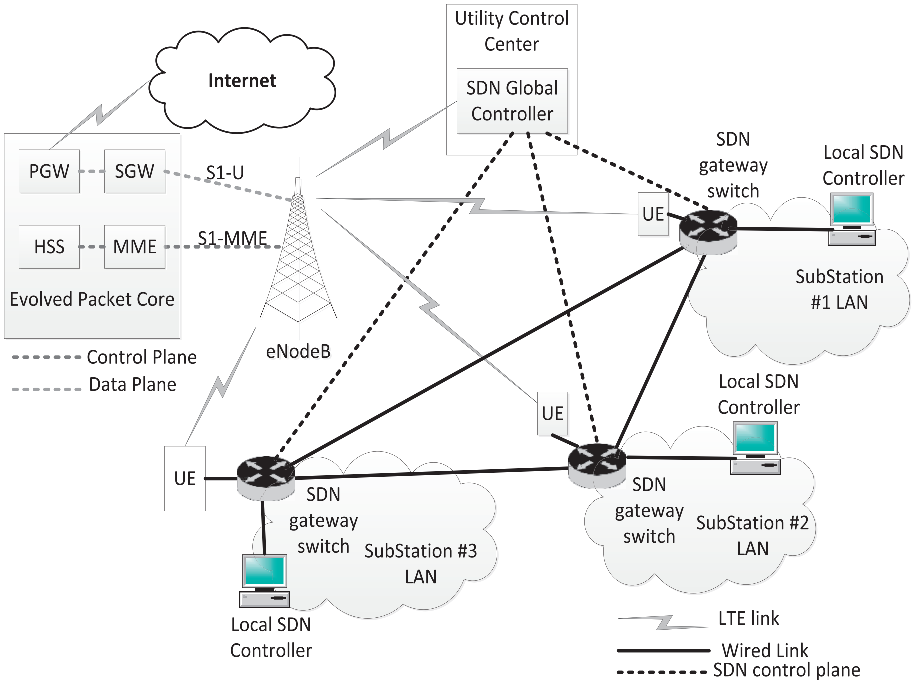


Fig. 3. Proposed SDN-based LTE redundant links model for substation communications.

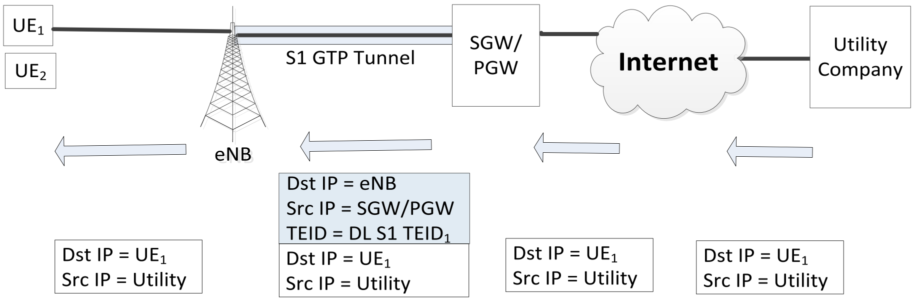


Fig. 4. LTE downlink traffic delivery.

link traffic is an IP address in the substation network, the PGW cannot recognize the appropriate tunnel identity and thus it fails to deliver this downlink traffic to the intended destination.

While a UE access list at the EPC network (Saputro et al., 2016) can tackle this issue, it still requires the LTE provider to know the substation network addresses to create the mapping. Besides the security concern of exposing the substation network addresses to the third party, it is also not practical in emergency situations where a quick response is required. Therefore, we propose a UE gateway application protocol that handles the UE's role as the gateway to the substation network. This application performs the following actions:

1. Reports its assigned UE IP address from the PGW to the SDN global controller;
2. Receives address mapping information from the SDN global controller. This address mapping contains the information about the destination substation IP address and its corresponding destination UE IP address;
3. Encapsulates and de-encapsulates the traffic between the communicating UEs. For this purpose, a new IP header is created, where the source and destination IP addresses are the UE's source IP address and UE's destination IP address respectively as depicted in Fig. 5. This way, the PGW can assign the appropriate TEID.

Note that since LTE network is an IP-based network, in case of GOOSE message, which is multicast data-link layer traffic, our proposed UE gateway application protocol has some additional steps before the encapsulation and after de-encapsulation processes as follows:

1. *At the source UE gateway:* It captures the GOOSE message at the data-link layer and embeds it in an IP datagram where the source and destination IP addresses in this datagram are the UE's source IP address and a predefined multicast IP address (i.e., a specific IP address that can be used to indicate a group of destinations) as the destination respectively.
2. *At the destination UE gateway:* After de-encapsulation and the gateway knows that the destination IP address is a multicast IP address, the gateway removes the IP header and then pushes the GOOSE traffic to the data-link layer for the data-link layer multicast.

We would like to note that a comprehensive consideration of security features for the framework is beyond the scope of this paper. We believe the deployment of middleware like C-DAX (Heimgaertner et al., 2015) can address these issues automatically. However, privacy and source anonymity as pointed out in case of IP address exposure needs to be studied on top of C-DAX services.

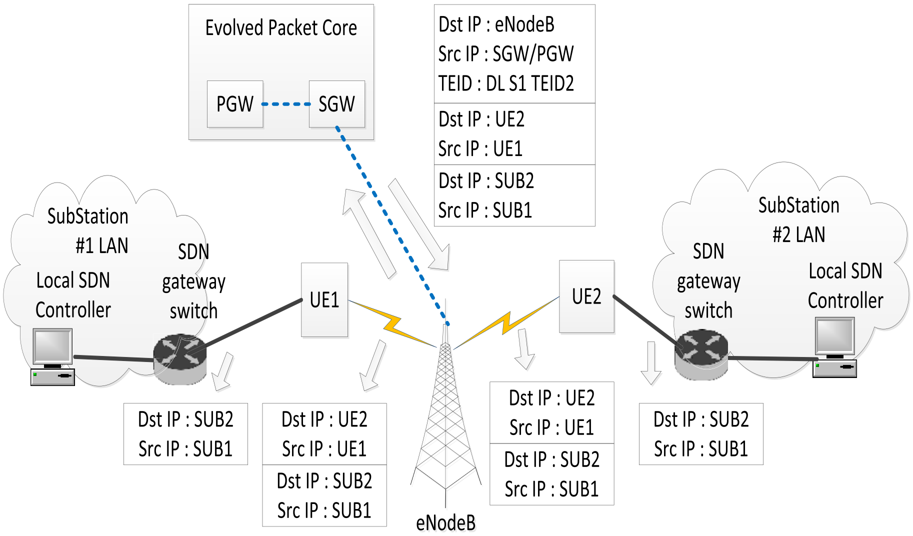


Fig. 5. Proposed LTE tunneling between UEs.

4.3. Redundant back-up link using IEEE 802.11

In cases where the disaster is stronger and the public LTE network is not available or partially available, there can be other potential solutions such as using LTE Device-to-device (D2D) communication (Nishiyama et al., 2014) or hybrid multi-hop WiFi communication and LTE (Saputro et al., 2019). In this section, we consider the case where battery-operated WiFi devices can be used to establish a multi-hop wireless backup link. In this case, each SDN switch is assumed to be equipped with a Wi-Fi Mesh enabled device that supports IEEE 802.11s standard (IEEE Std 802.11s, 2011) which can work with various MAC protocols such IEEE 802.11n. Note that for long-distance communications this can be the newly developed IEEE 802.11ah standard (IEEE Std 802.11ah, 2017). This device will be activated in case of emergency situations that cause a wired link failure.

However, before the wireless link can be used, the 802.11s devices need to find a route from one switch to another which is by default handled by the standard. Note that this process is a one-time process and can be done in advance to save time for the applications which require meeting certain end-to-end delay times.

4.4. Possible SDN-based failure detection mechanisms and recovery and our solution

Link failure detection and installing new flow entries in an SDN switch can be performed in a reactive fashion (also known as *restoration* approach) that involves the SDN controller in the process of finding new routes, or in a proactive fashion (also known as a *protection* approach) without SDN controller intervention. In a reactive solution, an SDN controller sends Link layer discovery protocol (LLDP) packets periodically. Thus, the recovery time will consist of LLDP packet transmission from Controller to switches, getting replies back, comparing the topology with previous state and updating the flows if necessary. In proactive solution, SDN switches are able to react whenever a failure is detected. In this case, alternative flow rules are installed to the SDN switches to be activated in case of a failure. Recovery time for proactive case depends on failure detection time and switch's flow rule reconfiguration (which is done automatically). As a comparison, reactive solution is slower but more dynamic, and proactive solution is faster but requires pre-configuration. Since meeting the delay requirements of Smart Grid applications (e.g., teleprotection) is crucial and the recovery time from a reactive solution might delay the packet transmissions, we opt for a proactive solution by exploiting the openflow *fast-failover group* (Openflow switch specification, 2011) and using the per-link BFD for the link-failure detection. An OpenFlow *fast-failover group* connects a couple of action buckets. Each action bucket consists of watch ports with specific actions for each port. If the highest priority watch port is active, the switch will perform the actions for that port.

In our work, a certain switch port in SDN gateway switch is assigned to a wireless communication channels. The priority of this wireless-assigned port is lower than the other switch ports that are assigned for the wired links (i.e., wired-assigned ports). When a higher priority port (i.e., the wired connection) is alive, the traffic will be forwarded through that port. However, whenever the *port-down status* for a wired-assigned port connected to other SDN gateway switch indicates that the port is down, the wireless-assigned port, which is the next higher priority port, will be used for forwarding the traffic. The SDN controller will pre-install these flow rules in the OpenFlow *fast-failover groups*. This way, we enable the SDN gateway switch to automatically forward the traffic, which is previously passed through the failed wired link, to the wireless link without any need to communicate with SDN controller. Additionally, we consider that the SDN switch also sends a *port-status* message to the SDN controller to trigger it to find any other possible routes (e.g., multi-hop paths through other wired-links) that may be better than using this temporary wireless link. If there is a better path, new flow rule can be disseminated by the SDN controller to the SDN gateway switch(es). This means that even though we opt to use proactive method as a solution right after the failure happens, we still use reactive solution by utilizing SDN Controller in order to have optimal path after a failure. Note that this cannot be done without SDN's capabilities.

Standard per-link BFD *liveness* mechanism is used to recognize a link failure. For each particular wired link, a BFD session is established and a periodic *hello* message is sent through that link. According to our configuration; if three messages in a row are not received by the other side, BFD assumes that a link failure occurs and then the corresponding *port-down status* is set to indicate a link failure.

4.5. Testing tool: mininet setup and its integration with ns-3

Even though there are many smart grid testbeds available (Cintuglu et al., 2017), none of them supports SDN-based infrastructure. Furthermore, the performance evaluation of the proposed SDN-enabled architecture in real substation to substation environment is not possible

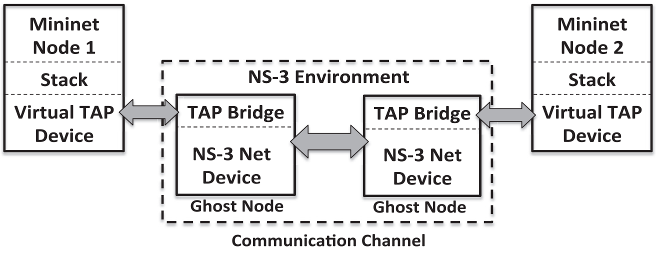


Fig. 6. Modeling channels in Mininet using ns-3 features (Aydeger et al., 2016).

since no utility has such an infrastructure which will be accessible for research. Therefore, we rely on emulators that are freely available such as Mininet (Lantz et al., 2010), an open source emulator that comes with OpenFlow protocol support and can work with internal as well as external SDN controllers.

Since Mininet can only mimic the behavior of protocols that are available in the Linux protocol stack, the support for backup wireless links and testing of LTE-based links in a realistic manner is not possible with the current environment. This does not only limit the research capabilities regarding SDN deployments but is also not flexible in terms of testing network resilience, fault-tolerance, real-time behavior and security when SDN is deployed. Therefore, there is a great need to integrate Mininet with one of the existing network simulators for comprehensively evaluating the effectiveness of SDN-based control on Smart Grid communication networks.

To this end, this paper proposes a novel mechanism that uses ns-3 communication channel among the Mininet nodes by bridging the capabilities of Mininet and ns-3, a C++ based discrete event simulator (Network simulator - ns - 3, 2017). Each device, link, protocol, application, etc. can be represented as objects and linked together to create network topologies. The proposed generic ns-3 integration model, is shown in Fig. 6. As can be seen, the *Tapbridge* object in ns-3, which effectively allows host systems and virtual machines running native applications and protocol stacks to integrate with a ns-3 simulation, is used. In our case, ns-3 connects to a *Virtual TAP Device* interface created on Mininet. Packets sent by Mininet host to the *Virtual TAP Device* are transmitted through a file descriptor to the ns-3 process. Next, they are forwarded down by *Tapbridge* to the *ns-3 Net Device* and transmitted over the ns-3 emulated channel. This allows us to analyze the behavior of native protocol suites (such as 802.11, LTE etc) in large-scale networks that may not be supported by Mininet.

In our work, either for IEEE 802.11 or the LTE network, both hosts and SDN switches are Mininet nodes. The integration with ns-3 is done at the switch node by connecting the virtual TAP device interface in the Linux to the ns-3 Tap-bridge in the ghost node. For each SDN switch, the connection from the virtual TAP device and Tap-bridge in the ghost node is created. The ghost node is a CSMA (Carrier Sense Multiple Access) network device model and is connected to either an IEEE 802.11 or a UE node that has two interfaces: a CSMA net device interface and an IEEE 802.11/LTE interface. As depicted in Fig. 7, the gateway application module described in Section 4.2 is built in ns-3 environment on top of the IP protocol of both interfaces to handle the traffic from/to the Mininet node and LTE. Hosts (i.e., the sender and receiver) are emulated in Mininet while the communication protocols are based on ns-3.

5. Experimental evaluation

5.1. Experiment setup

We utilized Mininet, ns-3, and FloodLight SDN controller (Project, 2014) to evaluate the performance of our proposed work. Network topology as in Fig. 2 was created in Mininet and integrated with ns-3 links as described in Section 4. While ns-3 provides pretty good capturing of the behavior in many cases, it is not perfect to reflect accurately the actual scenarios. However, it still does provide a good sense of the issues as studied in our previous work (Ozgur et al., 2016). Therefore, we decided to include experiments by generating background LTE traffic to investigate its impact on the delay. For testing the proposed framework, we used GOOSE, a data-link layer publisher-subscriber mechanism within IEC 61850 framework to ensure fast messaging within a 4 ms end-to-end delay enforced for teleprotection applications (Apostolov, 2012). We specifically consider Type 1A ‘Trip’ messages in our model. Example of such messages between substations are ‘block’ and ‘release’ binary messages (for our case, it would be necessary for substa-

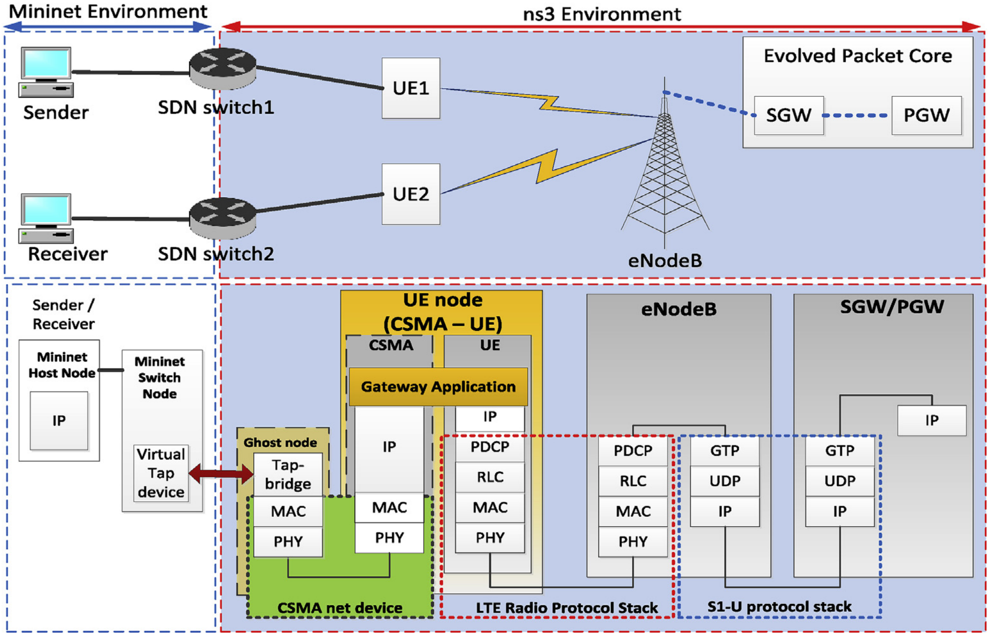


Fig. 7. The data plane integration of Mininet and ns3 LTE.

Table 1
Switching delay results (in ms).

Link Type	Switching delay (avg, std-dev, confidence interval 99%)		
	1s data generation frequency	20 ms data generation frequency	5 ms data generation frequency
Wired	(0.4, 0.2, 0.1)	(0.6, 1.0, 0.3)	(1.0, 1.4, 0.4)
WiFi	(1.1, 0.3, 0.2)	(1.3, 0.8, 0.2)	(2.9, 2.0, 0.6)
LTE	(1.1, 1.6, 1.1)	(21.2, 5.1, 1.5)	(36.5, 6.9, 2.1)

tions to block some of the devices in their network to enable urgent electricity need). These messages are mission critical and the most important fast binary messages. Thus, they have a strict and tight transfer time. In our experiments, a publisher in one substation is periodically sending multicast GOOSE messages to the subscriber in another substation through the SDN gateway switches. Our GOOSE packets were 20 bytes. Even though two different data generation frequencies are used in the experiments (i.e., 1s by considering steady state condition and 5 ms by considering a signal change as specified in Niejahr et al. (2010)), we also run experiments considering 20 ms data generation frequency. The simulation time is set to 2 min and the occurrence of the link failure is assumed to happen at the 60th second from the beginning of the simulation for the experiments with 1s data generation frequency. For 5 ms and 20 ms data generation frequencies, we set the simulation time to 10s and failure is assumed to be at 5th second. The BFD packet size is 66 bytes and its transmission rate is 1 ms. Note that we repeated each experiment 50 times for statistical significance and reported the average result.

5.2. Benchmark and metrics for testing

Fig. 2 represents the basic topology to test the effectiveness of the proposed SDN-based approach. Each substation connects to an OpenFlow-enabled gateway router. A double-link connection, which consists of the main wired-link and a backup-link, is used to support the resiliency of the communications between the substations through these gateway routers. The global SDN controller manages these gateway routers. The data transfer between substations uses the main wired-link. The backup-link, which can be wired or wireless, can be activated when there is a link failure in the main wired-link. In our experiments, we considered three different network topologies based on the communications technologies for the backup link as follows:

1. *Wired Topology*: This topology has double wired links to connect two SDN gateway switches for substation to substation communications. It is used as the benchmark for the wireless cases. We configured the throughput between nodes as 1 Gbps and the distance is set 100 m to be able to fairly compare with the wireless case.
2. *Proposed Wireless*: This topology used IEEE 802.11n wireless link as the redundant link to connect two SDN gateway switches in addition to the wired connection. The distance between the two switches is 100 m.
3. *Proposed LTE*: In this topology, LTE public network is used as the redundant link that connect two SDN gateway switches. The distance between a UE to the eNB is also 100 m in order to do a fair comparison with the proposed wireless case.

We considered the following performance metrics:

- *Switching Delay*: This metric represents the time required to restore the connection after a link failure occurs, which consists of a sequence of events: failure detection, switch flow-rule reconfiguration, and wireless/LTE link establishment until the new link is ready for transmitting the packet. Equation (1) represents the theoretical calculation of switching delay T_s, where T_{fd} represents the failure detection time, T_{config} represents the flow table configuration/update time from failed link to backup interface, and T_{link} represent the new link establishment time.

$$T_s = T_{fd} + T_{config} + T_{link} \tag{1}$$

However, the exact duration of each event would be very difficult to measure since it needs a timer within switch. Therefore, we approximated the switching delay T_s by subtracting the E2E delay of the first packet that is received at the destination after the link failure $D(T_{fail})$ from the average E2E delay of packets, depicted as $D_{E2E-avg}$, when these packets used the backup-link to reach the destination:

$$T_s = D(T_{fail}) - D_{E2E-avg} \tag{2}$$

- *End-to-end Delay (E2E delay)*: This metric represents the average ETE delay $D_{E2E-avg}$ of packets from the publisher to the subscriber host when the backup-link is used. We measured the $D_{E2E-avg}$ when the backup-link is fully functioning and there is no more packet affected by the link failure is still being transmitted. In other words, the measured E2E delays are purely coming from the newly transmitted packets after the backup-link has been completely established. Our empirical study shows that this situations starts around 1 s after the link failure at time T_{fail}. Equation (3) represents the computation of $D_{E2E-avg}$ where $D(t)$ represents individual packet delay generated at time t, and N is the total number of transmitted packets since $(T_{fail} + 1)$ until the end of the simulation time, $T_{\max}$.

$$D_{E2E-avg} = \frac{\sum_{t=1+T_{fail}}^{T_{max}} D(t)}{N} \tag{3}$$

- *Packet Loss*: This metric is used to assess the number of lost packets due to the link failure.
- *Number of Packets Exceeding the Delay Deadline*: This metric represents how many packets being sent cannot reach the destination within 4 ms deadline of teleprotection applications (Apostolov, 2012).

5.3. Performance results

The performance evaluations results are summarized in three tables based on the aforementioned metrics in Section 5.2.

5.3.1. Switching delay

We first compared the switching delay of our approach to that of Al-Rubaye et al. (2017). The results shown in Table 1 indicate that our wired case switching delay is less than or equal to 1 ms (e.g., 0.4, 0.6 and 1 ms respectively depending on the data frequency rate) while their result for the same process is about 6 ms. This shows a significant reduction which would be very critical for teleprotection applications. The main reason behind this is that we use fast-failover groups and do not require controller access in order to use a new path.

In the rest of this section, we analyze the switching delay results under a variety of configurations and link types. The results are shown in Table 1. Note that we also included the standard deviation and confidence interval (assuming 99% of all experiments) for the results in the same table. The results indicate that for 1s data generation frequency

Table 2
E2E delay results (in ms).

Link Type	E2E delay (avg, std-dev, confidence interval 99%)		
	1s data generation frequency	20 ms data generation frequency	5 ms data generation frequency
Wired	(0.22, 0.015, 0.007)	(0.14, 0.015, 0.004)	(0.16, 0.024, 0.011)
WiFi	(1.46, 0.06, 0.03)	(0.97, 0.082, 0.023)	(0.51, 0.07, 0.03)
LTE	(1.19, 1.01, 0.17)	(1.6, 1.1, 0.33)	(4.95, 7.72, 0.34)

Table 3
Number of packets exceeding 4 ms deadline.

Link Type	1s data generation frequency	20 ms data generation frequency	5 ms data generation frequency
Wired	none	3 over 20000 (0.01%)	38 over 47350 (0.08%)
WiFi	none	65 over 20000 (0.3%)	128 over 47350 (0.3%)
LTE	16 over 3000 (0.5%)	1862 over 20000 (9%)	14768 over 47350 (31%)

the switching delay does not have a significant impact on 4 ms deadline requirement for all link types. Considering that our 3 BFD message exchange loss is a signal of the link failure, possible link failure detection would take 3–4 ms. This time is eliminated in large data generation frequencies since it is very rare that the detection of BFD link failures would coincide with the time of the data generation. However, for 20 ms and 5 ms data generation frequencies, this is not valid.

While Wired and Wireless cases still have reasonable switching delays within the limits of 4 ms requirement, LTE has 36.5 ms delay which is far beyond the requirement. For 802.11, this time is a bit longer than the wired connection setup due to message exchanges for associations. However, for LTE, besides the connection setup, it is much longer since it involves the base-station as well as the EPC network. Furthermore, re-transmission protocols also have an impact on this delay. Finally, LTE base-station gets more data before it can handle the ones already in the queue. Such factors increase the overall switching delay significantly.

The main delay difference lies in the link establishment times since given the same Openflow switches, failure detection and flow-rule reconfiguration take same operations for any topology (*Wired*, *Proposed Wireless*, *Proposed LTE*) as shown in Equation (1). The number of link failures will also not cause any extra delay. Basically, the duration of network recovery mechanism with fast-failover openflow rules does not depend on whether the failure is on a singe link or multiple links. The total time to recover will be based on the delay for failure detection and for switching the interface. Therefore, the failure detection time for multiple failures will be the maximum of the delays caused by single failures. Whichever failure is longest to recover will cause all the nodes in the network to wait for it.

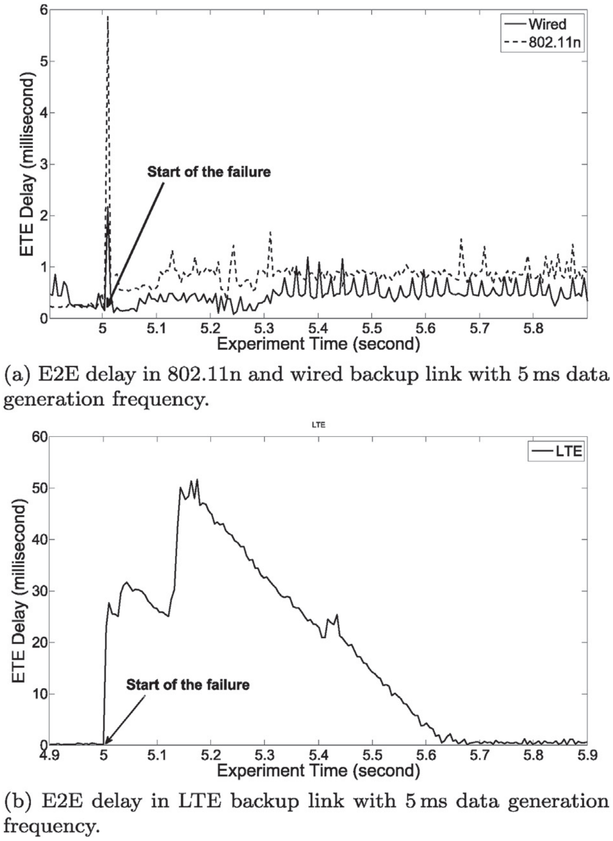
5.3.2. Packet loss

There is no packet loss for all link types and data rates which indicates that the proposed SDN-based approaches are reliable.

5.3.3. E2E delay & number of packets exceeding the delay deadline

Compared to the *wired* case, both *proposed wireless* and *proposed LTE* have higher E2E delay for both data rate cases. This is expected due to the limited bandwidth in both link types. However, it is interesting to see that E2E delay reduces with increased data generation frequency (i.e., 1s vs. 20 ms vs. 5 ms). This is related to increased utilization of the links in random access channel (WiFi). In this case, the channel is reserved continuously for the packets and thus there is less contention delay compared to 1 s case. However, this is not the case for LTE because it is not random access. It needs to schedule every packet which increases delay. LTE is slower since it will keep the packets in the queue and try to retransmit if there is packet in the queue. We observed that the proposed approaches are able to meet the delay requirements of 4 ms when GOOSE data generation frequency is 1s. This is due to the ability to switch in 1.1 ms time. Only a few packets (i.e., 0.5%) in LTE case were not able to meet the deadline due to increased E2E delay. In the case of 20 ms and 5 ms data generation frequencies, there are a few packets (i.e., 0.01%–0.3% and 0.08%–0.3% of the overall) missing the deadline for *wired* and *proposed wireless* due to increased switching delays at the SDN switches as shown in Table 2. However, for LTE, almost one-third of the packets missed the deadline for the 5 ms data generation frequency as shown in Table 3 due to increased switching and E2E delays. In particular, switching delay increased significantly, e.g., 36.5 ms for the 5 ms data generation frequency. Therefore, we decided to further look in to the reasons behind this.

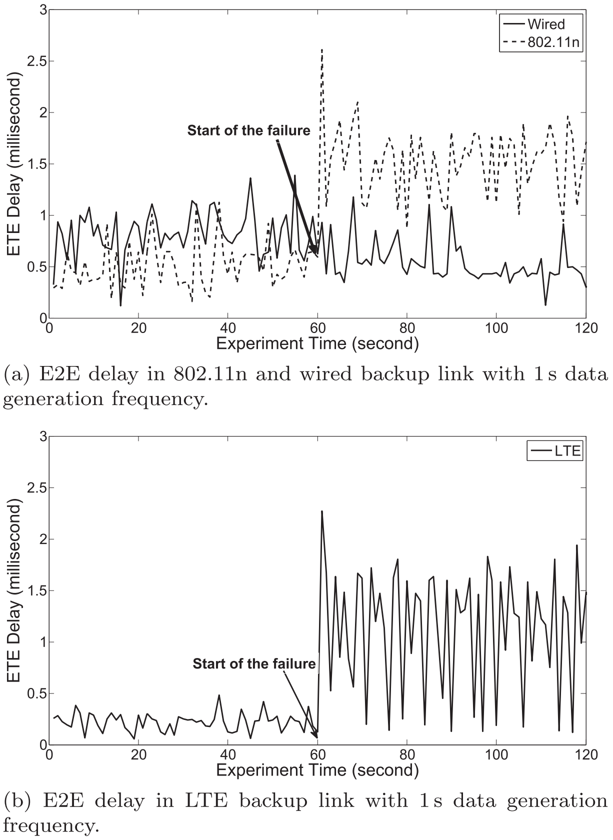
Specifically, for further investigation, we plotted E2E delays of GOOSE messages around the link failure time for the three link types under two data rates in Figs. 8 and 9. Two interesting results can be observed: First, while the link failure only has an impact on a small number of GOOSE messages around the link failure time for *wired* and *proposed wireless* (i.e., a very short transient state) as depicted in Figs. 8a and 9a in both data rates, this is not the case for the *proposed LTE* that experiences a long transient state for 5 ms data generation frequency as depicted in Fig. 8b and a very short transient state for 1s data generation frequency as depicted in Fig. 9b. Second, the GOOSE messages that experience the highest delay are not the first few messages after the backup links become active as in the *wired* and *proposed wireless* cases for 5 ms data generation frequency. This is an interesting behavior. We speculate that this behavior with 5 ms frequency can be attributed to the inherently different uplink (from UE to the base station) and downlink (from the base station to the UE) schemes, and the use of multiple layers of retransmission mechanisms in LTE. Typically, the LTE uplink traffic has a lower maximum throughput than the downlink traffic. When the link failure occurs, the activation mechanism for LTE as explained in Section 4.2 takes more time than the other link types and causes some GOOSE messages in the queue of the sender UE gateway for 5 ms data generation frequency. Eventually, some of these GOOSE messages cannot be received by the receiving UE gateway. In this case, LTE first attempts to retransmit the erroneous or lost messages at the medium access control layer. The Hybrid Automatic Repeated reQuest (HARQ) mechanism will handle this retransmission attempt and add around 8 ms of transmission delay (Zhang et al., 2012). This transmission delay can increase with several tens of milliseconds (Zhang et al., 2012) when the higher layer protocol, the radio link control (RLC) layer, is involved. This can happen, if after the maximum HARQ retry, the GOOSE message is still not correctly received by the receiving UE gateway. In this case, the Automatic repeat request scheme at the RLC layer will take over the retransmission attempts. Note that since many



(a) E2E delay in 802.11n and wired backup link with 5 ms data generation frequency.

(b) E2E delay in LTE backup link with 5 ms data generation frequency.

Fig. 8. Per message E2E delay of GOOSE messages through the simulation time for 5 ms data generation frequencies.



(a) E2E delay in 802.11n and wired backup link with 1 s data generation frequency.

(b) E2E delay in LTE backup link with 1 s data generation frequency.

Fig. 9. Per message E2E delay of GOOSE messages through the simulation time 1s data generation frequencies.

Table 4
LTE ETE Delays with Additional Background LTE Traffic, for 5 ms case.

Number of Pairs	Delay Increase %	Packet Loss %
100	2	0
140	2	5
200	15	60

more packets are waiting in the queue of the switch for 5 ms case than 1s case, the latter does not experience much of the similar problems of 5 ms case and thus transient state time is more consistent with those of *wired* and *proposed wireless*.

Impact of public LTE traffic: Note that the above experiments for LTE assumed that there is no other traffic using the network. However, in case of a natural disaster, there may be lots of devices trying to access the public LTE network at the same time since many people may want to reach their family or relative. This may cause a peak traffic that eventually may have a significant impact to the delay of the teleprotection traffic that passes through the network. Therefore, we also conducted experiments to assess the impact of public LTE usage on the performance. We considered the situation where many pair-devices (i.e., 100, and 200 pair-devices where a total of 200, and 400 UEs communicate simultaneously). The results are shown in Table 4. We noticed ETE Delay increases around 3% for 200 pair-devices and around 2% for 100 pair-devices for 5 ms data generation frequency respectively. LTE with less than 140 pair-devices, our proposal is still able to meet the delay requirements of Smart Grid Teleprotection application. However, whenever we increased the number of nodes more than 100 pairs, we observed some packet loss that would prevent meeting the GOOSE requirements. This could be happening because of congestion and scalability issues in LTE. Therefore, these results suggest that for guaranteed performance it will be required to use private LTE connections among the substations that are solely dedicated to teleprotection communication.

Impact of multi-hop wireless link: While the wireless case provides a promising solution, it suffers from limited transmission range and thus will only apply to cases where substations are close to each other. While there are new IEEE 802.11 standards such as IEEE 802.11ah with long distance coverage (e.g., more than 1mile), matching LTE coverage would only be possible with multi-hop scenarios where we can envision multiple relay nodes to enable such communication. These relay nodes can be picked from the existing AMI infrastructure which is prevalent. For instance, data collection points can provide this service. Therefore, we conducted experiments by considering multiple hops for wireless connection and measured the E2E delay. The results are shown in Table 5. As can be seen, after 2 hops, the E2E delay requirement could not be met. While E2E delay increases linearly for 1s data rate, for 5 ms data generation frequency, the situation is much worse. This case is suffering more due to the high data rate of the packet transmission which eventually increases the packets that are waiting in the queue. These delays are due to the half-duplex nature of the wireless communications and the increase in the hidden node problems along with the increase of the number of hops. Therefore, utilizing beyond 2 hops of links with IEEE 802.11n is not recommended to meet the delay requirements. Utilizing IEEE 802.11ah would cover a wider area and reduce the number of hops. Note that if there are multiple links failed simultaneously after the disaster, our solution will address them separately but in parallel. So in such cases, LTE or 2-hops solutions would apply in isolation. However, if the failures depend on each other, this creates a more complex problem which is out of scope of this paper.

Table 5
WiFi E2E delays with multiple hops.

Number of Hops	E2E Delay for 1s data generation frequency (ms)	E2E Delay for 5 ms data generation frequency (ms)
2	1.94	1.94
3	4.34	19.28
4	6.74	280.27

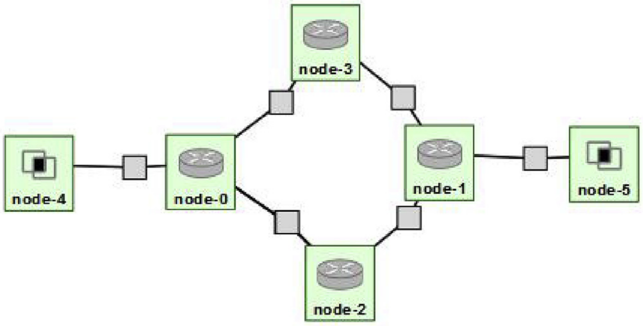


Fig. 10. Experimental network topology in GENI.

In addition to the multi-hop Wifi, we have also considered Device-to-device (D2D) LTE communications between substations in case of LTE network is down. We propose to integrate the possible D2D mechanism with relay devices as shown in Nishiyama et al. (2014) in our framework.

Experiments on GENI: We have also implemented our experiments on GENI testbed to observe switching delay performance in a realistic environment. GENI is a distributed virtual laboratory which provides access to real OpenFlow switches (Berman et al., 2014) and network infrastructure. We created a topology for our experiment in GENI as shown in Fig. 10. Basically, we created two wired links between node-0 to node-3 and disabled one of them in the middle of the experiment to observe fast-failover mechanism. We observed similar results as in the Mininet environment for the *wired* topology. Our switching delays were measured as 0.1 ms for 1 s data generation frequency, 0.3 ms for 20 ms case, and 0.5 ms for 5 ms case respectively. Note that these results are even better than our results in the Mininet environment (0.4 ms, 0.6 ms and 1 ms respectively. This can be attributed to the fact that Mininet mainly runs on a single virtual machine which may slow processing compared to an actual hardware-based switch. Nonetheless, the numbers are still consistent with each other indicating the reliability of our approach.

Furthermore, we also ran our experiments on GENI without the BFD enabled in OpenFlow switches to compare our approach with a no-protection system. We observed a single packet loss in the network with no failure detection scheme while there was no packet loss with BFD-enabled switches. This is an expected behavior since a switch recognizes the link failure after the first data packet is sent over the broken link.

6. Conclusion and future work

In this paper, we introduced how the emerging SDN paradigm could be considered as a viable technology for the Smart Grid communication architecture, which is currently under massive modernization effort by the utility providers. Specifically, we focused on GOOSE-based inter-substation communications and proposed an SDN-enabled framework when links fail. Our goal was to test the ability of SDN to recover failed links in real-time without losing any packets or significantly increasing the packet delay. We proposed to have IEEE 802.11 or LTE as a back-up link which can be activated from SDN Controller whenever the failure occurs. We developed a realistic framework which could integrate wireless channels to Mininet via ns-3. To this end, we introduced bridge nodes so that random access nature of Ethernet links can talk to LTE links that utilize a different MAC layer.

Evaluation results indicate that SDN can provide seamless resiliency in case of the availability of redundant wireless 802.11 or LTE links for real-time protocols such as GOOSE. More specifically, the results for the link failure indicate that with 1s data generation frequencies, teleprotection applications can use our *proposed wireless* approach which can meet the 4 ms deadline all the time. Furthermore, the *proposed LTE* can meet the deadline 99.5% of the time and thus it can also be a viable option if used in a private band. For 5 ms data generation frequency, we observed many packets exceeding the 4 ms E2E deadline, particularly with *proposed LTE* topology. However, GOOSE packets at a signal change are usually transmitted with lots of redundancy (Niejahr et al., 2010), and thus LTE might still be useable for such rates since the failure duration is less than a second and beyond that LTE can support ETE delays in normal operation.

As a future work, we plan to extend ns-3 capabilities to support 802.11ah and test our system with longer distance parameters. We also plan to integrate our framework into one of the remotely accessible Smart Grid Testbeds cited in Cintuglu et al. (2017) by replacing current switches with virtual SDN switches. Furthermore, we intend to implement WiFi or LTE connections on real devices such as USB dongles or USRPs.

Acknowledgment

This material is based upon work supported by the U.S. Department of Energy under Award Number DE-OE0000779. The views expressed are those of the authors only.

References

3GPP, 2011. Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN)(3GPP TS 36.300, Version 8.11. 0 Release 8). december 2009, ETSI TS 136. V8.

Akkaya, K., Uluagac, A.S., Aydeger, A., 2015. Software defined networking for wireless local networks in smart grid. In: Local Computer Networks Conference Workshops (LCN Workshops), 2015 IEEE 40th, IEEE, pp. 826–831.

Al-Rubaye, S., Kadhum, E., Ni, Q., Anpalagan, A., 2017. Industrial internet of things driven by sdn platform for smart grid resiliency. IEEE Internet Things J. 6 (1).

Apostolov, A.P., 2012. Iec 61850 communications based transmission line protection. In: 11th IET International Conference on Developments in Power Systems Protection (DPSP), pp. 1–6.

Aydeger, A., Akkaya, K., Cintuglu, M.H., Uluagac, A.S., Mohammed, O., 2016. Software defined networking for resilient communications in smart grid active distribution networks. In: 2016 IEEE International Conference on Communications (ICC), pp. 1–6.

Berman, M., Chase, J.S., Landweber, L., Nakao, A., Ott, M., Raychaudhuri, D., Ricci, R., Seskar, I., 2014. Geni: a federated testbed for innovative network experiments. Comput. Network. 61, 5–23.

Budka, K.C., Deshpande, J.G., Doumi, T.L., Madden, M., Mew, T., 2010. Communication network architecture and design principles for smart grids. Bell Labs Tech. J. 15, 205–227.

Chai, W.K., Wang, N., Katsaros, K.V., Kamel, G., Pavlou, G., Melis, S., Hoefling, M., Vieira, B., Romano, P., Sarri, S., Tesfay, T.T., Yang, B., Heimgaertner, F., Pignati, M., Paolone, M., Menth, M., Poll, E., Mampaey, M., Bontius, H.H.I., Develder, C., 2015. An information-centric communication infrastructure for real-time state estimation of active distribution networks. IEEE Trans. Smart Grid 6, 2134–2146.

Cintuglu, M.H., Mohammed, O.A., Akkaya, K., Uluagac, A.S., 2017. A survey on smart grid cyber-physical system testbeds. IEEE Commun. Surv. Tutor. 19, 446–464.

da Silva, E.G., Knob, L.A.D., Wickboldt, J.A., Gaspary, L.P., Granville, L.Z., Schaeffer-Filho, A., 2015. Capitalizing on sdn-based scada systems: an anti-eavesdropping case-study. In: Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on, IEEE, pp. 165–173.

Dorsch, N., Kurtz, F., Georg, H., Hgerling, C., Wietfeld, C., 2014. Software-defined networking for smart grid communications: applications, challenges and advantages. In: Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on, pp. 422–427.

Dorsch, N., Kurtz, F., Girke, F., Wietfeld, C., 2016. Enhanced fast failover for software-defined smart grid communication networks. In: Global Communications Conference (GLOBECOM). IEEE, pp. 1–6.

Dorsch, N., Kurtz, F., Wietfeld, C., 2018. Enabling hard service guarantees in software-defined smart grid infrastructures. Comput. Network. 147, 112–131.

Duong, V., Cueco, J., 2016. Bus protection a new and reliable approach. In: Protective Relay Engineers (CPRE), 2016 69th Annual Conference for, IEEE, pp. 1–11.

Ghosh, U., Dong, X., Tan, R., Kalbarczyk, Z., Yau, D.K., Iyer, R.K., 2016. A simulation study on smart grid resilience under software-defined networking controller failures. In: Proceedings of the 2nd ACM International Workshop on Cyber-Physical System Security. ACM, pp. 52–58.

Goldman, A.D., Uluagac, A.S., Beyah, R., Copeland, J.A., 2012. Plugging the leaks without unplugging your network in the midst of disaster. In: Local Computer Networks (LCN), 2012 IEEE 37th Conference on, IEEE, pp. 248–251.

Gyllstrom, D., Braga, N., Kurose, J., 2014. Recovery from link failures in a smart grid communication network using openflow. In: Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on, IEEE, pp. 254–259.

Heimgaertner, F., Hoefling, M., Vieira, B., Poll, E., Menth, M., 2015. A security architecture for the publish/subscribe c-dax middleware. In: Communication Workshop (ICCW), 2015 IEEE International Conference on, IEEE, pp. 2616–2621.

Hu, F., Hao, Q., Bao, K., 2014. A survey on software-defined network and openflow: from concept to implementation. IEEE Commun. Surv. Tutor. 16, 2181–2206.

IEC61850-90-1, 2010. Communication Networks and Systems for Power Utility Automation Part 90-1: Use of Iec 61850 for the Communication between Substations.

IEC61850-90-2, 2016. Communication Networks and Systems for Power Utility Automation Part 90-2: Using Iec 61850 for Communication between Substations and Control Centres.

Ieee Standard for Information TechnologyTelecommunications and Information Exchange between Systems - Local and Metropolitan Area Networksspecific Requirements - Part 11: Wireless Lan Medium Access Control (Mac) and Physical Layer (Phy) Specifications Amendment 2: Sub 1 Ghz License Exempt Operation, IEEE Std 802.11ah-2016 (Amendment to IEEE Std 802.11-2016, as Amended by IEEE Std 802.11ai-2016), 2017. , pp. 1–594.

Ieee Standard for Information TechnologyTelecommunications and Information Exchange between SystemsLocal and Metropolitan Area Networksspecific Requirements Part 11: Wireless Lan Medium Access Control (Mac) and Physical Layer (Phy) Specifications Amendment 10: Mesh Networking, IEEE Std 802.11s-2011 (Amendment to IEEE Std 802.11-2007 as Amended by IEEE 802.11k-2008, IEEE 802.11r-2008, IEEE 802.11y-2008, IEEE 802.11w-2009, IEEE 802.11n-2009, IEEE 802.11p-2010, IEEE 802.11z-2010, IEEE 802.11v-2011, and IEEE 802.11u-2011, 2011. , pp. 1–372.

Johnston, R.A., Hauser, C.H., Gjermundrod, K.H., Bakken, D.E., 2006. Distributing time-synchronous phasor measurement data using the gridstat communication infrastructure. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS06), vol. 10, p. 245b.

Kim, Y.J., Lee, J., Atkinson, G., Kim, H., Thottan, M., 2012. Sedax: a scalable, resilient, and secure platform for smart grid communications. IEEE J. Sel. Area. Commun. 30, 1119–1136.

Kriger, C., Behardien, S., Retonda-Modiya, J.-C., 2013. A detailed analysis of the goose message structure in an iec 61850 standard-based substation automation system. Int. J. Comput. Commun. Control 8, 708–721.

Kurtz, F., Dorsch, N., Bektas, C., Wietfeld, C., 2017. Synchronized measurement concept for failure handling in software-defined smart grid communications. In: Smart Grid Communications (SmartGridComm), 2017 IEEE International Conference on, IEEE, pp. 1–6.

Lantz, B., Heller, B., McKeown, N., 2010. A network in a laptop: rapid prototyping for software-defined networks. In: Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks. ACM, p. 19.

Lee, C., Shin, S., 2018. Fault tolerance for software-defined networking in smart grid. In: Big Data and Smart Computing (BigComp), 2018 IEEE International Conference on, IEEE, pp. 705–708.

Mackiewicz, R.E., 2006. Overview of iec 61850 and benefits. In: Power Systems Conference and Exposition, PSCE06. (IEEE PES 2006), IEEE, pp. 623–630.

Naik, P.K., Nair, N.K., Vyatkin, V., 2011. Sympathetic trip protection scenario in iec 61850. In: Australian Power Electronics Conference (AUPEC2011), Brisbane, Citeseer.

Network Simulator - ns - 3, 2017. .

Niejahr, J., Schuster, N., Spangler, M., 2010. Substation to substation (ss2ss) goose exchange for critical relay operations. In: CIGRE Canada, Conference on Power Systems.

Nishiyama, H., Ito, M., Kato, N., 2014. Relay-by-smartphone: realizing multihop device-to-device communications. IEEE Commun. Mag. 52, 56–65.

Niven-Jenkins, B., Brungard, D., Betts, M., Sprecher, N., Ueno, S., 2009. Requirements of an MPLS Transport Profile. Technical Report. .

Openflow Switch Specification, 2011. .

Ozgur, U., Tonyali, S., Akkaya, K., 2016. Testbed and simulation-based evaluation of privacy-preserving algorithms for smart grid ami networks. In: Local Computer Networks Workshops (LCN Workshops), 2016 IEEE 41st Conference on, IEEE, pp. 181–186.

Project, F., 2014. Floodlight Controller.

Rehmani, M.H., Davy, A., Jennings, B., Assi, C., 2018a. Software Defined Networks Based Smart Grid Communication: A Comprehensive Survey. arXiv:1801.04613.

Rehmani, M.H., Akhtar, F., Davy, A., Jennings, B., 2018b. Achieving resilience in sdn-based smart grid: a multi-armed bandit approach. In: 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft), IEEE, pp. 366–371.

Ren, L., Qin, Y., Wang, B., Zhang, P., Luh, P.B., Jin, R., 2017. Enabling resilient microgrid through programmable network. IEEE Trans. Smart Grid 8, 2826–2836.

Saputro, N., Akkaya, K., Uludag, S., 2012. A survey of routing protocols for smart grid communications. Comput. Network. 56, 2742–2771.

Saputro, N., Akkaya, K., Tonyali, S., 2016. Addressing network interoperability in hybrid ieee 802.11 s/lte smart grid communications. In: Local Computer Networks (LCN), 2016 IEEE 41st Conference on, IEEE, pp. 623–626.

Saputro, N., Akkaya, K., Adgin, R., Uluagac, S., 2019. Drone-assisted multi-purpose roadside units for intelligent transportation systems. In: 2018 IEEE 88th Vehicular Technology Conference: VTC2018-Fall, Chicago, USA.

Zhang, L., Okamawari, T., Fujii, T., 2012. Performance evaluation of end-to-end communication quality of lte. In: 75th Vehicular Technology Conference (VTC Spring). IEEE, pp. 1–5.

Zhang, X., Wei, K., Guo, L., Hou, W., Wu, J., 2016. Sdn-based resilience solutions for smart grids. In: Software Networking (ICSN), 2016 International Conference on. IEEE, pp. 1–5.

Abdullah Aydeger is a PhD student in Electrical and Computer Engineering in Florida International University, Miami, USA. He received his M.S. degree from Department of Computer Engineering at Florida International University in 2016 and his B.S. degree in Computer Engineering from Istanbul Technical University in 2013. Prior to his master degree, he did his internship at A∗STAR research institute in Singapore. His research interests include SDN, network security, and resiliency.

Nico Saputro received his PhD degree in Electrical Engineering from Florida International University, Miami, USA in 2016. He is a Postdoctoral Associate at the Department of Electrical and Computer Engineering at Florida International University Miami, FL. He is also a senior lecturer at the Mechatronics Engineering Bachelor Degree Program, at the Department of Electrical Engineering at Parahyangan Catholic University, Bandung, Indonesia. His research interests include security and privacy in a variety of wireless and cellular networks, communication protocols for Smart Grid, moving target defense, SDN, and Internet of Things. Dr. Saputro is a Fulbright Presidential Scholarship recipient (2009–2012) and a member of IEEE.

Dr. Kemal Akkaya is a full professor in the Department of Electrical and Computer Engineering at Florida International University. He leads the Advanced Wireless and Security Lab (ADWISE). His current research interests include security and privacy, internet-of-things, and cyber-physical systems. He is the area editor of Elsevier Ad Hoc Network Journal and serves on the editorial board of IEEE Communication Surveys and Tutorials. He has published over 120 papers in peer reviewed journal and conferences. He has received “Top Cited” article award from Elsevier in 2010. Dr. Akkaya is a senior member of IEEE.

A. Selcuk Uluagac is currently an Assistant Professor in the Department of Electrical and Computer Engineering at Florida International University, where he leads the Cyber-Physical Systems Security Lab (CSL). He has served as a member of the research faculty as a Senior Research Engineer in the School of Electrical and Computer Engineering at The Georgia Institute of Technology and prior to Georgia Tech, he was a Senior Research Engineer at Symantec. He is an IEEE senior member.