Int. J. of Thermal & Environmental Engineering
Volume 18, No. 1 (2021) 19-25

DSSNet Framework - Analysis of Combined Simulation and Emulation Results for Smart Grid Planning and Evaluation: A Survey

Usman Arshad [a]*, Muhammad Adnan [b], Farooq Mustafa Joyia [b], Waqas Nazir Awan [b], Muhammad Ali [b], Tayaba Abbasi [c]

[a] *Department of Electrical Engineering, University of Engineering & Technology, Lahore, Pakistan*
[b] *Department of Technology, The University of Lahore, Lahore, Pakistan*
[c] *Department of Electrical and Communication Engineering, United Arab Emirates University, Abu Dhabi, UAE*

Abstract

The recent advancements in power sector and technological revolution have enabled an environment for scientist and researchers to explore this area in new terms. Successful operations of modern grid systems highly rely on the communication architecture. New mechanisms for effective, reliable, and controllable network applications are being developed to enhance the efficiency of grid systems. This inclusion of communication networks in Smart Grid (SG) systems has opened the doors to apply Software Defined Networking (SDN) technology to enhance the efficiency and performance of SG systems. The Distribution System Simulator Network (DSSNet) relies on container based virtual time system used for synchronization among power simulation and network emulation. System scalability and usability can be enhanced with the distributed controller environment. This research work provides a framework for combining power distribution simulation with SDN emulation to analyze communication network applications for planning and evaluation of smart grid systems. The performance of DSSNet has been demonstrated and evaluated with a case study of demand and supply application. Demand and supply application tested with a large number of nodes demonstrate a new research finding of time delay with increased nodes. Finally, limitations and shortcomings of DSSNet have also been discussed which opens new doors for researchers to explore large network and more nodes-based SG systems.

Keywords: *Smart Grid, DSSNet, SDN Emulator, Power Simulator, ONOS, OpenDSS*

1. Introduction

The interconnection of electricity producers and consumers is termed as Electric Grid. An Electrical Grid (EG) consists of power stations, high voltage transmission lines from power stations to load centers, and distribution lines from load centers to the individual consumers for the power distribution [1]. At the time of initial development, EG was considered to be unidirectional system whose purpose was to control production and distribution of power according to the consumers' demand. With the passage of time, demand of power consumers increased, and power plants became very large in size. Further, dependability on fossil fuels driven plants has started to decrease due to uncertainty in the availability of fuels and their environmental impacts which leads to consider Renewable Energy (RE) resources like solar, wind, and hydel.

Latest advancements in electronics, control, and communication technologies have also impacted the development of conventional electrical grid systems. Now the systems have been developed to monitor real time usage, and demand and supply of power at the consumers' end. Advanced Metering Infrastructure (AMI) meters help in demand management at consumer side. Introduction of highly variable production from RE resources has changed the energy mix in many countries. The power generation, distribution, and consumption is becoming complex, therefore, requires more

* Corresponding author. Tel.: +92-345-4171094
E-mail: m.usman600@gmail.com

DOI: 10.5383/ijtee.18.01.003

diverse, controllable and coherent systems to integrate both demand and supply side advancements. SG comes out to be the versatile and robust solution to these complexities. SG is the form of EG which is capable for real time communications to and from production and consumption ends with an effective and centralized control over the distributed system. Latest discoveries in communication technologies and renewable power generation have motivated scientist and researchers to provide analysis and results towards the development of SG. Growth in SG modernization depends upon the convergence of operations and Information Technology (IT). Controlling the operations through SDN technology for secure and reliable operations and better and efficient control is the emerging topic of research in SG systems. SDN is an open-source, programmable approach for designing, building, and managing networks for better performance and monitoring [2]. It has the enhanced ability to separate both data and control planes and obtain an overall global view of the network/system.

Knowing the importance of real time data transmission, a system using OpenDSS was used for sending real-time signals from power simulation to a hardware [3]. Real-time simulations for hardware in the loop simulations have also been used for getting results closer to the real system [4]. The results obtained with this model are closer to the real system but dependent on specific hardware compatible with simulation environment. Engineers have also tried co-simulation between Power Sequence Load Flow (PSLF) and network simulator-2 (ns-2). In these simulations, events are sorted by timestamps for synchronization of messages between two simulators [5].

Another research very common in complex system modeling and design is the use of agent-based modeling and simulation. This platform uses agents as elements of co-simulation between power and communication system [6]. A federal approach of co-simulation of power and electrical systems is the use of multiple transmission and distribution power simulators with network simulator-3 (ns-3) as communication simulator [7]. Further improvement in the synchronization of the systems was suggested by the researchers [8], but still there is a challenge due to inherent difference in mechanism of simulation and emulation.

SDN technology, because of their ability to have a global view on entire network, can help to increase performance and security in large scale networks and even in smart grid based energy systems. Studies were carried out by researchers and utilization of hardware-in the-loop testbed with SDN platform for the development of micro-grid [9]. SDN was also proposed for scalable utility application deployment, however, the system was mainly limited to communication networks and devices [10]. Studies of combining SDN technology with high voltage solver are also available in literature [11]. For the network based smart grid systems, security against cyber-attacks is also the main concern while designing the system. A research shows use of SDN for the increase of performance of SCADA based networks model [12]. To cater for the impact of electricity black out in undue circumstances, live monitoring of data is of prime importance. The process of this live monitoring is termed as situational awareness and analysis of utility communication networks for situational awareness were carried out during blackouts [13].

2. System Description

With the advancements in the technology of power generation, transmission and distribution systems and communication systems, grid systems are also moving towards SG. With the complexities involved in new power system and communication system integration, system study approach uses a layered approach which makes it easier to understand and provide a better control for issues specific to the domain. The role of SDN based control layer is of vital importance. It provides a global overview of the overall system states and better security for the applications running on the system.

3. Model Design

A model that combines both power system simulation and SDN based communication system has been evolved in this study. System named DSSnet has been evolved that uses different open-source softwares like OpenDSS, Open Network Operating System (ONOS), and Open vSwitch. There are two basic components presented on top of two different operating systems that are windows and Linux. There will be a need of synchronization among two independent complex systems. For that purpose, a separate virtual clock system and coordinators have been placed in the system

4. System Design and Architecture

The system modeled above comprises of five main components: electrical power system simulator, power coordinator for controlling and interfacing simulations, communication network emulator, and network coordinator for interfacing software defined network, and lastly a virtual time system. Every system component in the architecture has a specific functionality described in detail in the following subsections.

4.1. Power Simulator

Power network has been simulated in DSSnet with power elements like relays, transformers, sensors, capacitors generators, and loads. Each Intelligent Electrical Device (IED) is modeled in simulator and network emulator. Other power systems may not be modeled in network emulator as they might exist only in power network. Power system simulator is designed for the simulations of utility distribution systems for the purpose of load modelling, fault studies, harmonic studies, and solving dynamic time step power flows. The time taken by simulations depends upon the type of simulation performed that may be of the order of milliseconds to a few seconds [14].

4.2. Network Emulator

Network emulator in DSSnet consists of software switches in Mininet platform to emulate the function of real SDN switches. All the hosts connected with open vSwitch topology are representation of IED in power simulator. Hosts have their namespaces and run real time processes for modeling the functioning of IEDs. All the elements of the modern power network that require communication, can be modeled in emulation including phasors, sensors, generators, and load meters like AMI meters [15]. Some benefits of this modelling are that we do not need to include all the elements of power network in the simulation and some of the elements can be presented in the emulator for faster processing and obtaining closer to real results. There are also some disadvantages associated with the emulation besides faster processing. Emulation can run up to a certain limited number of hosts and switches. Each host connected with the switch has its own virtual network adapter. Also, with the increase in the number of hosts beyond a certain limit, complexity of the system can increase to a very high level. Further, emulation unlike simulation can-not run with hundreds and thousands of hosts due to limited number of resources available in virtual environment. The solution to this shortcoming might include development of smaller distributed emulators for increasing scalability [16].

4.3. Network Coordinator

Network Coordinator on top of the emulator can be termed as central controller of network emulation. It has a global view of all the network devices in the emulation. Network devices and IEDs modelled in emulator are configured through network coordinator. The other purpose of the network coordinator is the communication and interfacing of the power simulation events managed on the power coordinator level. If a synchronization request is received at the coordinator, it interfaces with its counterpart and virtual clock of the system to control the overall DSSnet architecture.

4.4. Power Coordinator

Power coordinator has the same functionality on top of power simulator that network coordinator has in network emulator. The coordinator initiates simulation by setting circuit variables and provides Application Programming Interface (API) for modification and extraction of results from simulation. Some elements of the power grid may be modeled in power coordinator as a function of time. This timing functionality enables system to synchronize events of simulation with the emulation network output.

4.5. Virtual Time System

Emulator system clock always elapses with real world time clock but simulation system has its own system clock with respect to simulation environment. Therefore, pausing of emulation causes to pause the virtual clocks of each emulation entity. This situation causes a difference in the clock times of simulation, emulation, and real world clock. To overcome this situation, virtual clock system can be used that can be modeled according to the situation [17, 18]. Since the requirement of system includes ONOS and Mininet, the work of [19] had been extended for provision of virtual time system.

Virtual time system is generally used for the purpose of slowing down emulation so that emulator assumes that enough virtual resources are available. The other usage of virtual time system is the synchronization among emulation and simulation. In the system designed, each container or host is provided with its own virtual clock. With this type of provision, each element has the flexibility of starting, stopping, fastening or slowing down its own clock in order to synchronize with the simulation clock.

With the provision of each container clock, there arises the issue of consistency among all elements in the emulator. The issue is resolved with implementing a centralized timekeeper and by a two layered consistency mechanism.

5. Synchronization

Another main challenge of DSSnet is the effective synchronization among network emulator and power simulator. The challenge arises due to the reason of simulator using its own virtual simulation clock and emulator using system clock time. These clocks, if working independently, cause system errors. This phenomenon is explained with the help of an example.

As illustrated in Figure 1, if there are three discrete time events E_1, E_2, and E_3 that generate response of R_1, R_2, and R_3. E_1 occurs before E_2 and E_3 whereas R_1, R_2, and R_3 are the results of the events, respectively. Now, If R_1 is the response of E_1 and E_2 is dependent on R_1. If E_2 reaches before R_1, then there will be an error for the process. Delay and errors will escalate in the system as every next input and output is dependent on each other. To overcome this issue of errors due to poor synchronization of the events, virtual time system in Mininet platform has been developed.

Since the idea is to design a system where events are driven by network emulator, therefore, synchronization requests are originated at emulator side. Before the coordinators at both ends send signals for simulation to advance over a time interval, it must be ensured that emulator clocks have advanced up to the point where output for the simulation system has been generated. As the emulators are using, virtual clocks, simulation clocks are not about to pass the emulation systems. Figure 2 shows the execution cycle of DSSnet for some events.

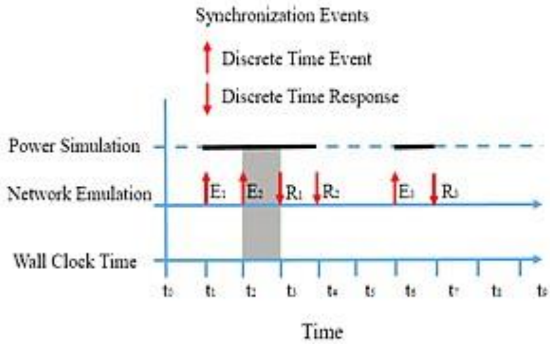

Fig. 1. The execution of system with respect to wall clock

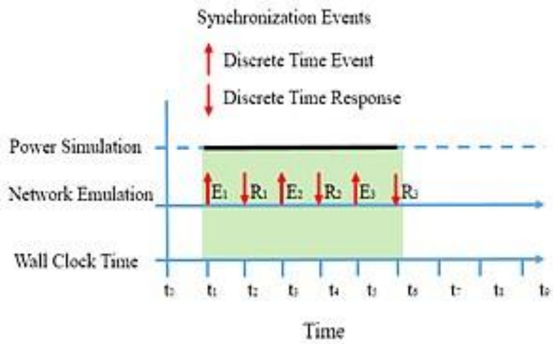

Fig. 2. Execution of DSSnet with respect to its own perceived time and virtual time elapsed in the simulation.

Time scale graph presented in Figure 2 can be expressed in the form of equation as:

$$Time_{Wall\_Clock} = \sum t_{E,i} + \sum t_{S,i} \tag{1}$$

$$Time_{DSSnet} = \sum t_{E,i} + \sum t_{S`,i} \tag{2}$$

$$ret = \frac{t_{S`,i}}{t_{S,i}} \tag{3}$$

Equation-1 describes the complete time of the system with respect to wall clock time which is the sum of emulation time ($t_{E,i}$) and simulation time ($t_{S,i}$). Equation-2 describes DSSnet virtual clock time that is the sum of emulation time ($t_{E,i}$) and the returned simulation virtual times ($t_{S`,i}$). Also, the value returned in Equation-3 can be one of the following cases:

a. (1-∞) if power simulator takes longer time to execute than real time, therefore, emulation virtual time becomes essential for synchronization of both the systems

b. (0-1) if power simulation system takes lesser or equal time for execution than the real time, it means that the system has real time simulation capability

c. 0 if the power system simulation time is not considered by the emulation, for example, computing the voltages and currents at speed of light

Every synchronization event occurs when there is influence of one on the other through coordinator. The optimization approach has been used in order to speed up the overall process timing. Some of the events that occur are dependent on previous output while others are not. Therefore, the main events queue has been divided into two different queues of blocking and non-blocking events. IEDs are able to pause the emulation system by sending synchronization messages and these messages may land in blocking queue. Other process using non-blocking queue can speed up the process of emulation, and hence the overall execution time of system.

6. Results and System Evaluation

6.1. Network Emulation and Virtual Time

DSSnet system mainly relies on the timing provided by virtual clock of system. There is a strict requirement of synchronization of events at both power and network coordinator ends. The process of synchronization requires virtual time to pause for specific intervals in order to maintain simulation and emulation synchronization. This virtual time system causes overhead in network emulation. Overhead in any network is not tolerable beyond certain specification as it causes network operations to slow down, especially if size of the network increases to hundreds to thousands of nodes that are emulated on a single machine [15]. System overhead was tested by pausing emulation hosts (IEDs) again and again. Number of hosts were changed from 10 to 500 in Mininet and for each number of hosts, emulation process was paused to 1000 times repeatedly. Cumulative Emulation overhead calculation for different hosts for the pausing operations have been plotted in Figure 3. From the figure, it has been noted that more than 90 percent operations take less than 100 milliseconds in 500 host case.

For, all other cases of lesser hosts more than 80 percent of operations took less than 50 milliseconds. It is also obvious from Figure 4 that average time for overhead increases almost linearly with the increase in number of hosts in emulation.

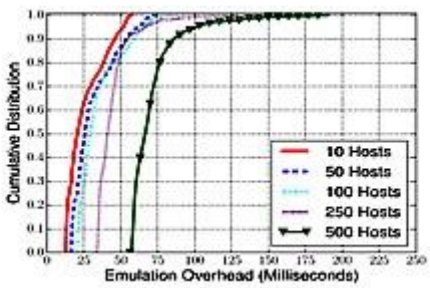

Fig. 3. Cumulative distribution of network emulation overhead by the operation of pausing virtual clock

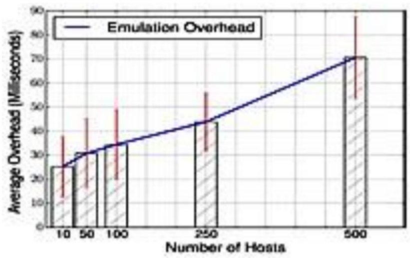

Fig. 4. Average increase in emulation overhead time with increase in number of hosts for network emulation

6.2. Accuracy Evaluation

Performance of every network is determined by two network flow characteristics known as throughput and latency from one end to the other end. To analyze the performance of network, two hosts connected with Open vSwitch were emulated with the link speed of 800Mbps and 10µs latency. Work of [20] was used to measure throughput and [21] was used to measure Round Trip Time (RTT) between the two hosts.

6.2.1 End-to-End Throughput

For testing throughput, *ipref* (a tool used to measure maximum achievable bandwidth of IP networks) was used to transfer data on TCP connection for the period of 30 seconds. For the first emulation purpose, system was tested without freezing while in the second run, the system was frozen for 1 second and the operation was repeated after every second for 64 times. The results of both the emulations were combined and it was observed that pausing the emulation causes 11% to 18% deviation from the baseline results. The results are depicted in Figure 5. The results may seem unsatisfactory in the first look, but these are very much dependent upon the OS scheduler. Just like throughput, same case for the evaluation of latency was adopted where 1000 pings were issued without freezing emulation first and then with a freeze duration of 0.1 seconds. First ping result was not considered for plotting to nullify the effect of ARP. Results are described in Figure 6, and it is observed that about 80% of the ping packets are received at around 0.2ms interval.

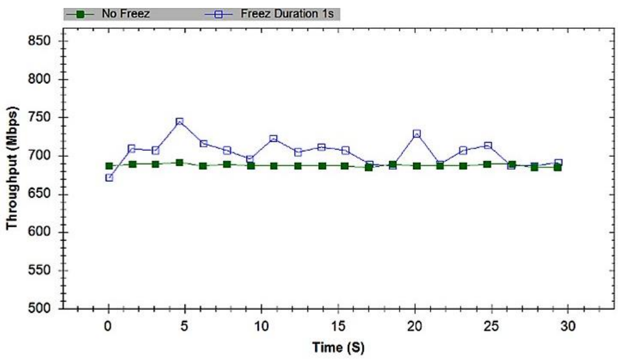

Fig. 5. Throughput comparison with freeze and no freeze emulation showing deviation

6.2.2 Accuracy Evaluation

Also, from the graph of freeze time interval it is obvious that freezing the emulation does not affect the overall end to end latency of the system in a prominent manner.

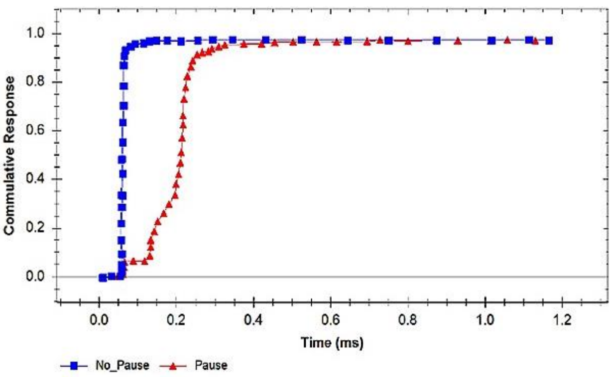

Fig. 6. Latency comparison with pause and not pause cases

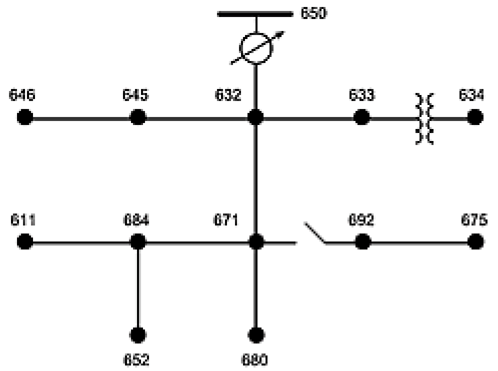

Fig. 8. IEEE 13 bus power distribution micro-grid model [22]

7. Demand Response Case Study

DSSnet has been designed for testing smart grid applications that apply to both power grid and communication network. A case study to analyze the behavior of micro-grid system using DSSnet and SDN enabled environment has been added in paper for better understanding of overall demand response of micro-grid.

7.1. Experimental Setup

A micro-grid model implemented in [22] using IEEE 13-bus system was modeled in OpenDSS shown in Figure 7. Base voltage of the system is set to 2.4kV and common coupling point of main power is at bus 650. A three-phase unbalanced with distributed energy source components has been added with sensors. Renewable energy generator in the form of wind turbine is added with bus 634 that has variable power output. The controller sends control messages and process the information to stabilize the system voltage. The control center computes each phase balance and sends messages to energy storage device. Model of communication network has been designed in ONOs on top of Mininet. The reason for using ONOS in simulation is that ONOS provide a Graphical User Interface (GUI) to give better understanding of topology of hosts and switches as well as overall network. Further, traffic flows and metrics can also be visualized in ONOS. One switch has been modeled for each bus and two additional switches for backup and multipath forwarding. Control plane is composed of three different ONOS instances where the cluster is overall connected to the communication network of micro-grid with a switch. Figure 8 describes exact state of the communication network topology.

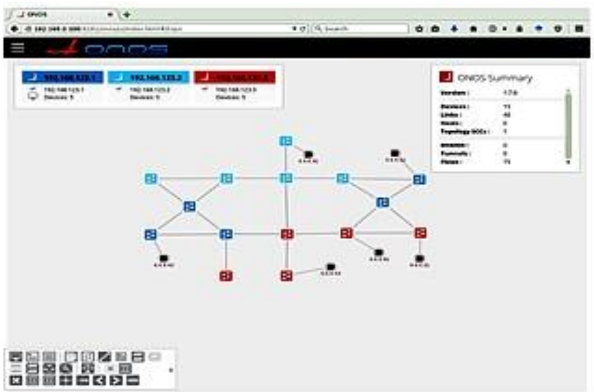

Fig. 7. ONOS Communication Network Topology on web-based GUI interface [22]

7.2. Experimental Results

The results extracted from [22] for single phase voltages at bus 652 and 675 are represented in Figure 9 for analysis. The demand response application suggests that voltage stability of actual power system does not change largely from simulated micro-grid. It is also important to mention that simulation system voltage response does not deviate more than 5% of the average value of actual system depicting that system is stable. These results are another motivation for evaluating and designing SDN based solutions for grid systems as well as communication network also. To observe the behavior of network system during link failure, primary link of the control center and energy storage location was broken manually at t-3.2s.

With global visibility of SDN enabled environment, system responds spontaneously over link failure of within negligible time (of the order of milliseconds). This self-healing topology configuration causes minimal deviation from the base case. In normal circumstances, where centralized view is not available, system convergence from a link failure takes time of tens of seconds. This delay due to link failure brings severe outage in the network system that causes power system to reach unstable state. But with the SDN enabled applications provided by ONOS, the effect of link failure is significantly reduced.

8. Conclusions

This article provides DSSnet as a testing platform that combines power system simulator and SDN based network emulator. It can be used to model power flows, communication networks, and smart grid applications. It can also be used for the evaluation of network applications for smart grid systems. DSSnet system can be used for planning and evaluation of more complicated applications with greater number of hosts and controllers before introducing it to the real systems.

9. Critical Analysis and Future Work

DSSnet, a system for combining electrical power simulations and network emulations in SDN environment, was tested and evaluated in this work. Overall characteristics and performance of system stands out to be good but still there are some limitations of the system that need to be addressed before implementing for the real time hardware applications. The first limitation stands out to be virtual time environment that is not possible for real hardware cases where clock time of the hardware cannot be stopped for synchronization with power system.

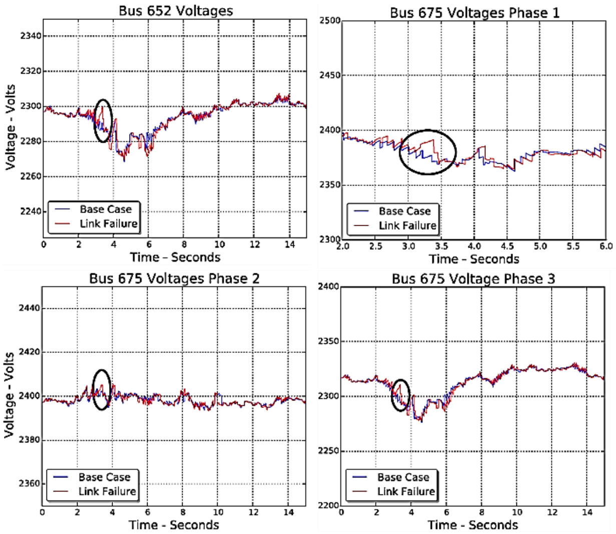

Fig. 9. Voltage response at bus 652 and 675 with link failure at t=3.2s. ONOS central controller with global view provides fast recovery from link failure state [22]

Secondly, the system has been tested for very limited number of host devices. As it was mentioned in Figure 7, the system was tested for maximum number of 500 hosts and there was an average overhead delay of about 70 milliseconds. This 70 milliseconds overhead delay may seem negligible for limited number of hosts but when real time system are modeled, there can be thousands of devices and that can be either Intelligent Electronic Devices (IEDs) or not. Further, their state might change continuously. It is also visible from Figure 7 that overhead delay keeps on increasing, although, sort of linear with increased number of hosts. Therefore, for more than hundreds of thousands of hosts, this delay might increase to the order of seconds to minutes that is unbearable. It may cause the system to an unstable state which is not affordable in practical or production environment.

One more limitation to the system is that no security mechanism for the power grid applications have been provided. With the help of real time network traffic obtained from DSSnet, data injection attacks must be studied that target power grid applications. There is a need to develop SDN based solutions to prevent intrusion and denial of service attacks. SDN can be helpful in finding vulnerabilities such as controller failure or any cyber-attack on the system for smart grid security, resilience, and intrusion detection. The above-mentioned limitations of the system can be used as reference for future research work and improvements in the design

References

[1] S. Kaplan, "Smart Grid. Electrical Power Transmission: Background and Policy Issues," *The Capital.Net, Government Series*, 2009, pp. 1-42.

[2] Open Networking Foundation, "Software Defined Networking," 2020, https://www.opennetworking.org

[3] D. Montenegro, M. Hernandez, G. Ramos, "Real Time OpenDSS framework for distribution systems simulation and analysis in *6th IEEE/PES Transmission and Distribution: Latin America Conference and Exposition*, 2012, pp. 1-5.

[4] C. Dufour, J. Belanger, "On the use of real-time simulation technology in smart grid research and development," *IEEE Transaction on Industry Applications*, 2014, vol. 50, no. 6, pp. 3963-3970.

[5] H. Lin, S. Veda, S. Shukla, L. Mili, J. Throp, "GECO: Global event-driven cosimulation framework for interconnected power system and communication network," *IEEE Transactions on Smart Grid*, 2012, vol. 3, no. 3, pp. 1444-1456.

[6] K. Hopkinson, X. Wang, R. Giovanini, J. Thorp, K. Birman, D. Coury, "EPOCHS: A platform for agent-based electronic power and communicaion simulation built from commercial off-the-shelf components," *IEEE transactions on Power System*, 2006, vol. 21, no. 2, pp. 548-558.

[7] S. Ciraci, J. Daily, J. Fuller, A. Fisher, L. Marinovici, K. Agarwal, "FNCS: A framework for power system and communication networks co-simulation," *In Proceedings of the Symposium on Theory of Modeling & Simulation – DEVS Integrative, Society for Computer Simulation International, San Diego, CA*, 2014, Article 36.

[8] S. Ciraci, J. Daily, K. Agarwal, J. Fuller, L. Marinovici, A. Fisher, "Synchronization algorithms for co-simulation of power grid and communication network," *IEEE 22nd International Symposium on Modelling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2014, pp. 355-364.

[9] L. Ren, Y. Qin, B. Wang, P. Zhang, P. Luh, R. Jin, "Enabling resilient micro-grid through programmable network," *IEEETransactions on Smart Grid*, 2017, vol. 8, no. 6, pp. 2826-2836.

[10] Y. Kim, K. He, M. Thottan, J. Deshpande, "Virtualized and self-configurable utility communications enabled by software-defined-networks," *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2014, pp. 416-421.

[11] X. Dong, H. Lin, R. Tan, R. Lyer, Z. Kalbarczyk, "Software-Defined networking for smart grid resilience: Opportunities and challenges," *In Proceedings of the 1st ACM Workshop on Cyber-Physical System Security (CPSS) ACM New York*, 2015, pp. 61-68.

[12] A. Goodney, S. Kumar, A. Ravi, Y. Cho, "Efficient PMU networking with software defined networks," *In IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2013, pp. 378-383.

[13] A. Sydney, D. Ochs, C. Scoglio, D. Gruenbacher, R. Miller, " Using GENI for experimental evaluation of software defined networking in smart grid," *Computer Networks 63*, 2014, pp. 5-16.

[14] R. Dugan, Reference guide, "The open distribution system simulator" 2013.

[15] B. Lantz, B. Heller, N. McKeown, "A Network in a Laptop: Rapid Prototyping for Software Defined Networks", *In Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, Hotnet-XI, New York, NY, USA*, ACM, 2010, pp. 19:1-19:6.

[16] Y. Zheng, D. Jin, D. M. Nicol, "Impacts of appliction look ahead on distributed network emulation", *In the proceedings of the 2013 Winter Simulation Conference (WSC)*, 2013, pp. 2996-3007.

[17] J. Lamps, D. M. Nicol, M. Caesar, "Timekeeper: A light weight virtual time system for linux", *In the proceedings of the 2nd ACM SIGSIM Conference on*

Principles of Adanced Discrete Simulation, SIGSIM PADS'14, New York, NY, USA, ACM, 2014, pp. 179-186.

[18] J. Yan, D. Jin, "A virtual time system for Linux-container-based emulation of software defined networks", *In proceedings of the 3rd ACM SIGSIM Conference on Principles of Advanced Discrete Simulation, SIGSIM PADS'15, New York, NY, USA,* ACM, 2015, pp. 235-246.

[19] J. Yan, D. Jin, "Vt-mininet: Virtual-time-enabled mininet for scalable and accurate software defined network emulation", *In proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research, SOSR'15, New York, NY, USA*, ACM, 2015, pp. 27:1-27:7.

[20] M. Gates, A. Warshavsky, ipref3, Reterived from http://software.es.net/iperf

[21] Y. Hideaki, iputils, Retrieved from http://www.skbuff.net/iputils

[22] C. Hanon, J. Yan, D. Jin, C. Chen, J. Wang, "Combining simulation and emulation systems for smart grid planning and evaluation", *ACM Transaction on Modelling and Computer Simulation 28, 4, Article 27,*Aug. 2018.