Communication Network Modeling For Simulation Of Wide Area Monitoring And Control Applications In Power Systems

by

SARANGA D. EDIRISINGHE DISSANAYAKE TENNAKOON MUDIYANSELAGE

A Thesis
submitted to the Faculty of Graduate Studies of The University of Manitoba,
in Partial Fulfillment of the Requirements for the degree of

Master of Science

Department of Electrical and Computer Engineering
University of Manitoba
Winnipeg, Manitoba R3T 5V6 Canada

Abstract

This thesis has mainly focused on investigating the effect of communication network on the power system operation. The main objective of this research has been to develop a set of communication network simulation tools and verify their suitability for realistic co-simulation of a power system and an associated data-communication network within a power system simulation environment. Based on a background study, a set of communication components have been developed for the PSCAD/EMTDC power system simulation software, which can simulate communication delay and packet losses. Furthermore, an analytical method based on queuing theory has also been developed to evaluate the communication delay and packet loss probability of a typical PMU-PDC communication network. Finally, the communication components developed in this thesis have been integrated into the simulation of a wide area power system application to investigate the effect of communication network parameters on the power system operation.

Acknowledgements

First and foremost I would like to express my heartiest gratitude to my advisor, Prof. Pradeepa Yahampath, for his excellent guidance, encouragement and unreserved help during my studies. His office door was always opened for me for discussions. Also, my sincere thanks goes to my co-advisor, Prof. Athula Rajapakse, for his invaluable assistance and support. My special thank also should go to other faculty members, specially to Prof. A. Alfa and Prof. M. Pawlak for their help in improving my knowledge in numerous ways.

I take this opportunity to thank my scholarship donor, Manitoba hydro for facilitating me financially to carry out my M.Sc.

Deepest gratitude are also due to the administrative staff of the Faculty of Graduate Studies and the Department of Electrical and Computer Engineering. Specially to our graduate student advisor, Ms. Amy Dario for all the help provided. A big thank you goes to all my colleagues in the CNER lab and those who supported me to complete the Chapter 5 of this thesis, specially to Mr. Dinesh Gurusinghe and Ms. Anupama Konara.

I am also greatly indebted to my sister and my beloved husband for their dedication, scarification, love and support given to me during my studies. Finally, my parents. They have been an inspiration throughout my life, and always supported my dreams and aspirations. I would like to thank them for their endless love and all they have done for me. I would like to dedicate this thesis to my beloved parents.

Contents

List of Tables

List of Figures

Acronyms

DTMC	Discrete Time Markov Chain
EMTDC	Electro Magnetic Transient Including Direct Current
FIFO	First In First Out
HVDC	High Voltage Direct Current
IED	Intelligent Electronic Device
LAN	Local Area Network
MAC	Media Access Control
PDC	Phasor Data Concentrator
PMU	Phasor Measurement Unit
PSCAD	Power System Computer Aided Design
PSS	Power System Stabilizer
RTU	Remote Terminal Unit
TCP/IP	Transmission Control Protocol /Internet Protocol
UDP/IP	User Datagram Protocol /IP
WAMPaC	Wide Area Monitoring, Protection and Control System
WAN	Wide Area Network

Chapter 1

Introduction

1.1 Overview

Wide area power system monitoring, control, and protection systems based on synchrophasor networks are currently emerging, and are being widely studied by electric utilities around the world. The traditional communication systems used for wide area control and monitoring in power systems are based on point-to-point communication links [1]. At present there exist no modeling tools in power system simulation software for realistically simulating the data network that is used to exchange the information between various devices in the system. Currently, most power system studies [2] either assume ideal communication links or use ad-hoc approaches (such as using stand-alone data network simulators for analyzing the communication aspects) and subsequently importing the results into the power system simulation. As most of the stability control and protection techniques require operation of the relays within less than 0.5 ms of an event, including the delay in the communication network, off-line simulations can be incorrect due to the integration of the results of two stand alone simulations and it can lead to errors in synchronization of the results [3]. Furthermore, ad-hoc approaches become intractable in Monte-Carlo simulations which are essential in analyzing the effects of random events in a power system. As more complex PMU networks evolve, the network size also grows. As a result, the study of such systems will require data network models which can be integrated into power system simulations.

Modeling and simulation of communication networks have been widely studied in recent years, particularly in relation to local area networks, wide-area networks, and more broadly the Internet [4], [5]. An industry standard software package for communication network simulation is the OPNET Modeler [6]. Most of the basic analytical tools, which are based on queuing theoretic models, are also applicable to control and monitoring applications in power systems. However, the behavior and the requirements of sources and sinks of data in such applications are generally different to those on the Internet (where most of the traffic consists of web-pages and multi-media). A study of data-traffic modeling in power system control applications is presented in [7]. Research into modeling and simulation of high-speed communication-based control, protection, and monitoring is currently emerging. A very recent work in this area can be found in [8], where OPNET Modeler [6] has been used to simulate a large PMU network in Nordic countries. The goal of that study has been to determine the requirements of transmission system operators, as well as the network delays and packet losses in a network of a large number of PMUs. However, in [8] the power system and the data network are not simulated as a single system. The goal of the research carried out in this thesis is to simulate both the power system and the associated data communication network in an integrated manner, so as to identify and characterize the impact of the data network on the operation of wide area protection and control applications.

The aspects of a communication link critical to power system protection applications are the packet delay (or latency), reliability, and network traffic characteristics. All of these are random functions of time. In a packet-based network, reliability is related to the packet loss probability. Both delay and packet-loss probability depend on congestion status of the network switches and channel bit-error rate (with high error rates, retransmission of critical data may be required). In time-critical applications, data packets which do not arrive within a time-out period contribute to packet losses. Another related issue is the maximum usable data-rate, which is directly determined by the bandwidth of the communication link. The maximum data rate directly impacts the communication delay. In general, the delay in a communication link is made-up of several components:

- Multiplexing and transition delay which occur during the exchange of data between a communication link and the data processing equipment
- Propagation delay which is determined by physical channel characteristics (e.g., optical fiber, co-axial cable, or wireless), and the associated physical distances
- Queuing delay which is due to waiting in queues in network components with buffers (e.g., switches, routers, etc)

This thesis will investigate these issues in relation to PMU networks so that they can be appropriately represented in the models as explained in Chapter 3.

1.2 Wide Area Power System

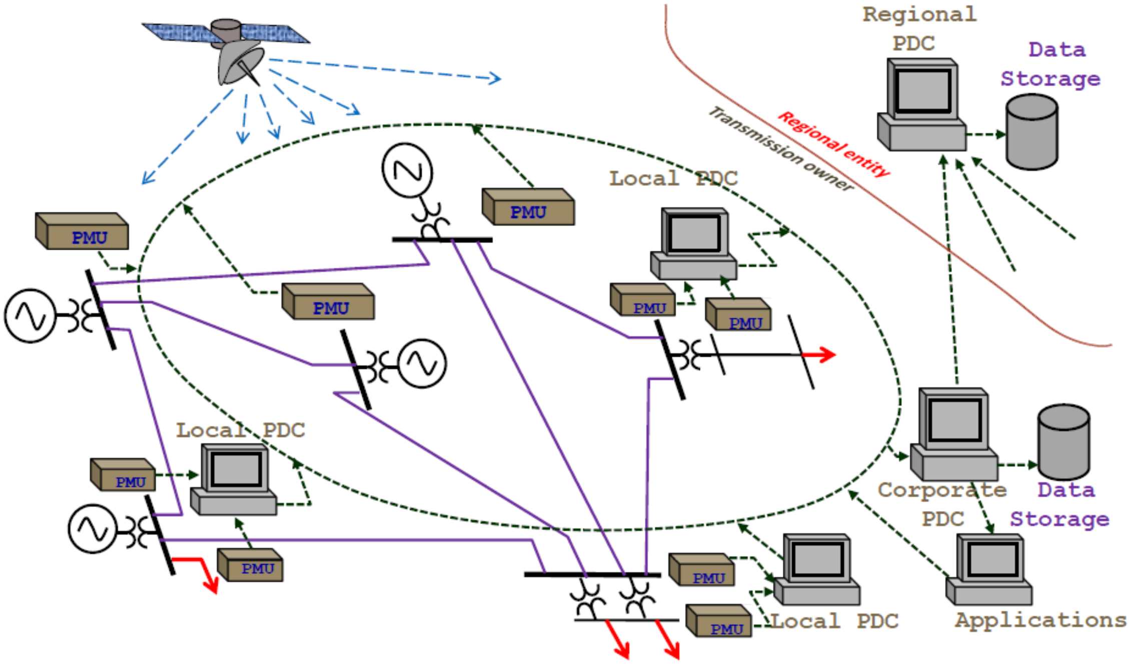


Figure 1.2.1: Typical wide area power system.

Recent advancements in communication technology and the installation of synchrophasor networks will lead to the fully automated monitoring, control and protection schemes in power systems. The main components of a wide area power system are the PMUs, a phasor data concentrator (PDC), and the associated communication network. Fig. 1.2.1 shows a typical wide area power system with main components.

Synchrophasor technology has improved the visualization of the state of a power system by providing a wide area view of the system. Synchropahsors provide a time tag with each measurement. Therefore, with synchronized measurements, we can obtain a true snap shot of the power system at a particular time instant. If the phasor measurement rate is sufficiently high, such as 60 measurements per second, it allows one to observe the dynamic operation of the system. The phase angle measurements can provide an idea about the power flow and the stability of a power system [9]. The, power utilities around the world are using phasor data for new protection and control applications [10]. In the beginning, the synchrophasor data were largely used for wide area monitoring, but nowadays the emerging interest is in protection and control applications. Hence, closed-loop control systems can be implemented using phase angle measurements, i.e., fully automated control systems [11]. The performance of the communication system is critical to the correct operation of closed-loop control in a wide area power system.

The Fig. 1.2.1 shows the hierarchical view of the synchrophasor network including the communication system. The PMUs in the substation are connected to local PDCs using a local area network (LAN) and these local PDCs are connected to a wide area network (WAN). On the other hand, some PMUs are directly connected to the WAN as well. The corporate PDC gathers phasor measurements from the WAN and sends to applications for data storage. The PDC can also send phasor measurements to regional or super PDCs. The global positioning system (GPS) is used to synchronize the phasor measurements to a common time base.

1.2.1 Phasor Measurement Unit

A PMU measures the state variables at certain nodes in a power system, and the state variables of other correlated nodes can be calculated with such PMU measurements. Therefore, the local area power system is observable with these PMU measurements [9]. The main idea behind the PMU based measurement is the projection of point-on-wave voltage and current signals onto a set of reference waveforms, to extract magnitude and phasor angle of voltage and current signals.

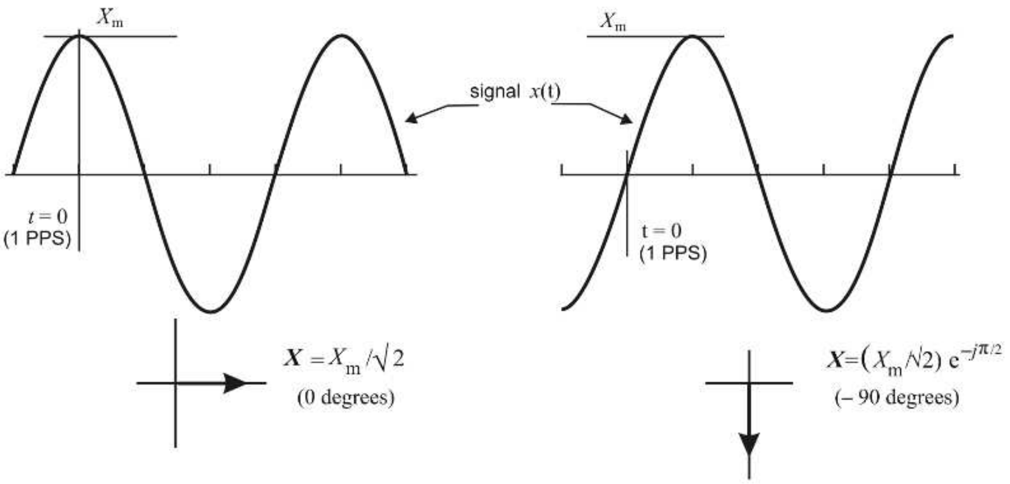


Figure 1.2.2: Convention for synchrophasor representation [13].

$$x(t) = X_m cos(\omega_0 t + \phi) = X_m cos(2\pi f_0 t + \phi) \tag{1.2.1}$$

where f_0 is the nominal angular system frequency (50 Hz or 60 Hz). In the general case frequency of the sinusoidal waveform $x(t)$ is also a function of time ($f(t)$). Then, the difference between the actual and nominal frequencies can be written as

$$g(t) = f(t) - f_0. \tag{1.2.2}$$

Therefore, the sinusoid $x(t)$ can then be written as [13]

$$\begin{aligned} x(t) &= X_m(t)cos(2\pi\int fdt+\phi) \\ &= X_m(t)cos(2\pi\int(f_0+g)dt+\phi) \\ &= X_m(t)cos(2\pi f_0 t+(2\pi\int gdt+\phi)) \end{aligned} \quad (1.2.3)$$

The syncrophasor representation for the waveform in Fig. 1.2.2 is shown in 1.2.4 [13].

$$X(t) = (X_m(t)/\sqrt{2})e^{(2\pi\int gdt+\phi)} \quad (1.2.4)$$

For the special case where $X_m(t)$ is constant and $g = \Delta f$ is a constant offset from the nominal frequency, the synchrophasor is shown in 1.2.5 [13].

$$\begin{aligned} \int g(t)dt &= \int \Delta f dt = \Delta f t \\ X(t) &= (X_m(t)/\sqrt{2})e^{(2\pi\Delta f t+\phi)}. \end{aligned} \quad (1.2.5)$$

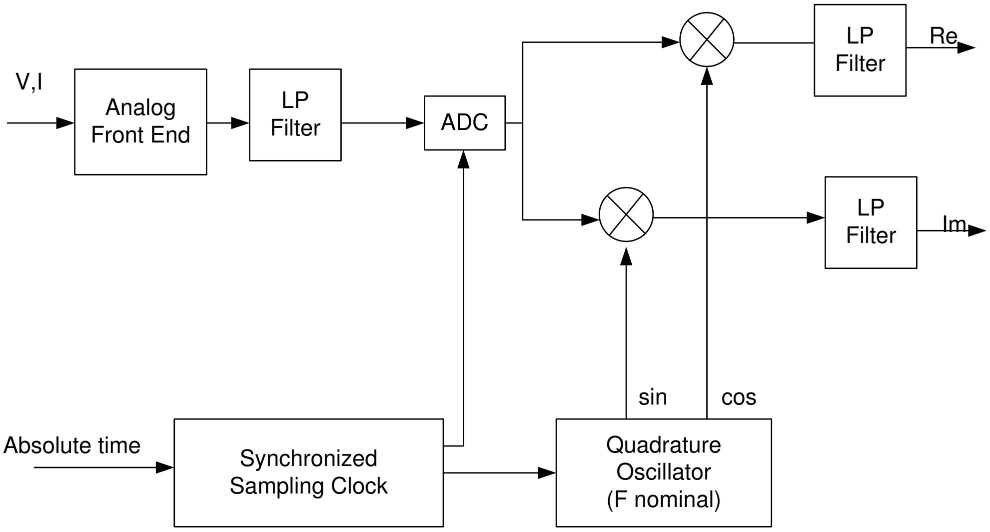


Figure 1.2.3: Block diagram of a PMU.

Fig. 1.2.3 shows the general structure of a PMU [13].

1.2.2 Phasor Data Concentrator

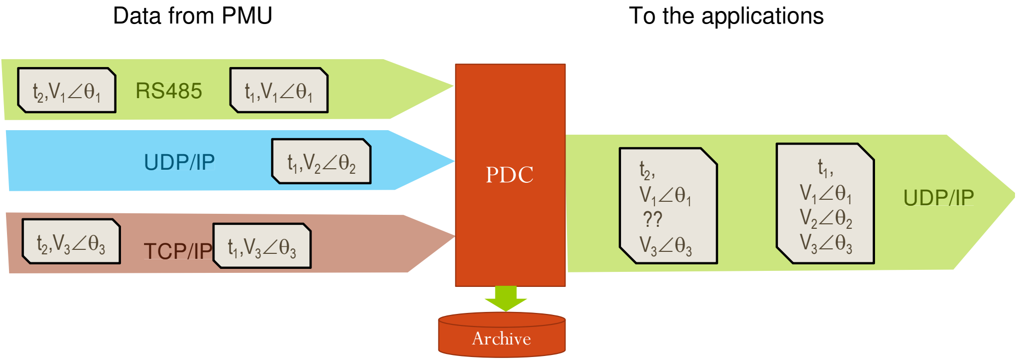


Figure 1.2.4: Block diagram of the PDC.

The most important function of a PDC is to collect the phasor measurements from a set of PMUs and align the measurements according to their GPS time stamp. If the PMU communication network was to experience abnormal delays or packet losses, some measurements can be lost. In such cases, the PDC performs quality checks by means of flags to indicate missing or corrupted data. Higher level PDCs keep an archive for offline and historical data analysis. The IEEE standard for synchropahsors [13] does not specify the protocol to be used for the communication between PMUs and PDC. Widely used communication protocols are TCP/IP, UDP/IP or serial communication, i.e., RS485. Fig. 1.2.4 shows the block diagram of the PDC with it’s functionalities.

1.2.3 Data Communication Network

A communication network is an essential component in wide area power system applications, because the loss of communication would leave the application with insufficient information to carry out its operations. For example, in a wide area control system, PMUs send phasor measurements to the PDC and the PDC forwards them to a wide area controller (WAC) for control actions. Then, the WAC sends control commands to the actuators, such as circuit breakers or generator controllers. That means, two way communication is required in such a WAN. Typically, the data network used by the control application may be

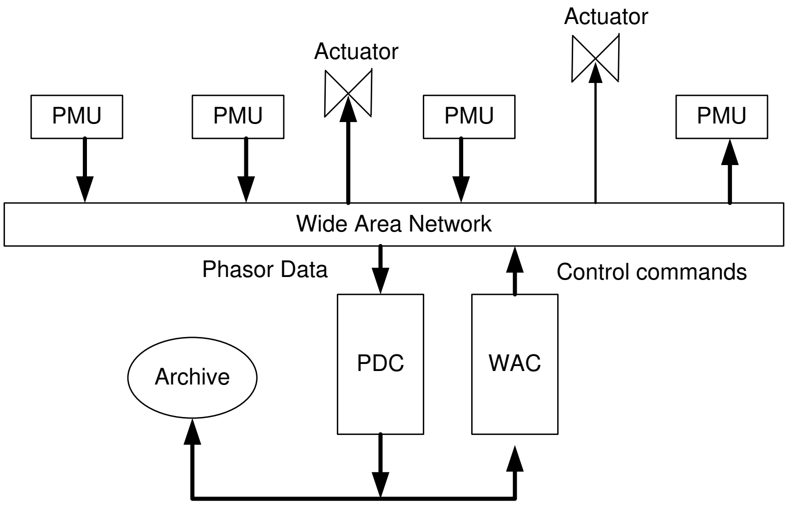


Figure 1.2.5: Wide area control with communication.

a part of the corporate data network of the power utility. Such a corporate network may carry other user traffic (e.g., web pages, e-mail). Hence, the behavior of the wide area control system can be highly unpredictable, and it is important to model and simulate the coupled data communication network to study the effect of network traffic characteristics on the power system operation.

1.3 Motivation and Objectives

At present, software for co-simulation of a power system and an associated data network do not exist. The main goal of this research has been to develop data network models which can be incorporated in to a power system simulation environment. These models should be able to simulate the realistic communication scenarios such as random delays and packet losses. The specific goals of this research are: (1) to develop a basic set of blocks which can be used to simulate a simple data network, (2) to verify the models by comparing the simulation results with results obtained by a queuing theoretic analysis of the data network, and (3) to demonstrate the usefulness of the developed software models by simulating a complete communication based power system application.

1.4 Methodology and Contribution

In carrying out this research, two key issues were identified.

- Modeling of a communication system within a power system.

  Network data traffic characteristics in power system applications are much different to those on the Internet because in the Internet the content is mainly web pages, e-mail or streaming multimedia. Towards this end, the goal was to investigate the statistical modeling of network data flow in a PMU network and thereby come up with appropriate models for communication components.

- Simulation of a communication network on a power system simulation software platform.

  In network simulation software such as OPNET, OMNET++ or network simulator 2 (NS2), there exist virtual machines representing various communication devices which transmit bits and data packets [6]. Our case is somewhat different from the situation in network simulation software, in that we have to simultaneously deal with both electrical signal such as currents and voltages that exist in the power system, and bit streams and data packets in the communication system. Therefore, an important goal has been to investigate how these two types can be integrated in a $co-simulation$.

Motivated by the PMU network currently in use at the Manitoba Hydro, this research has focused on an Ethernet-based transmission control protocol (TCP) /Internet protocol (IP) local area network which interconnects PMUs and a PDC. Mathematical models which accurately reflect the aforementioned data network characteristics under typical operating conditions have been investigated and modeled. Towards this end, a comprehensive study of TCP/IP protocol and PMU packet formats, as well as modeling co-axial-based Ethernet have been undertaken. The models have been developed and implemented in software (e.g., using FORTRAN programing language in PSCAD/EMTDC).

An analytical model was developed for a typical PMU-PDC communication network in which PMUs are connected to a central PDC in dedicated communication network. This analytical model can be used to determine the performance indicators such as packet-loss probability and communication delay. The analytical model was based on queuing theoretic considerations, Markov modeling, and cyclic-polling. The performance indicators predicted by the analytical model were then compared with those obtained by the simulation model. It was verified that simulation results closely agreed with the analytical counterparts.

1.5 Outline Of The Thesis

This thesis is organized as follows: Chapter 2 contains a literature review of the communication requirements of wide area power system applications and the methods that have been used in previous work to investigate the effect of data network on the operation of power system applications. In the same chapter, those schemes are compared with the approach proposed in this thesis. Chapter 3 presents an overview of the modeling of communication component in PSCAD/EMTDC platform, and the algorithms used to build those components. Chapter 4 presents queuing theoretic analytical model which uses cyclic polling method to evaluate the communication delay and packet loss probability of the PMU-PDC communication network. Chapter 5 presents the simulation results obtained by co-simulation of a simple wide area power system and the associated data network in PSCAD environment. Chapter 6 concludes the thesis and provides suggestions for the future work. Appendix A presents the FORTRAN codes of the new communication components developed in PSCAD/EMTDC. Appendix B presents the PSCAD/EMTDC model of the co-simulation.

Chapter 2

Survey of Related Work

2.1 Introduction

It is important to develop new communication technologies which can make power system operation more effective [16, 17]. A wide area power system is an example for the use of new communication technologies in applications such as protection and control. Before implementing new communication techniques in an existing power system realistically, it is required to perform a simulation studies to investigate the effect of the new technology on the power system operation. Also, before making any changes to the existing communication network or attempting performance improvements of the existing communication network by adding more bandwidth, one should always evaluate the effect of any proposed modifications. For these purposes, co-simulation of a power system and the underlying communication system is important. In literature, previous work has investigated the communication needs of a power system, proposed new communication technologies and protocols, and evaluated the effects of communication system performance on power system applications.

2.2 Communication Requirements in Power System Applications

Before implementing a platform for co-simulation of a power system and its underlying communication network, it is important to investigate the communication needs and communication bottlenecks that can be critical in power system applications. Authors in [14] have studied several wide are power system applications and their communication requirements as shown in Table 2.1. Table 2.1 considers communication requirements in terms of the data volume, end-to-end (ETE) delay, and the communication architecture where, N is the number of buses in the high voltage system, and n is a constant which is equal to (N/3).

Table 2.1: Communication requirement for some wide area power system applications [14]

.

Task	Amount of data	Latency	Bi-directional communication
PMUs added to conventional estimators	nx38 bytes in 4-10 seconds	Not critical, up to a second OK	Not required.
All-PMUs estimators	l .5Nx38x30= 1710 N bytes per sec	Less than 100 ms	Not required.
Seams	Nx38 bytes in 4-10 seconds	Not critical, up to a second	Not required.
CT and CVT calibration for all-PMU estimators	20,520 N In 12 hours	Not critical	Not required.
Adaptive dependability-security	Less than 100 bytes	Not critical, up to a second OK	Required to acknowledge change.
Monitoring approach of apparent impedance	Less than 100 bytes	Not critical, up to a second OK	Not required.
Supervision of back-up zones	Less than 1000 Bytes	50 msec.	Acknowledgment of data received is useful.
Adaptive loss-of-field	Less than 100 bytes	50 msec.	Acknowledgment of data received is relay necessary.
Adaptive out-of-step	nx(38)x30 bytes/second	50 msec	Not needed.
Intelligent islanding	nx100 bytes	50 msec.	Acknowledgment of data received is necessary.
Intelligent load shedding	nxl00x30 bytes	50 msec.	Acknowledgment of data received is not necessary.
Control of inter-area oscillations	25x38x30+ 5x38x30=3420 bytes/sec	200ms msec.	Acknowledgment of data received is not necessary.
Control of transient events	50x38x30 =57,000 bytes/sec	50 msec.	Acknowledgment of data received is not necessary.

Communication delay and packet loss probability are the most important performance measures that we should consider in the WAMPaC [10,18]. An important study of communication traffic in a typical utility environment can be found in [18] which presents communication latency and packet loss probability of an existing PMU network at the Manitoba Hydro.

Ethernet is the most widely used communication medium in power system communication in substations [19]. Fiber optic and SONET are also used as communication media in backbone communication networks. Satellite links are used to communicate measurements from remote substations to the central control center [20].

In wide area power systems, PMUs and remote terminal units (RTU) are the main traffic generation sources in a substation [21]. However, there can also be other data sources such as video units, intelligent electronic devices (IED), etc.

Synchrophasor deployment results in the generation of a large amount of data traffic across the power system communication network. It may cause bottlenecks in the communication system leading to unacceptable delays and packet losses. In [22], authors have studied the underlaying reasons for communication bottlenecks and has proposed an enhanced architecture to address this issue. They have mainly discussed the requirements of communication between a set of PMUs and the PDC, and conclude that when a PDC is connected to a set of PMUs which generates 60 packets/s, the communication network requires a WAN with an enhanced bandwidth and low latency intermediate nodes to communicate PMU measurements within 16 ms. Therefore, [22] proposed a communication network architecture which uses Ethernet based communication, TCP/IP protocol and an enhanced PDC model to achieve the communication time of 16 ms. It is emphasized in [22] that regardless of the technology in use, the communication system must be optimized to meet the requirements of the power system control and protection applications. Finally, in [18] it has been shown that the bandwidth required for PMU packet transmission is generally negligible compared to the bandwidth required for the transmission of control and protection signals.

2.3 Existing Methods

Previous work on the performance evaluation of communication networks in wide area power systems, by using a stand alone network simulation software appears in [1, 2, 8, 21, 23]. In this work they have proposed improvements to the wide area power system applications based on the simulation results obtained using stand alone network simulation software. Typically, most widely used network simulation software in power system studies is the OPNET [1, 23]. Other examples include OMNET++ [21] and NS2. Authors in [21] present performance aspects of the IP network infrastructures when utilized by both continuous PMU data streams and critical intelligent electronic device (IED)/remote terminal unit (RTU) data. They have used OMNET++ to simulate the communication network and have examined the ETE delay between PMUs and PDC in a wide area power system. Specially, [23] and [8] present the results of simulations performed using OPNET Modeler in order to determine the characteristics of communication delays incurred in wide area monitoring and control systems (WAMCS) which use multiple PMUs distributed over a large geographic area. Such systems normally include one or several PDCs that collect and sort the data from the PMUs. The results in [23] and [8] indicate that the configuration of central nodes such as the PDC needs to be optimized for the intended WAMCS. Furthermore, [1] addresses the analysis of PMU systems and two communication architectures, i.e., dedicated and shared communication network scenarios. The OPNET Modeler has been used to implement and analyze the the ETE delay between PMUs and PDC connected by those two communication architectures. That work specifically focuses on the transmission of phasor data to the PDC and control signals from the WAMCS back to substations devices.

The work in [24] presents an equation to calculate the communication delay between a PMU and PDC. This equation can also be used to approximately evaluate the ETE delay between PMUs and PDC in a particular wide area power system. Then the results can be subsequently imported into the power system simulation software such as PSCAD/EMTDC, but this calculation may not be 100% accurate since the aforementioned equation does not capture the randomness of the communication network.

2.4 Co-simulation Methods

As wide area measurements and control are being developed, the performance of the communication system becomes important. Furthermore, communication delays and packet losses affects the power system dynamics. Therefore, extensive integration of power systems and its communication infrastructure mandates the importance of studying the two types of systems as a single system. Authors in [25] and [26] present two co-simulation methods referred to as EPOCHS approach and ADEVS (A discrete event system simulator) approach respectively. In EPOCHS approach, two simulation software are used, one for the power system simulation and the other for the communication network simulation. Then, an interfacing software is used to combine the outputs from the two simulations. On the other hand ADVES approach uses the concept that power system primarily consists of elements that are modeled by continuous equations and a communication system has discrete event dynamics. Therefore, integrated operation of these two systems are studied using a hybrid modeling and simulation technique. The paper [26], discusses an approach based on the discrete event system specification (DEVS) that characterizes the interaction of the two systems formally to preserve simulation accuracy. The work in [3] proposes a power system / data network co-simulation framework which integrates a power system dynamic simulator and a network simulator using a synchronization mechanism. It is an extension of the EPOCHS and ADVES approach. Authors have chosen positive sequence load flow (PSLF) as the power system simulator and NS2 as the network simulator. As a case study an agent-based remote backup relay system has been simulated and validated using the co-simulation framework.

In the above mentioned co-simulation techniques, they have used two separate software platforms and techniques to simulate the power system and the communication network, subsequently combining the results of the two simulation using another third party software. However, the integration of the results of two stand alone simulations can lead to errors in synchronization of the results. In order to address this issue, the development of a single platform for co-simulation of both the power system and the data network is sought in this thesis.

Authors in [27] discuss the next generation communication requirements, technologies and architecture for the North America and Europe electric power grids. It says that, the power grid's existing communication architecture limits the control and protection schemes that can be implemented. In [18], the authors explain that if the utilities around the world use low bandwidth media such as microwave links to exchange PMU data, the number of connected PMUs should be planned properly in order to avoid packet losses and longer packet delays. In such scenarios, it is useful to run a co-simulation and evaluate the effect of the microwave links on the operation of the power system. A co-simulation will help to determine whether the PDC receives the PMU data over the microwave links before the PDC times-out and the loss probability of those PMU data packets.

The selection of a proper communication network topology, transmission media, and protocols is an important part in wide area monitoring, protection and control. The choices for network architecture, protocols etc., are hard to make without knowing the exact volume of data, requirements in latency, bandwidth and the quality of service, etc [2]. The goal of this research is to develop a software platform which can help make such decisions. The bandwidth/speed and the protocol can be easily modified according to the user requirements by changing user defined parameters in PSCAD communication components developed in this research.

In [28], authors propose a new WAMPaC application with a new communication routing algorithm, i.e., an adaptive PMU-based fault location estimation system with a fault-tolerance and load-balancing communication network. A co-simulation platform is needed to test this kind of new communication and power system applications together. The co-simulation platform developed in this research may also be used to verify the analytical method proposed in [17].

The packet losses and packet delay in the communication of the wide-area measurements systems affect the wide-area control of closed loop power system [10]. Therefore it is important design controllers of the closed loop control system, by considering the packet losses and time varying packet delays of the particular wide-area measurement system. A very recent work on this topic can be found in [10]. It presents a matrix inequality based method to design a controller for better performance of the wide area power system which can tolerate network induced delays and packet losses in the PMU-PDC communication network. In this study, they have assumed typical network delays and packet losses and they have used the two-area four-machine system [29] as the test case. However, the accuracy of such a study can be greatly improved by the co-simulation of the test case considered in [10].

Chapter 3

Communication Component Modeling in PSCAD/EMTDC

3.1 Introduction

This chapter presents the development of communication components for PSCAD/ EMTDC power system simulation software which enables the co-simulation of a power system and a communication system. PSCAD/EMTDC is power system simulation software which is widely used in industry and for academic purposes. It has important features such as user friendly graphical interface, a full library of advanced components, interactive control tools, the ability to create new components by the user and the ability to integrate with popular engineering simulation software such as MATLAB, etc. PSCAD can simulate power lines and cables, large non-linear industrial loads, transformers with saturation, power electronic systems and drives, asymmetrical faults, FACTS/HVDC systems, protection relay coordination, distributed power generation, rotating machines, embedded systems, etc [30–32]. PSCAD users can utilize the components in the master library, which are categorized in to 18 classes, to create the simulation cases. Furthermore, PSCAD allows the user to create new library components in addition to the master library.

PSCAD has been basically designed to simulate power systems. At present, it does

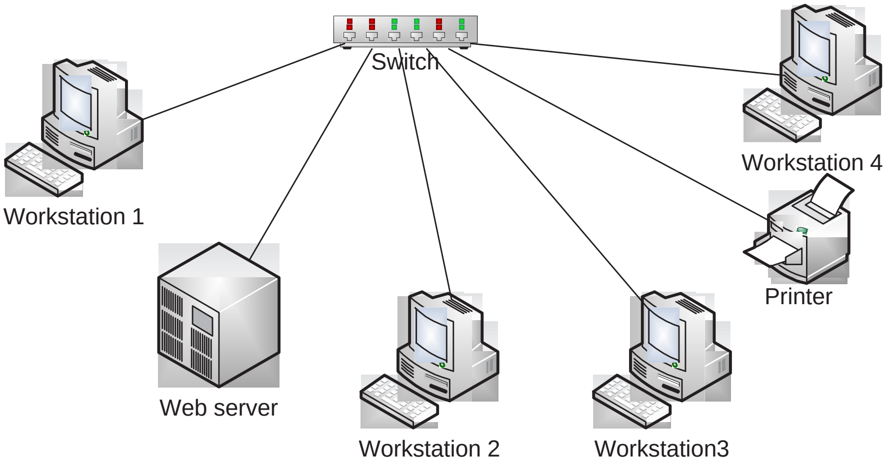


Figure 3.1.1: Typical data network.

not have any components which can be used to simulate data networks realistically. A typical communication network in which multiple data sources are connected to a switch using Ethernet cables is shown in Fig. 3.1.1. For simulation of such data network, it is required to have basic communication components such as cable links, switches and general data sources which can create packets at a data rate specified by the user. Such a set of components will allow the PSCAD user to easily simulate the effect of a data network on the operation of a power system application.

The main effects caused by the data network on the operation of power system applications can be in the form of packet delays and packet losses. Although, PSCAD has the capability to simulate the effect of constant delays by using the delay component, the delays associated with data communication networks are not constant and depend on various factors such as congestion in the network, cable length, the communication medium used (copper cables, fiber optic, satellite, microwave), etc. Therefore, it is important to simulate the random communication delays according to the network properties. On the other hand, packet loss can occur due to traffic congestion, switch buffer overflows and the network protocol. If the communication network is a shared network which uses carrier sense multiple access/ collision detection (CSMA/CD) protocol, the probability of loosing packets is high compared to the TCP/IP or user datagram protocol (UDP)/IP in the dedicated communication network [33].

3.2 Cable Model

3.2.1 Ethernet

Ethernet was developed in 1970’s by the Xerox Corporation in California. Ethernet is a family of computer networking technologies for local area networks (LANs). It was commercially introduced in 1980 and standardized in 1985 as IEEE 802.3. The Ethernet has largely replaced competing wired LAN technologies. The Ethernet standards comprise several wiring and signaling variants of the OSI physical layer [34] in use with the Ethernet. The original 10BASE5 Ethernet used coaxial cable as a shared medium. Later, the coaxial cables were replaced by twisted pairs and fiber optic links in conjunction with hubs or switches. The data rates were periodically increased from the original 10 megabits per second, to 100 gigabits per second [35].

Field	Preamble	Destination Address	Source Address	Type	Data	FCS
Bytes	8	6	6	2	46-1500	4

Figure 3.2.1: Ethernet frame.

If a system communicates over an Ethernet network, it divides a stream of data into shorter pieces called frames. Each frame contains information such as source and destination addresses and error-checking data, as shown in Fig. 3.2.1. Such information allows the detection of errors in data. Most of the power utilities around the world use Ethernet as the communication medium in the communication network which overlays the power system [36].

3.2.2 Bit Error Rate

In digital transmission, bit errors can occur due to channel noise, interference and due to synchronization errors. The bit error rate or the bit error ratio (BER) is the number of bit errors divided by the total number of transferred bits during a given time interval. The BER is a unit-less performance measure, often expressed as a percentage. It can be considered as an approximate estimate of the bit error probability. Basically there are two types of bit errors that can take place in communication channels.

- Random bit errors - bit errors which occur at random positions during the transmission
- Burst errors - a set of bit errors which occur sequentially during the transmission

3.2.3 PMU Data Packet Format

A communication system plays a vital role in wide area power system applications such as wide area monitoring, control, and protection [37]. Therefore, in this research it was decided to use a typical wide area power system application to investigate the effect of a data network on the power system operation using the PSCAD/EMTDC simulation platform. Hence, the PMUs are considered as the main data source. Therefore, it is important to consider the PMU data packet format.

This section presents the format of messages to and from a PMU or PDC for use in real-time communication of phasor data. If the PMU is to be used with other systems, to which the phasor information is to be transmitted in real time, implementation of this packet format is required for conformance with the standard IEEE C.37.118.2 [13]. The PDC identifies a PMU data frame of each PMU in the network, by using the IDCODE of the PMU and details such as the status, frame size, and second of century (SOC), etc, that are embedded in a data frame for the proper interpretation of PMU data. The real-time phasor data frame shall consist of binary data ordered as shown in Table 3.1 [13]. All fields have a fixed length and no delimiters shall be used. The frame starts with SYNC, FRAMESIZE, IDCODE, SOC and FRACSEC, and terminates with a CRC-CCITT. The PSCAD model for a cable link explained in 3.2.4, will accept data packets from a PMU, which has been created according to the specifications of the IEEE standard for the synchrophasors [13] and encapsulated in an Ethernet frame as shown in the Fig. 3.2.2.

Table 3.1: PMU data frame organization.

Number	Field	Size(Bytes)	Description
1	SYNC	2	frame type and version number
2	FRAMESIZE	2	number of bytes in the PMU data frame
3	IDCODE	2	PMU ID
4	SOC	4	time stamp
5	FRACSEC	4	fraction of second
6	STAT	2	Bit-map flags
7	PHASORS	4*PHNMR or 8*PHNMR	phasor data
8	FREQ	2/4	frequency
9	DFREQ	2/4	rate of change of frequency
10	ANALOG	2*ANNMR or 4*ANNMR	analog data
11	DIGITAL	2*DGNMR	digital data
12	CHK	2	CRC-CCITT

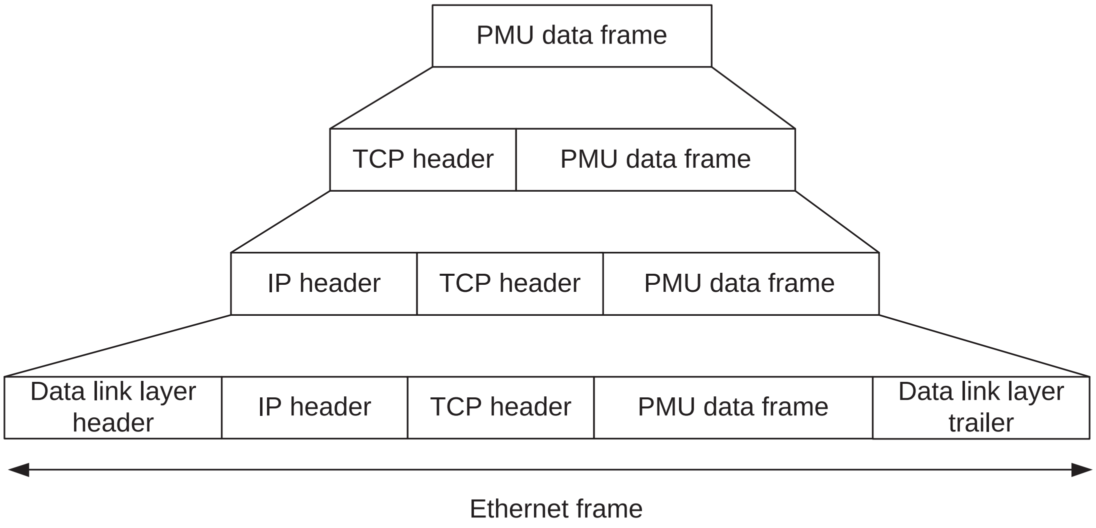


Figure 3.2.2: PMU data packet encapsulation in to an Ethernet frame.

3.2.4 The PSCAD Cable Link Model

The PSCAD model developed in this research for the cable link is shown in Fig. 3.2.3. A user can specify the cable length, the supporting data rate, the desired protocol to be used over the network (either TCP/IP or UDP/IP), the type of cable link (either Ethernet or fiber optic) and the bit error rate of the link. This PSCAD model is capable of simulating the effect of the packet delay due to factors such as the cable length and the time it takes to insert all the bits in an Ethernet frame to the cable. The packet delay in a cable link can be defined as

$$T = \text{propagation delay} + \frac{L}{R}, \tag{3.2.1}$$

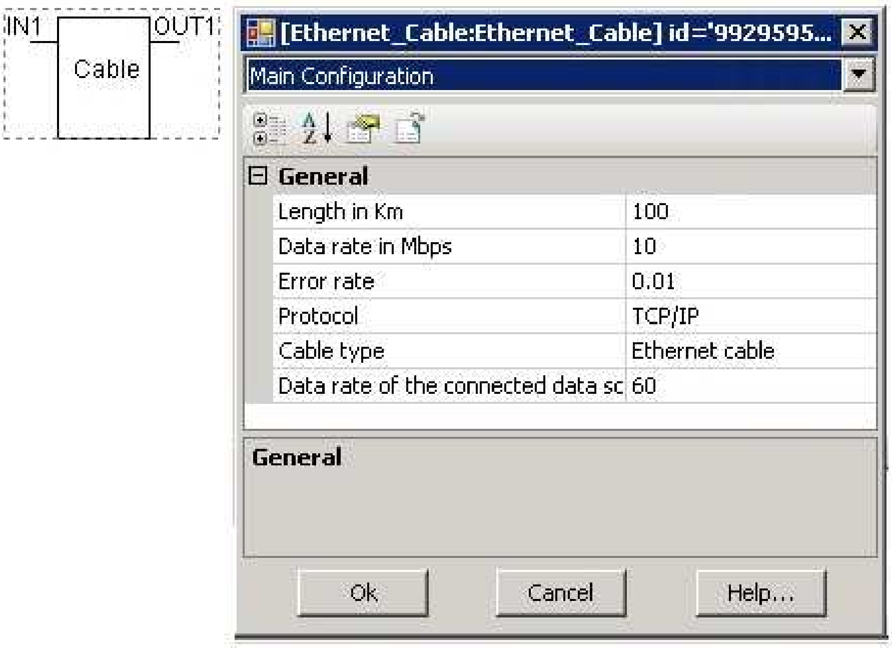
where L is the number of bits in the frame and R is data rate. The propagation delay is defined as the delay of bits propagating between the transmitter and the receiver [38]. The propagation delay will be calculated according the distance that user specifies. The value of L can be calculated as

$$\begin{aligned} L = {} & \text{bits in the PMU data message + bits in TCP/IP or UDP/IP header} \\ & \text{+ bits in IP header + bits in the data link layer header and trailer}. \end{aligned} \tag{3.2.2}$$

Furthermore, the model also simulates random bit errors as explained in section 3.2.2 according to the user specified bit error rate. Fig. 3.2.4 shows the steps used to model the cable link component and Algorithm 1 describes the algorithm used to implement the cable link in PSCAD/EMTDC.

Algorithm 1 Ethernet Cable Model

```
Input:
I: Input data array. i.e., PMU data frame    X: PMU data frame size
Y: STAT bit value
P: P=1, if user defined the protocol as TCP/IP, else P=0
E: Bit error probability
D: Length of the cable
R: Data rate
C: Cable type, either Ethernet or fiber optic
T1: Time that PMU data frame entered to the Ethernet cable
Add the relevant no of bits as header and trailer bits to X according to the value of the variable P and calculate the value of the L.
Calculate the no of bits to be erroneous according to the value of E i.e.  number of erroneous bits=L*E
for i= number of erroneous bits do
    Select a bit position
    Flip the bit
end for
IF STAT=15 (STAT=15 means new PMU data packet has arrived)
Calculate the latency(T). i.e., the summation of L/R and propagation delay according to D and C.
Store the output data array(A) .i.e the input data array (I) after the bit errors
Calculate the data output time(T2)=T+T1
IF simulation time equals the output time(T2), display the output data array(A)
Output: output data array(A) and output time(T2).
```



Length in Km	100
Data rate in Mbps	10
Error rate	0.01
Protocol	TCP/IP
Cable type	Ethernet cable
Data rate of the connected data sc	60

Figure 3.2.3: PSCAD cable link model.

3.3 Switch Model

3.3.1 Overview of a Switch

In a network, an increased amount of traffic can be generated as more computers are added. In such cases, switches can divide a large network into smaller sub networks without affecting the performance of the network. A switch works as a filter to control the usage of the devices on the network. It works to inspect the packets and data that is communicated on the network between the devices. A switch can benefit a network because it works to control the bandwidth of the network and helps to keep it stable.

There are two types of switches: a ”cut-through switch” and a ”store-and-forward” switch. A ”cut-through switch” forwards the packets before they have been received. This helps to reduce the amount of time that the switch takes to operate. However, this method is not very reliable. It has been known to have packets that have been corrupted, thus affecting the throughput of the network [39]. A ”store and forward” switch sends the packets to a station where they are held temporarily until needed. The information in a

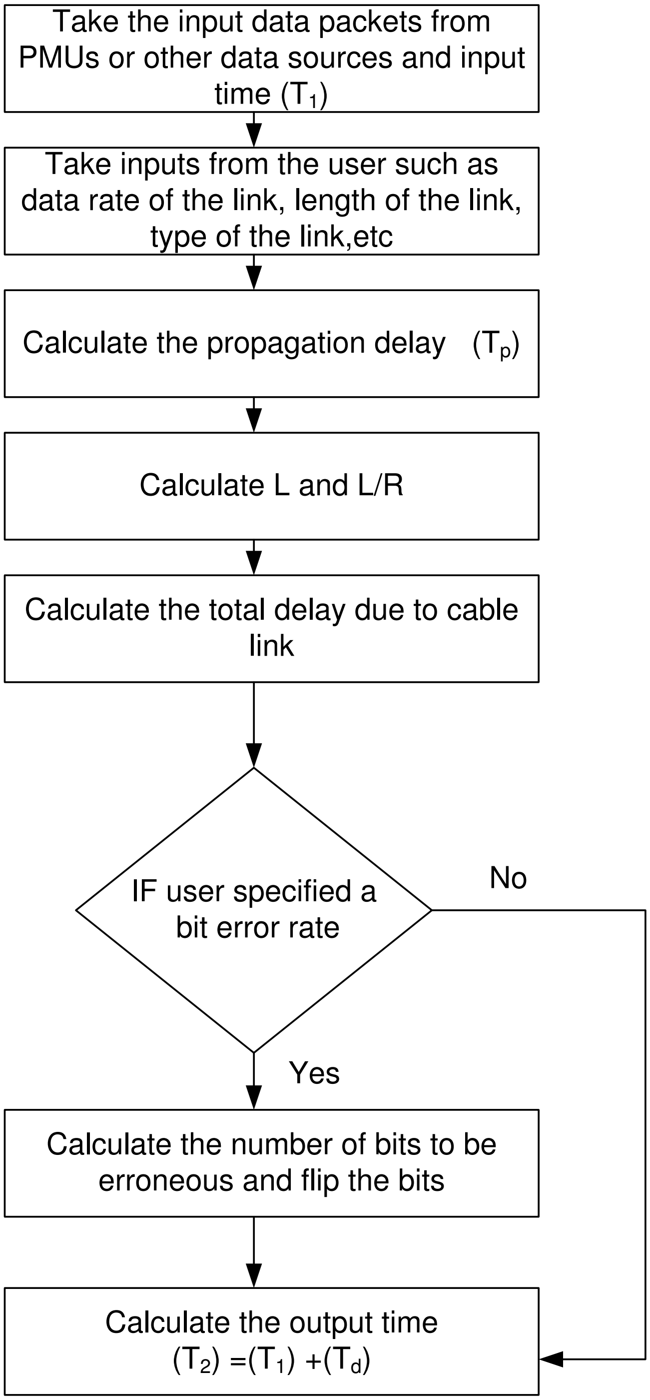


Figure 3.2.4: PSCAD cable link implementation.

packet is verified before it is forwarded to the network. This type of switch is preferred in network settings where there are high error rates. Such switches are also preferred as they are more reliable. This is because the packets are completely analyzed before they are forwarded thus improving the reliability [40].

A switch is added to the network by plugging various devices into the ports on the switch as shown in Fig. 3.1.1. When the switch gets the first bit of information from a device, such as a computer, it reads the MAC address of that device. The MAC address is the media access control number which is an ID number that is assigned to each network adapters and is preset by the manufacturer. The switch works to filter the information that is communicated among all of the devices. The switch has enough memory to retain this information in its memory bank.

3.3.2 Latencies Associated with a Switch

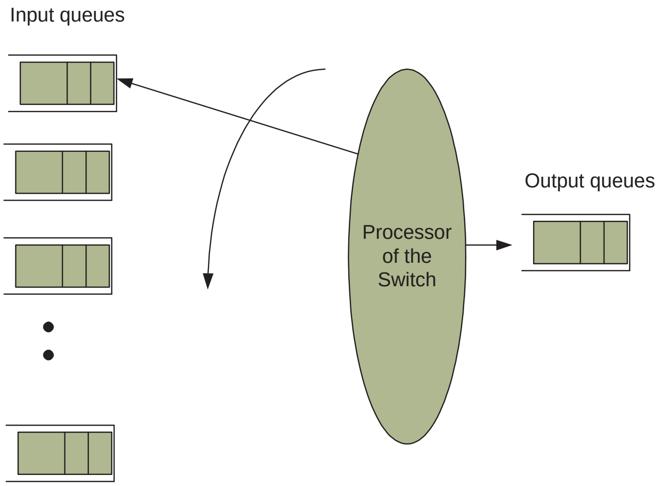
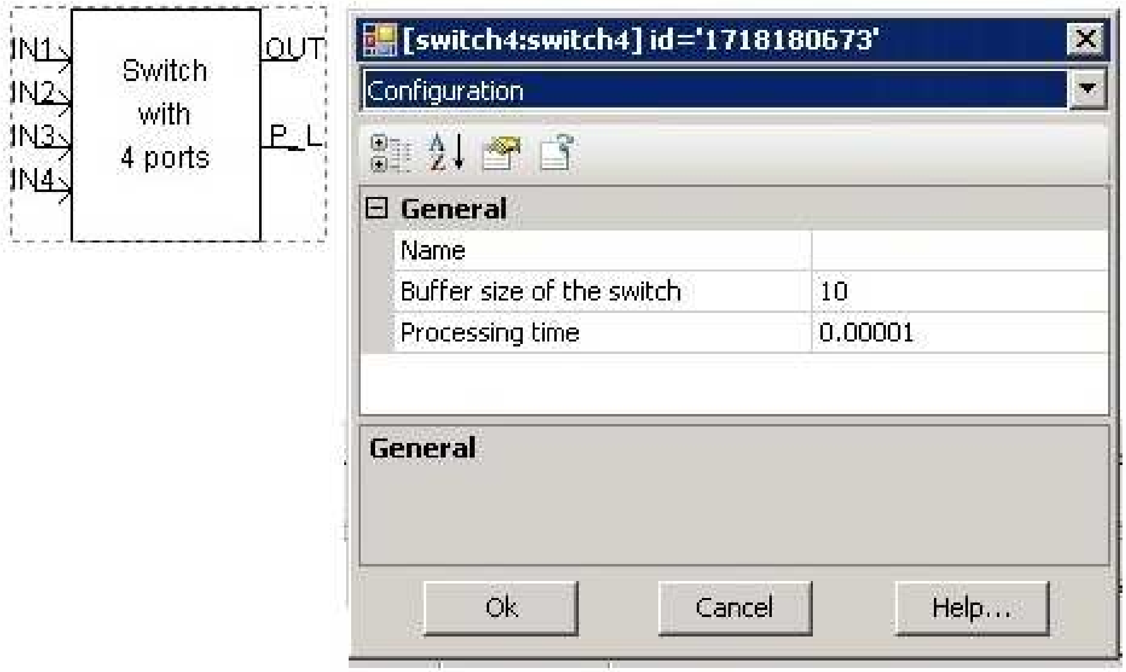


Figure 3.3.1: Queuing model of a switch.

This section explains the sources of latency on a network with switches and describes how cumulative latency due to the switch is calculated. The packet delay due to a switch in a communications network is defined as the time it takes for a packet to travel through the switch. Switched networks have two sources of latency: 1) switch fabric processing time, 2) frame queuing [41]. Switch fabric latency is deterministic and queuing latency is non-deterministic. Switches use queues as shown in Fig. 3.3.1, in conjunction with a store and forward mechanism to eliminate the problem of frame collisions that used to exist on broadcast Ethernet networks [42]. Queuing introduces a non-deterministic factor to the packet delay, since it can often be very difficult to predict the exact traffic patterns on a network. The data packets are stored in a buffer inside the switch. If the this buffer is full with data packets, switch discards the next data packet it receives leading to a packet loss.

3.3.3 The PSCAD Switch Model

The PSCAD model developed in this thesis for switch is shown in Fig. 3.3.2. A user can specify the buffer size and the processing time of the data packets in the buffer, i.e., switch fabric processing time. Normally, switch fabric processing time is in the micro second range. This PSCAD model for Ethernet switch is capable of simulating the delay due to the processing time, queuing in the switch buffer, and packet losses due to switch buffer overflow. Fig. 3.3.3 shows the steps used to model the switch component and Algorithm 2 describes the exact procedure used to implement the functionality of the switch.



[switch4:switch4] id='1718180673'

General

Name	
Buffer size of the switch	10
Processing time	0.00001

Figure 3.3.2: PSCAD switch model.

Algorithm 2 Ethernet Switch Model

Input:
B: Buffer size
P: Processing time
T1: Data input time
I: Data packet

```
IF number of packets in the buffer equals zero then
    Take the input data and send them to the output port after the processing time (P)
    Output time (T) = T1+P
    number of packets in the buffer=number of packets in the buffer+1
ELSE
    Output time(T) = T1+P*(number of packets in the buffer+1)
    number of packets in the buffer=number of packets in the buffer+1
END IF
IF Simulation Time equals the output time (T)
    Display input data array (I).
    number of packets in the buffer=number of packets in the buffer-1
END IF
IF number of packets in the buffer is greater than B then
    Packet loss=Packet loss+1
END IF
```

Output: Data output and Packets loss.

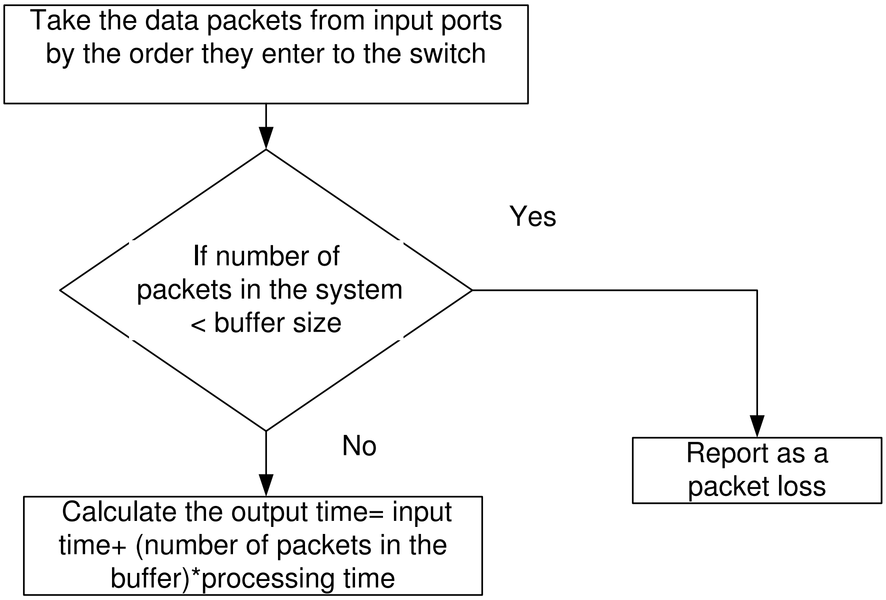


Figure 3.3.3: PSCAD switch implementation.

3.4 Data Source

Usually, data sources other than PMUs appear in a power system application, examples of which include RTUs, video units, protective relays, and intelligent electronic devices (IED) [23]. The presence of such devices along side PMUs can have an impact on the operation of wide area power system applications. When modeling a general data source, the key parameters are the data rate of the source, length of the data packet in bits, and a statistical model for packet generation, i.e., random packet generation or uniform packet generation, as shown in the Fig. 3.4.1 and Fig. 3.4.2 respectively.

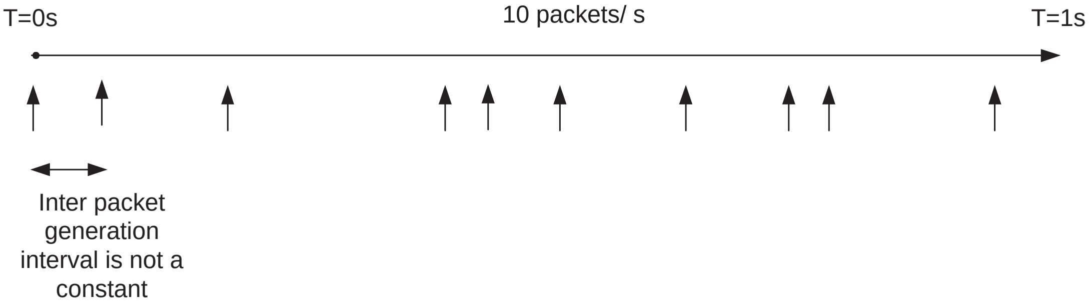


Figure 3.4.1: Random packet generation example with $\lambda = 10$.

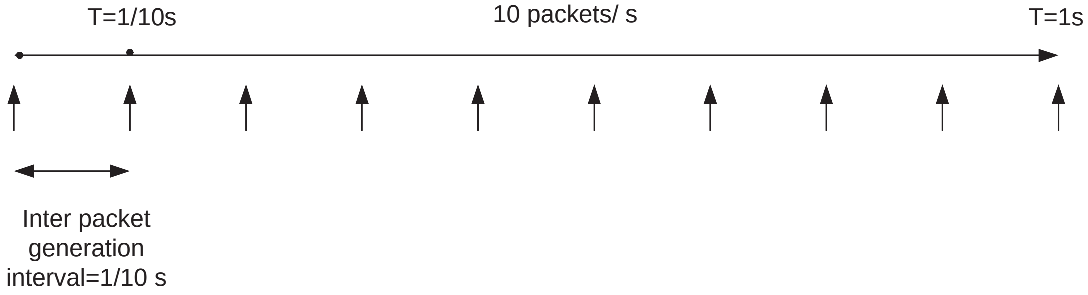


Figure 3.4.2: Uniform packet generation. In this example data rate is 10 packets/s.

3.4.1 Random Packet Generation Model

In the PSCAD model for data source described in section 3.4.3, an exponential distribution has been used to generate data packets at random time intervals. Let λ be the rate parameter or the average rate of packets per unit of time and X be an exponential random variable. Then the probability density function $f(x)$ can be written as,

$$f(x) = \lambda e^{-\lambda x} \tag{3.4.1}$$

where $\lambda > 0$. The cumulative density function $F(x)$ is given by,

$$F(x) = 1 - e^{-\lambda x}, x \geq 0. \tag{3.4.2}$$

$$X = F^{-1}(u). \tag{3.4.3}$$

where, u is a uniform random variable between 0 and 1 [44]. Therefore, we have,

$$X = \frac{\text{-log u}}{\lambda}. \tag{3.4.4}$$

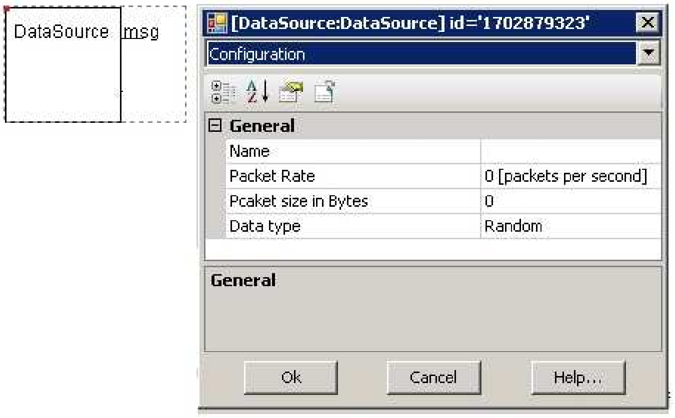
Finally, X is the exponential random variable.

3.4.2 Uniform Packet Generation Model

Let λ is the rate parameter. The time interval between two consecutive packets is constant in uniform packet generation as shown in Fig. 3.4.2. Then the time interval between two consecutive packets is $1/\lambda$.

3.4.3 The PSCAD Data Source Model

The PSCAD model for a general data source is shown in Fig. 3.4.3. In this model the user can specify the data rate, packet length, and the packet generation model (uniform or random). The algorithm used to implement the data source is shown in Algorithm 3.



[DataSource:DataSource] id='1702879323'

Configuration

General

Name	
Packet Rate	0 [packets per second]
Pcaket size in Bytes	0
Data type	Random

General

Figure 3.4.3: PSCAD data source model.

Algorithm 3 Data Source Model

Input:

T: Data source type (uniform or random)
R: Packet rate
S: Packet length
C: Counting variable

```
IF simulation time equals zero then
    count=0
    output=0
END IF
IF Type(T) is Uniform then
    IF Simulation Time == (1/R)*count then
        Generate the data packet according to the user specified packet length (S)
        count=count+1
    END IF
ELSE Type (T) is random then
    IF Simulation Time == (1/R)*count + random number between 0 and 1/R, generated according to 3.4.4.
        Generate the data packet according to the user specified packet length (S)
        count=count+1
    END IF
END IF
```

Output: Data packet.

Chapter 4

Queuing-Theoretic Modeling of a PMU Communication Network

4.1 Introduction

With emerging synchropahsor technology, systems with a large number of PMUs will likely be deployed in the near future. As the number of data sources in a system increases, the congestion in network routers and switches may lead to unacceptable delays and packet-losses at the PDC and other control devices. However, in the wide area power system studies no analytical method exist to calculate the communication delay of a PMU-PDC communication network including the queuing delay of network switches. Therefore, this chapter presents an analytical approach to study the delay and packet loss characteristics of a PMU-PDC communication network using queuing theoretic tools [46–49]. Also, this analytical method is useful to verify the accuracy of the new communication components developed to PSCAD/EMTDC in this thesis. There are two types of PMU-PDC communication network architectures, i.e., dedicated network and shared network [21, 52]. Reliable communication of PMU measurements is important to deploy the wide area protection and control. For that most of the wide area studies consider a dedicated network architecture rather than a shared network architecture, since in a dedicated network, each PMU has a dedicated communication link to the central PDC. Therefore, a centralized dedicated PMU-PDC communication architecture in which the control center switch cyclically allocates its processing time among the substation switches in the network is considered in this study. The time-sharing behavior of this PMU-PDC network is modeled using the cyclic polling approach [50]. Then a discrete-time Markov chain is set up to derive expressions for communication delay and packet-loss probability of PMU-PDC links. We also present numerical results obtained by simulation of the data network in PSCAD to validate the accuracy of the proposed model. The computational complexity of our model is linear in the number of PMUs in the network. Therefore, it can be particularly useful in the study of WAMPaCS which include a large number of PMUs.

4.2 System Model and Assumptions

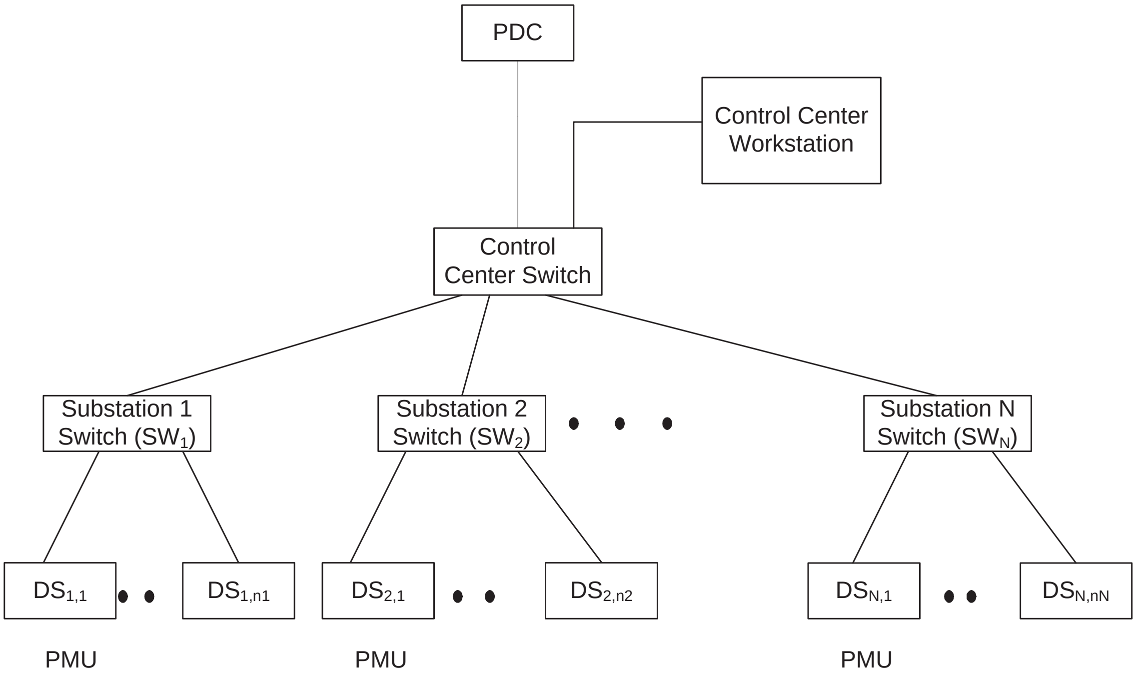


Figure 4.2.1: Block diagram of a PMU-PDC communication system.

The system model under consideration is shown in Fig. 4.2.1. In this network model, PMUs and other data sources such as RTUs and video units [21] are linked to a *switch* in each substation. All substation switches are linked to the PDC and the control center workstation via a switch in the control center. The control center switch will route the packets from the PMUs to the PDC while the packets from RTUs and video units are routed to the control center workstation.

All communication links are assumed to be standard 10BASE5 Ethernet links, also known as the *thick Ethernet*. The number of PMUs and other data sources connected to each sub station switch is not necessarily a constant and can vary according to the application requirements. At least one PMU is connected to each substation switch. Let the number of substation switches in the network be N. Since PMUs typically generate data at a constant rate [1], it is assumed that inter-arrival time of packets from a PMU is constant. On the other hand, the inter-arrival times of packets from other RTUs and video units are assumed to be random. It is also assumed that all substation switches have an identical buffer size of B packets, i.e., a switch can keep a maximum of B data packets, including the one being served. In the set-up considered, each substation switch forwards data packets in its queue (buffer) to the control center switch in a first-in-first-out (FIFO) order. In this analytical model, a *cyclic polling system* is used to represent the process by which the control center switch accesses or *serves* the packets buffered in different substation switches. Hence, this type of setup can be represented by a vacation queue model [55], and the PMU-PDC network can be decomposed in to a single server queue with visits and

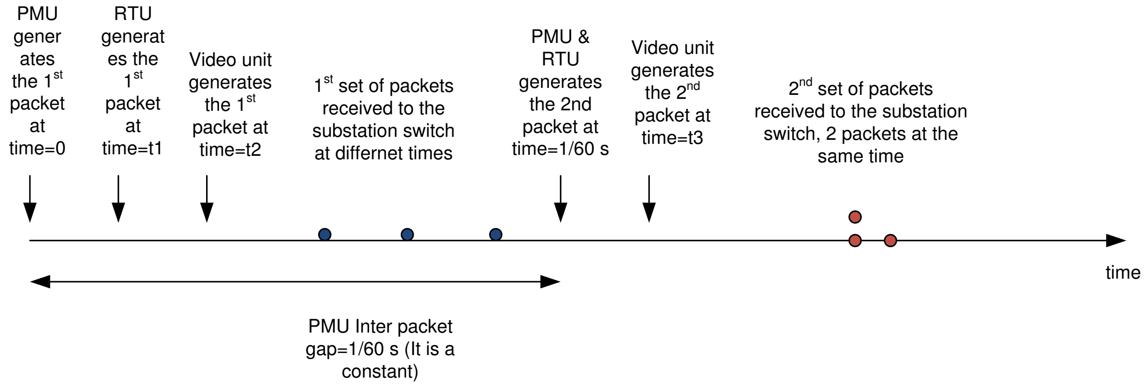


Figure 4.2.2: Timing diagram.

vacation periods, where the vacation period for each substation switch is the sum of all the service periods of all the other switches.

4.2.1 Cyclic Polling System

Polling models have been widely studied in the last two decades related to various areas. As a result, a large body of knowledge exists in this area. These models serve as powerful tools for the performance analysis in a wide variety of important applications. A polling system is defined as a single resource of service shared by multiple queues. The range of application in which polling models can be used is very broad. They include computer communications, robotics, traffic and transportation, manufacturing, production, e-mail distribution, etc.

Among polling models, the most popular one is the cyclic polling model. It was first used in the analysis of time-sharing computer systems. In cyclic polling system, each queue in the polling system is visited cyclically. In addition to the modeling aspects of polling systems, we are also interested in the aspects of performance improvements and system optimization issues. Traditionally, these are the ultimate goals of any modeling and analysis effort.

4.2.2 Phase Type Distribution

A phase-type distribution is a probability distribution that results from a system of one or more inter-related Poisson processes occurring in sequence, or phases. The sequence in which each of the phases occurs may itself be a stochastic process. The distribution can be represented by a random variable describing the time until absorption of a Markov process with one absorbing state. Each of the states of the Markov process represents one of the phases. It has a discrete time equivalent, i.e., the discrete phase-type distribution. The set of phase-type distributions is dense in the field of all positive-valued distributions, i.e., it can be used to approximate any positive-valued distribution.

Definition: Phase type distributions are distributions of the time until absorption in an absorbing DTMC. If after an absorption the chain is restarted, then it represents the distribution of a renewal process. Consider an $(n_t + 1)$ absorbing DTMC with state space $0, 1, \ldots, n_t$ and let the state 0 be the absorbing state. Let T be an m dimension sub-stochastic matrix with entries

$$\mathbf{T} = \begin{bmatrix} t_{1,1} & \cdots & \cdots & t_{1,n_t} & \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{n_t,1} & \cdots & \cdots & t_{n_t,n_t} & \end{bmatrix}.$$

and also let

$$\boldsymbol{\alpha} \;\; = \;\; [\alpha_1, \alpha_2, ..., \alpha_{n_t}],$$

and $\alpha \leq 1$. In this context, α_i is the probability that the system starts from a transient state i, 1≤ i $\leq n_t$ and $t_{i,j}$ is the probability of transition from a transient state i to a transient state j with $0 \leq T\mathbf{1} \leq 1$ and at least one new state less than one. We say the phase type distribution is characterized by (α, T) of dimension n_t [51]. The transition matrix P of this absorbing Markov chain is given as

$$\mathbf{P} = \begin{bmatrix} 1 & 0 \\ \mathbf{t} & T \end{bmatrix}.$$

, where **t**=**1**-T**1** and **1** is the identity matrix.

4.2.3 Vacation Queues

Vacation queues are a very important class of queues in real life, especially in telecommunication systems where medium access control (MAC) is a critical component of managing a successful network. By definition a vacation queue is a queuing system in which the server is available only a portion of the time. At other times it is busy serving other stations or just not available maybe due to maintenance activities (either routine or due to a breakdown). Polling systems are a class of queuing systems in which a server goes to serve different queues based on a schedule, which implies that the server is not always available to a particular queue. Hence when the server is attending to other queues then it is away on vacation as far as the queue not receiving service is concerned. Vacation queues are used extensively to approximate polling queues.

4.3 Problem Formulation

4.3.1 Communication Delay

The total communication delay experienced by a packet transmitted from a data source to its destination can be given by [24]

$$T = T_p + T_q + L/R, \tag{4.3.1}$$

where T_p is the propagation delay, T_q is the waiting time spent in the buffers, L is the size of an Ethernet frame, and R is the data rate in bits/s. Except T_q, it is straightforward to compute other parameters in (4.3.1). In this chapter, our goal is to derive expressions for the mean waiting time in the queue or *queuing delay* $\bar{T}_q = E\{T_q\}$ and total average delay $\bar{T} = E\{T\}$ for communication between a given PMU and the PDC. Furthermore, queuing delay in a system as shown in Fig. 4.2.1, can be decomposed in to 2 parts, i.e., queuing delay in the substation switch (T_{q_1}) and queuing delay in the control center switch (T_{q_2}). In the following it is assumed that the switch-over time of the processor in the control center switch to be negligibly small. It is also assumed that, if the switch buffer is full, any data packets arriving from connected data sources are discarded, leading to *packet-losses*.

4.3.2 Discrete Time Markov Chain (DTMC) Model

In this section, we set-up arrival, service, and vacation processes of the DTMC used to represent the system shown in Fig. 4.2.1. Consider the first substation switch SW_1 which contains n_1 data sources, $DS_{1,1}, \ldots, DS_{1,n_1}$. As an example, let $n_1 = 3,$ i.e., $DS_{1,1}$, is the PMU connected to the switch 1, $DS_{1,2}$ is the RTU and $DS_{1,3}$ is the video unit. However, the packets sent by different data sources will arrive at the switch at different times, since all data sources do not generate data packets at same and constant intervals. Therefore, there is a probability that more than one packet arrive at the SW_1 at the same time instant as illustrated in Fig. 4.2.2.

Arrival Process

Let the probability of p packets arriving at the i^{th} switch SW_i at the time instant t be $D_{i,p}$, where $p = 0, \ldots, n_i$. Let's assume that the packets form a data source arrives at the switch according to the geometric distribution. Then, arrivals to each substation switch is a batch Markovian arrival process (BMAP) [51]. Hence, $D_{i,0}, \ldots, D_{i,n_i}$ can be expressed as

$$D_{i,0} = \prod_{j=1}^{n_i}(1 - a_{i,j}) \tag{4.3.2}$$

$$D_{i,p} = \sum_{j=1}^{n_i}\prod_{j=1}^{p} a_{i,j}\prod_{k=p}^{n_i}(1 - a_{i,k}), \text{for p=}1, \ldots, n_i - 1, \tag{4.3.3}$$

$$D_{i,n_i} = \prod_{j=1}^{n_i} a_{i,j}, \tag{4.3.4}$$

where, $a_{i,j}$ is the probability that a data packet arrives from j-th data source to SW_i, $i = 1, \ldots, N$, and $j = 1, \ldots, , n_i$. Since, $\sum_{j=0}^{n_i} D_{i,j} = 1$, this arrival process is stochastic.

Service Process

In the following the queuing delay has been included in the substation switch (T_{q1}) as the service time of each substation switch. The procedure explained in [46] is used to calculate

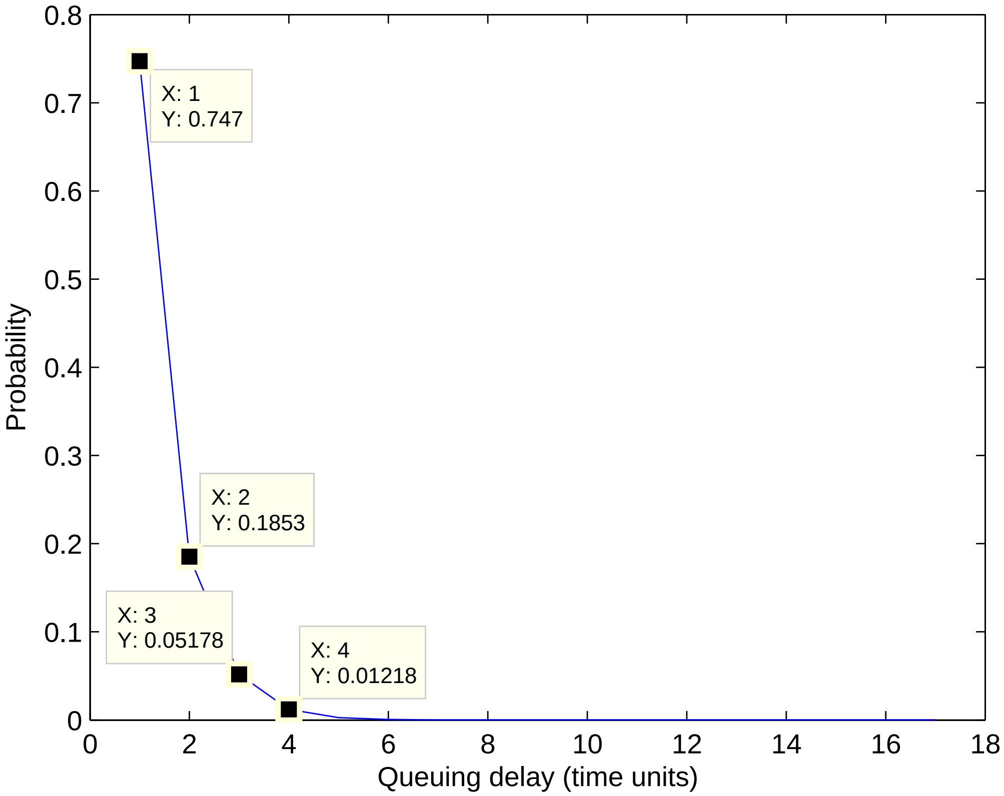


Figure 4.3.1: Queuing delay distribution for the switch SW_1.

the queuing delay distribution for each substation switch is shown in Fig. 4.3.1. T_{q1} which was calculated using the method in [46] for the example explained in section 4.3.2. Suppose we discretized the range of T_{q1} into J time steps, so that $T_{q1} \approx cJ$, where c is an integer. Then, T_{q1} can be represented by a discrete phase (PH) type distribution with geometric service completion ($\mathbf{S}_i$, $\boldsymbol{\beta_i}$) of order cJ, where $\mathbf{S}_i$ is the $cJ \times cJ$ matrix which represents the transition states in phase type distribution

$$\mathbf{S}_i = \begin{bmatrix} 0 & 1-b_{1,i} & 0 & \cdots & \cdots \\ 0 & 0 & 1-b_{2,i} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & 1-b_{(cJ-1),i} \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix}.$$

The probabilities of getting absorbed from each state are given by the $cJ \times 1$ matrix

$\mathbf{s}_i = [b_{1,i}, \cdots, b_{(cJ-1),i}, 1]^T$, where $b_{j,i}$ is the probability of completing the service in the j time units in the i^{th} switch which can be taken from the queuing delay distribution for each substation switch. The initial distribution is given by $\boldsymbol{\beta_i} = [1\ 0\ \cdots\ 0]$ with the dimension $1 \times cJ$.

Vacation Process

The vacation time is modeled as an exhaustive single vacation type. That is, the server goes on a vacation as soon as the system is empty. After the server returns from a vacation, it starts to serve the packets waiting in the queue or if the queue is empty, waits for the first packet to arrive [51]. Therefore we have modeled the vacation time as a discrete phase type distribution given by ($\mathbf{V}_i, \boldsymbol{\alpha_i}$) of order k_i for the i-th switch. Once we calculate the queuing delay distribution for each substation switch using the method in [46], we can calculate the vacation time for each substation using the Monte-Carlo method [54]. The order, k will vary according to the number of data sources in the system, since the vacation of a given substation switch depends on the time spent by the server (i.e., processor of the control center switch) on other substation switches. If i^{th} substation switch SW_i has k_i time-units of vacation time, it can be represented as follows by using a phase type distribution

$$\begin{aligned} \mathbf{V}_i &= \begin{bmatrix} \mathbf{0}_{k_i-1,1} & \mathbf{I}_{k_i-1,k_i-1} \\ 0 & \mathbf{0}_{1,k_i-1} \end{bmatrix}, \\ \mathbf{v}_i &= [\mathbf{0}_{1,k_i-1} \quad 1]^T, \text{ and } \boldsymbol{\alpha}_i = [1 \quad \mathbf{0}_{1,k_i-1}], \end{aligned}$$

where **I** denotes the identity matrix and **0** denotes a null matrix.

DTMC for SW_i

With arrival, service, and vacation processes as defined above, the DTMC can be set up as a BMAP/PH/1 queue with PH vacation. Using the results in Sec. 5.2.3.1 in [51], we can show that the state transition matrix of this DTMC can be given by $\mathbf{P}_i$ as shown in (4.3.5)

$$\begin{array}{ll}
\text{States} \rightarrow & \qquad (0,1) \qquad (0,2) \qquad\qquad (1,1) \qquad\qquad\qquad (1,2) \qquad\quad (m,1) \qquad\qquad\qquad\qquad (m,2) \\
\mathbf{P}_i = & \left[\begin{array}{cccccc}
\begin{bmatrix} D_{i,0} & 0 \\ D_{i,0} \otimes \mathbf{v}_i & D_{i,0} \otimes \mathbf{V}_i \end{bmatrix} & \begin{bmatrix} D_{i,1} \otimes \boldsymbol{\beta} & 0 \\ D_{i,1} \otimes \mathbf{v}_i \boldsymbol{\beta} & D_{i,1} \otimes \mathbf{V}_i \end{bmatrix} & \cdots & \begin{bmatrix} D_{i,m} \otimes \boldsymbol{\beta} & 0 \\ D_{i,m} \otimes \mathbf{v}_i \boldsymbol{\beta} & D_{i,m} \otimes \mathbf{V}_i \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \cdots \\
\begin{bmatrix} 0 & D_{i,0} \otimes (\mathbf{s} \boldsymbol{\alpha}_i) \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} D_{i,0} \otimes \mathbf{S} + D_{i,1} \otimes \mathbf{s} \boldsymbol{\beta} & 0 \\ D_{i,0} \otimes \mathbf{v}_i \boldsymbol{\beta} & D_{i,0} \otimes \mathbf{V}_i \end{bmatrix} & \cdots & \begin{bmatrix} D_{i,m-1} \otimes \mathbf{S} + D_{i,m} \otimes \mathbf{s} \boldsymbol{\beta} & 0 \\ D_{i,m-1} \otimes \mathbf{v}_i \boldsymbol{\beta} & D_{i,m-1} \otimes \mathbf{V}_i \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \cdots \\
\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & D_{i,0} \otimes (\mathbf{s} \boldsymbol{\alpha}_i) \\ 0 & 0 \end{bmatrix} & \cdots & \begin{bmatrix} D_{i,m-2} \otimes \mathbf{S} + D_{i,m-1} \otimes \mathbf{s} \boldsymbol{\beta} & 0 \\ D_{i,m-2} \otimes \mathbf{v}_i \boldsymbol{\beta} & D_{i,m-2} \otimes \mathbf{V}_i \end{bmatrix} & \ddots & \cdots \\
\vdots & \ddots & & \ddots & \ddots & \cdots
\end{array}\right]
\end{array} \tag{4.3.5}$$

(see next page), where $m = n_i$ and the number-pairs above the matrix indicate possible states. As an example, in $\mathbf{P}_i$, (0,1) represents the state in which there are no packets in the switch SW_i and the server is in the serving mode, and (0,2) represents the state in which there are no packets in the switch and the server is on vacation. Since the system can have a maximum of B packets in the switch, the last two states in $\mathbf{P}_i$ are $(B,1)$ and $(B,2)$.

4.4 System Behavior Analysis

In this section, we develop an expression for the mean waiting time in the queue (or the queuing delay) and packet loss probability. In order to do this, we should find the stationary distribution of the above DTMC given by $X_i = (X_{i,1}, X_{i,2}, \ldots, X_{i,B})$, where $X_{i,j}$ is the probability that system has j packets in the i-th switch, under the stationarity condition X_i=$X_i P_i$. Note that the packet-loss probability is $P_L = X_{B+1}$. For SW_i,

$$P_{L_i} = X_{i,B} \sum_{j=1}^{n_i} D_{i,j}, \tag{4.4.1}$$

and in order to find the queuing delay, we can apply a suitably modified version of the well known *Little's law*

$$L_q = \lambda W_q, \tag{4.4.2}$$

where, L_q is the mean number of packets in the queue, λ is the arrival rate of data packets, and W_q is the mean waiting time in the queue. The mean number of packets waiting in the queue to get the service can be expressed as

$$L_{q_i} = \sum_{u=1}^{B} (u-1) X_{i,u}, \tag{4.4.3}$$

Since, the all switches have identical buffer sizes, (4.4.3) is the same for all substation switches. If the system experiences packet losses, then we should consider the effective arrival rate,

$$\lambda' = (1 - P_{L_i})\lambda, \tag{4.4.4}$$

instead of λ, in (4.4.2). For j^{th} data source in i^{th} switch $DS_{i,j}$, $\lambda_{i,j}$=$\lambda a_{i,j}$ packets/ms and queuing delay $W_{q_{(i,j)}}$ is approximated as

$$W_{q_{(i,j)}} = \frac{\sum_{u=1}^{B} (u-1) X_{i,u}}{(1 - P_{L_i})(\lambda a_{i,j})}. \tag{4.4.5}$$

Likewise, we can calculate the packet loss probability using (4.4.1) and the queuing delay (4.4.5) for all data sources connected to any substation switch. We can then calculate the total delay using (4.3.1). Note that in this model, (4.4.5) gives summation of T_{q1} and T_{q2}, as we have included T_{q1} in the service time. Note that, when the number of data sources connected to the substation switch n_i increases, the number of terms in (4.3.2)-(4.3.4) increase linearly. Therefore, the complexity of calculating the queuing delay and packet loss probability using the proposed model increases only linearly in the number of data sources in the network.

4.5 Numerical Results

In order to evaluate the DTMC model developed in this research numerically, following system parameters are used.

- $N = 3$ (SW_1, SW_2, and SW_3),
- $n_1 = 3$ (1 PMU, 1 RTU and 1 video unit),

- $n_2 = 2$, (1 PMU and 1 RTU),
- $n_3 = 4$, (2 PMUs, 1 RTU and 1 video unit) and
- $B = 8, 10, 15$.

According to Fig. 4.3.1, we model the service time for SW_1 as a discrete phase type distribution ($\mathbf{S}$, $\boldsymbol{\beta_1}$) of order 4, with the time step c equal to 1, and $b_{1,1} = 0.747$, $b_{2,1} = 0.1853$, $b_{3,1} = 0.0518$, $b_{4,1} = 0.012$. The same procedure is followed to calculate the service time for SW_2 and SW_3. Then, using the Monte-Carlo method, we estimated the average vacation times for SW_1, SW_2 and SW_3 to be $k_1 = 8$ ms, $k_2 = 10$ ms, and $k_3 = 9$ ms respectively and modeled these vacation times by a separate phase type distribution, ($\mathbf{V}_i, \boldsymbol{\alpha_i}$) of order k_i.

In this section, we present numerical results for packet loss probability, queuing delay $\bar{T}_q$, and total communication delay $\bar{T}$ of each data source connected to three substation switches SW_1, SW_2, and SW_3. In order to find the total delay $\bar{T}$, we require the propagation delay between each data source and the PDC. The propagation delay of 10BASE5 Ethernet is 4.33 ns/meter (IEEE 802.3). The data rate has been assumed to be $R = 1$ Mbps. The Ethernet frame size L for PMU, RTU and video unit is taken as 100 bytes, 700 bytes and 1100 bytes respectively [21]. In this analytical model, we assume a standard PMU data rate of 60 packets/s [13]. RTU data rate and video unit data rate are assumed as 30 packets/s and 200 packets/s respectively [21].

Table 4.1: Numerical results based on the analytical model for SW_1. $a_{1,1} = 0.2, a_{1,2} = 0.1, a_{1,3} = 0.7, D = 100$ km.

	Queuing Delay $\bar{T}_q$ (μs)			Total Delay $\bar{T}$ (ms)		
	B=8	B=10	B=15	B=8	B=10	B=15
PMU	11.1519	12.5317	16.8433	1.2112	1.212	1.2168
RTU	27.8797	31.3292	42.1084	6.0279	6.0313	6.0421
Video Unit	4.2892	4.8199	6.4782	9.2043	9.2048	9.2065

Table 4.1 presents the queuing delay and total delay corresponding to 3 data sources connected to the SW_1, for three buffer sizes, where D is the distance between the data

Table 4.2: Numerical results based on the analytical model for SW_2. $a_{2,1} = 0.67, a_{2,2} = 0.33, D = 200$ km.

	Queuing Delay $\bar{T}_q$ (μ s)			Total Delay $\bar{T}$ (ms)		
	B=8	B=10	B=15	B=8	B=10	B=15
PMU	3.1417	4.1844	6.9781	1.6031	1.6042	1.6070
RTU	6.3787	8.4955	14.1676	6.4064	6.4085	6.4142

Table 4.3: Numerical results based on the analytical model for SW_3. $a_{3,1} = 0.2, a_{3,2} = 0.2, a_{3,3} = 0.1, a_{3,4} = 0.6, D = 325$ km.

	Queuing Delay $\bar{T}_q$ (μs)			Total Delay $\bar{T}$ (ms)		
	B=8	B=10	B=15	B=8	B=10	B=15
PMU	13.1998	15.0301	18.1934	2.1122	2.1160	2.1272
PMU	13.1998	15.0301	18.19341	2.1122	2.1160	2.1272
RTU	25.9245	27.0640	36.7860	6.9259	6.9341	6.9578
Video Unit	3.6385	4.7809	8.1103	10.1036	10.1048	10.1081

sources and the control center. It is important to notice that the queuing delay depends on the values for packet arrival probability, $a_{1,j}$. $a_{1,j}$ for PMU, RTU and video are calculated as 60/290, 30/290 and 200/290 respectively, according to the data rate of the each data source. Notice that a higher $a_{1,j}$ results in lower queuing delay compared to a lower $a_{1,j}$. The reason for this is that the queuing delay decreases with the higher packet arrival probability. If the packet arrival probability is high, then the probability of being served by the processor of the control center switch is also high. Therefore, the mean waiting time in the queue (queuing delay) decreases. Similarly, Table 4.2 and Table 4.3 present the performance measures for data sources connected to SW_2 and SW_3 respectively.

As can be seen in Tables 4.1, 4.2, and 4.3, queuing delay of each data source increases with the buffer size. The effect of the buffer size on the queuing delay is presented in Fig. 4.5.1. When the buffer size is large, it gives more room for data packets and as a result, eventually the mean waiting time increases. Also it is clearly noticeable that a small buffer size allows a smaller number of packets to be served, so that the queuing delay is less compared to higher buffer sizes, but packet losses will be higher. Furthermore, with any buffer size, a switch with more data sources experiences a longer queuing delay compared to a switch with a smaller number of data sources. As an example, consider SW_2 with 2 data sources and SW_3 with 4 data sources. The PMU connected to SW_2 has less queuing delay compared to the queuing delay of the 1-st PMU connected to SW_3 for $B = 8$, $B = 10$ and $B = 15$.

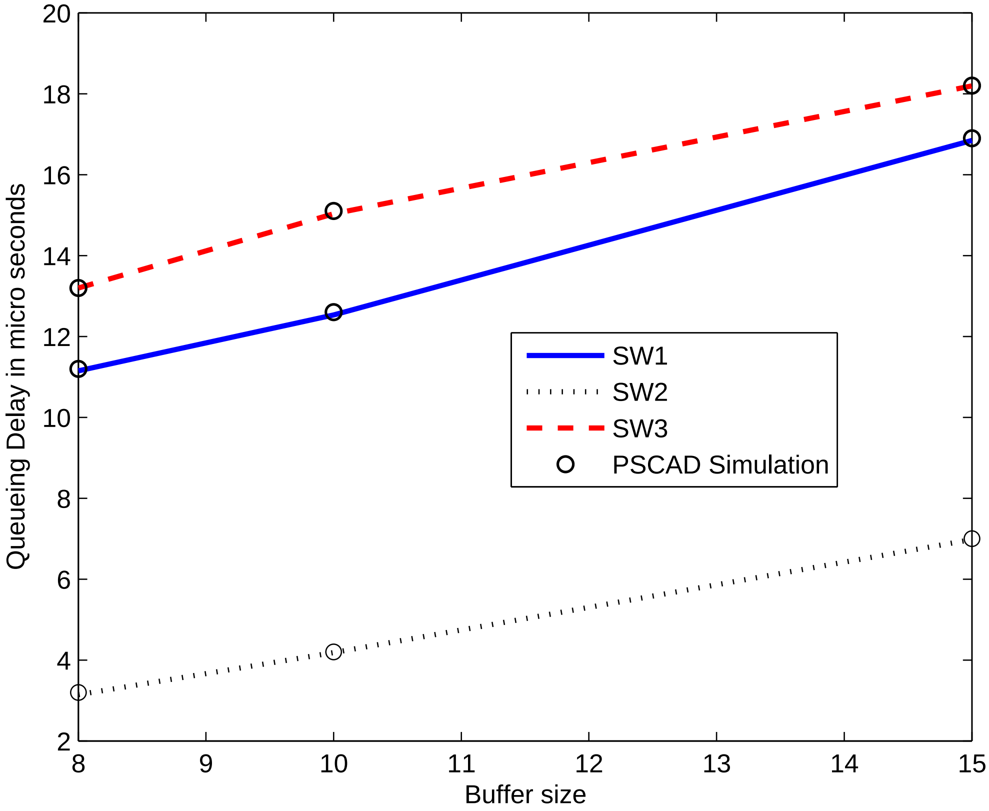


Figure 4.5.1: Queuing delay vs buffer size (in packets) for the 1^{st} PMU in each substation switch.

In order to verify the validity of the modeling procedure presented in this chapter, we next compare the network performance parameters calculated from our analytical model against those estimated by PSCAD simulations as shown in Figs. 4.5.1 and 4.5.2. According to Fig. 4.5.2, it is clear that if the number of PMUs connected to a substation is large, the packet loss probability increases. For example, SW_2 has only 2 data sources connected to it and therefore it has the lowest packet loss probability, while SW_3 with 4 data sources has the highest packet loss probability. This effect can be clearly seen in Fig. 4.5.2. Also, in order to present the effect of buffer size on the packet loss probability, we have also con-

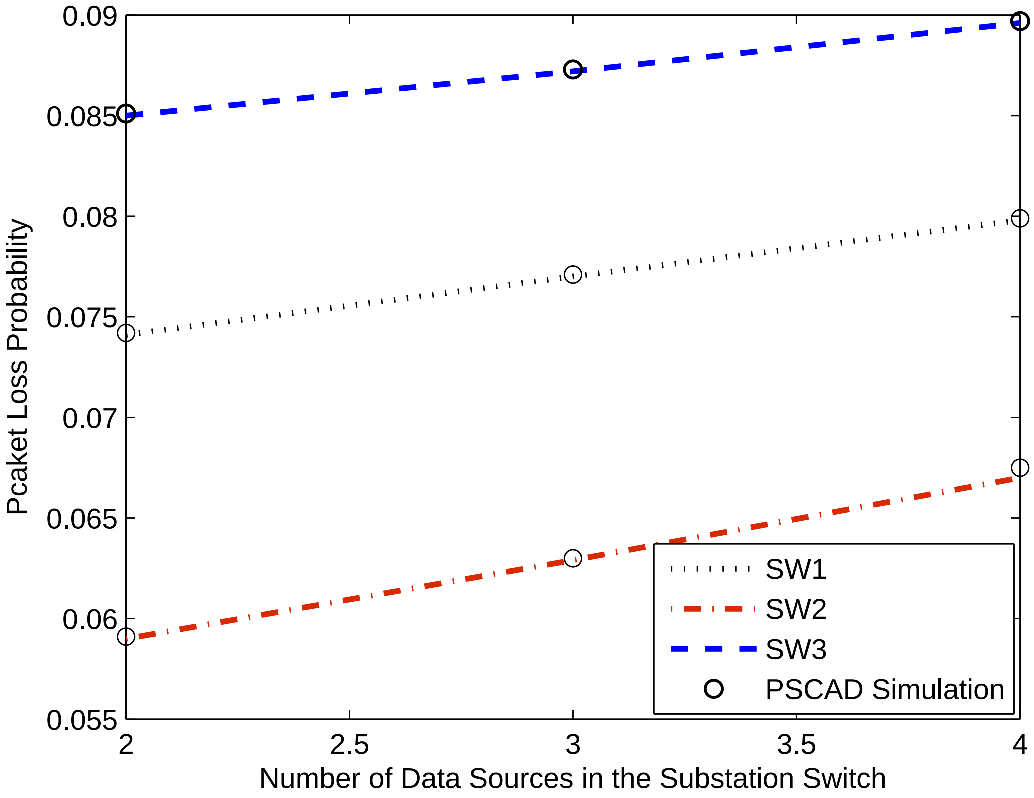


Figure 4.5.2: Packet loss probability vs no. of data sources in the substation switch.

sidered three buffer sizes in Fig. 4.5.2. As these results confirm, the proposed analytical model predicts the impact of buffer size on the packet loss probability, i.e., increasing the buffer size leads to a decrease in packet-loss probability. The PSCAD simulation results agree with the results of the proposed analytical model.

4.6 Conclusions

A novel queuing-theoretic model based on cyclic polling for a centralized PMU-PDC communication system has been proposed. Using simulations we have demonstrated that this model can be used to predict the critical performance metrics of a PMU network such as the packet delay and packet loss probability. The model can be used to study the impact of the number of data sources in a given network on the system reliability in terms loss probability and delay which can have an adverse effect on the underlying WAMPaCS performance. It can also be useful to the network designer in choosing the appropriate buffer size for switches in order to ensure the required level of reliability. Furthermore, the proposed modeling approach in general can be useful in developing software tools for co-simulation of a power system and a communication network that overlays it.

Chapter 5

Co-Simulation

5.1 Introduction

The goal of this chapter is to demonstrate the effectiveness of software tools developed in Chapter 3 by simulating a simple power system application. A system designer must be able to determine the impact of a given communication network infrastructure on critical power system applications that rely on the communication network and identify the potential bottlenecks. Power system simulation is a commonly used tool for assessing the operational characteristics of various control and protection schemes. A comprehensive analysis of a power system with communication based protection and control can only be carried out by the co-simulation of the power system and the data communication network. ”Two area four generator system” presented in [56] is selected as the wide area power system application to integrate the communication components developed in this thesis into the power system simulation. Two area four generator system is modeled in PSCAD with some modifications to the system as in [57].

As most of the stability control and protection applications in power systems require response time of less than few hundred milliseconds after a disturbance such as a fault, the typical communication delays in a data network can be very significant. Also, the loss of data packets and bit errors may cause some serious issues on the power system operation.

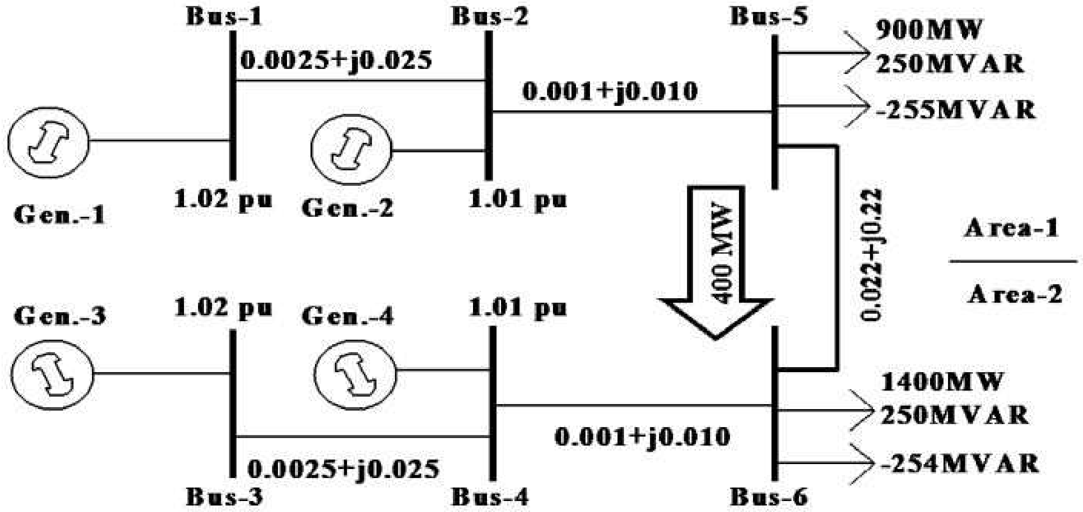


Figure 5.2.1: Single line diagram of two area four generator power system taken from [57].

In the this chapter, the effect of various communication conditions on the power system operation will be demonstrated based on co-simulation results.

5.2 Power System Application

Two area four generator system is a simple benchmark power system [56] with two generators (Gen.-1, and Gen.-2) in ”Area-1” and two generators (Gen.-3, and Gen.-4) in ”Area-2”. Gen.-1 and Gen.-2 are connected to the other two generators in Area-2, through a tie-line as shown in Fig. 5.2.1.

Initial power flow solution for the two area four generator system is given in Table 5.1. Loads are considered to be constant admittance type loads. Power transfer from Area-1 to Area-2 is 400MW. All four generators are modeled as 6^{th} order models. IEEE type AC4A exciters are added to all four generators. Dynamic data for generator and exciter models are given in [57].

This power system is linearized around the operating point shown in Table 5.1 for small signal stability analysis. Small signal stability in a power system is the ability of the power system to operate in synchronism when it is subjected to small disturbances [56]. Poorly

Table 5.1: Initial power flow solution.

Bus Number	Terminal Voltage (pu)	Angle (deg.)	Real Power (MW)	Reactive power (Mvar)
1	1.02	64.25805	664.211	-12.865
2	1.02	55.05614	664.4	492.182
3	1.02	0.00	564.708	-17.736
4	1.02	-7.82345	500	484.847
5	0.9786	47.68	900	250
6	0.9765	-13.69	1400	250

damped electromechanical oscillations are a common small signal stability problem that prevails in a power system. In the power system we considered, a small signal oscillations (stability issue in the system) can be seen when a disturbance is applied to the 2^{nd} generator (Gen.-2). Electromechanical modes in the two area four generator power system were identified using the eigen analysis of the linearized system. A poorly damped mode was revealed having a frequency of 0.3337 Hz, and a damping ratio of 0.0237.

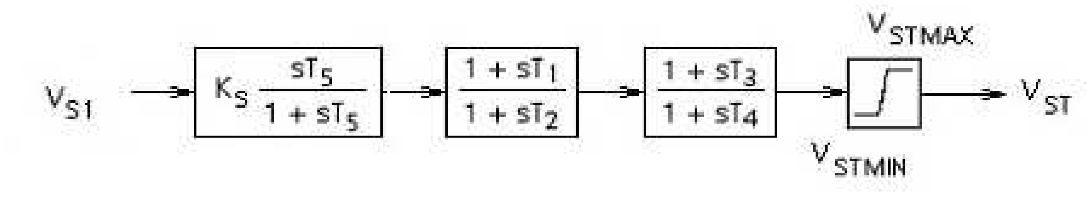


Figure 5.2.2: Power system stabilizer.

Power system stabilizers (PSS) can be used to damp-out the electromechanical oscillations. PSS are the conventional and cost effective solution for providing damping to the electromechanical oscillations. A simple speed sensitive power system stabilizer model is given in Fig. 5.2.2. It consists of a washout filter and two lead lag blocks and a gain. Washout filter, which is a high pass filter, passes-out the oscillations in speed signal without affecting the generator terminal voltage for steady changes in speed. In Fig. 5.2.2,

- $k_s =$ Gain [pu]
- $T_1, T_3 =$ Lead time constants [s]

- $T_2, T_5 =$ Lag compensating time constants [s]
- $T_5 =$ Washout time constant [s]
- $V_{s_1} =$ Input
- V_{STMAX} and $V_{STMIN} =$ Maximum and minimum output limits [pu]
- $V_{ST} =$ Output [pu]

The most commonly used input (reference signal) for the PSS is the generator speed signal. Conventional PSS uses local speed of the generator (speed of the generator where PSS is installed). With the implementation of PMU network in the power system, remote signals (speed of another generator in the system apart form the generator which PSS installed) can also be used for the damping of electromechanical oscillations. In certain situations, if damping of electromechanical modes is impossible with local signals, application of remote signals can be an advantage. Analysis of participation factors shows that in this power system, remote signals can be used to achieve a higher damping. In order to damp-out the electromechanical oscillations of the two area four generator system, PSS is located at the Gen.-4 using the speeds of generator 3 (remote signal) and generator 4 (local signal). The addition of these two speeds are given to the PSS as the input signal for the PSS. The PSS design parameters are selected to increase the damping of the system to 15%. The time constant T_5 of the washout filter was selected as 10s, so that the signal with frequency of interest is unchanged. The values used to design the PSS are as follows,

- $k =$1.8
- $T_1 = T_3 =$0.725
- $T_2 = T_4 =$0.317
- $T_5 =$10 s
- $V_{STMAX} =$+5

- V_{STMIN} =-5

5.3 Communication Network

Communication network appropriate for the co-simulation of the selected power system application is designed using the new communication components in PSCAD. Each generation station is considered as a substation. The PMU models are connected to Bus-1, Bus-2, Bus-3, and Bus-4 and they are named as PMU_1, PMU_2, PMU_3, and PMU_4 respectively. The data source component is used to represent the data sources typically appear in a utility substation environment such as RTUs, IEDs and video units [21]. Video units are considered as the random packet generation sources while RTUs and IEDs are considered as the uniform packet generation sources. All the data sources including a PMU in each substation are connected to the substation switch and substation switches are connected to a switch in the control center. The communication network suitable for the power system application described in section 5.2 is shown in Fig. 5.3.1. The type of data source, packet rate, and packet size of each data source is shown in Table 5.2.

Table 5.2: Details of the data sources.

Data source	Type	Packet rate (packets/s)	Packet size (Bytes)
DS_1	RTU	30	600
DS_2	RTU	30	600
DS_3	Video unit	200	1100
DS_4	RTU	30	600
DS_5	Video unit	30	600
DS_6	RTU	30	600
DS_7	RTU	30	600
DS_8	Video unit	200	1100
DS_9	RTU	30	600
DS_{10}	RTU	40	600
DS_{11}	IED	50	800
DS_{12}	Video unit	200	1100

The lengths of the cable links used to connect PMUs and other data sources with the control center are as follows.

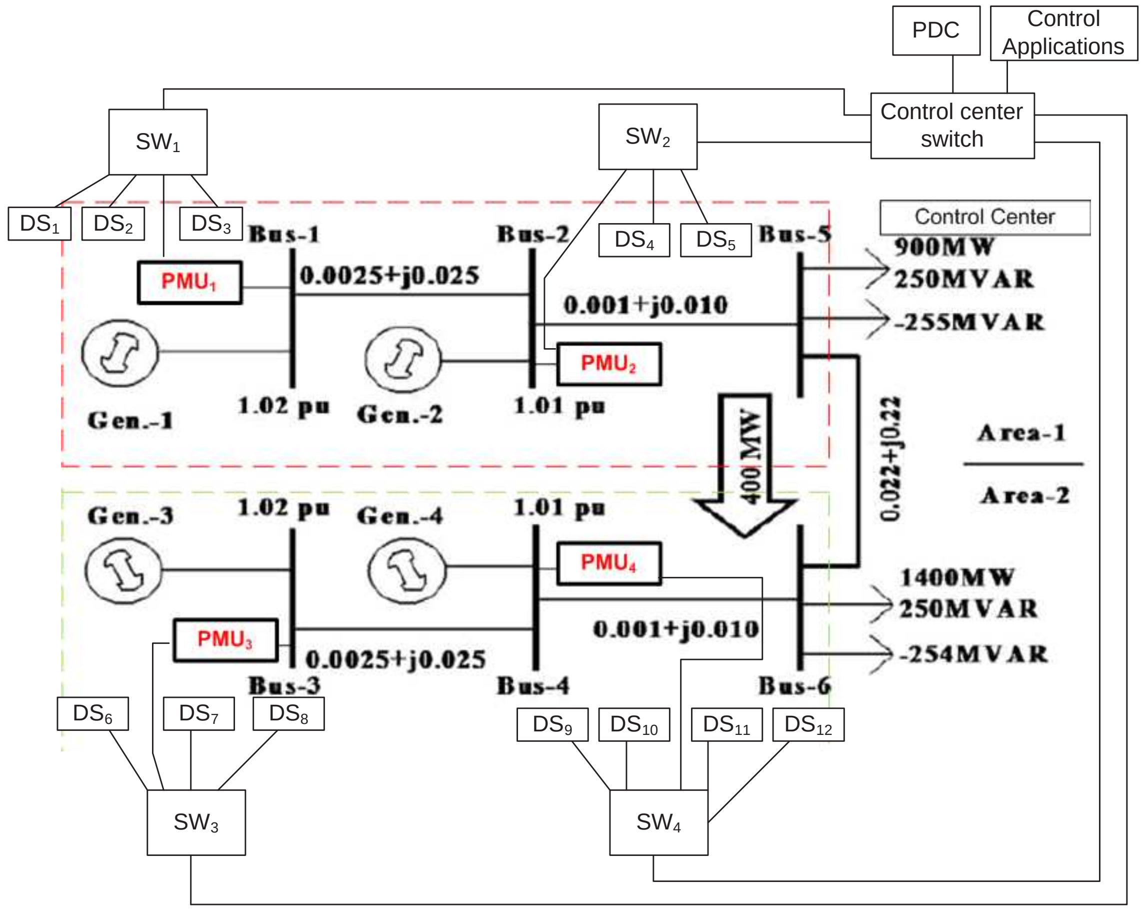


Figure 5.3.1: Block diagram of the communication network.

Table 5.3: Cable lengths in the communication network.

Cable Name	Length
l_8	100 m
l_4, l_6, l_{16}, l_{12}, l_{22}	200 m
l_2, l_{10}, l_{15}	250 m
l_3, l_5, l_9,l_{21}	300 m
l_1, l_7, l_{11},l_{14}	350 m
l_{13}	400 m
l_{17}	35 km
l_{18}	10 km
l_{19}	255 km
l_{20}	230 km

5.4 Co-Simulation Results

The PMUs in substations measure the current and voltage of the substation generator bus along with the speed of the the generator. Then, the data packets with these PMU measurements are transmitted to the control center via the communication system. Other data sources such as RTUs, IEDs and video units also send their data packets to the control center. PMUs and other data sources in the substation are connected to the substation switch using Ethernet cable links ($l_1, l_2, ..., l_{16}$) and substation switches are connected to the control center using fiber optic cables ($l_{17}, l_{18}, l_{19}, l_{22}$). Fiber optic links are used for the backbone communication links to provide more bandwidth. Data rate of the Ethernet cables and fiber optic cables are considered as 1 Mbps and 10 Mbps respectively. Control center is assumed to be located near the load center of 900MW. In the control center, system is monitored by looking at the phasor data extracted by the PDC. If a disturbance occurred at any point in the system, small signal oscillations can be seen. These small signal oscillations will be detected from the PMU measurements at the control center. When a stability issue or a low damping ratio is detected from the PMU measurements, control center decides the control action and apply the control action according to the type of disturbance. Then a control signal is transmitted to the relevant actuating device through the communication network. In order to damp-out the small signal oscillations in the two area four generator system, the appropriate control action is the application of the 3^{rd} and 4^{th} generator speeds to the PSS installed at the generator 4. Also, control or actuating device is the PSS installed in the generator 4. The speed of the generator 3 which is measured by PMU_3 will be received by the PDC in the control center via the communication network. That speed signal will be send to the PSS installed in generator 4 at substation 4. On the other hand, the speed of the generator 3 acts as the remote signal to the PSS. This is how the co-simulation works in order to damp out the small signal oscillations of the two area four generator system with the aid of associated communication network. The process of co-simulations is shown in Fig. 5.4.1 as a block diagram.

5.4.1 Case 1: Communication Network Under Normal Conditions

The effect of communication network on the damping of the oscillations is presented in this section. First, the behavior of the power system with no packet losses and bit errors

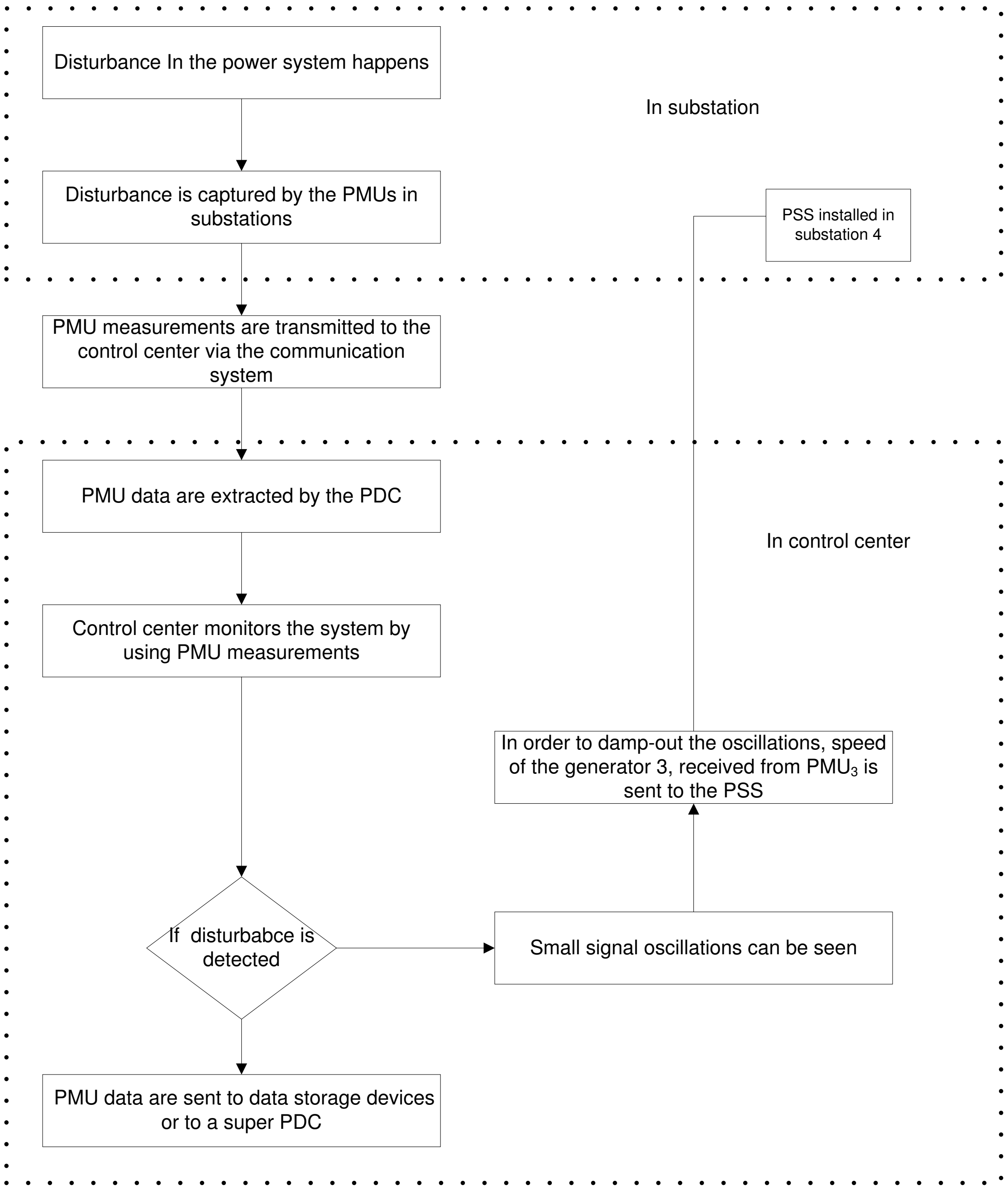


Figure 5.4.1: Block diagram of the co-simulation process for the two area four generator system.

is demonstrated. In this section, the bit error rates of the cable links are considered to be zero, i.e., a communication network without bit errors in the cable links. To avoid packet losses due to the switch buffer over flow, the buffer sizes of each substation switch and control center switch are set to serve all the packets entering the switch. Buffer sizes of the switches in the network are listed below.

- buffer size of the substation 1 switch (B_1)=100 packets
- buffer size of the substation 2 switch (B_2)=100 packets
- buffer size of the substation 3 switch (B_3)=100 packets
- buffer size of the substation 4 switch (B_4)=1000 packets
- buffer size of the control center switch (B_c)=1000 packets

Speed of Gen.-3 is measured under 3 conditions at the substation 3, i.e., without PSS (to demonstrate small signal oscillations), with PSS by using the speed of Gen.-4 as the input to the PSS (to demonstrate the damping out of oscillations using local signal only), and with PSS by using the summation of the speeds of Gen.-3 and Gen.-4 as the input to the PSS (to demonstrate the damping out of oscillations using local signal and remote signal). Also, speed of Gen.-3 measured at the control center to demonstrate the proper functioning of the communication network. Fig. 5.4.2 shows the speed of generator 3 at substation 3. This is the speed that PMU_3 measured at substation 3. Fig. 5.4.3 shows the speed of generator 3 at the control center, i.e., the speed measurement received to the control center from PMU data packets and via the communication network. Fig. 5.4.4 clearly reveals that small signal oscillations after applying a disturbance to the two area four generator power system, is properly captured from the active power measurements of the generator 4. It can be seen that, disturbance (sudden change in the excitation voltage of Gen.-2) has taken place at time=1s. Then small signal oscillations have occurred due to that disturbance. After that, PSS has damped out the oscillations in order to maintain the stability of the power system.

Fig. 5.4.4 demonstrates the small signal oscillations and the damping out of the oscillations after activating the PSS using local and remote signals. Fig. 5.4.2 and Fig. 5.4.4 clearly show that damping is improved after applying the remote signal to the PSS. That is an advantage of the implementation of synchrophasor based wide area power systems.

Communication delay of the associated communication network is shown in Fig. 5.4.6 and Fig. 5.4.7 for the speed signals of generator 3 and 4. It is clearly shown that communication delay of the speed signal of generator 4 is larger compared to the communication delay of speed signal of the generator 3. In chapter 4, (in Fig. 4.5.1) it is shown, when the buffer size increases, queuing delay also increases. Since, B_4 is high compared to the B_3, data packets from PMU_4 has longer queuing delays compared to the data packets from PMU_3. Also, Fig. 5.4.8 shows packet loss of each substation switch in the network. Since we set the buffer sizes of switches to serve all the packets enter to the the switches, no packet has been lost.

Fig. 5.4.3 and Fig. 5.4.5 clearly shows that, small signal oscillation can be detected properly at the control center, if the communication network perform without packet losses and bit errors in the cable links. There is no effect from the communication delay on identifying the status of the power system.

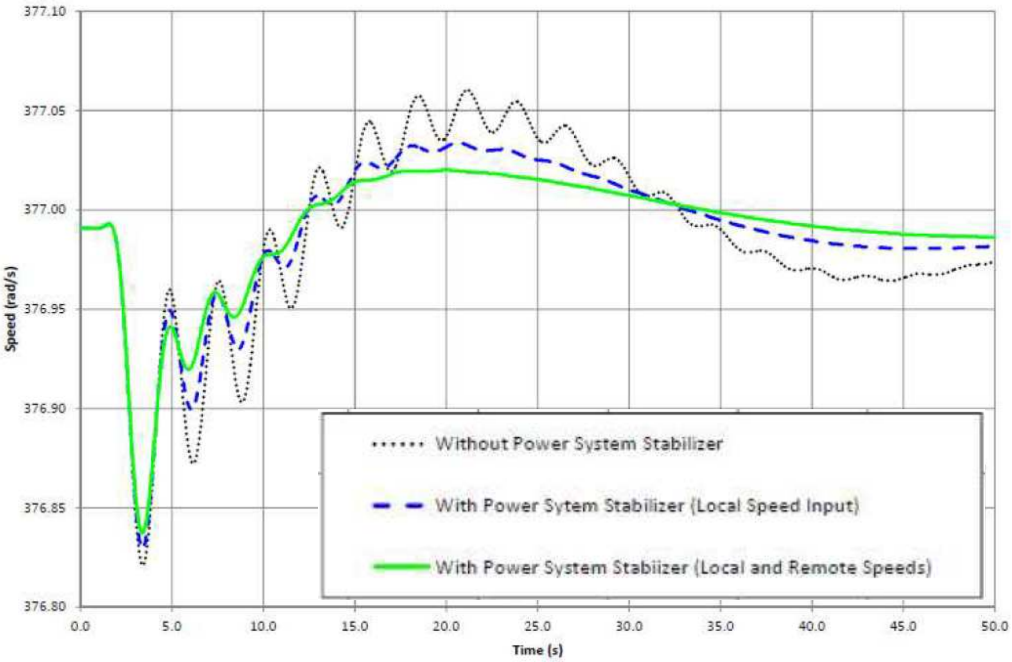


Figure 5.4.2: Speed of generator 3 measured at substation 3.

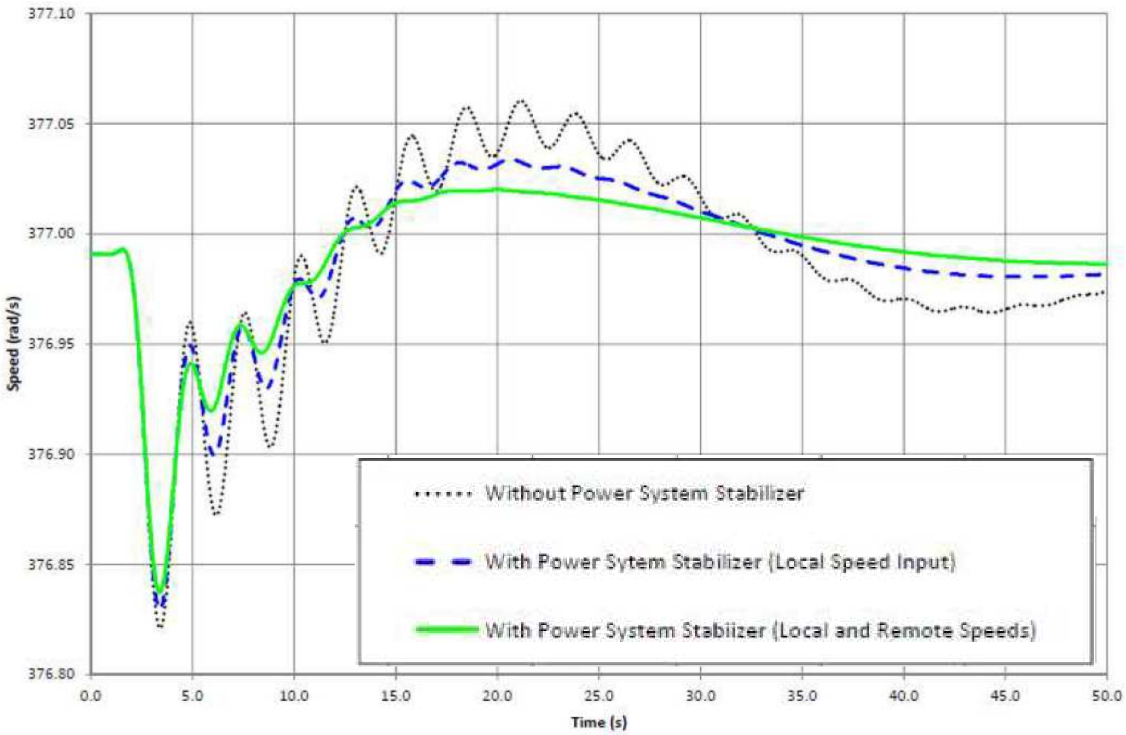


Figure 5.4.3: Speed of generator 3 measured at control center.

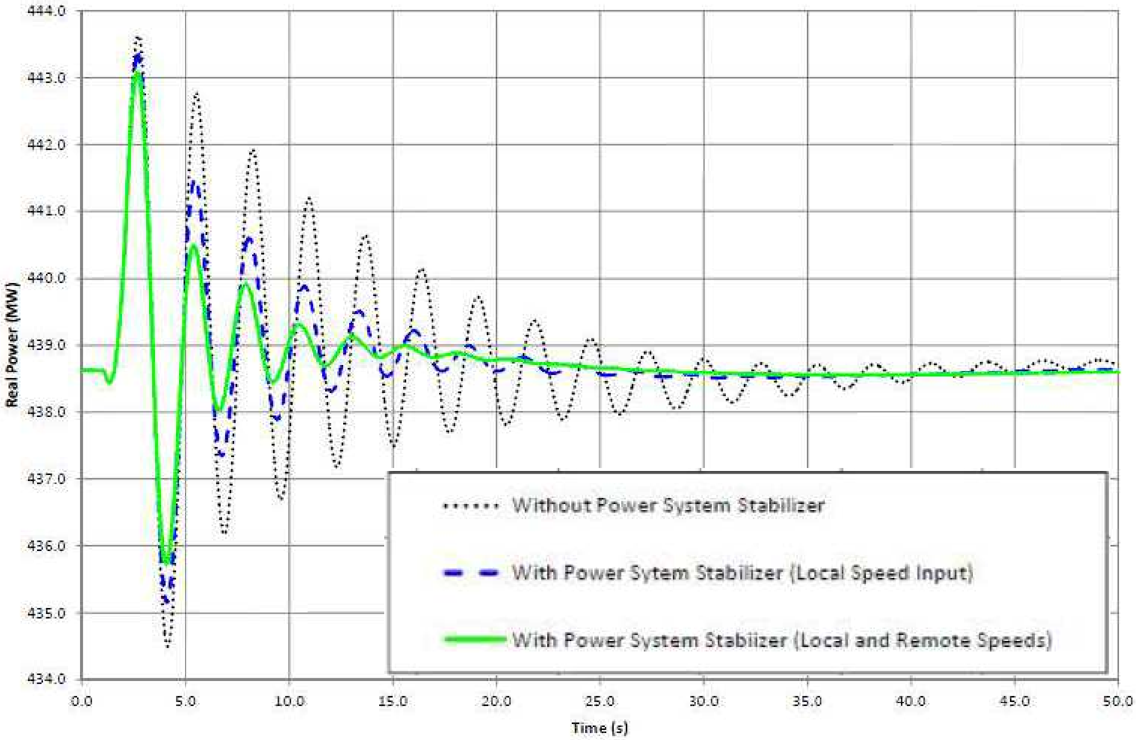


Figure 5.4.4: Power of generator 4 measured at substation 4.

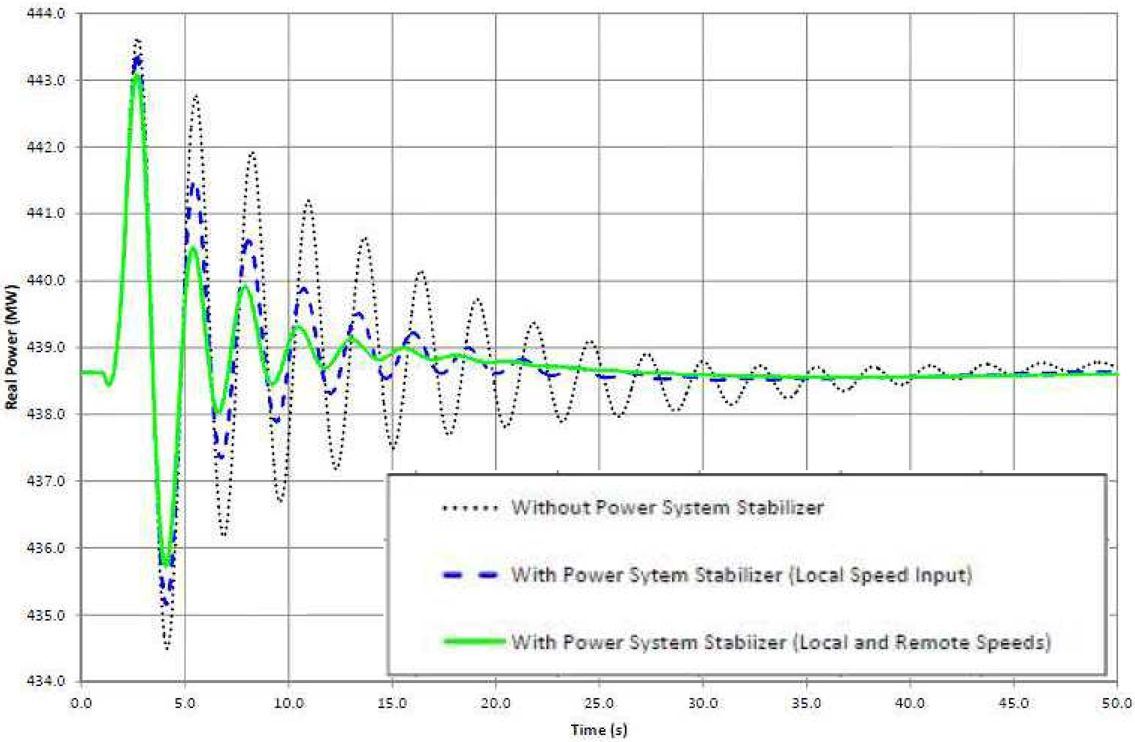


Figure 5.4.5: Power of generator 4 measured at control center.

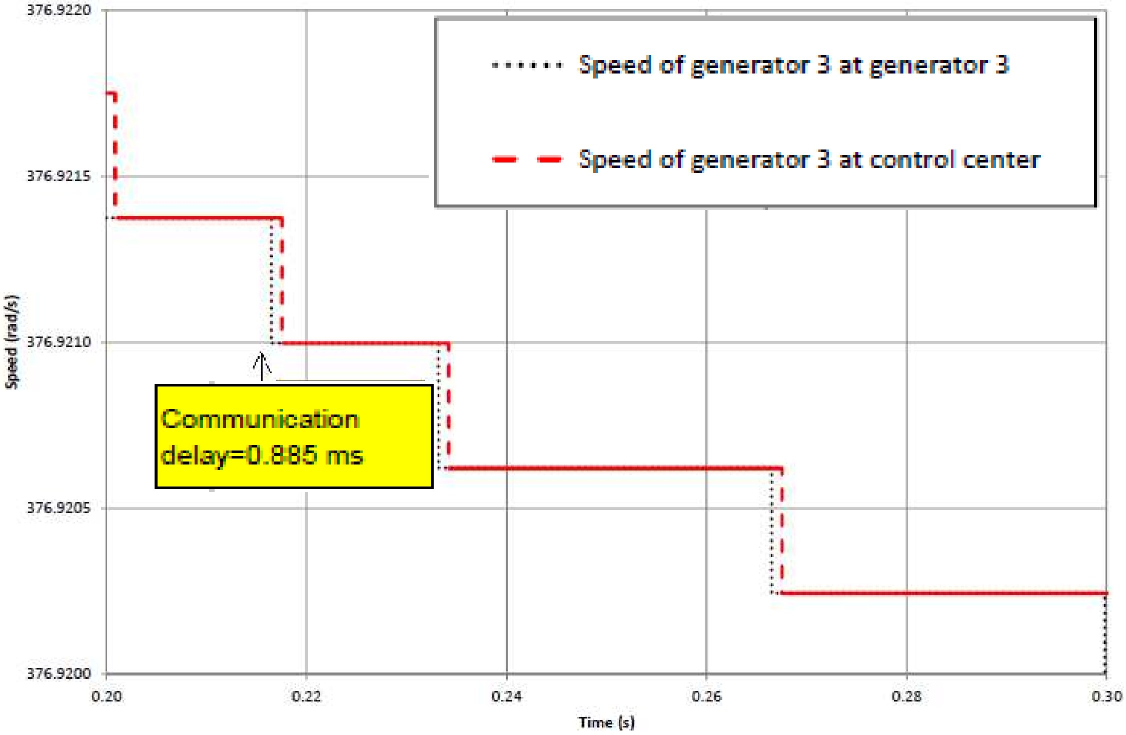


Figure 5.4.6: Speed of generator 3 with communication delay.

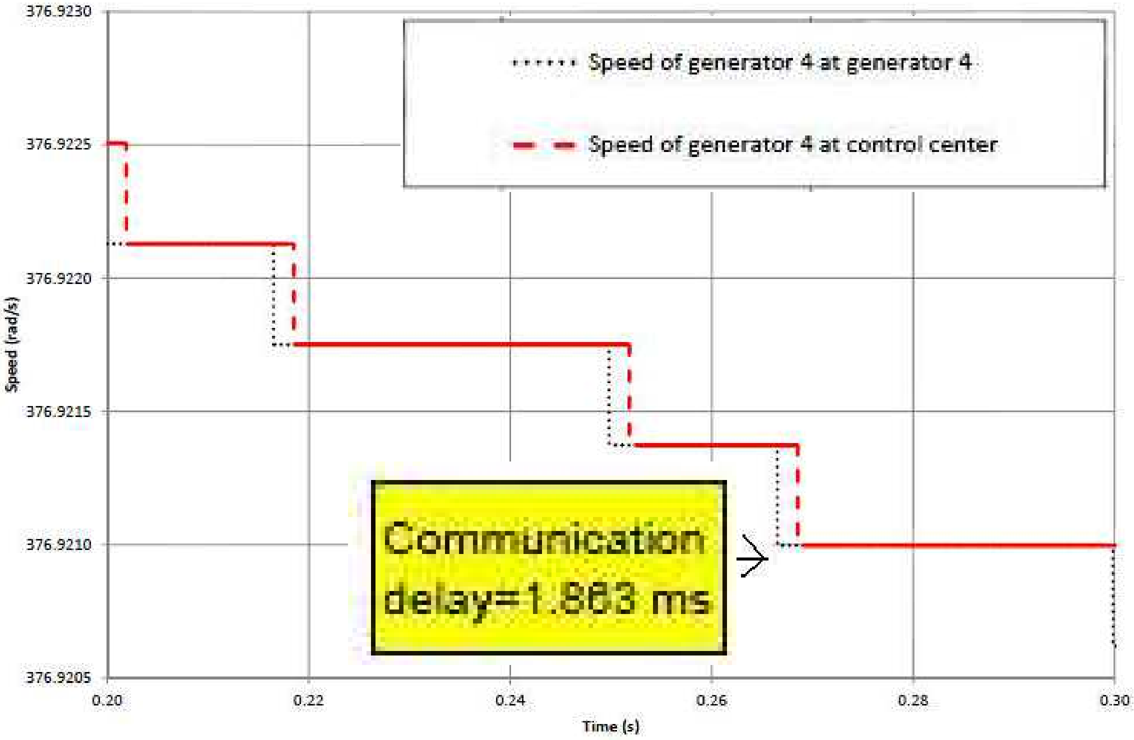


Figure 5.4.7: Speed of generator 4 with communication delay.

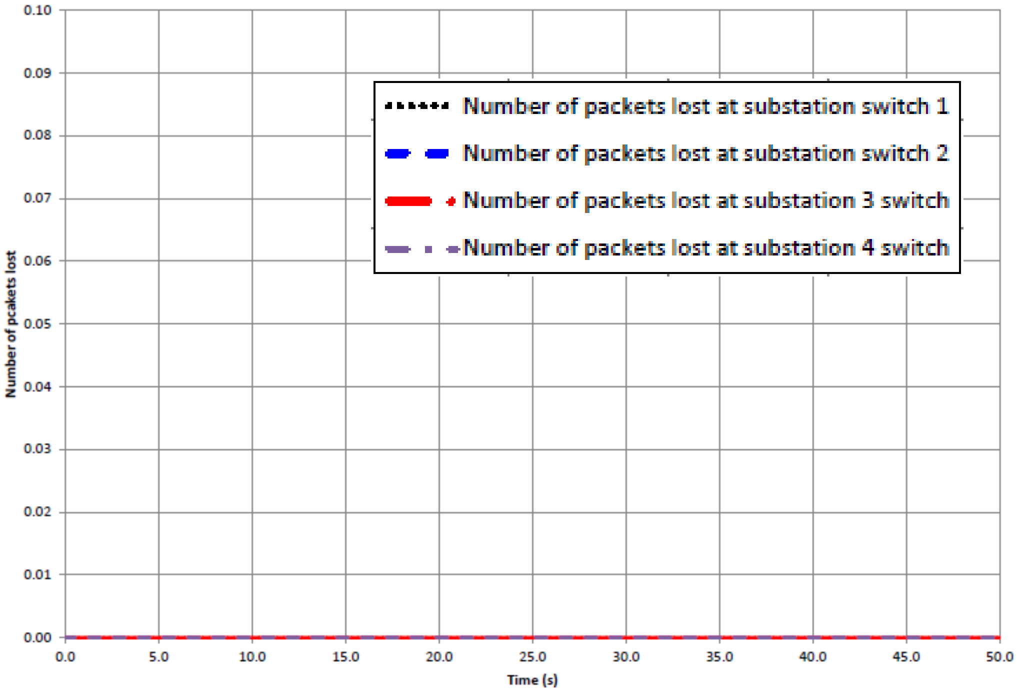


Figure 5.4.8: Number of packets lost under normal operation of the communication network.

5.4.2 Case 2: Communication Network Under Worst Conditions

Two scenarios are considered to demonstrate the effect of communication network on the damping-out of the oscillations.

Effect of Packet Losses

The data packets from PMU_3 will be lost due to the switch buffer overflow when more data sources are connected to the substation 3 switch. Six more data sources that generate 250 packets/s are connected to substation switch 3. Then, data packets from PMU_3 can be lost, due to high traffic to the switch. Therefore, the remote signal to the PSS is lost. Hence, PSS cannot provide the stability of the power system by damping-out the oscillations due to the loss of remote signal and system becomes unstable. This scenario is shown in Fig. 5.4.9 and Fig. 5.4.10 from the power system point of view, i.e., speed signal shows an abnormal variation due to packet loss and variations in active power of generator 4 shows that system becomes unstable due to the packet loss occurred at time=5.667s. Results indicate that loss of data due to switch buffer overflow has a critical effect on the power system operation, since loss of data would leave the power system with insufficient data to carry out its operation and as a result the stability has been lost.

Furthermore, co-simulation of the power system and the underlying communication network helps to identify the potential bottlenecks of the communication system that affects the power system application and design the communication network appropriately. If we want to design a communication network without packet losses, the bottlenecks in this communication network are the buffer size of the substation switch and the number of data sources connected to each substation switch. Therefore, Table 5.4 presents the safe margins for communication network parameters (Buffer size of the substation switch and the number data sources connected to the substation switch) in order to avoid the packet losses in the substation switches.

Also, it is important to decide the control action to maintain the stability of the power system, after identifying this kind of abnormal behavior in communication network. Therefore we have implemented a control operation for the input signal of the PSS in order to maintain the stability of the power system, if a packet loss occurred in the communication network. Algorithm 4 describes the proposed control operation to regulate the input speed signal within 10%. The packet lost occurred at time= 5.667 (before the implementation of control algorithm at the input to the PSS) and time= 12.4834 (after the implementation of the control algorithm to the PSS) is shown in Fig. 5.4.11.

Algorithm 4 Control action to maintain the stability of the power system after packet losses

```
Input:
I: Speed signal of generator 3 (remote signal to the PSS)
IF the remote speed signal for the PSS is less than 0.9 pu
speed = 0.9 pu
ELSE
pass the received remote speed signal to the PSS
Output: Output of the PSS.
```

If remote signal for the PSS is lost, PSS will rectify the effect of the data loss due to switch buffer overflow using the proposed control algorithm. It is shown in Fig. 5.4.10, and Fig. 5.4.9 by comparing the speed signal of generator 3 and power of generator 4 with the implementation of control action and without the control action described in Algorithm 4. As shown in Fig. 5.4.10, although the system stability has been lost before introducing the control action to the PSS input, stability has been maintained with the implementation of control action. Above correction is justified because the generator will be tripped by under speed protection if the speed drops below a threshold value.

Effect of Bit Errors

In order to investigate the effect of bit errors in the cable links, bit error rate of the link l_8 is set to 0.0001. As shown in Fig. 5.4.12 and Fig. 5.4.13 more oscillations can be seen due to the bit flip which has been taken place at time= 22.067s. But the system has not become unstable due to the bit errors. To eliminate the effect of bit errors, again we have implemented the control operation described in Algorithm 5. After we apply the control

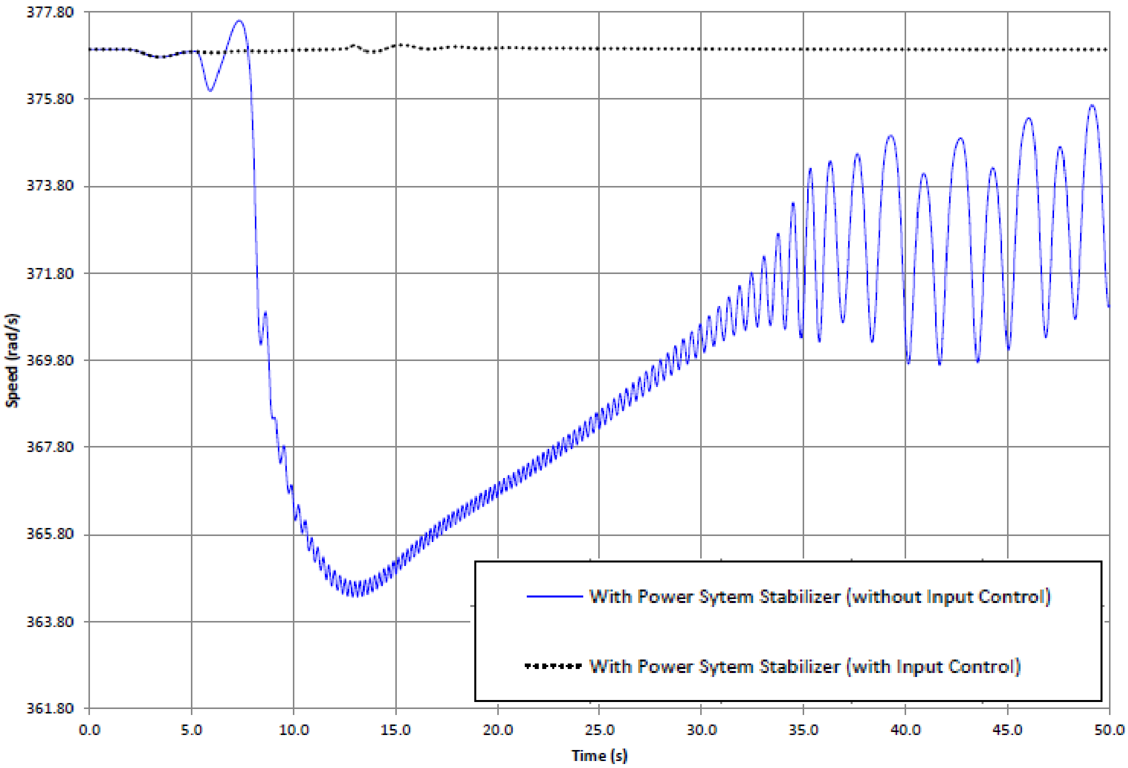


Figure 5.4.9: Effect of packet loss on speed of generator 3.

Table 5.4: Safe margins to avoid packet losses.

Switch	Buffer size by keeping number of data sources constant	Number of data sources by keeping buffer size constant
Substation 1 switch	25	By adding 6 more data sources with data rate of 250 packets/s
Substation 2 switch	18	By adding 9 more data sources with data rate of 250 packets/s
Substation 3 switch	25	By adding 6 more data sources with data rate of 250 packets/s
Substation 4 switch	32	By adding 17 more data sources with data rate of 250 packets/s

action to the PSS, again a bit error has occurred at time= 9.2167s and PSS has rectified the oscillations due to the bit flip using the control algorithm implemented at the input to the PSS signal.

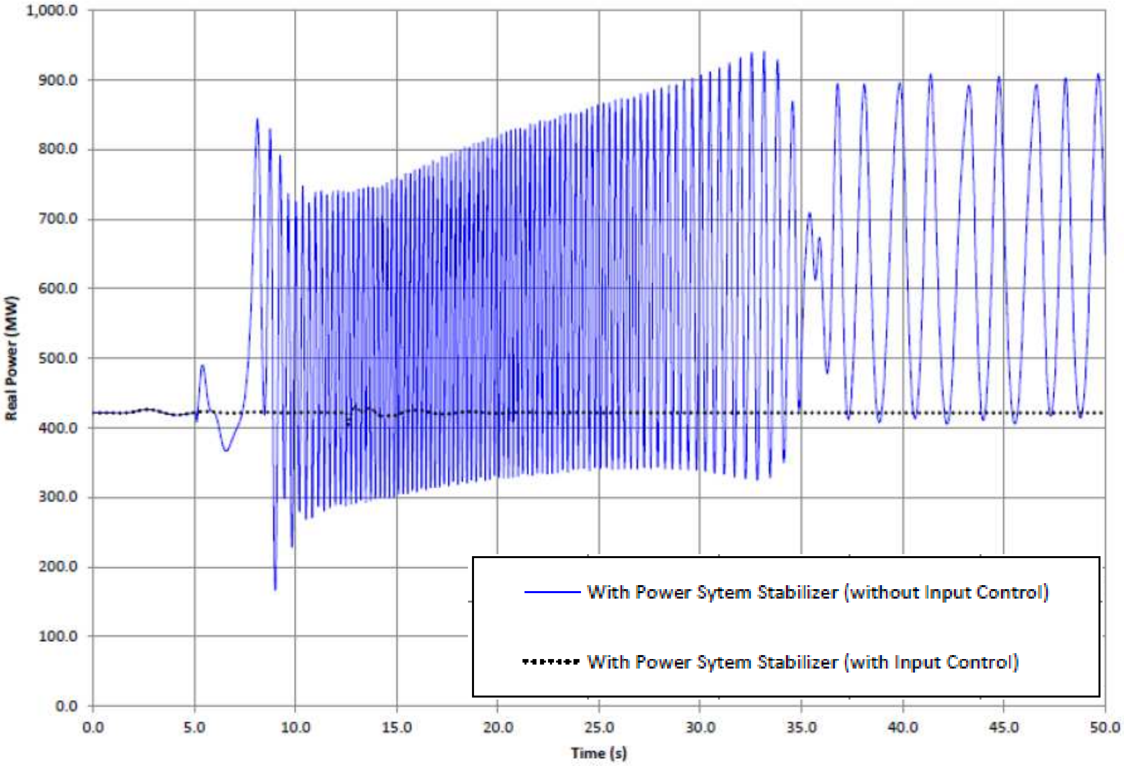


Figure 5.4.10: Effect of packet loss on active power of generator 4.

5.5 Conclusions

Using the co-simulation of a selected power system application and the underlying communication network, we have demonstrated that effect of various conditions of the communication network on the operation of two area four generator power system. Since we used the fiber optic links with 10 Mbps data rate for the communication links between substation switches and the control center, the effect of the propagation delay due to the distance between substations and control center is negligible on the monitoring, control, and protection applications of this power system. However, packet losses and bit errors have a considerable effect on the stability of the power system, and thereby, we have come up with an algorithm as a solution for such scenarios. In the communication network point of view, it is clearly shown that buffer size of the substation switch and the number of data sources connected to each substation switch are the bottlenecks identified in the communication network. These bottlenecks have been identified by the co-simulation.

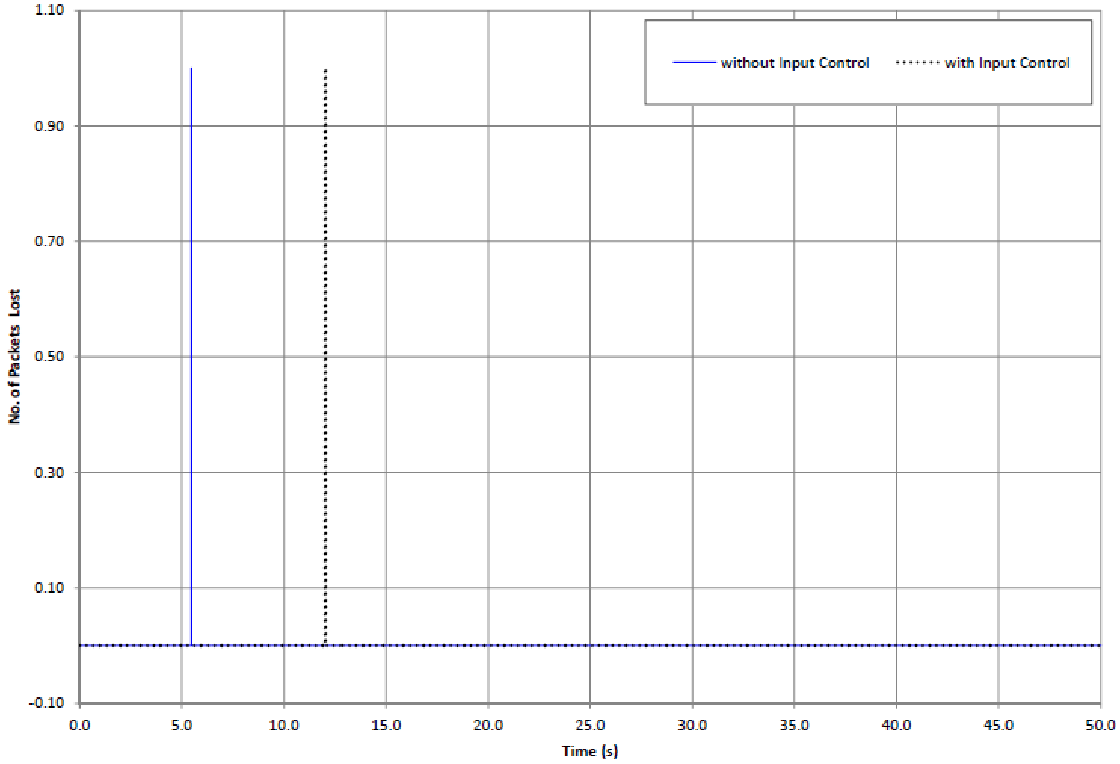


Figure 5.4.11: Packet loss in the communication network.

Algorithm 5 Control action to maintain the stability of the power system after bit errors take place

Input:

I: Speed signal of generator 3 (remote signal to the PSS)

```
IF the remote speed signal for the PSS is outside the limit of 0.9 pu or 1.1 pu
speed = 0.9 pu or 1.1 pu
ELSE
pass the received remote speed signal to the PSS
```

Output: Output of the PSS.

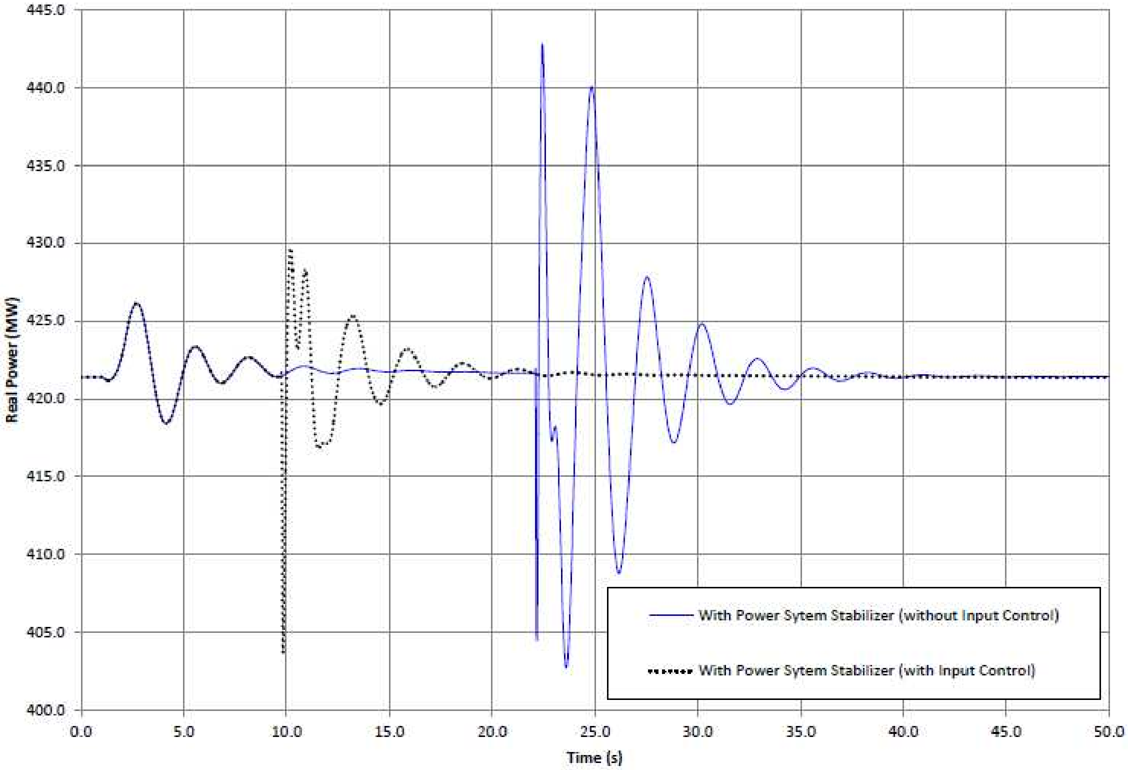


Figure 5.4.12: Effect of bit errors on active power of generator 4.

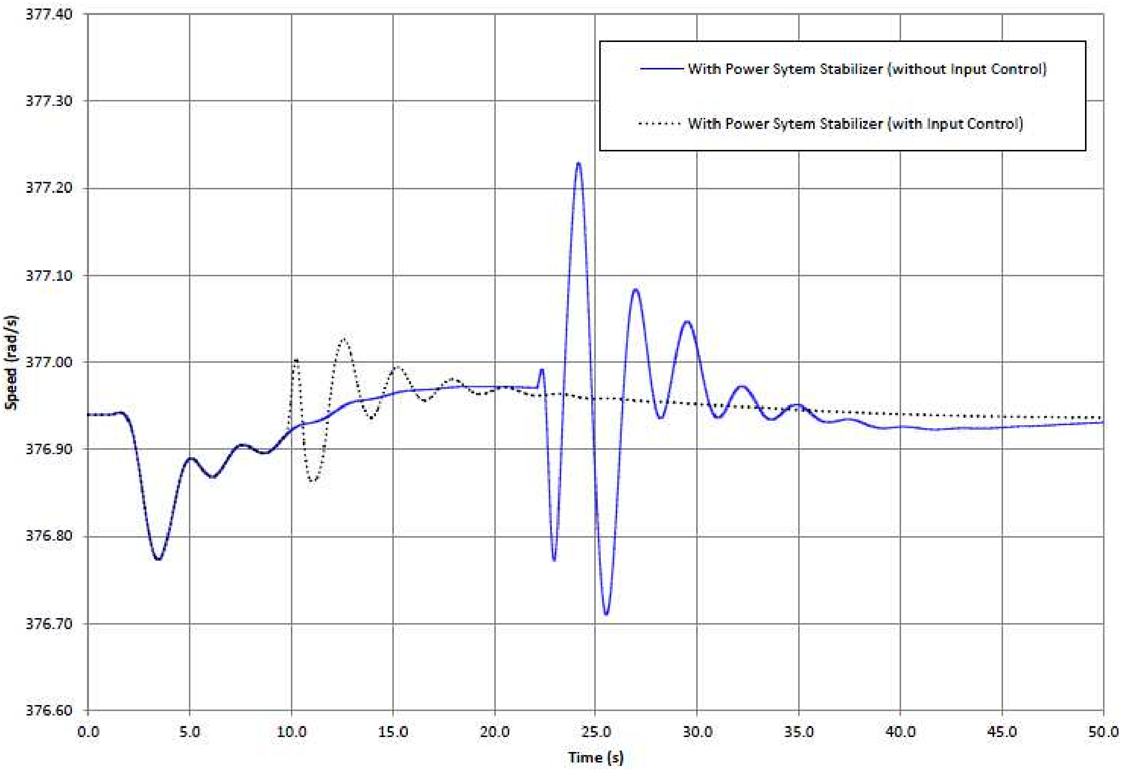


Figure 5.4.13: Effect of bit errors on speed of generator 3.

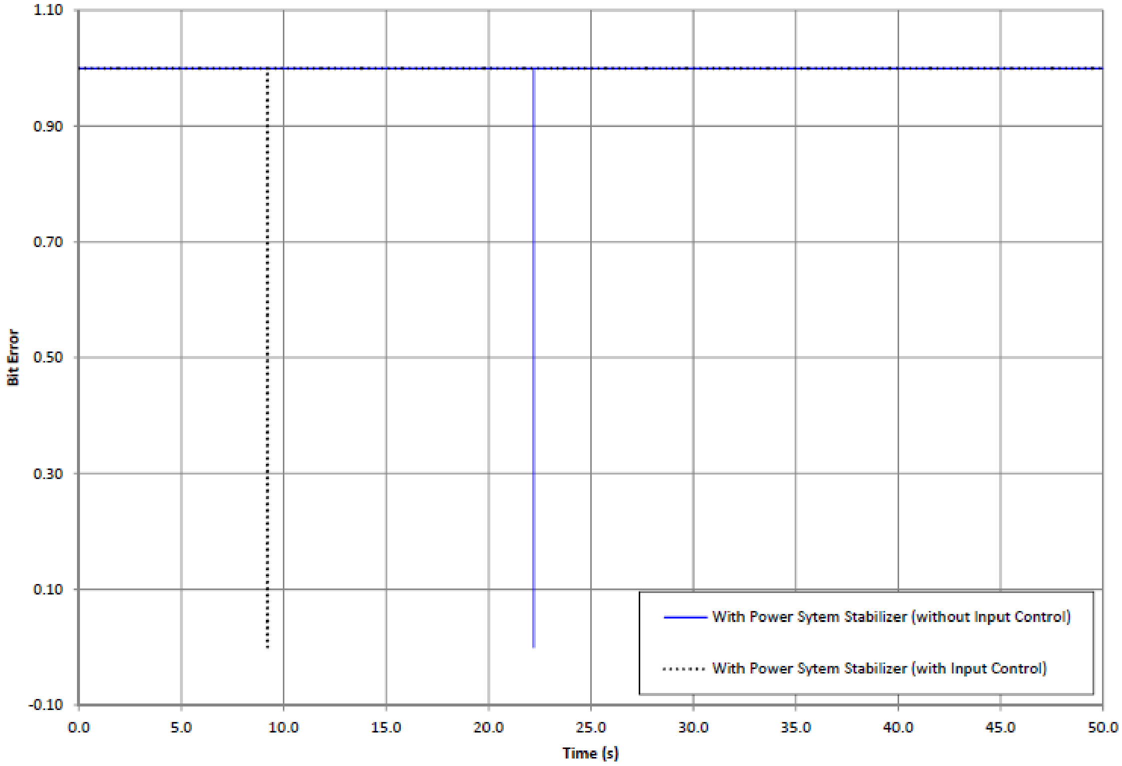


Figure 5.4.14: Bit error occurred to the 218^{th} bit in the data packet of PMU_3.

Chapter 6

Conclusion and Future Work

This thesis has mainly focused on investigating the effect of communication network on the power system operation. Hence, a background study and a literature review of the communication needs of power system applications has been carried out in order to identify the potential bottlenecks and the characteristics of a communication network that have major effects on wide area power systems. Based on that study, a set of communication components have been developed for the PSCAD/EMTDC power system simulation software, which can simulate the communication delay and packet losses. Furthermore, an analytical method based on queuing theory has also been developed to evaluate the communication delay and packet loss probability of a typical PMU-PDC communication network. Finally, the communication components developed in this thesis have been integrated into the simulation of a wide area power system application investigate the effect of communication network parameters on the power system operation.

It has been shown that co-simulation of the wide area power system and its underlying communication network clearly demonstrate the effect of the communication network parameters on the power system operation. Co-simulation also helps to identify the potential bottlenecks of the communication system and optimize the operation of the power system application with the available communication resources. The analytical method this thesis using queuing theory can be used for calculation of communication delay and packet loss probability of a given PMU-PDC communication network with any number of PMUs or other data sources such as RTUs, IEDs, etc. The PSCAD simulation results agree with the results of the proposed analytical model and the application of those components has been demonstrated.

The research work presented in this thesis can be expanded to develop a library of network modeling tools to support the simulation of various industry standard communication protocols (e.g. IEC 61850 used in intra-substation networks), network architectures, and transmission technologies (e.g. wireless) widely used in the power industry.

Bibliography

[1] M. Chenine, E. Karam and L. Nordstrom, “Modeling and Simulation of Wide Area Monitoring and Control Systems in IP-based Networks,” *IEEE Power Energy Society General Meeting, PES ’09.*, pp. 1-8, July. 2009.

[2] Yi Deng, Hua Lin, A.G. Phadke, S. Shukla, J.S. Thorp, and L. Mili, “Communication network modeling and simulation for Wide Area Measurement applications,” *IEEE PES 2012*, pp. 1-6, Jan. 2012.

[3] Hua Lin, S. Sambamoorthy, S. Shukla, J. Thorp, and L. Mili, “Power system and communication network co-simulation for smart grid applications,” *2011 IEEE PES, Innovative Smart Grid Technologies (ISGT)*, pp. 1-6, Jan 2011.

[4] D. Bertsekas and R. Gallager, “Data Networks,” *2nd Edition, Prentice-Hall*.

[5] N. Ince and A. Brag, “Recent Advances in Modeling and Simulation Tools For Communication Networks and Services,” *Springer Science*,2007.

[6] OPNET Inc., “OPNET Modeler,” *www.opnet.com.*

[7] Z. Wang, A. Scaglione, and R. J. Thomas, “Generating statistically correct random topologies for testing Smart Grid communication and control networks,” *IEEE Transactions on Smart Grid*, pp. 28-39, June 2010.

[8] M. Chenine and L. Nordstrom, “Modeling and Simulation of Wide-Area Communication for Centralized PMU-Based Applications,” *IEEE Transactions on Power Delivery*, vol. 26, no. 3, pp. 1372-1380, July. 2011.

[9] Hongga Zhao, “A New State Estimation Model of Utilizing PMU Measurements,” *International Conference on Power System Technology 2006*, pp. 1-5, Oct. 2006.

[10] S. Wang, X. Meng and T. Chen, “Wide-Area Control of Power Systems Through Delayed Network Communication,” *IEEE Transactions on Control Systems Technology*, vol. 20, no. 2, pp. 495-503, Mar. 2012.

[11] R. Lira, C. Mycock, D. Wilson and H. Kang, “PMU performance requirements and validation for closed loop applications,” *IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies (ISGT Europe)*, pp. 1-7, Dec. 2011.

[12] Zhenyu Huang, J. F. Hauer and K. E. Martin, “Evaluation of PMU Dynamic Performance in Both Lab Environments and under Field Operating Conditions,” *IEEE Power Engineering Society General Meeting, 2007*, pp. 1-6, June. 2007.

[13] “IEEE Standard for Synchrophasor Data Transfer for Power Systems,” *IEEE Std C37.118.2-2011 (Revision of IEEE Std C37.118-2005)*, pp. 1-53, 2011.

[14] A. G. Phadke and J. S. Thorp, “Communication needs for Wide Area Measurement Applications,” *5th International Conference on Critical Infrastructure (CRIS)*, pp. 1-7, Sept. 2010.

[15] M. Chenine, Kun Zhu and L. Nordstrom, “Survey on Priorities and Communication Requirements for PMU-based Applications in the Nordic Region,” *IEEE PowerTech, 2009 Bucharest*, pp. 1-8, July. 2009.

[16] G.N. Ericsson, “Modeling of power system communications-recognition of technology maturity levels and case study of a migration scenario,” *IEEE Transactions on Power Delivery*, vol. 19, no. 1, pp. 105-110, Jan. 2004.

[17] G.L. Yu, B.H. Zhang, H. Xie and C.G. Wang, “Wide-Area Measurement-Based Nonlinear Robust Control of Power System Considering Signals’ Delay and Incompleteness,” *IEEE Power Engineering Society General Meeting, 2007*, pp. 1-8, June 2007.

[18] K. Narendra, T. Weekes, “Phasor Measurement Unit (PMU) Communication Experience in a Utility Environment,” *Conference on power systems, CIGRE Canada*, pp. 6-13, Oct. 2008.

[19] D. Dolezilek, “Case Study Examples of Interoperable Ethernet Communications within Distribution, Transmission, and Wide-Area Control Systems,” *IEEE International Conference on Communications Workshops*, pp.1-7, May 2010.

[20] V. Skendzic and R. Moore, “Extending the Substation LAN Beyond Substation Boundaries: Current Capabilities and Potential New Protection Applications of Wide-Area Ethernet,” *IEEE Power Systems Conference and Exposition*, pp. 641-649, Nov. 2006.

[21] M. Chenine, I. Al Khatib, J. Ivanovski, V. Maden and L. Nordstrom, “PMU Traffic Shaping in IP-Based Wide Area Communication,” *5th International Conference on Critical Infrastructure (CRIS)*, pp. 1-6, Sept. 2010.

[22] K.A. Fahid, P. Gopalakrishnan and S. Cherian, “ A High Performance Network Communications Architecture for Synchrophasor Data Transfer in Wide Area Monitoring, Protection and Control Applications,” *Bulk Power System Dynamics and Control - VII. Revitalizing Operational Reliability, 2007 iREP Symposium*, pp. 1-4, Aug. 2007.

[23] M. Chenine and L. Nordstrom, “Investigation of Communication Delays and Data Incompleteness in Multi-PMU Wide Area Monitoring and Control Systems,” *International Conference on Electric Power and Energy Conversion Systems, EPECS ’09.*, pp. 1-6, Nov. 2009.

[24] B. Naduvathuparambil, M. C. Valenti and A. Feliachi, “Communication delays in wide area measurement systems,” *Proceedings of the Thirty-Fourth Southeastern Symposium on System Theory*, pp. 118-122, 2002.

[25] K. Hopkinson, et al., “EPOCHS: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components,” *IEEE Transactions on Power Systems*, vol. 21, pp. 548-558, 2006.

[26] J. Nutaro, et al., “Integrated Hybrid-Simulation of Electric Power and Communications Systems,” *IEEE Power Engineering Society General Meeting, 2007.*, pp. 1-8, 2007.

[27] C. H. Hauser, D. E. Bakken, and A. Bose, “A Faliure to communicate,” *IEEE Power and Energy Magazine*, pp. 47-55, Mar. 2005.

[28] C. Chuang, J. Jiang, Y. Wang, C. Chen and Y. Hsiao, “An Adaptive PMU-based Fault Location Estimation System with a Fault-Tolerance and Load-Balancing Communication Network,” *IEEE Lausanne Power Tech*, pp. 1197-1202, July 2007.

[29] P. Kunder, “Power System Stability and Control,” *New York: McGraw-Hill*, 1994.

[30] M.R. Iravani, “Modeling and simulation of a static transfer switch,” *IEEE Power Engineering Society Winter Meeting*, vol. 2, pp. 643-646, 2001.

[31] M. R. Aghaebrahimi and R.W. Menzies, “A customized air-core transformer for a small power tapping station,” *IEEE Transactions on Power Delivery*, vol. 13, no. 4, pp. 1265-1270, Oct. 1998.

[32] G. J. Li, T. T. Lie, C.B. Soh, and G. H. Yang, “Decentralised nonlinear H infin; control for stability enhancement in power systems,” *IEEE Proceedings on Generation, Transmission and Distribution*, vol. 146, no. 1, pp. 19-24, Jan. 1999.

[33] “IEEE Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN - Specific Requirements - Part 15: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs) ,” *IEEE Std 802.15.1-2002*, pp. 1-473, 2002.

[34] J. D. Day and H. Zimmermann, “The OSI reference model ,” *IEEE Proceedings*, vol. 71, no. 12, pp. 1334-1340, March. 1983.

[35] L. Zier, W. Fischer and F. Brockners, “Ethernet-based public communication services: challenge and opportunity ,” *IEEE Communications Magazine*, vol. 42, no. 3, pp. 88-95, March. 2004.

[36] D. Dolezilek, “Case Study Examples of Interoperable Ethernet Communications within Distribution, Transmission, and Wide-Area Control Systems,” *IEEE International Conference on Communications Workshops (ICC)*, pp. 1-7, May. 2010.

[37] M. M. Eissa, M. E. Masoud and M. M. M. Elanwar, “A Novel Back Up Wide Area Protection Technique for Power Transmission Grids Using Phasor Measurement Unit,” *IEEE Transactions on Power Delivery*, vol. 25, no. 1, pp. 270-278, Jan. 2010.

[38] A. Veluswami, Qi-Jun Zhang and M. S. Nakhla, “A neural network model for propagation delays in systems with high speed VLSI interconnect networks,” *IEEE Proceedings of the Custom Integrated Circuits Conference*, pp. 387-390, May. 1995.

[39] N. M. A. Ayad and F. A. Mohamed, “Performance analysis of a cut-through vs. packet-switching techniques,” *IEEE Proceedings on Computers and Communications*, pp. 230-234, Jul. 1997.

[40] S. Amstutz, “A New Store and Forward Message Switching System,” *IEEE Transactions on Communication Technology*, vol. 19, no. 4, pp. 528-529, Aug. 1971.

[41] S. Varadarajan and T. Chiueh, “EtheReal: a host-transparent real-time Fast Ethernet switch,” *Sixth International Conference on Network Protocols*, pp. 12-21, Oct. 1998.

[42] Cheung-Wing Chan and Tak-Shing Yum, “An Algorithm for Detecting and Resolving Store-and-Forward Deadlocks in Packet-Switched Networks,” *IEEE Transactions on Communications*, vol. 35, no. 8, pp. 801-807, Aug. 1987.

[43] A. J. Lawrance, “Is there life after bit error rate or before?,” *IEEE International Symposium on Circuits and Systems*, pp. 185-188, May. 2006.

[44] Luc Dvroye, “Non-uniform random variate generation,” *Springer-Verlag*, 1986.

[45] Mu Wei and Zhe Chen, “Distribution System Protection with Communication Technologies,” *IEEE 36th Annual Conference on Industrial Electronics Society- IECON*, pp. 3328-3333, Nov. 2010.

[46] B. Van Houdt and C. Blondia, “Polling Systems: Applications, Modeling, and optimization,” *Stochastic Models*, vol. 38, no. 10, pp. 1750-1760, Oct. 1990.

[47] B. Van Houdt and C. Blondia, “The Delay Distribution of a Type k Customer in a FCFS MMAP[K]/PH[K]/1 Queue,” *Journal of applied Probability (JAP)*, vol. 39, no. 1, Mar. 2002.

[48] I. Frigui, R. Stone and A.S. Alfa, “Message delay for a priority-based automatic meter reading network,” *Computer communications*, vol. 20, pp. 38-47, 1994.

[49] A. Kohnheim and B. Meiser, “Waiting lines and times in a system with polling,” *JACM.*, vol. 21, 1974.

[50] H. Levy and M. Sidi, “The Waiting Time Distribution of a Type K customer in a Discrete Time FCFS MMAP[K]/PH[K]/c(c=1,2) Queue Using QBD,” *IEEE Transactions on Communications*, vol. 20, no. 1, pp. 55-69, 2004.

[51] Attahiru Sule Alfa, “Queueing Theory for Telecommunications: Discrete Time Modelling of a Single Node System,” *Springer, New York*, 2010.

[52] J. D. Decotignie, “Ethernet-Based Real-Time and Industrial Communications,” *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1102-1117, June. 2005.

[53] T. Xu and X. Yin and D. You and Y. Li and Y. Wang, “A novel communication network for three-level wide area protection system,” *IEEE PES 2008*, vol. 39, no. 1, pp. 1-8, Jul. 2008.

[54] S. Ross, “Simulation,” *Academic Press*, 4th Edition, 2006.

[55] Tsung-Yin Wang, Jau-Chuan Ke and Fu-Min Chang, “A Discrete-Time Queue with Modified Vacation Policy,” *Fourth International Joint Conference on Computational Sciences and Optimization (CSO)*, pp. 132-136, Apr. 2011.

[56] P. Kundur, “Power System Stability and Control,” *McGraw-Hill*, 2010.

[57] N. Kshatriya, U. D. Annakkage, A. M. Gole, and I. T. Fernando, “A comparison of the use of participation factors and residues for design and location of power system stabilizers,” *X SEPOPE*, 2006.

Appendix A

A.1 Fortran Code for the Cable Link

The main Fortran code is named as Ethernet Cable, and two sub programs are used for the data recreation (when a bit error occur) and to convert decimal to binary values.

```
SUBROUTINE EthernetCable(IN1,OUT1,D,R1,R2,P,Er,Ch)
INCLUDE ’nd.h’
INCLUDE ’s1.h’     ! TIME variables
INCLUDE ’s2.h’
INCLUDE ’emtconst.h’
INCLUDE ’emtstor.h’

INTEGER IN1(12), OUT1(12), R2
REAL Ch,D, R1, P, Er, L,Prop_const
REAL u, u2, no_of_bits, u1,u3
INTEGER Inter_output_array(12)
INTEGER i, k, fliping_packet, Tot_bits
INTEGER My_NSTORI
REAL My_NSTORF
REAL propDelay, delay, TotalDelay, T_old
```

```
INTEGER Fliping(NINT(8*(IN1(2))*Er))
INTEGER Fliping_Bits(8*IN1(2))4
INTEGER Fliping_Bits2(8*IN1(2))
INTEGER a, c, b
My_NSTORI=NSTORI
My_NSTORF=NSTORF
NSTORF=NSTORF+20
NSTORI=NSTORI+20
seed = 7654321

IF (TIME==0) THEN
DO k=1,12
OUT1(k)=0
END DO
STORI(My_NSTORI + 22) = 3337
END IF

u1=ran(seed)
fliping_packet=NINT(u1*R2)
T_old=STORF(My_NSTORF)
no_of_bits=NINT(8*(IN1(2))*Er)
Tot_bits=8*IN1(2)

a=9
c=17
b=3328

DO i=1,Tot_bits
```

```
Fliping_Bits(i)=1
END DO

DO i=1,no_of_bits
STORI(My_NSTORI+22)=MOD(((a*STORI(My_NSTORI+22))+c),b)
u3=(STORI(My_NSTORI+22))
u2=(u3/b)
Fliping(i)=NINT((8*IN1(2))*u2)
DO k=1,Tot_bits
IF (k==Fliping(i)) THEN
Fliping_Bits(k)=0
END IF
END DO
END DO

!data re creation
IF ((IN1(6)>14)) THEN
IF ((STORI(My_NSTORI))==fliping_packet) THEN
CALL data_recreation(IN1, Er, Fliping_Bits, Inter_output_array)
DO k=1,12
STORI(My_NSTORI+k)=Inter_output_array(k)
END DO
STORI(My_NSTORI)=0
ELSE
DO k=1,12
STORI(My_NSTORI+k)=IN1(k)
END DO
END IF
```

```
!Calculation of communication delay
IF (P==0) THEN
L=(IN1(2)*8)+300
ELSE
L=(IN1(2)*8)+200
END IF

IF (Ch==0) THEN
Prop_const=2165E-9
ELSE
Prop_const=1000E-9
END IF

propDelay=(D*1000/500)*Prop_const delay=propDelay+(L/(R1*1E6))
TotalDelay=TIME+delay
TotalDelay=(NINT(TotalDelay*1E6))*(1E-6)
STORF(My_NSTORF)=TotalDelay
STORI(My_NSTORI)=STORI(My_NSTORI)+1
END IF

IF ((TIME .GE.((T_old)-0.000001)) .AND. (TIME .LE. ((T_old)+0.000001))) THEN
DO i=1,12
OUT1(i)=STORI(My_NSTORI+i)
END Do

ELSE
DO i=1,12
OUT1(i)=0
```

```
END DO
END IF

RETURN
END
```

A.1.1 Fortran Code for Data Recreation

```
SUBROUTINE data_recreation(IN1, Er, Fliping_Bits, Inter_output_array)

INCLUDE ’nd.h’
INCLUDE ’s1.h’ ! TIME variables
INCLUDE ’s2.h’
INCLUDE ’emtconst.h’
INCLUDE ’emtstor.h’

INTEGER IN1(12),k
REAL u,u2,Er
INTEGER nu
INTEGER i,y,sum_of_binary, sum_of_binary1, sum_of_binary2
INTEGER, DIMENSION(0:15) :: array1
INTEGER, DIMENSION(0:15) :: array3
INTEGER, DIMENSION(0:31) :: array2
INTEGER Fliping(NINT(8*(IN1(2))*Er))
INTEGER W,nu1,nu2
INTEGER Inter_output_array(12)
INTEGER nu3
INTEGER Fliping_Bits(8*IN1(2))
```

```
DO W=1,12

IF (W==1) THEN
nu=(IN1(1))
CALL DEC_to_bin_1(nu, array1)
DO i=1,16
IF (Fliping_Bits(i)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
END IF
END DO

sum_of_binary =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary =sum_of_binary +y
END DO

Inter_output_array(1)=sum_of_binary

END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==2) THEN
```

```
nu=(IN1(2))
CALL DEC_to_bin_1(nu,array1)
DO i=1,16
IF (Fliping_Bits(i+16)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

sum_of_binary =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary =sum_of_binary +y
END DO

Inter_output_array(2)=sum_of_binary
END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==3) THEN
nu=IN1(3)
CALL DEC_to_bin_1(nu, array1)
DO i=1,16
```

```
IF (Fliping_Bits(i+32)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

sum_of_binary =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary =sum_of_binary +y
END DO

Inter_output_array(3)=sum_of_binary

END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==4) THEN
nu1=IN1(4)
nu=(nu1/32767)
nu2=MOD(nu1,32767)
CALL DEC_to_bin_1(nu, array1)
CALL DEC_to_bin_2(nu2, array3)
```

```
DO i=1,16
IF (Fliping_Bits(i+48)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

DO i=1,16
IF (Fliping_Bits(i+64)==0) THEN
IF (array3(i-1)==0) THEN
array3(i-1)=1
ELSE
array3(i-1)=0
END IF
ELSE
array3(i-1)=array3(i-1)
END IF
END DO

sum_of_binary1 =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
```

```
sum_of_binary1 =sum_of_binary1 +y
END DO

sum_of_binary2 =0
DO k=1,16
y=(2**(k-1))*array3(k-1)
sum_of_binary2 =sum_of_binary2 +y
END DO

Inter_output_array(4)=(sum_of_binary1*32767)+sum_of_binary2

END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==5) THEN
nu1=IN1(5)
nu=(nu1/32767)
nu2=MOD(nu1,32767)
CALL DEC_to_bin_1(nu, array1)
CALL DEC_to_bin_2(nu2, array3)

DO i=1,16
IF (Fliping_Bits(i+80)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
```

```
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

DO i=1,16
IF (Fliping_Bits(i+96)==0) THEN
IF (array3(i-1)==0) THEN
array3(i-1)=1
ELSE
array3(i-1)=0
END IF
ELSE
array3(i-1)=array3(i-1)
END IF
END DO

sum_of_binary1 =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary1 =sum_of_binary1 +y
END DO

sum_of_binary2 =0
DO k=1,16
y=(2**(k-1))*array3(k-1)
sum_of_binary2 =sum_of_binary2 +y
END DO
```

```
Inter_output_array(5)=(sum_of_binary1*32767)+sum_of_binary2

END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==6) THEN
nu=IN1(6)
CALL DEC_to_bin_1(nu, array1)

DO i=1,16
IF (Fliping_Bits(i+112)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

sum_of_binary =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary =sum_of_binary +y
END DO
```

```
Inter_output_array(6)=sum_of_binary

END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==7) THEN
nu=IN1(7)
CALL DEC_to_bin_1(nu,array1)

DO i=1,16
IF (Fliping_Bits(i+128)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

sum_of_binary1 =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary1 =sum_of_binary1 +y
END DO

Inter_output_array(7)=sum_of_binary1
```

```
END IF


!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==8) THEN
nu=IN1(8)
CALL DEC_to_bin_1(nu, array1)

DO i=1,16
IF (Fliping_Bits(i+144)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

sum_of_binary =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary =sum_of_binary +y
END DO

Inter_output_array(8)=sum_of_binary
```

```
END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==9) THEN
nu=IN1(9)
CALL DEC_to_bin_1(nu,array1)

DO i=1,16
IF (Fliping_Bits(i+160)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
END IF
END DO

sum_of_binary1 =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary1 =sum_of_binary1 +y
END DO

Inter_output_array(9)=sum_of_binary1

END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
IF (W==10) THEN
nu=IN1(10)
CALL DEC_to_bin_1(nu, array1)

DO i=1,16
IF (Fliping_Bits(i+176)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

sum_of_binary =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary =sum_of_binary +y
END DO

Inter_output_array(10)=sum_of_binary

END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==11) THEN
```

```
nu=IN1(11)
CALL DEC_to_bin_1(nu, array1)

DO i=1,16
IF (Fliping_Bits(i+192)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

sum_of_binary =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary =sum_of_binary +y
END DO

Inter_output_array(11)=sum_of_binary

END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IF (W==12) THEN
nu=IN1(12)
```

```
CALL DEC_to_bin_1(nu, array1)

DO i=1,16
IF (Fliping_Bits(i+208)==0) THEN
IF (array1(i-1)==0) THEN
array1(i-1)=1
ELSE
array1(i-1)=0
END IF
ELSE
array1(i-1)=array1(i-1)
END IF
END DO

sum_of_binary =0
DO k=1,16
y=(2**(k-1))*array1(k-1)
sum_of_binary =sum_of_binary +y
END DO

Inter_output_array(12)=sum_of_binary

END IF

END DO

RETURN
END
```

A.1.2 Fortran Code for the Decimal-To-Binary Conversion

```
SUBROUTINE DEC_to_bin_1(nu,array1)

INCLUDE ’nd.h’
INCLUDE ’s1.h’ ! TIME variables
INCLUDE ’s2.h’
INCLUDE ’emtconst.h’
INCLUDE ’emtstor.h’

INTEGER nu
INTEGER, DIMENSION(0:15) :: array1

array1(0) = MOD(nu, 2) ! This is actually MOD(number / 1, 2),
array1(1) = MOD(nu / 2, 2) ! but that’s silly and wasteful.
array1(2) = MOD(nu / 4, 2)
array1(3) = MOD(nu / 8, 2)
array1(4) = MOD(nu / 16, 2)
array1(5) = MOD(nu / 32, 2)
array1(6) = MOD(nu / 64, 2)
array1(7) = MOD(nu / 128, 2)
array1(8) = MOD(nu / 256, 2)
array1(9) = MOD(nu / 512, 2)
array1(10) = MOD(nu / 1024, 2)
array1(11) = MOD(nu / 2048, 2)
array1(12) = MOD(nu / 4096, 2)
array1(13) = MOD(nu / 8192, 2)
```

```
array1(14) = MOD(nu / 16384, 2)
array1(15) = MOD(nu / 32768, 2)

RETURN
END

SUBROUTINE DEC_to_bin_2(nu2,array3)

INCLUDE 'nd.h'
INCLUDE 's1.h' ! TIME variables
INCLUDE 's2.h'
INCLUDE 'emtconst.h'
INCLUDE 'emtstor.h'

INTEGER nu2
INTEGER, DIMENSION(0:15) :: array3

array3(0) = MOD(nu2, 2) ! This is actually MOD(number / 1, 2),
array3(1) = MOD(nu2 / 2, 2) ! but that's silly and wasteful.
array3(2) = MOD(nu2 / 4, 2)
array3(3) = MOD(nu2 / 8, 2)
array3(4) = MOD(nu2 / 16, 2)
array3(5) = MOD(nu2 / 32, 2)
array3(6) = MOD(nu2 / 64, 2)
array3(7) = MOD(nu2 / 128, 2)
array3(8) = MOD(nu2 / 256, 2)
array3(9) = MOD(nu2 / 512, 2)
array3(10) = MOD(nu2 / 1024, 2)
```

```
array3(11) = MOD(nu2 / 2048, 2)
array3(12) = MOD(nu2 / 4096, 2)
array3(13) = MOD(nu2 / 8192, 2)
array3(14) = MOD(nu2 / 16384, 2)
array3(15) = MOD(nu2 / 32768, 2)

RETURN
END
```

A.2 Fortran Code for the Switch

SUBROUTINE $switch_G(IN1, IN2, IN3, IN4, OUT1, OUT2, OUT3, OUT4, P_L, B)$

```
INCLUDE ’nd.h’
INCLUDE ’s1.h’ ! TIME variables
INCLUDE ’s2.h’
INCLUDE ’emtconst.h’
INCLUDE ’emtstor.h’

INTEGER IN1(12), IN2(12),IN3(12), IN4(12),B,k,i
REAL P_L, P1, no_of_packets_prev
INTEGER OUT1(12), OUT2(12), OUT3(12), OUT4(12)
INTEGER OUT1_prev(12), OUT2_prev(12), OUT3_prev(12), OUT4_prev(12)
INTEGER My_NSTORI
REAL My_NSTORF
REAL input_time1,input_time2,input_time3,input_time4
REAL output_time_prev1,output_time_prev2,output_time_prev3,output_time_prev4
```

```
REAL no_of_packets_lost1,no_of_packets_lost2,no_of_packets_lost3,no_of_packets_lost4

My_NSTORI=NSTORI
My_NSTORF=NSTORF
NSTORF=NSTORF+100
NSTORI=NSTORI+(B*12)+200

output_time_prev1=STORF(My_NSTORF+1)
output_time_prev2=(STORF(My_NSTORF+2))
output_time_prev3=STORF(My_NSTORF+3)
output_time_prev4=STORF(My_NSTORF+4)
no_of_packets_prev=STORF(My_NSTORF)

DO k=1,12
OUT1_prev(k)=STORI(My_NSTORI+k)
OUT2_prev(k)=STORI(My_NSTORI+k+15)
OUT3_prev(k)=STORI(My_NSTORI+k+30)
OUT4_prev(k)=STORI(My_NSTORI+k+45)
END DO

P1=0.00001

!!!!!!!!!!!!!!!!!!!FOR IN1!!!!!!!!!
IF (IN1(6)==15) THEN

IF ((STORF(My_NSTORF)) .LT. B) THEN
STORF(My_NSTORF)=(STORF(My_NSTORF))+1
input_time1=TIME
```

```
STORF(My_NSTORF+1)=input_time1+(P1*(STORF(My_NSTORF)))
STORF(My_NSTORF+6)=STORF(My_NSTORF+6)+1

DO k=1,12
STORI(My_NSTORI+k)=IN1(k)
END DO

ELSE
no_of_packets_lost1=(STORF(My_NSTORF))-B+1

END IF

END IF

IF (STORF(My_NSTORF+6) .GE. 1) THEN
IF ((TIME .GE. (output_time_prev1-0.0000001)) .AND.
(TIME .LE.(output_time_prev1+0.0000001))) THEN

DO i=1,12
OUT1(i)=OUT1_prev(i)
END DO
STORF(My_NSTORF)=STORF(My_NSTORF)-1

ELSE
DO i=1,12
OUT1(i)=0
END DO
```

```
END IF
END IF

!!!!!!!!!!!!!!!!!!!FOR IN2!!!!!!!!!
IF (IN2(6)==15) THEN

IF ((STORF(My_NSTORF)) .LT. B) THEN
STORF(My_NSTORF)=STORF(My_NSTORF)+1
input_time2=TIME
STORF(My_NSTORF+2)=input_time2+(P1*(STORF(My_NSTORF)))
STORF(My_NSTORF+7)=STORF(My_NSTORF+7)+1

DO k=1,12
STORI(My_NSTORI+k+15)=IN2(k)
END DO

ELSE
no_of_packets_lost2=(STORF(My_NSTORF))-B+1

END IF
END IF

IF (STORF(My_NSTORF+7) .GE. 1) THEN
IF ((TIME .GE. (output_time_prev2-0.0000001)) .AND.
(TIME .LE.(output_time_prev2+0.0000001))) THEN

DO i=1,12
OUT2(i)=OUT2_prev(i)
```

```
END DO

STORF(My_NSTORF)=STORF(My_NSTORF)-1

ELSE

DO i=1,12

OUT2(i)=0

END DO


END IF

END IF


!!!!!!!!!!!!!!!!!!!!FOR IN3!!!!!!!!!

IF (IN3(6)==15) THEN


IF ((STORF(My_NSTORF)) .LT. B) THEN

STORF(My_NSTORF)=STORF(My_NSTORF)+1

input_time3=TIME

STORF(My_NSTORF+3)=input_time3+(P1*(STORF(My_NSTORF)))

STORF(My_NSTORF+8)=STORF(My_NSTORF+8)+1


DO k=1,12

STORI(My_NSTORI+30+k)=IN3(k)

END DO

ELSE

no_of_packets_lost3=(STORF(My_NSTORF))-B+1


END IF

END IF
```

```
IF (STORF(My_NSTORF+8) .GE. 1) THEN
IF ((TIME .GE. (output_time_prev3-0.0000001)) .AND.
(TIME .LE.(output_time_prev3+0.0000001))) THEN

DO i=1,12
OUT3(i)=OUT3_prev(i)
END DO
STORF(My_NSTORF)=STORF(My_NSTORF)-1
ELSE
DO i=1,12
OUT3(i)=0
END DO

END IF
END IF

!!!!!!!!!!!!!!!!!!!!!!!!FOR IN4!!!!!!!!!
IF (IN4(6)==15) THEN

IF ((STORF(My_NSTORF)) .LT. B) THEN
STORF(My_NSTORF)=STORF(My_NSTORF)+1 input_time4=TIME
STORF(My_NSTORF+4)=input_time4+(P1*(STORF(My_NSTORF)))
STORF(My_NSTORF+9)=STORF(My_NSTORF+9)+1

DO k=1,12
STORI(My_NSTORI+45+k)=IN4(k)
END DO
ELSE
```

```
no_of_packets_lost4=(STORF(My_NSTORF))-B+1

END IF
END IF

IF (STORF(My_NSTORF+9) .GE. 1) THEN
IF ((TIME .GE. (output_time_prev4-0.0000001)) .AND.
(TIME .LE.(output_time_prev4+0.0000001))) THEN

DO i=1,12
OUT4(i)=OUT4_prev(i)
END DO
STORF(My_NSTORF)=STORF(My_NSTORF)-1
ELSE
DO i=1,12
OUT4(i)=0
END DO

END IF
END IF

STORF(My_NSTORF+12)=(STORF(My_NSTORF+12))+no_of_packets_lost1+
no_of_packets_lost2+no_of_packets_lost3+no_of_packets_lost4
P_L=STORF(My_NSTORF+12)

RETURN
END
```

A.3 Fortran Code for the Data Source Component

```
SUBROUTINE DataSource(msg,Data_valid,R,L,T)
INCLUDE ’nd.h’
INCLUDE ’s1.h’ ! TIME variables
INCLUDE ’s2.h’
INCLUDE ’emtconst.h’
INCLUDE ’emtstor.h’

INTEGER msg(12)
INTEGER Data_valid,T,i,count,count_old
REAL R,L,x,u,u2,new_time,My_NSTORF
INTEGER My_NSTORI,w
INTEGER u3
INTEGER*4 timeArray(3) ! Holds the hour, minute, and second
INTEGER a,c,b

My_NSTORI=NSTORI
My_NSTORF=NSTORF
NSTORF=NSTORF+20
NSTORI=NSTORI+20

call itime(timeArray) ! Get the current time

a=9
c=17
b=187
```

```
    IF (TIME==0) THEN
count=0
count_old=0
STORI(My_NSTORI+2)=timeArray(2)
STORI(My_NSTORI+3)=timeArray(2)
STORI(My_NSTORI+3)=MOD(((a*STORI(My_NSTORI+3))+c),b)
msg(2)=NINT(L)
END IF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
w=NINT((1/R)*10000)
x=(1/R)
IF (T==0) THEN !uniform
IF ((TIME .GE.((x*STORI(My_NSTORI))-0.0000001)) .AND.
(TIME .LE. ((x*STORI(My_NSTORI))+0.0000001))) THEN
DO i=1,12
msg(i)=1000
END DO
msg(6)=15
msg(2)=NINT(L)
STORI(My_NSTORI)=STORI(My_NSTORI)+1
Data_valid=1
msg(3)=1000+STORI(My_NSTORI+3)

ELSE
DO i=1,12
msg(i)=0
END DO
```

```
msg(6)=14
msg(2)=NINT(L)
Data_valid=0
END IF
END IF

IF (T==1) THEN !random
new_time=((x*STORI(My_NSTORI))+STORF(My_NSTORF+3))
IF ((TIME .GE.(new_time-0.000001)) .AND. (TIME .LE. (new_time+0.000001))) THEN
DO i=1,12
msg(i)=1000
END DO
STORI(My_NSTORI+2)=MOD(((a*STORI(My_NSTORI+2))+c),b)
STORF(My_NSTORF+3)=(STORI(My_NSTORI+2)*100/b)*(1/(100*R))
msg(6)=15
msg(2)=NINT(L)
STORI(My_NSTORI)=STORI(My_NSTORI)+1
Data_valid=1
msg(3)=1000+STORI(My_NSTORI+3)
ELSE
DO i=1,12
msg(i)=0
END DO
msg(6)=14
msg(2)=NINT(L)
Data_valid=0
END IF
END IF
```

```
RETURN
END
```

Appendix B

B.1 PSCAD Model Of The Two Area Four Generator System With Communication Network

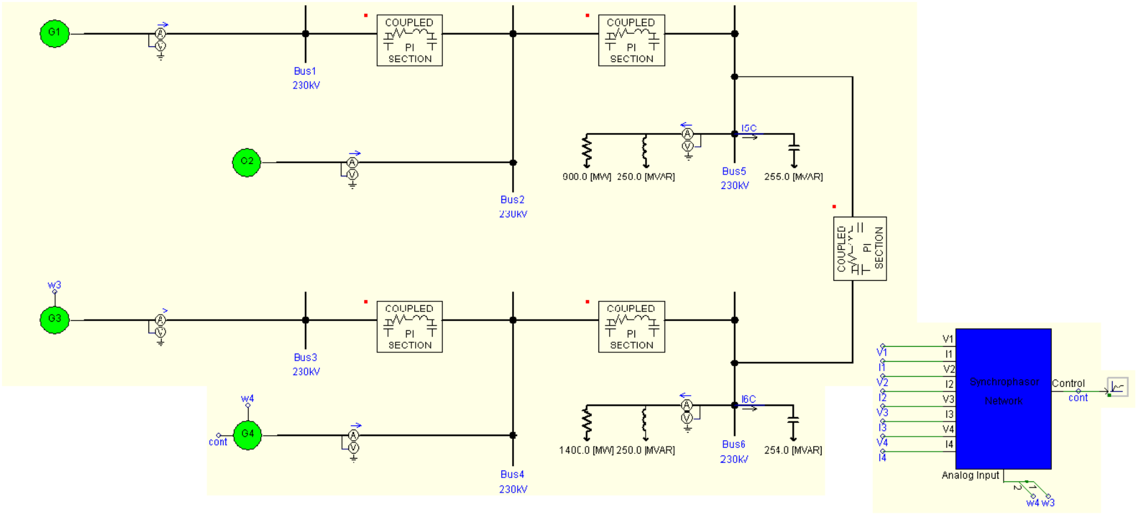


Figure B.1.1: Two area four generator system in PSCAD.

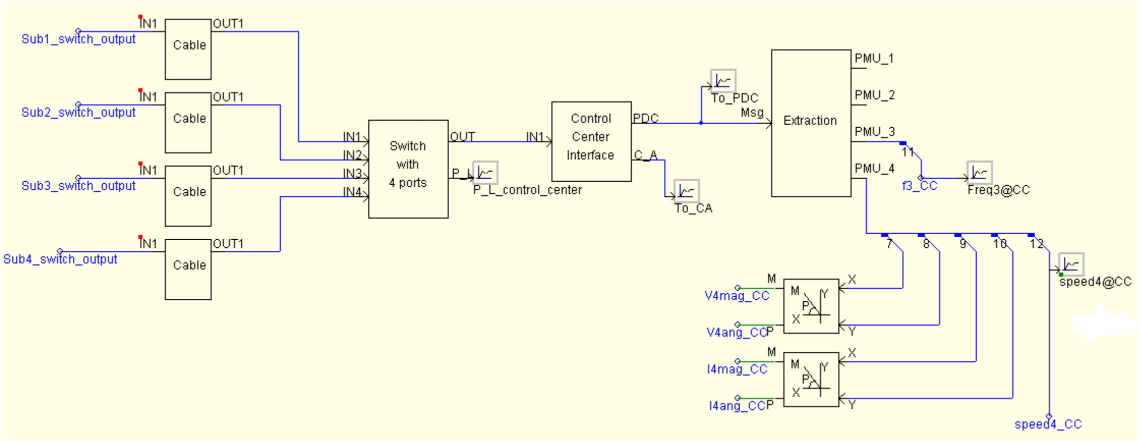


Figure B.1.2: Communication network for two area four generator system in PSCAD.

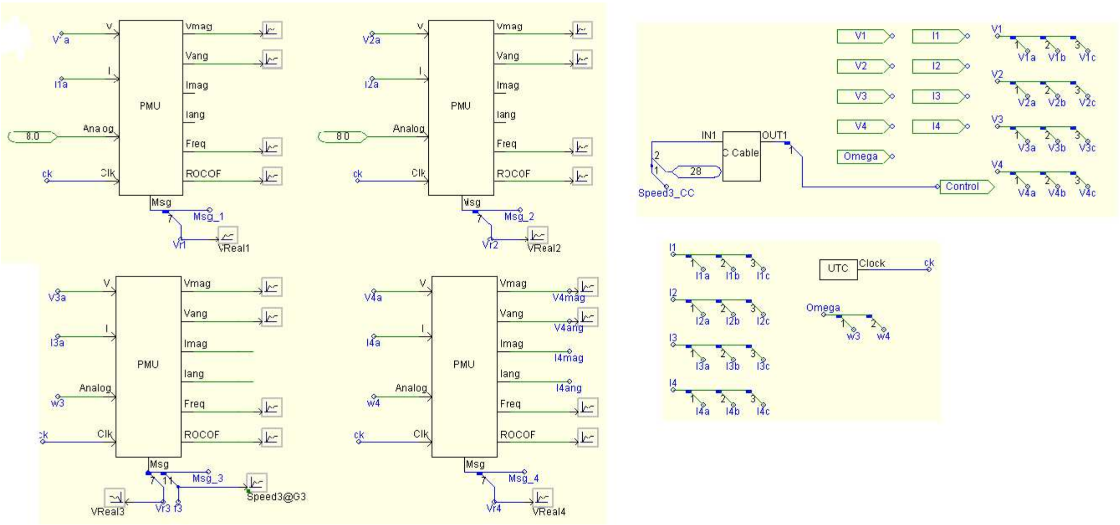
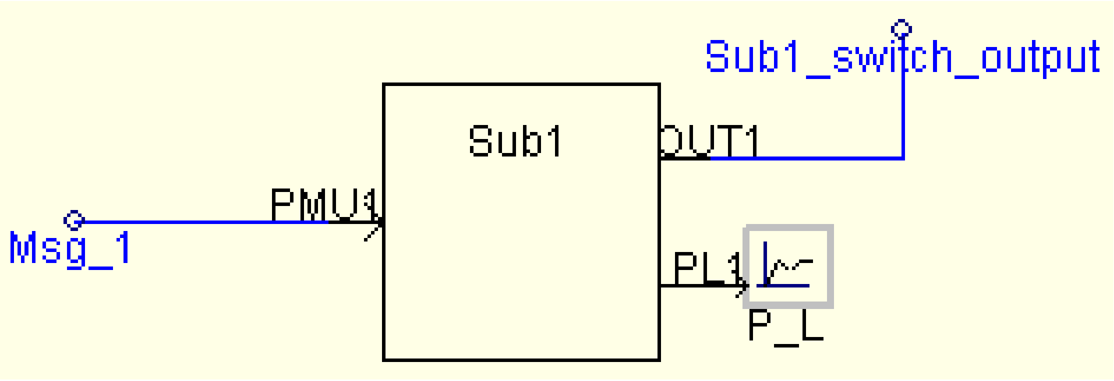
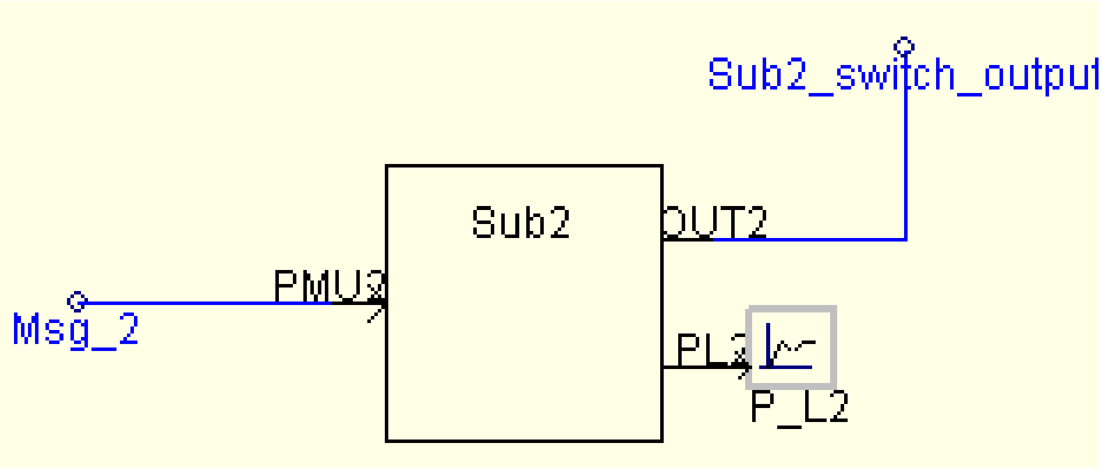
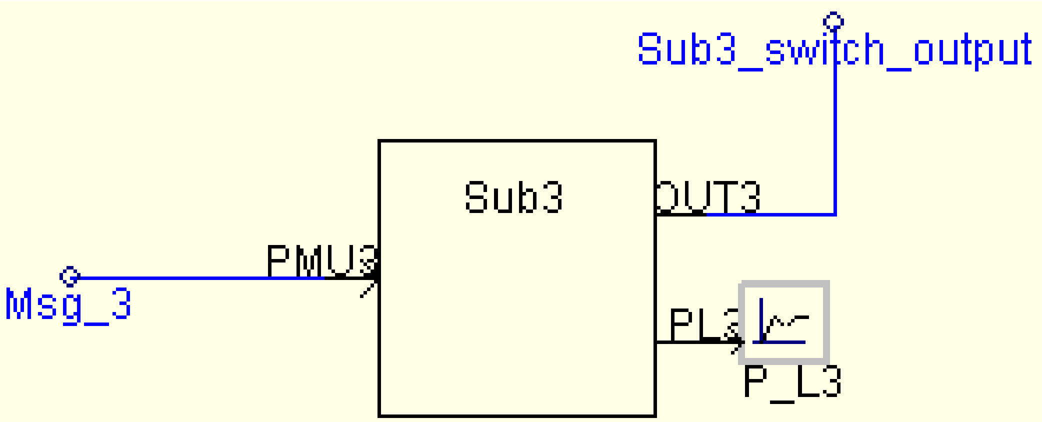
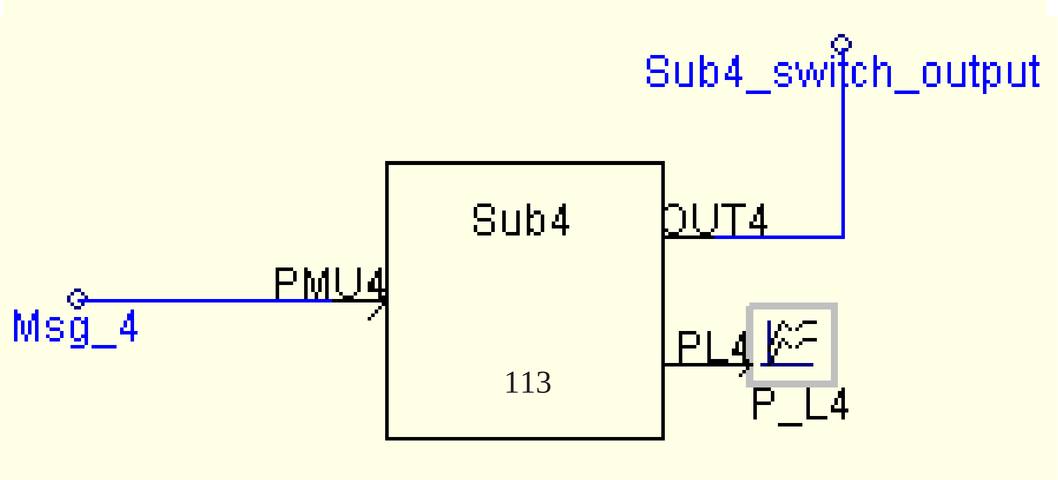


Figure B.1.3: PMU network in PSCAD.


Sub1_switch_output
Sub1
OUT1
PMU1
Msg_1
PL1
P_L


Sub2_switch_output
Sub2
OUT2
PMU2
Msg_2
PL2
P_L2


Sub3_switch_output
Sub3
OUT3
PMU3
Msg_3
PL3
P_L3


Sub4_switch_output
Sub4
OUT4
PMU4
Msg_4
PL4
P_L4

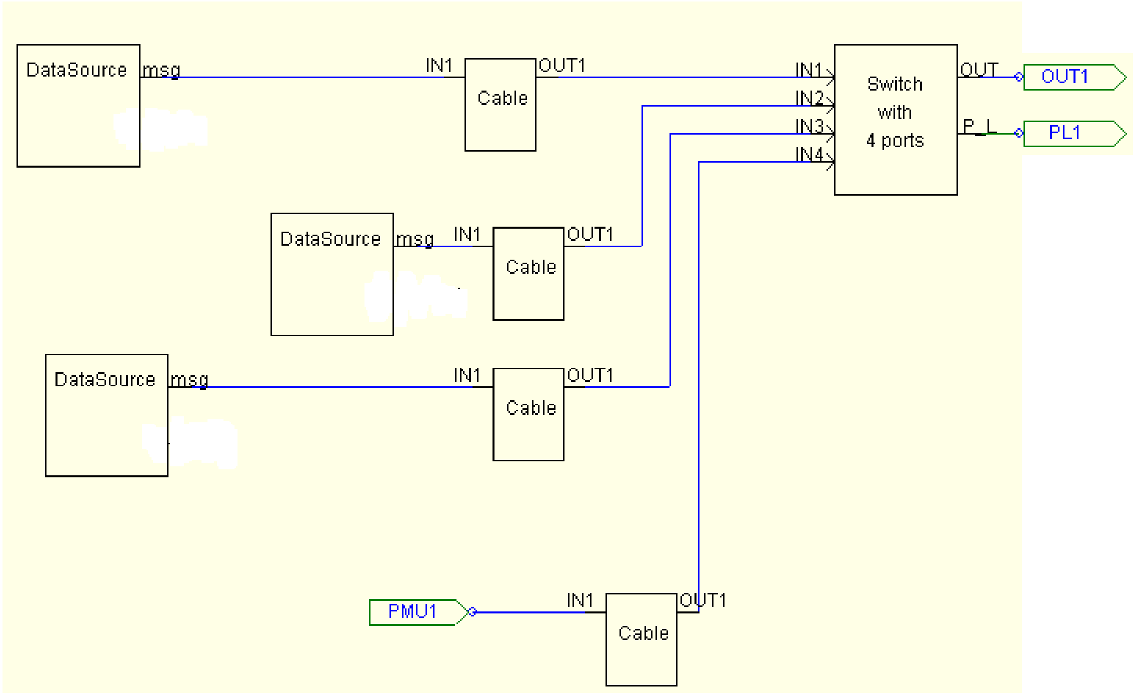


Figure B.1.5: Model of Substation 1 in PSCAD.

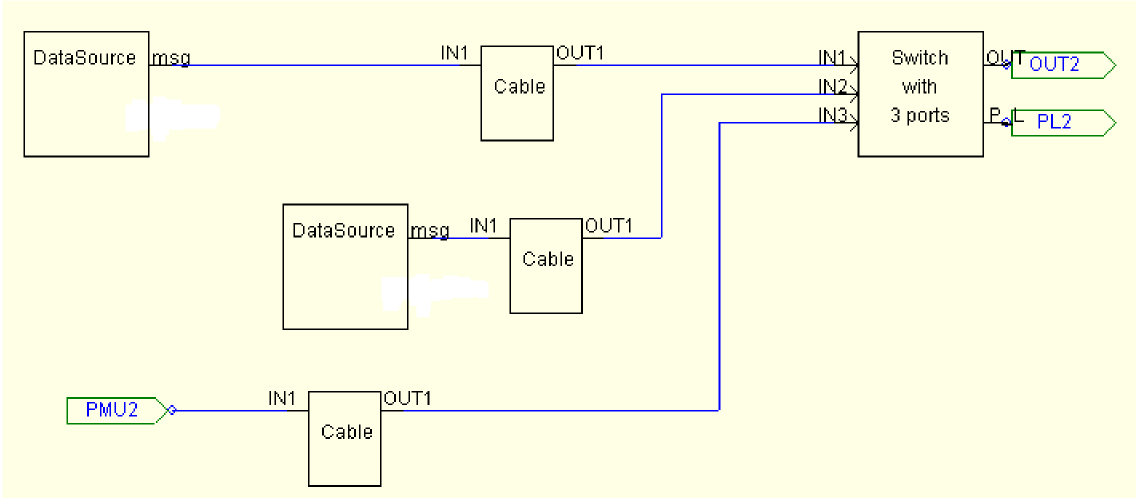


Figure B.1.6: Model of Substation 2 in PSCAD.

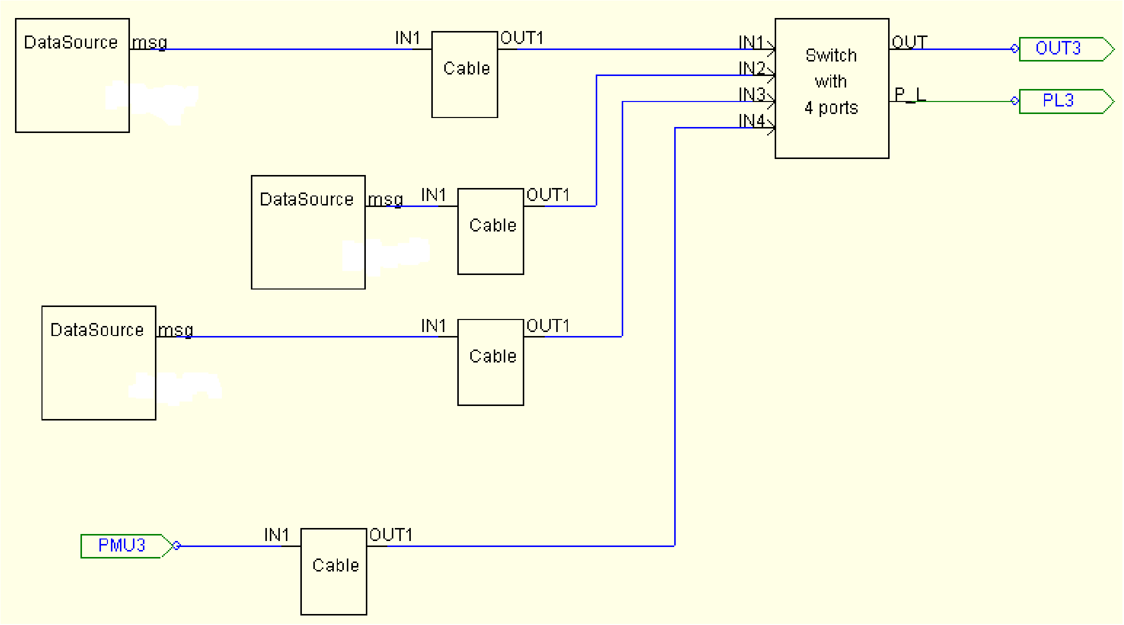


Figure B.1.7: Model of Substation 3 in PSCAD.

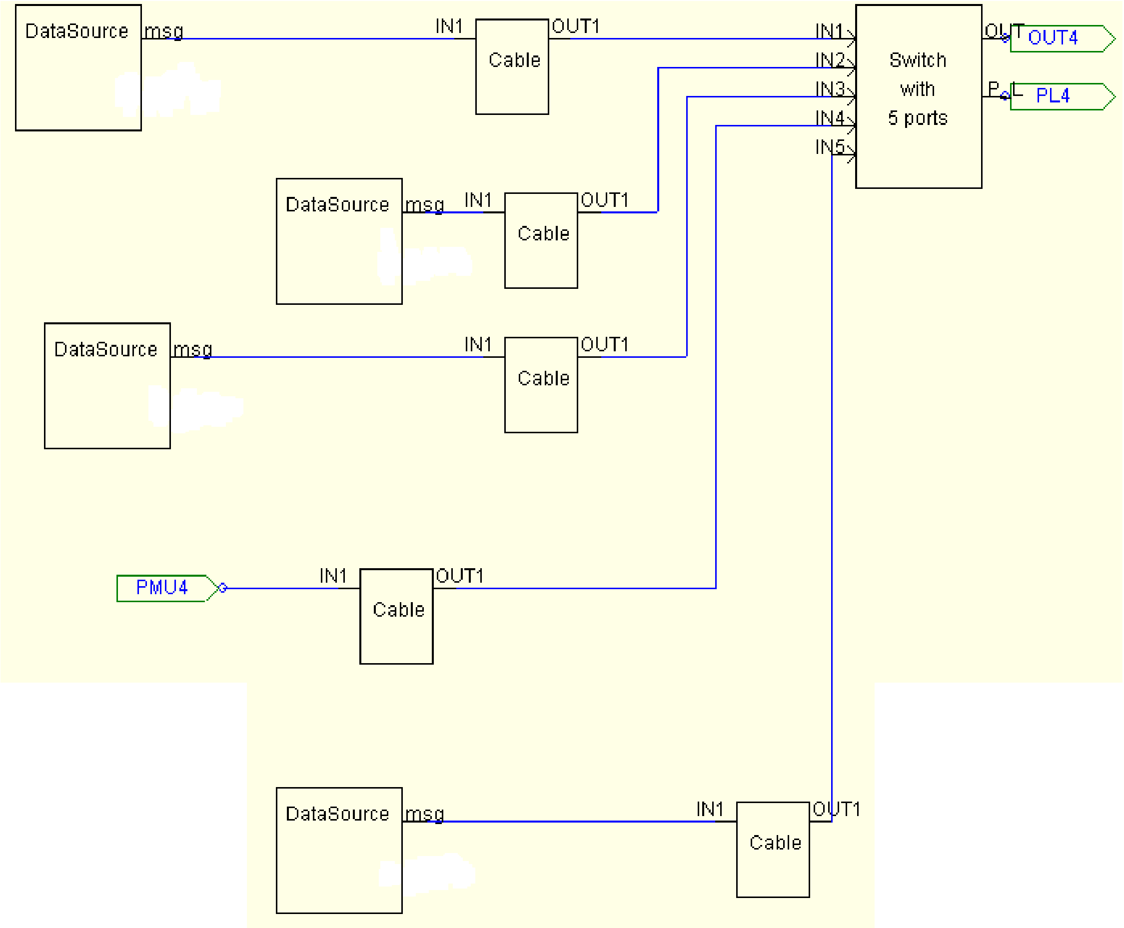


Figure B.1.8: Model of Substation 4 in PSCAD.