Software defined neighborhood area network for smart grid applications

Nafi, Nazmus S, Ahmed, Khandakar, Gregory, Mark and Datta, Manoj (2018) Software defined neighborhood area network for smart grid applications. Future Generation Computer Systems, 79 (Part 2). 500 - 513. ISSN 0167-739X

Accepted Manuscript

Software defined neighborhood area network for smart grid applications

Nazmus S. Nafi, Khandakar Ahmed, Mark A. Gregory, Manoj Datta

PII: S0167-739X(17)31100-7
DOI: https://doi.org/10.1016/j.future.2017.09.064
Reference: FUTURE 3720

To appear in: *Future Generation Computer Systems*

Received date : 27 May 2017
Revised date : 2 September 2017
Accepted date : 24 September 2017

Please cite this article as: N.S. Nafi, K. Ahmed, M.A. Gregory, M. Datta, Software defined neighborhood area network for smart grid applications, *Future Generation Computer Systems* (2017), https://doi.org/10.1016/j.future.2017.09.064

Software Defined Neighborhood Area Network for Smart Grid Applications

Nazmus S. Nafi, *Student Member*, *IEEE*, Khandakar Ahmed, *Member*, *IEEE*, Mark A. Gregory, *Senior Member*, *IEEE*, Manoj Datta, *Member*, *IEEE*

***Abstract*—Information gathered from the Smart Grid (SG) devices located in end user premises provides a valuable resource that can be used to modify the behavior of SG applications. Decentralized and distributed deployment of neighborhood area network (NAN) devices makes it a challenge to manage SG efficiently. The NAN communication network architecture should be designed to aggregate and disseminate information among different SG domains. In this paper, we present a communication framework for NAN based on wireless sensor networks using the software defined networking paradigm. The data plane devices, such as smart meters, intelligent electronic devices, sensors, and switches are controlled via an optimized controller hierarchy deployed using a separate control plane. An analytical model is developed to determine the number of switches and controllers required for the NAN and the results of several test scenarios are presented. A Castalia based simulation model was used to analyze the performance of modified NAN performance.**

***Index Terms*— Advanced metering infrastructure, neighborhood area network, smart grid, software defined networking, wireless sensor network**

1. Introduction

The conventional power grids are on the verge of being upgraded to become Smart Grids (SG). Any changes in power grids are challenging due to its diversified and enormous network size, functionality, and standard specifications. The power grids need to accommodate state of the art SG applications and implement the applications for the benefit of consumers, business, industry, utility service providers and other stakeholders. Challenges remain before the development and implementation of SG applications will proceed apace, including selecting the appropriate communication technologies, network architecture, security, and regulation. SGs will be capable of transmitting electrical power both in forward and reverse direction in the distribution domain [1]. Large scale projects to generate power from renewable sources [2][3] are increasing the motivation to upgrade the power grid into a SG. Adding more renewable energy sources with synchronization with the existing power generators may reduce the substantial domestic and commercial demand load and decrease costs over time. Smart distribution grid applications [4] such as advanced meter infrastructure (AMI) [5], demand response (DR) [6], distributed energy resources (DER) [7] and vehicle to grid (V2G) [8] have the potential to contribute to the day-to-day fluctuating power demands. With modeling of the SG applications it is becoming evident there is a need for new technologies to be used to facilitate increased communications capability.

The SG transformation process will be focused on the design of the communications network. Real-time monitoring and control of the large-scale intelligent device implementation requires improved traffic engineering and data management, with latency becoming a critical factor. Also, to improve security, efficiency, and the reliability of the power network, a robust communication network that enables autonomous system operations is a necessity. Developing a communications network and systems for SGs could be facilitated by the emergence of the software defined networking (SDN) paradigm.

Networking devices in the current power grids are generally designed to serve an individual or selected number of applications. The primary objective of the networking devices is to enable machine to machine (M2M) communication, using an approach known as hardware-centric networking. Hardware-centric networking faces scalability and controllability challenges because the devices are generally static and cannot be dynamically updated with changed or new features. In an emergency scenario for example, it is difficult to re-configure network settings in a timely and effective manner. Over time, the non-adaptive network configuration of large M2M networks is prone to security degredation. The lack of a real-time grid monitoring capacity contributes to poor quality of service (QoS). The limited network control capability has driven the move towards SG communication networks utilizing SDN because it offers improved network control and programmability. SG communication networks that are based on SDN provide a more efficiently, flexible and dynamic environment for M2M communications [9]. The SDN based communications network separates the control and data planes to improve the control mechanism whilst

Nazmus S. Nafi is a PhD candidate with the School of Engineering, RMIT University, Melbourne, Australia (e-mail: nazmusshaker.nafi@rmit.edu.au).
Khandakar Ahmed is with the College of Engineering and Science, Victoria University, Melbourne, Australia (e-mail: khandakar.e.ahmed@ieee.org).
Mark A. Gregory is with the School of Engineering, RMIT University, Melbourne, Australia (e-mail: mark.gregory@rmit.edu.au).
Manoj Datta is with the School of Engineering, RMIT University, Melbourne, Australia (e-mail: manoj.datta@rmit.edu.au).

reducing the data plane dependence on vendor specific networking devices in the SG [10][11]. Smart meters and sensors nodes installed in the neighborhood area networks (NANs) would be connected to SDN enabled switches that provide the data layer traffic forwarding. Implementing SDN in a wireless environment consisting of a large number of wireless sensor nodes or smart meters is a significant challenge. For example, identifying the communications framework, the network architecture, and the number of switches and controllers required in the wireless NAN is an important step. This paper focuses on managing a network of wirless sensor devices utilizing an SDN enabled NAN. According to [12] network control is fully programmable in SDN and it is isolated from the packet forwarding mechanism. An SDN controller usually has the functionality to be the centralized network controller or part of a hierarchy of distributed controllers and can deploy multiple packet forwarding schemes or flows via the managed SDN enabled switches [13][14][15]. SDN has the potential to support information-centric networks (ICNs) where the network complexity could be reduced and network manageability increased based on identifying information flows and aggregation points [16]. Thus, SDN becomes an appropriate candidate for SG communication networks, where a huge number of M2M devices are controlled, monitored, and smart data aggregation is mandatory to run the delay sensitive applications. Moreover, inter-domain communication within the SG would require a sophisticated packet classifier and network address translators or firewalls which could be managed and configured using an SDN switch [17]. SDN could exploit the features of ICNs [18] such as content query, content-id based routing or in-network content retrieval to form groups of similar traffic sets and disseminate data through the cross-domain devices of the SG [19][20]. Another advantage of using SDN in SGs would be with virtual networking. SDN overcomes the limitations of a conventional virtual local area network (VLAN) or a virtual private network (VPN) due to its improved feture set, capabilities and ability to be reconfigured quickly. The concept of a virtual power plant (VPP) [21] could be one outcome of the virtualization capabilities provided by SDN.

According to Gungor et. al. [22] WSNs could be a potential technology that could be used in any subdomain of the power grid. Other wireless technologies such as WiMax and Wi-Fi could be incorporated to build a robust heterogeneous network (HetNet). To keep the scope within the bounds of the design of an SDN based WSN for SG NAN, no further discussion on SDN based SG HetNet has been included in this paper, it is left for future work. The most common applications within a NAN are smart metering, DR, and distributed automation [23]. To deploy these applications, a large number of end user devices or smart meters and several data concentrators or aggregators are required. In a SG NAN, the data rate varies between 100 Kbps to 10 Mbps and data is expected to travel to network boundary points over distances of up to 10 Km (max) [23]. Among the available wireless sensor technologies ZigBee mesh networks are widely accepted. The IEEE standards association has released standards through the IEEE802.15.4g task group (TG4g) [24] that provide specifications to support a large number of smart devices deployed in diverse geographical topologies using minimal infrastructure. The system design for our proposed framework considers the specifications released by TG4g. Figure 1 shows a conceptual network architecture of a heterogeneous communication network for SGs based on the SDN paradigm. As shown in the Figure 1, the different SG domains such as Generation, Distribution or customer premises could be modeled with SDN cross domain content-based networking properties. Distributed controllers manage the data plane devices in the different domains of the SG. In the NAN, data related to the power grid will be retrieved using sensor devices, smart meters and intelligent electronic devices. A well designed SDN would utilize the retrieved data to deploy useful SG applications such as Automatic Meter Reading (AMR), outage management and distribution automation. In the distribution grid, SG applications like vehicle to grid (V2G), Grid to vehicle (G2V) or substation automation could be deployed via SDN based SG wide area networks.

Figure 2 shows the conceptual architecture of the SDN based NAN. The sensor nodes are deployed in the customer premises to create home area networks (HAN), building area networks (BAN) or industrial area networks (IAN) and when combined a single NAN is created. Within a NAN, the sensor nodes and smart meters are connected to the SDN enabled switches (SDSWs) as shown in Figure 2. The switches are managed by a hierarchy of controllers, with the NAN controller/gateway forming the control plane entry point to the NAN. The switches are associated with SDN controllers that manage traffic flows by providing traffic flow packet forwarding instructions. Packet forwarding instructions received from the controller are stored in a flow table within a switch.

The research presented was motivated by the observation that there is a need for SDN based SG NANs to facilitate devices connected by wireless or wired communication technologies. Particularly, the motivation of this work is to enable SDN based M2M communication in the NAN using WSN. This paper highlights the controller distribution within the NAN domain to optimize efficiency and reduce costs. The research presented in this paper includes:

- A novel SDN based NAN communication framework for WSN connected devices
- Consideration of delay sensitive and delay tolerant SG applications
- Analysis of the network performance
- A novel algorithm that improves traffic flows over the WSN
- An analysis of the number of controller and switches required in the NAN to support the WSN connected devices

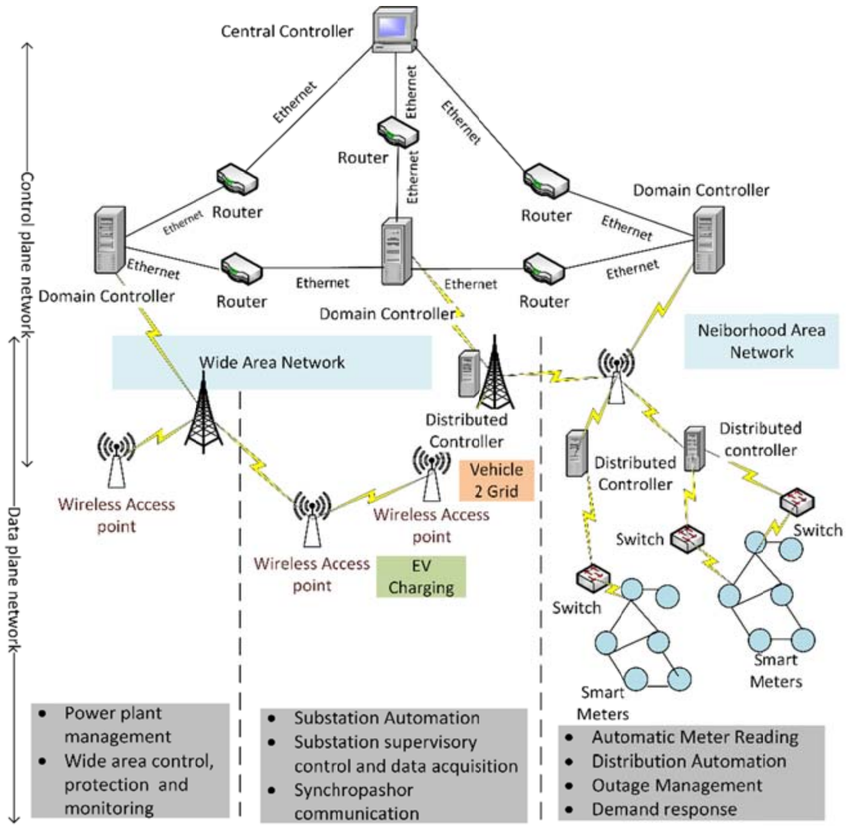


Figure 1: SDN based network architecture of a heterogeneous communication network for SG

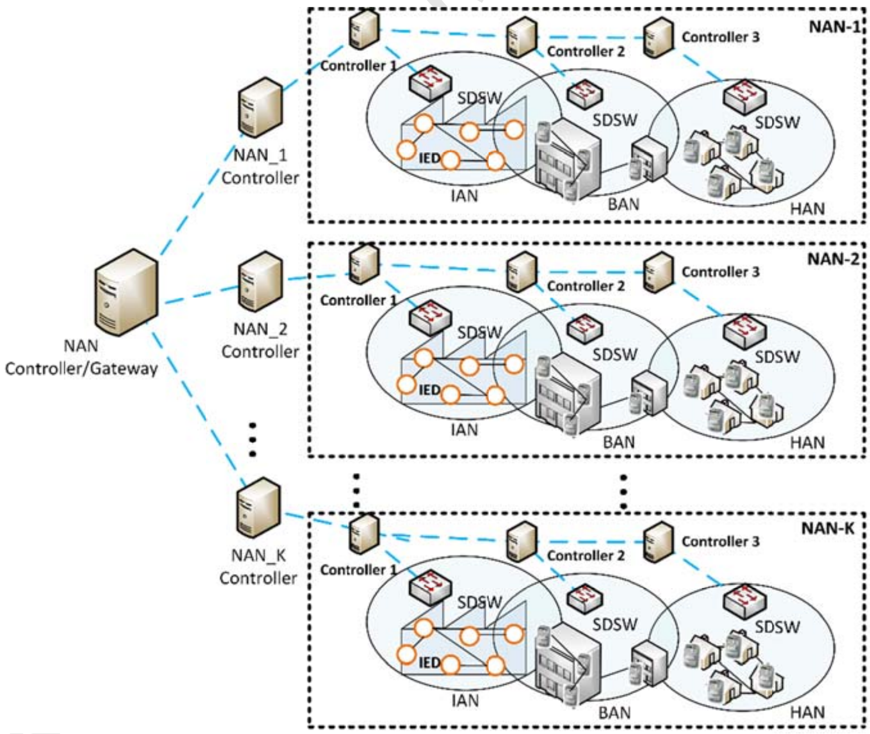


Figure 2: SDN based NAN architecture

The rest of the paper is organized as follows. Section 2 provides details on the SDN based SG NAN communication framework and describes the distribution grid architecture. Section 3 presents the analytical model used to determine the number of controllers and switches for a NAN. Section 4 exhibits the simulation model and analyzes the results. The conclusion and future research directions are presented in Section 5.

2. SDN BASED SG NAN COMMUNICATION FRAMEWORK

2.1 Overview

SG NAN and Field Area Network (FAN) applications that are AMI or smart meter based are typically semi-periodic. For example, AMR, DM, electric vehicle (EV) charging and discharging and micro-grid management. DR is responsible for load balancing between the power generator and consumers by utilising load control programs. The load control could be based on dynamic energy unit pricing or based on a more advanced technique like remote load control (RLC). Examples of well know dynamic price programs are time-of-use (TOU), real-time pricing (RTP), critical peak pricing (CPP) and peak time rebates (PTR). The characteristics of an RLC program can vary based on properties such as interruptible loads, reducible loads, partially interruptible loads. More details on the DR programs could be found in [25]. Distribution automation (DA) [26] should be designed for near real-time information about the grid operation and is used to improve distribution network monitoring and control by utilizing power distribution control, monitoring and management devices such as voltage regulators, fault detectors, capacitor bank controllers, and reclosers. A DA system can enable useful applications like Volt/VAR control, fault detection, isolation and restoration and distribution system monitoring and maintenance. Volt/VAR control adjusts the voltage and balances the load factor to reduce energy loss. In distribution monitoring and maintenance, sensor data is used to monitor the status (open/closed) of various distribution network equipment. With fault detection isolation and restoration, a section of the grid could be isolated if there is an occurrence of a fault and automatic restoration systems can be used to minimize the service interruption.

Meter reading applications send scheduled meter data to the meter data management system (MDMS) via smart meters installed in the customer premises. Both MDMS and clients could use the data for billing inquiries, and verify outage and restoration events. Also, meter data could be used to monitor the health of the smart meter (e.g., connection status, hardware configurations), on grid events (e.g., software upgrades), or to generate event alarms (voltage distortions, outage, scheduled maintenance). In the case of an outage event, an outage management application needs to send the 'last gasp' alarm from the affected smart meters to the control center. As there might be no power available except the charge stored in the capacitor of the affected smart meter, outage alarms are required to be transmitted within few 10s of milliseconds [27]. It is crucial to meet the reliability requirements when a large number of smart meters are trying to access the network simultaneously. Based on [28], in rural, suburban and urban areas the meter density could be 100, 800 and 2000 per Km^2 respectively. Table I summarizes the properties of four major SG applications in the NAN domain and shows their communication requirements. Based on the delay variability of these applications, we have categorized existing SG AMI applications in two groups - delay sensitive and delay tolerant applications. Application characteristics and requirements [29][30] are also summarized.

2.1 Communication Model

The OpenFlow protocol is the functional open standard protocol for controller to switch communications that is acknowledged by the SDN research community [31]. OpenFlow provides the linkage between the programmable control plane and switches used to connect the data plane together. OpenFlow can be used to design test facilities for centralized and distributed control mechanisms in large networks. Usually, the communication devices in an SDN serve different functions depending on whether the devices are part of the control plane or the data plane. The control plane is used to facilitate network services and applications, flow management, monitoring and maintenance. The network devices in the data plane provide flow transport between the end point devices and up stream systems that today might be found in the Cloud. The SDN paradigm is appropriate for SGs as it facilitates connecting end point devices using WSN in a NAN, provides the flexibility to adjust to an increasing number of M2M devices being added to the NAN and can be quickly upgraded to support new or enhanced SG applications. The proposed communication framework presented in this paper can be used to implement a SG NAN. The communications model permits the flexible deployment of innovative SG applications. The authors in [32] focused on developing an SDN based routing protocol for AMI applications in a wireless environment. This work provides an interesting early insight into how an SDN based SG NAN using WSNs might route traffic.

Table I Major SG application within NAN domain

AMI applications	*Characteristics*	*Data Sending Interval*	*Data Rate*	*Data Size (bytes)*	*Delay*	*Reliability*
Outage management	Event based, Delay Tolerant	1 per meter per power lost	56kbps	25	2s	>98%
Demand response (DR)	Semi periodic (Delay tolerant)/Event based (Mission Critical)	1 per device per Broadcast request	14-100kbps per node	100	500ms-1min	>99.5%
Distribution automation	Semi-Periodic, Delay Sensitive	1 per device per 12 h	9.6-56kbps	150-200	25 _ 100 ms	>99.5%

Meter reads	Periodic, Delay Tolerant	5min, 10min, 15min, 30min, 1Hr	10Kbps to 128Kbps	200	2-15s	>98%

To develop a framework for the SDN-based SG NAN using WSN, the first task is to determine the control and data plane characteristics. At the time of network initialization, flows are defined via the controller and stored in the NAN switches. The flow commands are stored in all switches whilst the corresponding applications are active in the network, and might be replaced or removed if there was no demand for the flow table entries. As shown in Figure 3 (a), there are two delay tolerant SG application packets generated from IEDs and smart meters. Upon arriving at the switch, packets are placed into a flow lookup queue. If the flow is matched with a flow entry stored in the flow table, the corresponding switch forwards the packet to the desired destination by placing the packet into a queue called the immediate buffer (shown in Figure 3(b)). If the flow is not matched, information from the packet is sent to the nearest controller with a request for a flow instruction to be provided for the packet type. After retrieving a flow command for the packet, the packet is placed into a buffer responsible for handling delayed packets, which are then forwarded at the earliest opportunity. Note that, flow commands could be modeled in such a way that switches can unicast, multicast or broadcast the packet based on the SG application specifications.

Data from the smart meters is sent to the switch via unicast packet transmission. The packet format of the unicast packets is shown in Figure 3 (c). Upon receiving this packet, the switch uses an application classification module to determine the application name and identification number (shown in Figure 3 (d)). Next, the switch looks for a flow match with an entry in the flow table. If the flow is found, the related action corresponding to this flow instruction is performed. The packet is treated as resolved and sent to the network layer. In the case of a table miss event, at first, the switch inserts the packet into the queue of an unresolved packet buffer. Next, a control packet is generated which contains the preamble fields of the SG application packet. This packet is called '*Packet_In Request*' and sent to the controller containing only the packet header fields (shown in Figure 3 (e)). Only the header fields of the packet are sent to avoid overloading the control channel. Upon receiving the control packet, '*Packet_In Request*' (Figure 3 (e)), the controller replies to the corresponding switch with a new flow command by attaching a field in the acknowledgment packet. This packet is called '*Packet_out Response'* and contains the flow command being sent to the switch. The SG application packet with the new flow command is shown in Figure 3 (f). At this stage, the switch updates its flow table and forwards the packet. Algorithm 1 and Algorithm 2 summarized the process.

Algorithm 1: Application classifier module at the application layer of the cluster switch

input: ***Read*** $SDSG_{gen\ pk}$
if $pk_{val} < 0$ then
  $pk_{val} \leftarrow$ ***A1, A2, … , An***
  $new\ pk_{app\ id} \rightarrow SDSG_{gen\ pk}$
  $new\ pk_{app\ name} \rightarrow SDSG_{gen\ pk}$
  $set\ new\ pk_{app\ pk} = SDSG_{gen\ pk}$
  $SDSG_{app\ pk} \rightarrow$ ***pk forward buffer***
else ***pk destination → Controller***
end if
initialize ***flow table query***
move $SDSG_{app\ pk}$ ***from queue***
read app_{id} **and** app_{name}
if app_{id}**and** app_{name} = ***true*** then
  Get Action
  if ***int i*** then
    for **i**=***0, i < action, i++*** do
      set pk destination
    end for
  end if
  if ***int j*** then
    for ***j=0, j < action, j++*** do
    set pk destination **'-1'**
    end for
  else $SDSG_{control\ pk}$ = ***Preamble*** + $SDSG_{app\ pk}$
  end if

25. else ***Drop pk***
end if

In the control plane of the SDN based NAN, SG applications could be configured and modified. The primary task of the controller is to provide flow control message to the switches. Adding and updating network services and applications can be carried out using the controller hierarchy. SG applications can be added and updated via the controllers which can be used to push updates out to switch application-classifier modules. The proposed framework utilises WSNs, so an association process of the NAN controllers to local switches is handled based on Received Signal Strength Indicator (RSSI) and Round-Trip Time (RTT). The details of the controller association process can be found in [33].

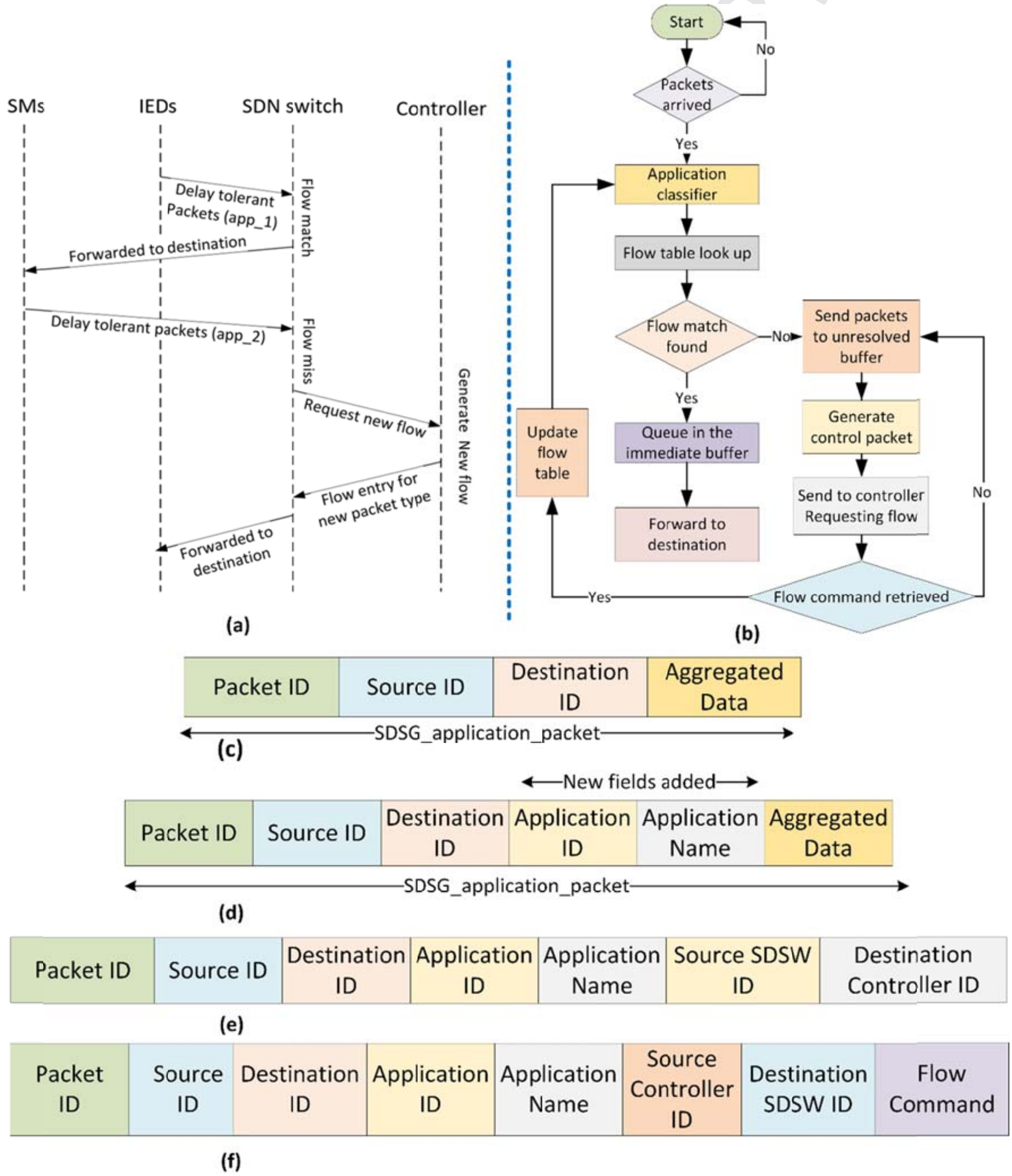


Figure 3: (a) Message exchange between network devices, (b) application layer packet flow protocol, (c) application layer packet format, (d) packet format after application classifiation (e) packet format with flow command request to the controller (f) packet format with flow command form the controller

2.1 SDN based Distribution Grid Architecture

In order to enable distributed control over a large area, a sector based grid topology has been considered. The grid topology selection for cluster communication was influenced by the configuration of the power distribution system. According to [34] distribution systems are mostly radial where a certain number of service transformers are connected to a single feeder. A sample of distribution network architecture is shown in Figure 4 (a). A service transformer of 50 MVA can usually provide ten service drop connections [34]. Based on this estimation, if a radial system consists 108 service transformers, it can provide connections to 1080 smart meters. The radial system shown in Figure 4 (a) can be represented as an 8x2 grid where each service transformer has ten service drop connections attached to ten smart meters. Now, if each service transformer is equipped with one SDN switch and responsible for an individual sector, it can be concluded that there are 108 sectors. The distribution feeder is connected with service transformers and distributed in a radial manner. Each service transformer then delivers power to the home premises. Figure 4 (b) shows a section of the customer premises where each home is connected to the service transformer via an individual service drop. Smart meters and IEDs are installed at each home and aggregate data for further processing. The meter data is sent to the corresponding switches and forwarded to the appropriate authority to enable various SG applications.

Algorithm 2: Handling packets with forwarding instruction provided to the controller

Read $SDSG_{control\ pk\ reply}$ *pk fields*
Get app_{id} app_{name} *Action*
Set New flow=Action
Set app_{id} **and** app_{name} → *New flow*
for ***i=1, i<unsolved_queue*** do
  $temp_{pk}$=***unsolved_queue***[***i***]
  if app_{id}= $temp_{pk\_app\_id}$ **and** app_{name}= $temp_{pk\_app\_name}$ then
    Move $temp_{pk}$ → *forwarding buffer*
  end if
end for

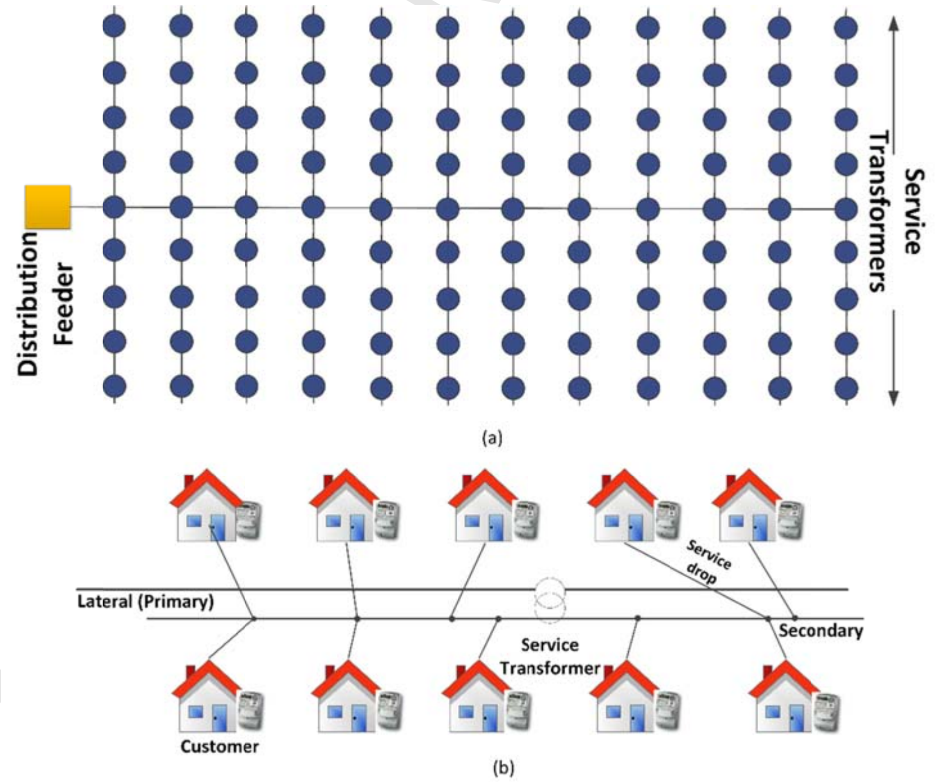


Figure 4: (a) Radial configuration of a single distribution feeder with connected services transformers [34] (b) Customers premises services drop connections from the service transformer.

3 Optimization: Distributed Controllers and Switches

As mentioned earlier, in the proposed framework, a group of IEDs and smart meters will form clusters and a single switch will be responsible for each cluster. In this context, it will be a basic requirement to determine the appropriate number of switches and controllers for large NANs with WSNs. It's important to identify the number of switches that will be deployed in the distribution grid to maximize the network throughput and efficiency (e.g. packet success rate). Also, to establish robust communication infrastructure and achieve full control via SDN controllers, intelligent implementation of distributed controllers in the control plane is essential. Furthermore, a novel mathematical model is proposed to determine the minimum number of controllers and switches that can serve the proposed framework with maximum efficiency.

Let's assume, α is the total processing time of a single controller to resolve an incoming flow request and to disseminate the traffic to the appropriate destinations. The total processing time depends on three parameters: (a) flow request delay, (b) associated communication delay, and (c) flow request-response delay. Flow request delay is the period consumed by the switch whilst identifying that flow instructions are not in the flow table for a new flow and sending the flow information to the controller. The communication delay is a summation of the switch store and forwarding delay and the propagation delay. Lastly, the flow request response time is consumed as the controller processes the flow information for the incoming instruction request packets. This can be written as:

$$\propto = \propto_{FR} + \propto_{FRR} + \propto_{CD1} + \propto_{CD2} \tag{1}$$

Where, $\propto_{FR}$, $\propto_{FRR}$, $\propto_{CD1}$ and $\propto_{CD2}$ represents flow request delay, flow request response time, communication delay from switch to controller, and controller to switch, respectively. Considering,

$$\propto_{CD1} \cong \propto_{CD2} \cong \propto_{CD} \tag{2}$$

Thus,

$$\propto = \propto_{FR} + \propto_{FRR} + 2 \propto_{CD} \tag{3}$$

Now, assume α_{PD} is the propagation delay and α_{SF} is the store and forwarding delay of the switches. The store and forwarding delay is the time period taken by the switch to place the packet into the queue of immediate buffer and to transmit the packet at the earliest opportunity. Thus, the total communication delay can be represented as

$$\propto_{CD} = \alpha_{PD} + \alpha_{SF} \tag{4}$$

If there are *h* number of hops between a switch and a controller then,

$$\Delta_{PD} = \sum_{i=1}^{h} \alpha_{PD_i} \tag{5}$$

$$\Delta_{SF} = \sum_{i=1}^{h} \alpha_{SF_i} \tag{6}$$

Therefore, (4) can be re-written as

$$\propto_{CD} = \sum_{i=1}^{h} (\alpha_{PD_i} + \alpha_{SF_i}) \tag{7}$$

For, simplification consider $\alpha_{PD_i} \cong \alpha_{PD_{i+1}} \cong \cdots \cong \alpha_{PD_{i+h}} \cong \alpha_{PD}$ and $\alpha_{SF_i} \cong \alpha_{SF_{i+1}} \cong \cdots \cong \alpha_{SF_{i+h}} \cong \alpha_{SF}$. Therefore, (7) can be expressed as

$$\propto_{CD} = (\alpha_{PD} + \alpha_{SF})*\text{h} \tag{8}$$

To determine the value of $\propto_{FRR}$ in (3), consider that ξ is the flow resolve rate per controller and η is the number of switch assigned per controller. Then, the $\propto_{FRR}$ *can be written as*

$$\propto_{FRR} = \frac{\eta\beta}{\xi} \tag{9}$$

Now, by replacing (8) and (9) in (3)

$$\propto = \propto_{FR} + \frac{\eta\beta}{\xi} + 2(\alpha_{PD} + \alpha_{SF})*\text{h} \tag{10}$$

Hence,

$$\eta = \frac{[\alpha - \propto_{FR} - 2(\alpha_{PD} + \alpha_{SF})h]\xi}{\beta} \tag{11}$$

for inter-sector communication, sector based distance (SBD) routing protocol (cf. Section II in [35]) has been considered. According to SBD, switches continue forwarding their flow request to their immediate neighbor switch, which lies on the same row in the virtual grid until the packet reaches the switch that is in the same column as the controller. The flow request will then be forwarded vertically up or down until it reaches the controller. Figure 5 shows the routing of packets from switch to the controller with SBD routing protocol.

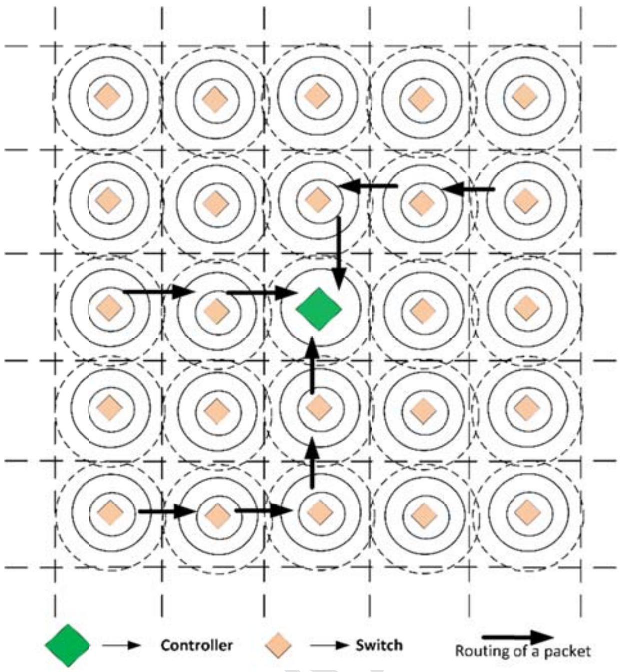


Figure 5: Routing of packets from switch to controller based on SBD

Lemma 3.1:

Let, there are η *number of switches uniformly distributed in a* $(\sqrt{\eta}\, X \sqrt{\eta})$ *grid with one switch each cell as shown in*

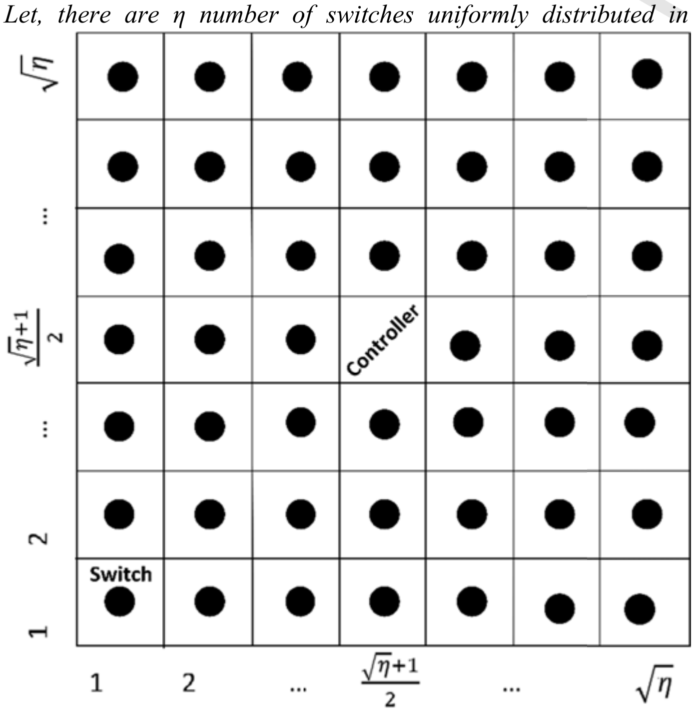


Figure 6 A controller is deployed in the middle of the grid and all η *switches are assigned to this controller. Then, the average number of hops 'h' from a switch to controller is* $\sqrt{\eta}/2$ *(approximately)*

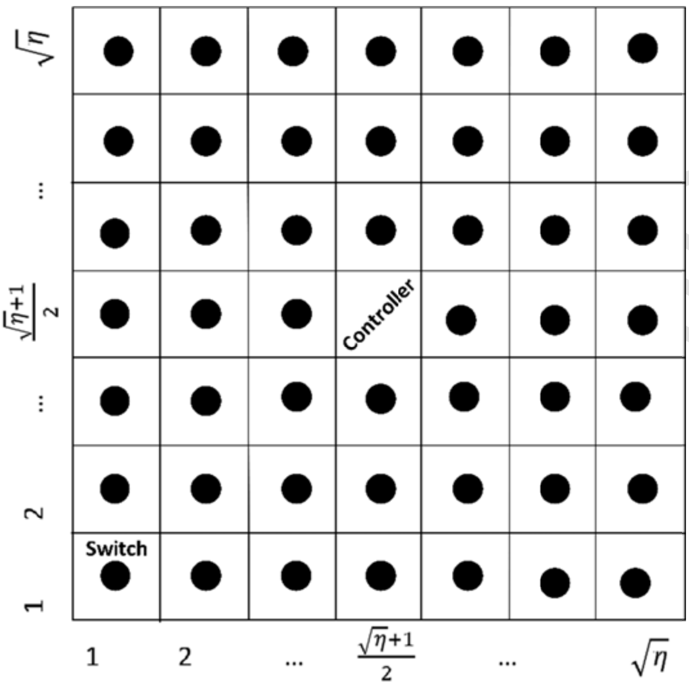


Figure 6: Hop counts in a virtual grid

So, (11) can be rewritten as

$$\eta = \frac{[\alpha - \propto_{FR} - (\alpha_{PD} + \alpha_{SF})\sqrt{\eta}]\xi}{\beta} \tag{12}$$

We can write (12) into a binomial quadratic equation, i.e.,

$$\eta\beta + \sqrt{\eta}(\alpha_{PD} + \alpha_{SF})\xi + (\propto_{FR} - \alpha)\xi = 0 \tag{13}$$

The solution of equation (10) will derive the minimum number of switches under a single controller within the NAN. Hence, the number of switches per controller would be

$$\eta = \left\{\frac{-(\alpha_{PD} + \alpha_{SF})\xi + \sqrt{((\alpha_{PD} + \alpha_{SF})^2\xi^2) - 4\beta(\propto_{FR} - \alpha)\xi\}}}{2\beta}\right\}^2 \tag{14}$$

Here, β will depend on the number of smart meters within the NAN. To determine the optimal number of switches of the NAN, assume, λ is the packet arrival rate (per sec) at the switch, m is the number of smart meters under each switch, and p is the packet generation rate per second at each of the smart meters.

$$\lambda = m * p \tag{15}$$

If a switch S_w can handle ε packets per second and the total number of smart meters in a NAN is M, the outcome of the following convex optimization problem would derive the optimal number of switches required in the network

$$\text{Minimize } S_w = \frac{M}{m}$$
$$\text{s.t. } m \le \varepsilon/p \tag{16}$$

Now, the optimal number of controllers C_t would be

$$C_t = \frac{S_w}{\eta} \tag{17}$$

4 SIMULATION MODEL

The proposed communication framework of SDN based WSN for SG NAN was implemented using Castalia, which is an OMneT++ platform simulator that provides radio and wireless channel models. To derive accurate and realistic performance of the model, a popular suburb named Fitzroy in Melbourne, Australia has been considered. It is assumed that all of the houses in this suburb have smart meters installed. The total land area of the selected region in Fitzroy is about 1.28 Km^2 as shown in Figure 7. Based on simulation parameters in Table II, the transmission power the whole area has been divided into 96 sectors each covering ($100x100\ m^2$). It was found that with a transmit power of 4.5 dBm the maximum line of sight distance can reach up to 300 m [36]. Based on our observation, there are approximately 1000 smart meters uniformly distributed within this suburb. Thus, it can be assumed that the average number of smart meters in each sector would be around 10 to 11. Please note that the traffic profiles of the SG applications are considered based on Table I. The simulation scenario was varied with the different seed value, and total 100 runs were considered to provide results that included average and 95% confidence intervals. The simulation parameters are summarized in Table II.

Google Earth Image (≈1000 smart meters) Goggle Map image (1.28 square Km)

Figure 7: Distribution of Smart Meters in the Fitzroy suburb

Figure 8 shows the relationship between the processing time and number of the switches per controller based on the derived equation of the optimized number of distributed controllers and switches. Among the actively running SG applications, delay sensitive applications require maintaining strict delay boundary of processing time. Keeping this requirement in mind, the simulation model considered the average delay of 110 ms (average delay requirement of delay sensitive applications) as the minimum required processing time to determine the optimum number of controllers. Based on the analysis, it has been identified that the number of switches that can be handled by a single controller would be about 30. Accordingly, the considered topology of WSN would require four controllers in total if each sector of the NAN is equipped with a single switch. However, the optimal number of controllers is derived based on the proposed mathematical model and in a practical scenario, the number may increase due to other factors such as environmental issues, terrain pattern, and placement of the switches at homes.

Table II SIMULATION SETTINGS

Simulation parameter	***Details***
Size of simulation area	$1600x800\ m^2$
Number of Smart meters (n)	1000
Radio Range (EN, SDSW, Controller)	~50 m [37]
Transmission Power	4.5 dBm

One hop distance between switches	200 m
Propagation delay	66 µsec
Storing and forward delay	10 ms
Flow resolve rate per controller	100000 [38]
Flow request response time	100 µsec [38]
Number of clusters	96
Number of considered service transformer	100
Delay requirement- App1* (Distribution automation)	100 ms (Delay sensitive)
Delay requirement- App2* (Demand Response- mission critical)	200 ms (Delay sensitive)
Delay requirement-App3* (Demand response programs)	500 ms (Delay tolerant)
Delay requirement-App4* (Outage management)	1 s (Delay tolerant)
Delay requirement-App5* (Meter reading)	1s (Delay tolerant) [1]
*representative applications to reflect different delay requirements	

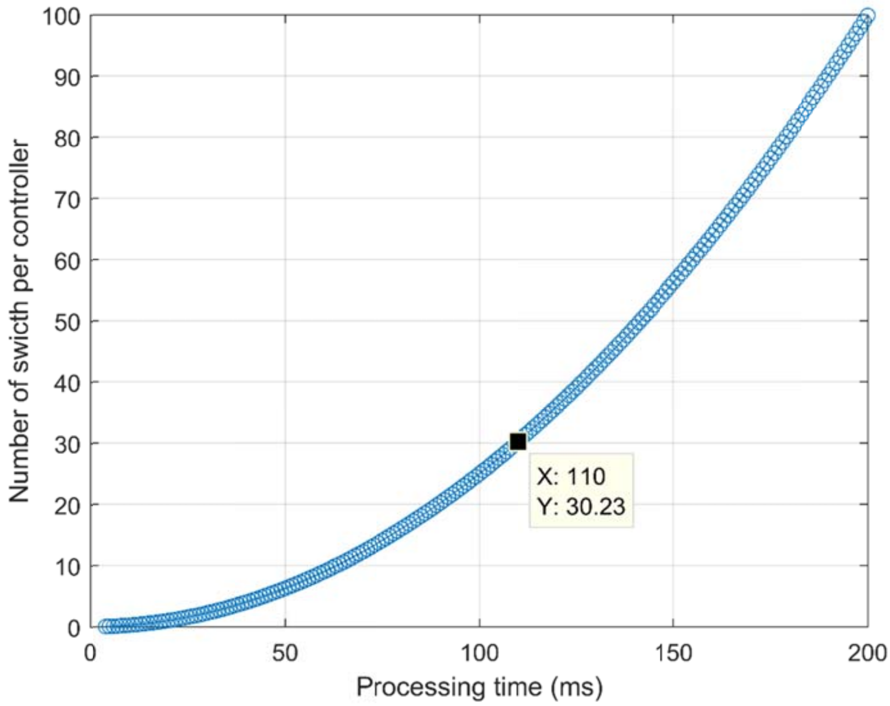


Figure 8: Variation of number of switch per controller in terms of processing delay

Three different simulation scenarios have been developed based on the network configuration. The first scenario only considers predefined flows for all SG applications running in the NAN domain. This network configuration can be regarded as the best-case scenario of the network. Any incoming packets at the switch will arrive within the flush timeout of the defined flows in the flow table. Flush-out time corresponds to period after which a flow instruction is removed from the flow table due to inactivity of the application. Hence, there will be no delay associated with flow table query. Based on the mentioned SG applications in the table, the network was configured, and the end to end delays have been measured to ensure successful deployment of the proposed framework.

To evaluate the performance of proposed model in the worst conditions, the second scenario of the simulation model considers no flows defined for any SG applications. Thus, each time a new application packet arrives at the switch, it is forwarded to the associated controller to get assigned with the new flow type. After successful initialization of flows, the packets are forwarded to the corresponding destinations.

Figure 9 shows the average end-to-end delay of different applications during the simulation runtime. The end-to-end delay corresponds to the period taken by a particular packet associated with an individual application traveling from the smart meters/IEDs to the NAN controllers. Further, the best case and worst case scenario in terms of average delay. Note that, in best case scenario (Figure 9 (a)); all SG application had predefined flows stored in the switches whereas in worst case scenario (Figure 9 (b)) there were no flows stored at any switches in the network. The delay sensitive applications suffer the most in the second case as the overall average delay increases. The increase derived here represents the flow processing time of the network (shown in Figure 9(c)).

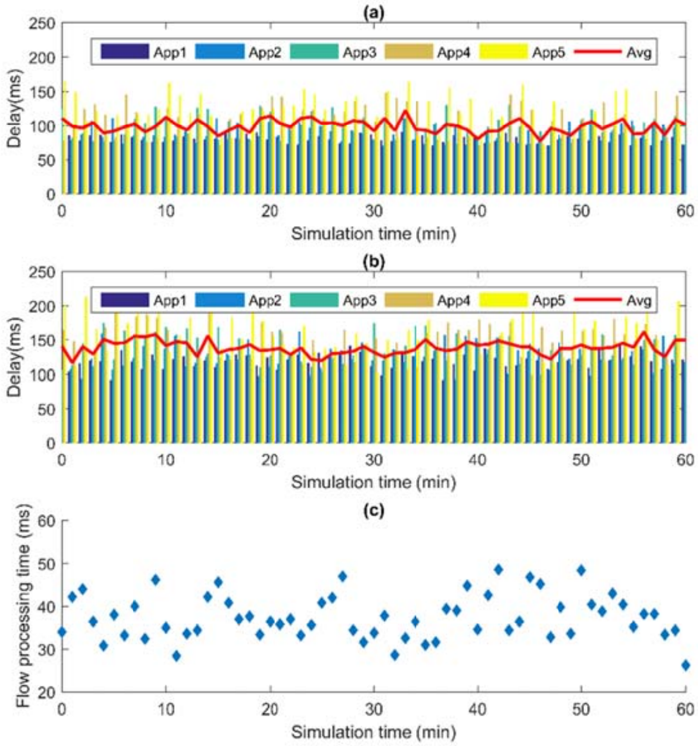
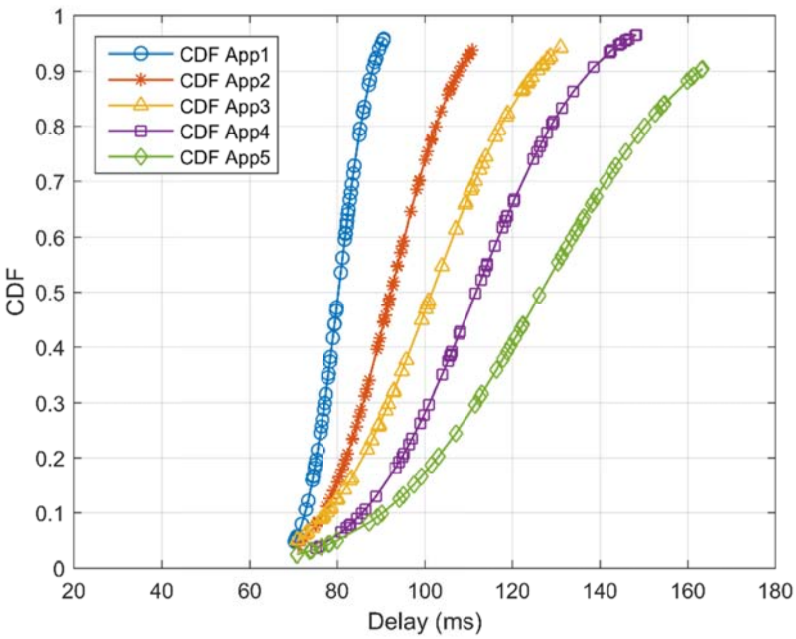
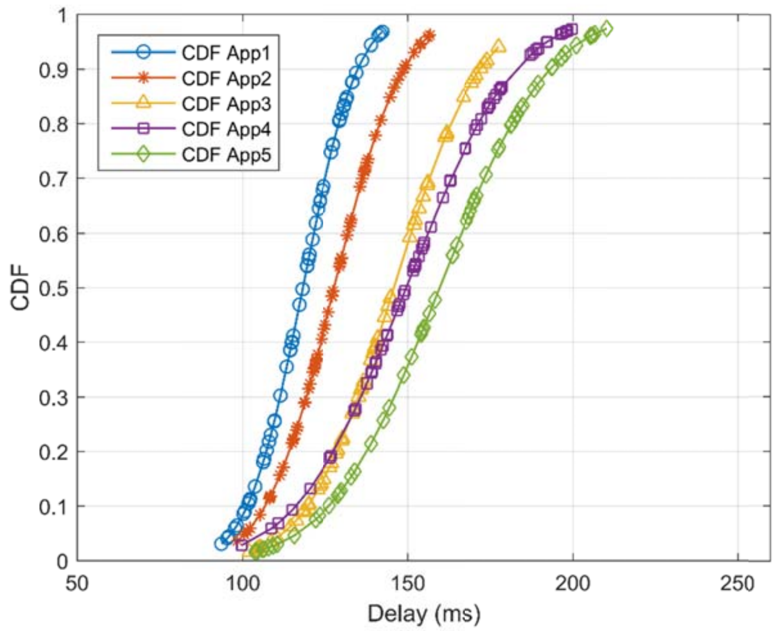


Figure 9: (a) Delay profile of different applications and total average delay in the best-case scenario (b) delay profile of different applications and total average delay in the worst case scenario and (c) flow processing time



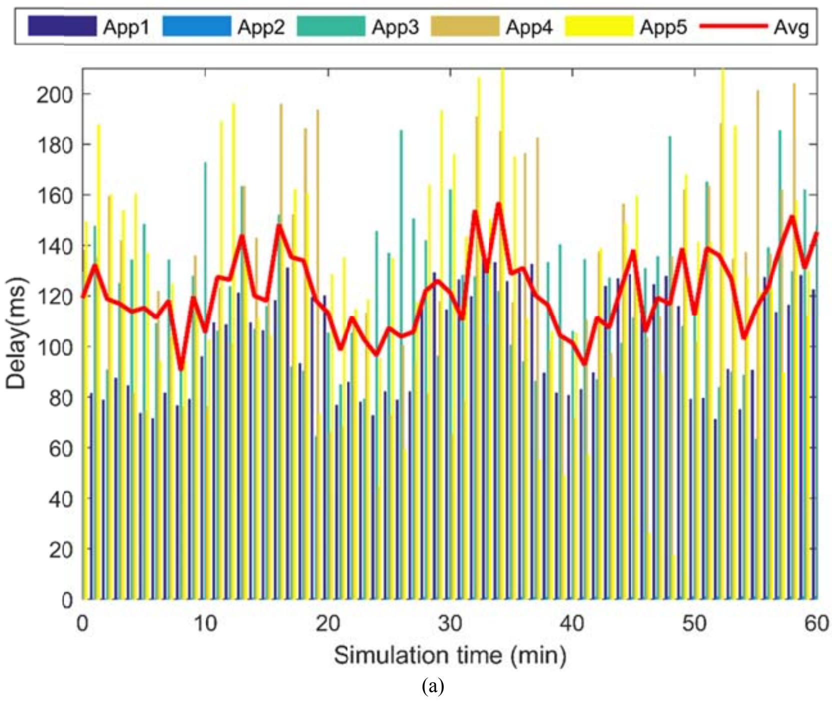
(a)



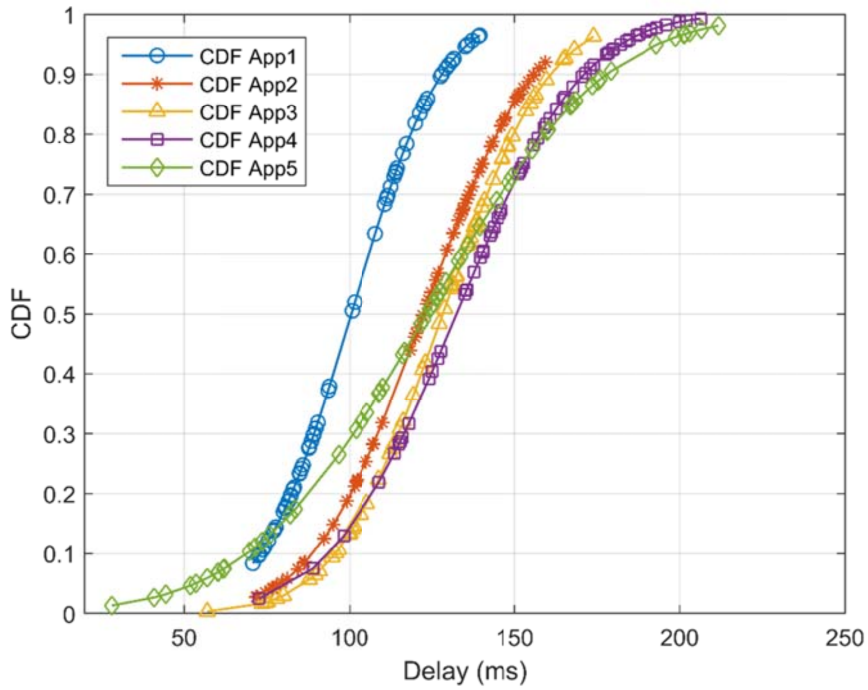
(b)

Figure 10: (a) Packet delivery time taken by different applications in the best-case scenario (b) Packet delivery time taken by different applications in the worst-case scenario

Figure 10 shows the packet delivery time and the probability of successful packet delivery for different SG applications in both best and worst case. The Cumulative Distribution Function (CDF) graph in Figure 10 (a) shows that in the best-case scenario, all packets were delivered to the specified destinations within the delay boundary of the associated applications. In case of the delay sensitive applications, such as App1 and App2, packets are successful in maintaining strict delay boundary of 100 ms and 200 ms. Due to the additional flow set up required for the delay sensitive applications (such as distribution automation (App 1)), in the worst-case scenario (shown in Figure 10 (b)), the network fails to deliver delay sensitive packets within the required delay boundary of 100 ms. The additional flow processing time contributes to the increased delays for all of the active SG applications in the worst-case scenario.



(a)



(b)

Figure 11: (a) Delay profile of different applications in random scenario (b) Packet delivery time taken by different applications in random scenario

The third scenario considers random packet generation from different SG applications. Here, in this case, predefined flows are assigned to some of the applications (both delay sensitive and delay tolerant). Further, different simulation scenarios were created where some predefined applications were varied. The packet generation rate of the different applications was varied on the basis of periodicity and stochasticity. As shown in Figure 11 (a) and Figure 11 (b), it is interesting to note that if the application is a delay sensitive application and the simulation stochastically generates an App1 packet, the network fails to deliver the packet within the required delay boundary. This is due to the randomly generated packet being placed in the long unresolved buffer queue. At times the network transmission processes exceeded the queue flush out time and it was necessary to reassign flow commands each time the SG event triggers. The randomly generated delay sensitive application packets were to be delivered within a strict time limit.

Based on the discussed simulation scenarios mentioned earlier, it is evident that it would not be realistic to adopt any of these techniques as there will be some delay sensitive SG applications which require confirmed packet delivery within strict delay boundary. Also, if the applications require flow information every time the packets arrive at the switch, the additional control channel overhead would be overwhelmed with high delay. A simple solution is found in the fourth simulation scenario. The end-to-end delay requirement of delay sensitive SG applications within the NAN usually remains between 25-100 ms (i.e. distribution automation). Also, mission critical data related to demand response has a delay boundary of 200 ms. To handle the identified challenge, two different types of flows named as *pro-active flow* and *reactive flow* have been assigned to handle the delay sensitive and delay tolerant applications respectively. In the case of proactive flows, the SDN controller pushes the flow to the SDN switch right after deploying any delay sensitive application within the network. This reduces the time required to resolve a delay sensitive application packet by avoiding control channel communication between SDN switch and SDN controller. On the other hand, reactive flows are unicast only to the corresponding SDN switch on demand basis. It is to be noted that the proactive flows are permanent in contrast to the reactive flows.

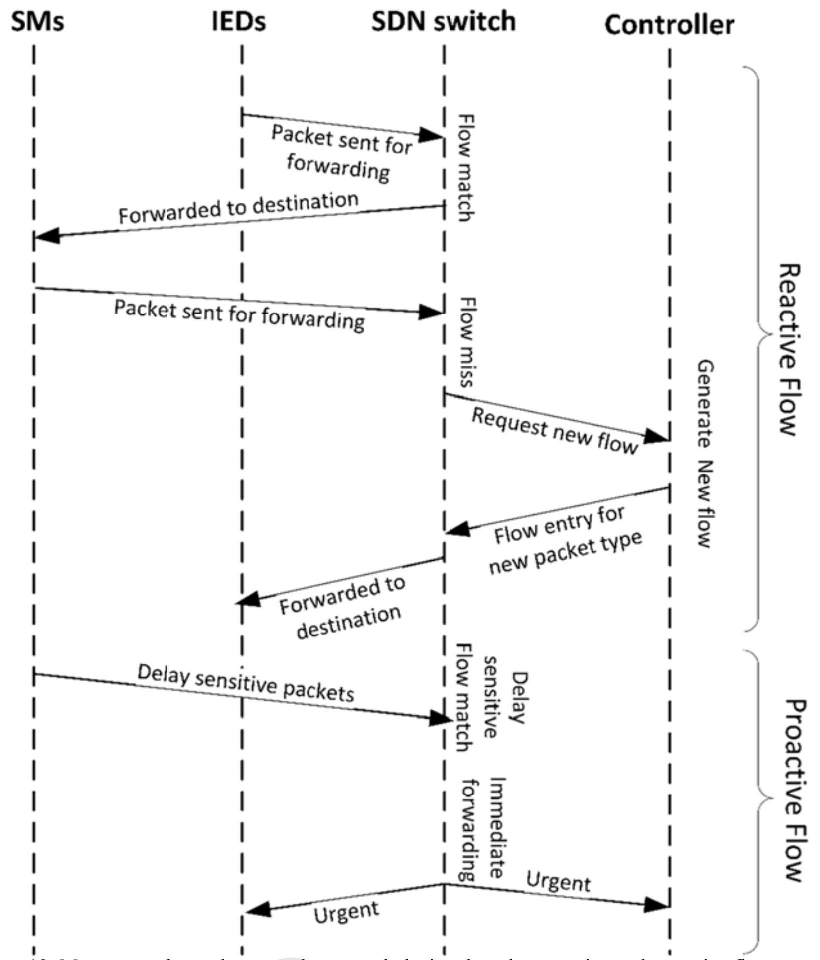
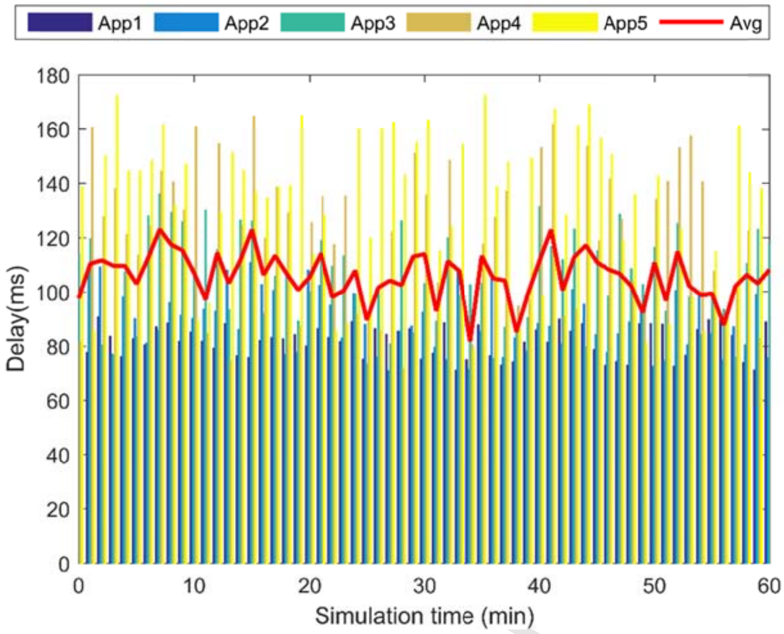


Figure 12: Message exchange between the network devices based on reactive and proactive flow types

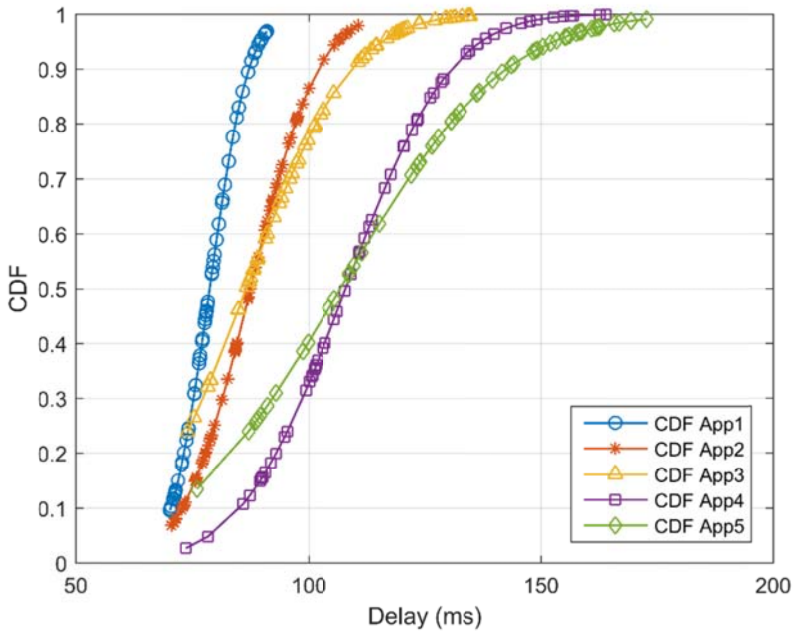
In the case of the delay sensitive applications whenever a packet arrives at the switch it's being broadcast to the associated switches based on the predefined proactive flow command. In this framework, proactive flow command is used to handle all the delay sensitive SG applications. On the other hand, upon receiving any application packet for the first time at the switch, the controller pushes a reactive flow into the switch to handle the packet. After getting assigned to a particular reactive flow command, similar application packets could be forwarded in a manner. Furthermore, for any delay sensitive application, packets are broadcast to all of the switches with infinite flush out time. In the case of a proactive flow the controller pushes flow instructions to the switches immediately after deploying a new application whereas, in the case of a reactive flow, the controller replies only after receiving a control packet requesting flows sent by the switch. Figure 12 shows the handshaking messages between the network devices based on reactive and proactive flow types.

Figure 13 (a) shows the delay profile of the SG applications while implementing proactive and reactive flows. It can be observed that the average total delay for all of the active applications is around 106 ms. Here, the delay sensitive applications were assigned with proactive flows and their delay requirement is met successfully. On the other hand, with reactive flows the delay tolerant applications also successfully meet the delay requirement. Figure 13 (b) shows the probability of packet delivery in terms of delay for different applications. It can be observed from the CDF graph (Figure 13 (b)) that the proposed proactive and reactive flows could successfully handle all of the active SG applications considered in this simulation model.

To determine the optimal point, simulation scenarios were developed by varying the total number of deployed controllers. Figure 14 shows the number of controllers required and the associated average delay statistics while varying the number of controllers in the simulation scenario (worst-case). The selected SG applications in this study utilized a maximum transmission delay in the worst-case scenario and the optimal point found could handle the delay without the need to implement proactive flows. The outcome reveals that the optimal point is reached with six controllers and the average delay was at the minimum value (106 ms). Thus, based on the simulation outcome, it is evident that the proposed mathematical model derived a near realistic estimation of an optimal number of controllers required in the NAN.



(a)



(b)

Figure 13: (a) Delay profile of different applications in case of proactive and reactive flows (b) Packet delivery time taken by different applications in case of proactive and reactive flows

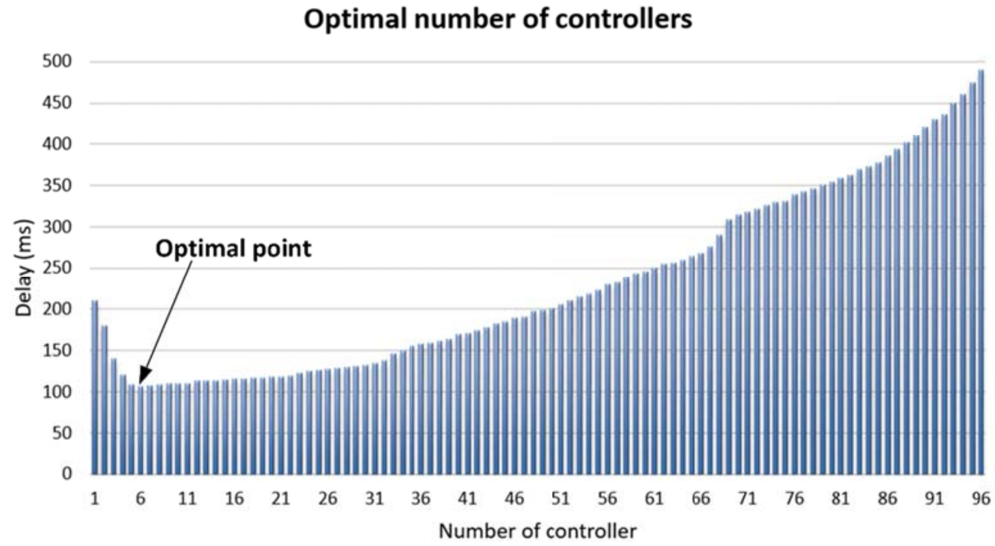


Figure 14: Optimal number of deployed controllers

An SDN based SG communication network would provide a significant upgrade to capability and flexibility with the NAN by being able to add and upgrade network services and applications. Since, the primary objective of the proposed SDN based SG communication framework is to develop a novel new approach to SG networking, the initial work targets the key NAN domain. Future challenges include integrating different SG domains into the SDN based SG communication network utilizing the appropriate transmission technologies for each domain type. Once modeling of the enhanced SDN based SG communication network is completed, performance benchmarking would assist with a determination of the validity of the approach.

1 Conclusion

The emerging SGs and SG applications is providing the motivation needed to upgrade SG communication networks. In this paper, a novel approach for handling the SG applications within a NAN domain is proposed based on the SDN paradigm. The proposed framework focuses on SDN based SG NANs and is is advantageous because of the flexibility and programmability provided by SDN and its ability to manage a large number of M2M devices. The paper sheds light on the crucial challenge of finding the number of controllers required for a WSN based NAN domain. The mathematical model presented was developed based on WSNs using the cc2350 standard. Deployments of proactive and reactive SG applications eliminate the difficulty of managing the delay sensitive and delay tolerant applications. The approach used to determine the number of controllers needed in a NAN with WSN connected devices provides a means to ensure that overdesigning and overprovisioning controllers does not occur. The continuation of this work will include implementing the framework using a testbed. Cross-domain communication in SGs utilising SDN multi-controller communications remains another area for further investigation.

2 Appendix

Proof of Lemma 3.1

Proof: Let's assume the total number hop counts in a virtual grid is H, where the hop counts of switches S_{w1}, S_{w2}, S_{w3}, …, S_{wn} to reach the controllers are denoted with $h_1, h_2, h_3, \ldots, h_\eta$

Hence,

$$H = (h_1 + h_2 + h_3 + \cdots + h_{\sqrt{\eta}}) + (h_{\sqrt{\eta}+1} + h_{\sqrt{\eta}+2} + h_{\sqrt{\eta}+3} + \cdots + h_{2\sqrt{\eta}}) + (h_{2\sqrt{\eta}+1} + h_{2\sqrt{\eta}+2} + h_{2\sqrt{\eta}+3} + \cdots + h_{3\sqrt{\eta}}) + \cdots + (h_{(\sqrt{\eta}-1)(\sqrt{\eta}+1)} + h_{(\sqrt{\eta}-1)(\sqrt{\eta}+2)} + h_{(\sqrt{\eta}-1)(\sqrt{\eta}+3)} + \cdots + h_\eta) \quad (18)$$

Then,

$$H = \left\{(\sqrt{\eta}-1) + (\sqrt{\eta}-2) + \cdots + \sqrt{\eta} - \left(\frac{(\sqrt{\eta}+1)}{2}\right) + \cdots + (\sqrt{\eta}-2) + (\sqrt{\eta}-1)\right\} + \left\{(\sqrt{\eta}-2) + (\sqrt{\eta}-3) + \cdots + \sqrt{\eta} - \left(\frac{\sqrt{\eta}+1}{2}\right) - 1 + \cdots + (\sqrt{\eta}-3) + (\sqrt{\eta}-2)\right\} + \left\{(\sqrt{\eta}-3) + (\sqrt{\eta}-4) + \cdots + \sqrt{\eta} - \left(\frac{\sqrt{\eta}+1}{2}\right) - 2 + \cdots + (\sqrt{\eta}-4) + (\sqrt{\eta}-3)\right\} + \cdots + \left\{(\sqrt{\eta}-1) + (\sqrt{\eta}-2) + \cdots + \sqrt{\eta} - \left(\frac{\sqrt{\eta}+1}{2}\right) + \cdots + (\sqrt{\eta}-2) + (\sqrt{\eta}-1)\right\} \quad (19)$$

$$H = \left\{\eta - \frac{(\sqrt{\eta}+1)^2}{4}\right\} + \left\{\eta - \frac{(\sqrt{\eta}+1)^2}{4} - \sqrt{\eta}\right\} + \left\{\eta - \frac{(\sqrt{\eta}+1)^2}{4} - 2\sqrt{\eta}\right\} + \cdots + \left\{\eta - \frac{(\sqrt{\eta}+1)^2}{4}\right\} \quad (20)$$

Thus, the average number of hops from a switch to controller would be

$$h = H/\eta \cong \sqrt{\eta}/2 \quad (21)$$

3 References

[1] R. Khan, J. Khan, "A comprehensive review of the application characteristics and traffic requirements of a SG communications network," *Computer Networks*, Volume 57, Issue 3, Pages 825-845, ISSN 1389-1286, February, 2013.
[2] A. Molderink, V. Bakker, M. G. C. Bosman, J. L. Hurink, and G. J. M. Smit, "Management and control of domestic SG technology," *IEEE Transaction on Smart Grid*, 1(2):109119, 2009.
[3] B. Kroposki, R. Margolis, G. Kuswa, J. Torres, W. Bower, T. Key, and D. Ton, "Renewable systems interconnection: Executive summary," *Technical Report NREL/TP-581-42292*, U.S. Department of Energy, 2008.
[4] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, G. P. Hancke, "A Survey on SG Potential Applications and Communication Requirements," *IEEE Transaction on Industrial Informatics*, vol.9, no.1, pp.28-42, Feb. 2013.
[5] F. Benzi, N. Anglani, E. Bassi, and L. Frosini, "Electricity smart meters interfacing the households," *IEEE Transaction on Industrial Electronics*, vol. 58, no. 10, pp. 44874494, Oct. 2011.
[6] R. Deng, Z. Yang, M. Chow, J. Chen, "A Survey on Demand Response in SGs: Mathematical Models and Approaches," *IEEE transaction on Industrial Informatics*, vol.11, no.3, pp.570-582, June 2015. doi:10.1109/TII.2015.2414719.
[7] F. Andren, R. Brundlinger, T. Strasser, "IEC 61850/61499 Control of Distributed Energy Resources: Concept, Guidelines, and Implementation", *IEEE Transaction on Energy Conversion*, vol.29, no.4, pp.1008-1017, Dec. 2014. doi: 10.1109/TEC.2014.2352338.
[8] L. Igualada, C. Corchero, M. Cruz-Zambrano, F.-J. Heredia, "Optimal Energy Management for a Residential Microgrid Including a Vehicle-to-Grid System," *IEEE Transactions on Smart Grid*, vol.5, no.4, pp.2163-2172, July 2014, doi: 10.1109/TSG.2014.2318836.
[9] N. Dorsch, F. Kurtz, H. Georg, C. Hagerling, C. Wietfeld, "Softwaredefined networking for SG communications: Applications, challenges and advantages," *IEEE International Conference on Smart Grid Communications(SmartGridComm)*, 3-6 Nov, 2014.
[10] A. Cahn, J. Hoyos, M. Hulse, E. Keller, "Software-defined energy communication networks: From substation automation to future SGs," *IEEE International Conference on SG Communications (SmartGridComm)*, Vancouver, BC, 2013, pp. 558-563., 2013, doi: 10.1109/SmartGridComm 2013.6688017.
[11] L. Ren, Y. Qin, B. Wang, P. Zhang, P. B. Luh, R. Jin, "Enabling Resilient Microgrid through Programmable Network," *IEEE Transactions on Smart Grid*, vol.PP, no.99, pp.1-1, 2016, doi: 10.1109/TSG.2016.2589903.
[12] I. Petri, Z. Mengsong, A.R. Zamani,J. Diaz-Montes, O. Rana, M. Parashar, "Integrating Software Defined Networks within a Cloud Federation," *15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, pp.179-188, doi: 10.1109/CCGrid, May 2015.
[13] Z. Guo, M. Su, Y. Xu, Z. Duan, L. Wang, S. Hui, and H. J. Chao, "Improving the performance of load balancing in software-defined networks through load variance-based synchronization," Computer Net-works, vol. 68, pp. 95–109, 2014
[14] M. Cello, Y. Xu, A. Walid, G. Wilfong, H. J. Chao and M. Marchese, "BalCon: A Distributed Elastic SDN Control via Efficient Switch Migration," 2017 IEEE International Conference on Cloud Engineering (IC2E), Vancouver, BC, 2017, pp. 40-50. doi: 10.1109/IC2E.2017.33
[15] S. Schmid, J. Suomela, "Exploiting locality in distributed SDN control", Proc. 2nd ACM SIGCOMM Workshop Hot Topics Softw. Defined Netw., pp. 121-126, 2013.
[16] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, B. Ohlman, "A survey of information-centric networking," *IEEE Communication Magazine*, 50 (7), 2636, 2012.
[17] L. E. Li, Z. M. Mao, and J. Rexford, "Software-defined cellular networks," *Proc. of European Workshop on Software Defined Networks (EWSDN)*, Darmstadt, Germany October 2012.
[18] L. Veltri et al., "Supporting information-centric functionality in software defined networks," *Proc. IEEE International Conference on Communications (ICC)*, Ottawa, Canada, June 2012.
[19] T. Baker, B. Al-Dawsari, H. Tawfik, D. Reid, and Y. Ngoko, "GreeDi: An energy efficient routing algorithm for big data on cloud," Ad Hoc Networks, vol. 35, pp. 83–96, 2015.
[20] T. Baker, Y. Ngoko, R. Tolosana-Calasanz, O. F. Rana and M. Randles, "Energy Efficient Cloud Computing Environment via Autonomic Meta-director Framework," 2013 Sixth International Conference on Developments in eSystems Engineering, Abu Dhabi, 2013, pp. 198-203. doi: 10.1109/DeSE.2013.43
[21] S. Lukovic et al., "Virtual power plant as a bridge between distributed energy resources and SG," *Proc. of 43rd Hawaii International Conference on System Sciences (HICSS)*, Kauai, HI, USA, January 2010.
[22] V. C. Gungor, L. Bin, G.P. Hancke, "Opportunities and Challenges of WSNs in SG," *IEEE Transactions on Industrial Electronics*, vol.57, no.10, pp.3557-3564, doi: 10.1109/TIE.2009.2039455, Oct 2010.
[23] M. Kuzlu, M. Pipattanasomporn, S. Rahman, "Communication network requirements for major SG applications in HAN, NAN and WAN," *Computer Networks*, Volume 67, Pages 74-88, ISSN 1389-1286, July 2014, [online] http://dx.doi.org/10.1016/j.comnet.2014.03.029
[24] K. H. Chang, B. Mason, "The IEEE 802.15.4g standard for smart metering utility networks," *IEEE Third International Conference on SG Communications (SmartGridComm)*, 5-8 Nov. 2012. doi: 10.1109/Smart-GridComm.2012.6486030.
[25] J. Vardakas, N. Zorba, C.V. Verikoukis, "A Survey on Demand Response Programs in SGs: Pricing Methods and Optimization Algorithms," *IEEE Communications Surveys and Tutorials*, vol.17, no.1, pp.152-178, Firstquarter 2015, doi: 10.1109/COMST.2014.2341586.
[26] R.W. Uluski, "The role of Advanced Distribution Automation in the SG," *Power and Energy Society General Meeting*, 2010 IEEE, vol., no., pp.1-5, 25-29 July 2010 doi: 10.1109/PES.2010.5590075.
[27] *Machine-to-Machine (M2M) communications study report*, IEEE 802.16 Broadband Wireless Access Working Group, Tech. Rep., 2010.
[28] Guidelines for assessing wireless standards for SG applications, NIST Priority Action Plan 2 (PAP2), Tech. Rep., 2010.
[29] D.F. Ramirez, S. Cespedes, C. Becerra, C. Lazo, "Performance evaluation of future AMI applications in SG Neighborhood Area Networks," *IEEE Colombian Conference on Communications and Computing (COLCOM)*, pp.1-6, 13-15 May 2015, doi: 10.1109/ColCom-Con.2015.7152093.
[30] Q. D. Ho, Y. Gao and G. Rajalingham, "Wireless Communications Networks for the SG," *SpringerBriefs in Computer Science*, 2014, DOI 10.1007/978-3-319-10347.
[31] K. K. Yap, M. Kobayashi, R. Sherwood, T. Y. Huang, M. Chan, N. Handigol and N. McKeown, "OpenRoads: Empowering Research in mobile networks, ACM SIGCOMM Computer Communication Review", vol. 40, no. 1, pp. 125-126, 2010.
[32] J. Kim, F. Filali and Y. B. Ko, "A lightweight CoAP-based software defined networking for resource constrained AMI devices," *IEEE International Conference on Smart Grid Communications (Smart-GridComm)*, Miami, FL, 2015, pp. 719-724, 2015, doi: 10.1109/SmartGrid-Comm.2015.7436386.

[33] K. Ahmed and M. Gregory, "Distributed Efficient Similarity Search Mechanism in Wireless Sensor Networks," Sensors, vol. 15, no. 3, pp. 5474–5503, May 2015.

[34] "Power Delivery Systems," in Power Distribution Planning Reference Book, Second Edition, ed: CRC Press, 2004.

[35] K. Ahmed and M. A. Gregory, "Optimized TDMA based distance routing for Data Centric Storage", *IEEE 3rd International Conference on Networked Embedded Systems for Every Application (NESEA)*, Liverpool, pp. 1-7., 2012, doi: 10.1109/NESEA.2012.6474019.

[36] N. S. Nafi, K. Ahmed, M. Datta and M. A. Gregory, "A novel Zigbee based pilot protection scheme for smart distribution grid," *Telecommunication Networks and Applications Conference (ATNAC)*, Australasian, Southbank, VIC, 2014, pp. 146-151. doi: 10.1109/ATNAC.2014.7020889.

[37] K. Mahmood, A. Chilwan, O. Osterbo and M. Jarschel, "On the Modeling of OpenFlow-based SDNs: The Single Node Case," *Proceedings of Computer Science and Information Technology (CS and IT)*, Vol.4, pp 207-214, 2014, doi 10.5121/csit.2014.41120.

[38] K. Holmberg, A. Adgar, A. Arnaiz, E. Jantunen, J. Mascolo, and S. Mekid, E-maintenance. Dordrecht: Springer, 2010.

4 BIOGRAPHIES

Nazmus Shaker Nafi (S'14) received the B.Eng. (Hons.) degree in Communication from the International Islamic University Malaysia (IIUM), in 2010, and the M.Phil. degree in Computer engineering from the University of Newcastle, Newcastle, NSW, Australia, in 2013. Currently, he is a Ph.D. candidate in the RMIT University, Melbourne, Australia. His current research interests include Smart Grid communication systems, wireless network architecture, Machine to Machine, Software Defined Networks and Internet of Things.

Khandakar Ahmed (M'11) is a Lecturer in Discipline of IT, School of Engineering and Science, Victoria University, Melbourne, Australia. He completed his PhD Degree from the RMIT University, Australia in December 2014. During his PhD, he was an active scholar in the network research group of School of Engineering, where he explored Data-Centric Storage (DCS) in Wireless Sensor Network (WSN). Before joining Victoria University, Khandakar was a full-time Lecturer with the School of IT and Engineering (SITE), Melbourne Institute of Technology (MIT), Melbourne and a researcher in Australia-India Research Centre for Automation Software Engineering (AICAUSE), RMIT University (2016-2017). He has been with AICAUSE as a Post-Doctoral researcher before he takes the full-time position at MIT during 2015-2016. He received his M.Sc. in Networking and e-Business Centred Computing (NeBCC) under the joint consortia of University of Reading, UK; Aristotle University of Thessaloniki, Greece and Charles III University of Madrid (UC3M), Spain in 2011. Khandakar completed his B.Sc. (Eng.) Degree in Computer Science & Engineering from Shahjalal University of Science and Technology (SUST), Bangladesh, with an excellent CGPA of 3.92/4.0. He is also serving as the member of the editorial board of Australian Journal of Telecommunications and the Digital Economy (AJTDE). He has also been serving as a reviewer for several international A* journals and conferences.

Mark A Gregory (SM'99) is a Fellow of the Institute of Engineers Australia and a Senior Member of the IEEE. Mark A. Gregory received a PhD from RMIT University, Melbourne, Australia in 2008, where he is an Associate Professor in the School of Engineering. In 2009, he received an Australian Learning and Teaching Council Citation for an outstanding contribution to teaching and learning. He is the Managing Editor of two international journals (AJTDE and IJICTA) and the General Co-Chair of ITNAC. Research interests include telecommunications, network design and technical risk.

Manoj Datta (S'06, M'11) received the D.Eng. degree in interdisciplinary intelligent system engineering from University of the Ryukyus, Japan, in 2011. From 2011 to 2012, he was a Post-Doctoral Research Fellow at the Japan Society for the Promotion of Science (JSPS), Japan. During 2012 - 2013, he was an Assistant Professor at the Gifu University, Gifu, Japan. He is currently a lecturer of Electrical Power and Energy Engineering at the RMIT University, Melbourne, Australia. His research interests include micro-grid modelling, grid integration of DGs, and intelligent ancillary services in smart grid. Dr. Datta was the recipient of the University of the Ryukyus President's award for his outstanding D.Eng. thesis. He is a member of the Institute of Electrical Engineers, Japan.

- A novel SDN-based NAN framework built upon WSN
- Five different types of SG applications have been considered based on their traffic characteristics to model the SDN-based NAN
- Performance of the network has been analyzed to validate the feasibility of the framework.
- Smart algorithms have been developed and introduced in the properties of WSN at the application layer to accommodate the SDN features.
- Additionally, to drag a solution specifying the optimal number of controller and switches, a novel analytical model has been developed based on the communication traffic cost.