Seamless Bulk Electric Grid Management

Final Project Report

Power Systems Engineering Research Center

Empowering Minds to Engineer the Future Electric Energy System

Seamless Bulk Electric Grid Management

Final Project Report

Project Team

Anjan Bose, Washington State University

Thomas Overbye, University of Illinois at Urbana-Champaign

Santiago Grijalva, Georgia Institute of Technology

Graduate Research Students

Yannan Wang and Fan Ye
Washington State University

Komal Shetye, Richard Macwan and Frank Borth
University of Illinois at Urbana-Champaign

Leilei Xiong
Georgia Institute of Technology

PSERC Publication 15-08

November 2015

For information about this project, contact:

Anjan Bose
Regents Professor
School of Electrical Engineering and Computer Science
Washington State University
Pullman, WA 99164-2752
Phone: 509-335-1147
E-mail: bose@wsu.edu

Power Systems Engineering Research Center

The Power Systems Engineering Research Center (PSERC) is a multi-university Center conducting research on challenges facing the electric power industry and educating the next generation of power engineers. More information about PSERC can be found at the Center's website: http://www.pserc.org.

For additional information, contact:

Power Systems Engineering Research Center
Arizona State University
527 Engineering Research Center
Tempe, Arizona 85287-5706
Phone: 480-965-1643
Fax: 480-965-0745

Acknowledgements

This is the final report for the Power Systems Engineering Research Center (PSERC) research project entitled "Seamless Energy Management Systems" (PSERC project S-62G). The project was sponsored by EPRI. We express our appreciation to EPRI and to Paul Myrda for his support and guidance for the project. We also express our appreciation for the support provided by PSERC's industry members, as well as EPRI members, and by the National Science Foundation's Industry / University Cooperative Research Center program.

Executive Summary

Seamless Bulk Electric Grid Management

The main impediment in designing the framework for the next generation Energy Management System (EMS) and analytics is not the lack of ideas but a way to compare these ideas and determine which ideas are better than others. Ultimately, a flexible platform is needed that can simulate the layers of high voltage hardware, the IT hardware, and the various software packages that forms the base on which the applications for operating the grid can run. Such a simulation platform is needed to test myriads of ideas from individual components (e.g., a new electronic controller) to operational procedures (e.g., fast wide-area control schemes to avoid blackouts), but will require significant resources and time to put together. In this project we proposed the building of a simple platform that incorporates the different layers without the multiplicity of details to determine the feasibility and complexity of building the simulation platform that can mimic a real continent wide grid interconnection.

The base for such a platform has to be the simulation of the power grid itself that can produce Phasor Measurement Unit (PMU) measurements with fidelity (such as the time granularity of a transient stability program). On top of this layer has to be the communication layer needed to move the data from where they are produced to the applications. The management of the data, both static and real-time, is another layer on which the applications layer must reside. The combination of these layers makes up the total infrastructure.

The power grid simulation of the platform uses the PowerWorld Dynamics Studio (DS) program to simulate the transient stability behavior of the power grid and produces the streaming output of the PMU data in standard format that would normally be measured. The same program can also accept control signals that affect the transient behavior. This work was conducted at the University of Illinois at Urbana-Champaign (UIUC). A test power system of 42 buses was used. A particular fault on one transmission line was studied for transient behavior. If the fault is cleared in 50ms, the two generators at the nearby bus goes unstable, but if one of the generators is subsequently tripped, the other one remains stable. Such a generator tripping would require a special protection scheme.

The communication network is simulated by using NS3, a discrete-event network simulator for Internet systems, targeted primarily for research and educational use. Such a communication network overlays the power grid. Different architectures of the communication networks for the 42 bus power system are tested. These different architectures can also have different bandwidths and different phasor data concentrator (PDC) processing times for the PMU data passing through the PDCs. This work was conducted by Washington State University (WSU). It is shown that some of these networks with certain characteristics can guarantee the latency of the control signal that trips the generator to keep the system stable, while other designs of the communication network do not. The architecture of the network and the PDC processing times had the most influence on the communication delays.

New types of EMS applications are also tested on this platform. A decentralized control of frequency is tested by dynamically changing the power agreement between agents after a disturbance. This work was conducted by the Georgia Institute of Technology (GT). The same 42 bus test system was used to generate the PMU data. Then, the decentralized frequency controller was shown to successfully control the frequency with communications between seven agents that rapidly converges on a new power agreement that brings the frequency back to normal.

This project shows that platforms such as this one can be built to co-simulate the behavior of the power grid, the communication network, and the applications. Although more elaborate simulation platforms will be needed to test production grade designs of controller hardware and software, the models and algorithms are all shown to be feasible.

Table of Contents

List of Figures

1 Introduction to Seamless EMS Systems

Decision makers for the bulk electric grid have come to rely on various computational methods and software applications to help plan and operate the power system. Due to its large size, complex emerging behavior (such as increased integration of wind and demand response), and possibility of events, even experienced personnel can struggle with making decisions quickly while meeting the objectives of system security and reliability. Hence, the decision-making process must be equipped with enhanced applications that address emerging behavior and that exploit new data and computation capabilities.

Currently, at the ISO level, there are two primary groups of decision makers: one that deals with control and operations, and a separate group that deals with transmission and generation planning. Both perform similar functions albeit with different objectives and on different time scales. The primary objective of the control and operations group is to meet the current demand and ensure the minute-to-minute reliability of the overall system, including voltage management, constraint handling, etc. The primary objective of the planning group is to meet the future demand and ensure the future reliability of the system.

Although they share similar responsibilities and often run similar algorithms, planning and operations each utilize their own specific software applications, models, and data formats [1]. This practice of each group working in isolation has led to two seams or rifts in the ISO community. The first seam is the difficulty in comparing application results due to the utilization of different models in operations and planning. A unified network model for both planning and operations is necessary in order to achieve interoperability of the two groups at the ISO level [2]. The second seam that is a focus of this project is the repetitiveness of simulations. For instance, contingency analysis (CA) needs to be performed across multiple time points for operations and multiple scenarios for planning. For instance, in the case of operations, the same list of contingencies (which may include thousands of plausible events) is resolved every 2-5 minutes while the system may have experienced small changes in generation and demand. An integrated multi-temporal, multi-scenario algorithm that takes advantage of existing information from the system and that is less "brute force" is required. Furthermore, this algorithm must be applicable to both planning and operations to more efficiently handle studies of large number of scenarios or contingencies. Throughout this report we use contingency analysis as an example of an application that requires massive scenario analysis. Other applications, such as Voltage Security Assessment (VSA), Dynamic Security Assessment (DSA), and Available Transfer Capability (ATC) fall within the same category and are examples of security applications. Future power system operation may also require the calculation of economic optimization applications, such as Security-Constrained Economic Dispatch (SCED), Security Constrained Optimal Power Flow (SCOPF), and Security-Constrained Unit Commitment (SCUC) to be used under massive scenario analysis for look-ahead operation. While we do not include specifics of these application in this project, we use contingency analysis as an example to provide insight into software architecture issues of seamless EMS equipped with look-ahead capabilities.

The third seam is the continued utilization of legacy code in today's software applications, which rely primarily on sequential computing rather than taking advantage of modern high performance computing technology.

There exists a need for next generation power system management tools that can simulate the different layers such as high voltage hardware, the IT hardware, the various software packages that forms the base on which the applications for operating the grid can run. We propose to build a simple platform that incorporates the different layers without the multiplicity of details to determine the feasibility and complexity of building the simulation platform that can mimic a real continent wide grid interconnection. This report introduces three parts, including PMU data generation, network simulation and corresponding application:

- **PMU Level Grid Simulations and Cases (UIUC)**. The University of Illinois portion of this project consists of providing a simulation platform and dynamics cases that can generate simulated PMU data at sampling rates of 30 or 60 times per second. The data are sent from PowerWorld Dynamics Studio (DS), which runs an interactive transient stability level simulation and can export data in C37.118 format. The 42 bus system is shown in Figure 1.1.

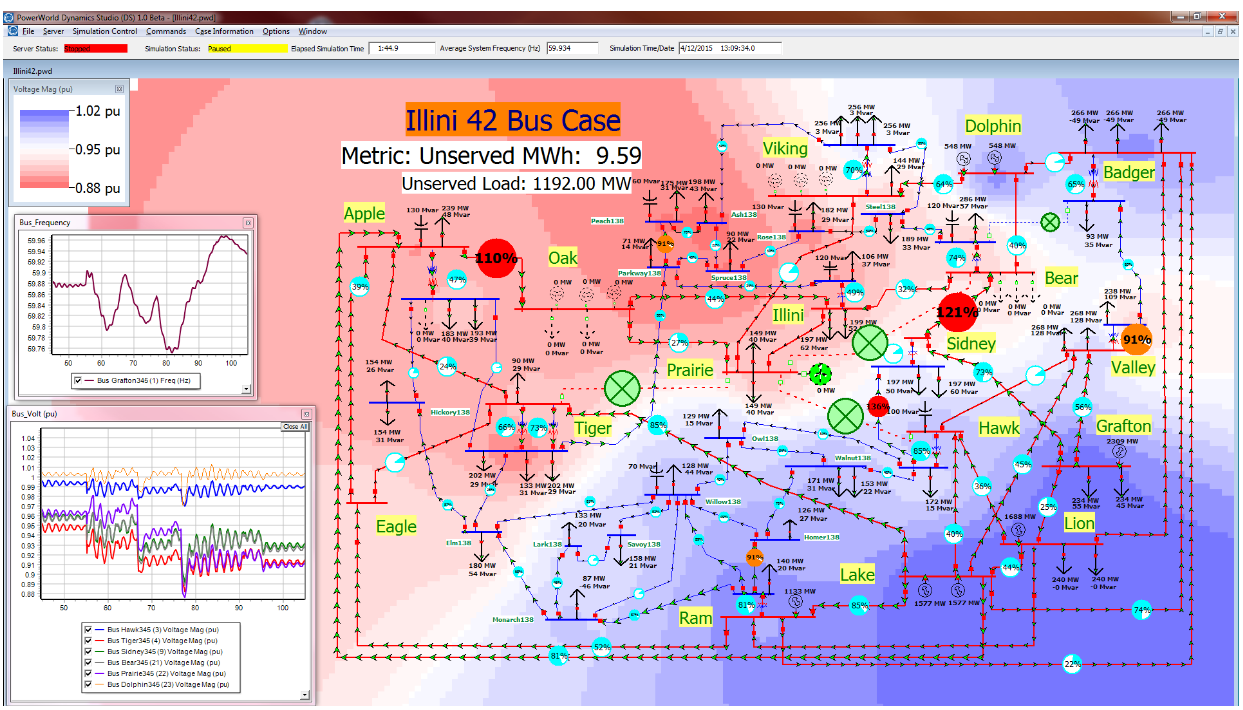

Figure 1.1 42 Bus Case for PMU Data Generation

- **Communication Network Simulation (WSU)**. The task of WSU part is to simulate the communication network between substations and the control center (CC). As far as we know, the data would have delay due to the imperfect network, which would have impact on the smart grid operation or control. The input of this part is PMU data from UIUC part, and the output is the same streaming data with delays.

- **Decentralized Application (GT).** The Georgia Tech team provides an overview of decentralized control applications for the future grid, the development of a power-communication co-simulator, and an example of decentralized power agreement protocol including the effect of communication delays,

Based on the mentioned above, the overview of PMU data path is shown as below:

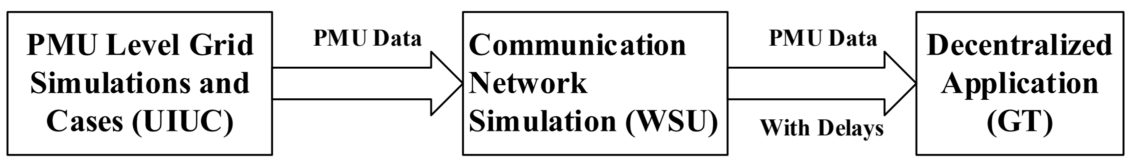

Figure 1.2 PMU Data Path from UIUC to WSU and to GT

2 PMU Level Grid Simulations and Cases (University of Illinois at Urbana-Champaign)

2.1 Introduction

As the smart grid moves forward, there is a need for flexible and interactive power system simulation environments in which new ideas for grid communication, control, analytics and visualization can be prototyped. A nice description of the role simulators can play in smart grid development is presented in [3]. Power systems has a long history of interactive simulation environments, with key distinctions often associated with the simulation time frame of the associated underlying dynamics. In this chapter an interactive environment for simulating power system dynamics on what we'll call the PMU time frame (power system cycles and slower) is presented. This simulation is then used in other chapters, such as to demonstrate the impact of communication delays.

In order to put this in context, Figure 2.1 (derived from Fig. 1.2 of [4]) shows the wide variety of time frames that might need to be considered in developing simulations for smart grid applications. However, in order to make the simulation computationally tractable and to simplify the modeling, the time frame of interest needs to be considered. Dynamics significantly faster than the time frame of interest can be represented by algebraic constraints and those significantly slower can be considered constant.

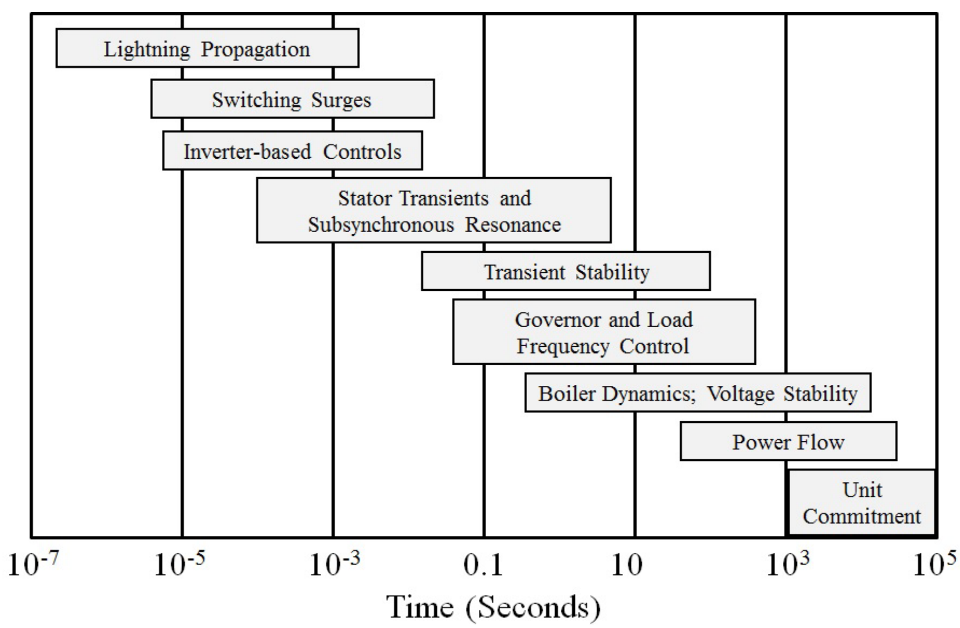

Figure 2.1 Power System Time Frames

The first interactive digital simulations were operator training simulators (OTSs) with [5] providing an early example. With this approach, the power system was assumed to have a uniform, but not constant, frequency. Dynamics with time frames longer than about one second were considered, such as generator boiler-turbine governors and automatic generation control, but the network equations were solved using a power flow. As the name implies, OTSs were often used to train operators. Slightly longer-term simulations, which used a constant frequency power flow assumption, were used to teach students and nontechnical professionals about the operation of the power grid with [6] providing an example. Such packages often ran substantially faster than real-time to teach concepts such as loop flow and interconnected operation. Because of the lack of dynamics, they could efficiently solve interconnect size systems with tens of thousands of buses. On an even longer time frame, [7] was used to teach market operations, working with a discrete, often one hour simulation step-size. In such market simulations the power flow was often not explicitly solved.

All of the preceding methods assume that even a large network can be modeled as algebraic constraints with speed of light considerations ignored. To represent very fast dynamics, such as for lightning propagation, switching surges and hardware-in-the-loop, simulations based on the electromagnetic transient approach of [8] have been developed. In this approach, the transmission lines are modeled with the differential equations associated with the voltage and current relationships in inductors and capacitors. By using Trapezoidal integration techniques, the models reduce to a network of coupled current sources and shunt resistances in which transmission line propagation delays can be considered explicitly. However, with simulation step sizes of microseconds they are often limited to smaller systems unless using large amounts of parallel computation.

The interactive simulation environment presented here sits between the extremely short time frame of [8] and the uniform frequency model of [5]. That is, simulating the system with a step size on the order of ¼ or ½ cycles (e.g., 0.004 seconds). In power systems this is known as transient stability time frame, but since it corresponds to the sampling frequency of PMUs, a complementary name is the PMU time frame. In this time frame, the dynamics of the generator machines, exciters, governors and stabilizers can be represented, along with dynamic models for the load (such as for induction motors). Hence during disturbances, each bus has a unique frequency, yet the transmission network equations are still represented as algebraic constraints. This time frame also allows for detailed modeling of the interaction of the power system with its underlying communication and control systems [40], [9], [10], and [11]. Cyber security issues in the communication system can also be considered [12].

The contribution of this section is a description of an interactive PMU time frame simulation environment used in this project, focusing on the visualization and test case development considerations. One use of such an environment is to provide a flexible platform that can be used to simulate the different layers such as the communication layer, and application testing layer, of smart grid hardware that might compose a next generation energy management system (EMS). The section is organized as follows. Section II provides a brief description of the power system simulation environment. Section III describes the development of example cases for control, communication and visualization prototyping. Section IV focus on the visualization of time-varying information, while Section IV presents the integration of this platform with a data historian program, for additional visualizations and analytics, using the C37.118 protocol.

2.2 A PMU Timeframe Interactive Simulation

While changes to the grid are resulting in more concern about dynamic issues in power system operation, the widespread deployment of PMUs is greatly increasing knowledge about power system dynamics in this PMU time frame, and allowing for the possibility of more closed-loop control. Hence, there is a need for smart grid prototyping and teaching environments modeling these power system dynamics.

In order to avoid the complexity and cost of writing a transient stability simulation from scratch, in an approach similar to what was presented in [11], the core smart grid dynamics simulation environment described here (abbreviated as DS) utilizes a commercial transient stability package as its simulation engine [13]. This provides the advantages of allowing it to represent hundreds of different power system models, import and export case models in industry standard formats, and efficiently solve large power system cases. The DS is able to communicate with other packages either using C37.118 or with command protocol allowing for interactive control. Hence, it provides an extensible environment that can be used to simulate the communication and control systems, such as from [9].

The DS is configured to run either in real-time, or either faster or slower than real-time up to computational limitations. A modest PC can solve systems with several thousand buses in real-time, and can simulate the small systems described here at more than 100 times real-time.

One aspect of the DS is its focus on interactive power system simulation and visualization. For instance Figure 2.2 shows the one-line diagram for a small, fictitious 345/138 kV system in which the per unit voltage magnitudes are represented using a color contour [14]. During an interactive simulation, the one-line contour can be updated at a user selected rate of up to about 10 Hz (depending on contour resolution and machine speed), allowing for good visualization of power system voltage effects. By varying this rate, it is possible to compare how a one-line might respond when driven by PMU data, versus one driven by SCADA data in which the refresh rate would be once every few seconds.

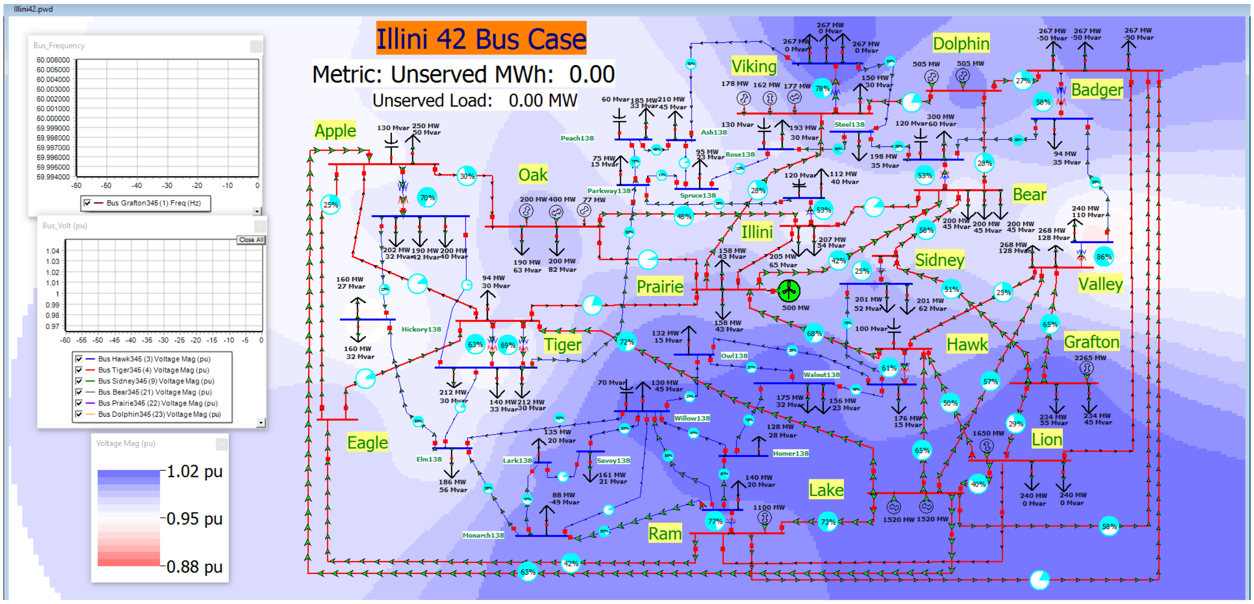

Figure 2.2 Small System with Voltage Contour

Another feature of the DS is the ability to display strip-charts of a wide variety of system quantities. Figure 2.3 demonstrates this functionality on a scenario that takes the Figure 2.2 case, and over the course of about one minute models the impact of a tornado moving through a substation, sequentially opening three 345 kV transmission lines, and taking a 500 MW wind farm off-line. In Figure 2.3, both strip-charts are displaying one minute of data, with the top chart showing the system frequency and the bottom one showing several of the bus voltage magnitudes. This scenario is setup as sort of a game in which the goal for the user is to interactively modify the system as the simulation progresses in real time, by control actions such as shedding load, to prevent a voltage collapse. Given the oscillations on the system, this scenario was designed to have important dynamics within the PMU time frame.

In order for such an environment to be useful, cases and scenarios are needed, as well as the ability to interact with other packages such as the WSU communication simulation package. Issues associated with the case and scenario design are covered next, while the use of C37.118 data to interface with other packages, is covered in the subsequent section.

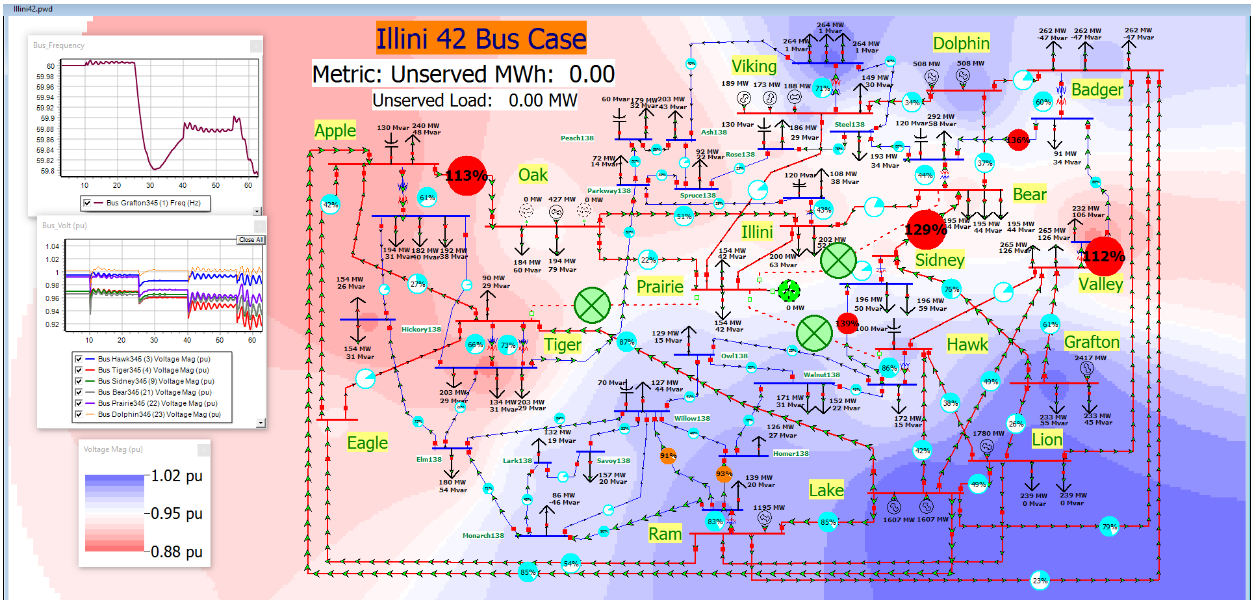

Figure 2.3 42 Bus Case with Several Lines and Generators Outaged

2.3 Development of Smart Grid Dynamics Case

In order to prototype PMU time frame control and communication in such an interactive DS environment, appropriate power system cases are needed, ideally ones that can be publicly shared. While some power flow cases are available, such as the ones at [15], few public cases with power system dynamics are widely available. The ones that are available often have quite simplistic dynamic models, such as representing the generators with classical models. Large-scale dynamics models can sometimes be obtained under non-disclosure agreements (NDAs); usually the NDAs restrict publication of detailed system results. Therefore there is a need for more realistic synthetic cases. This section discusses issues associated with the creation of these dynamic power system simulation cases, presenting case study results for the development of the cases used here, including the case from the previous section and a slightly modified version of the case used in the communication delay studies in the next chapter.

The 42 bus case shown in Figure 2.2 was developed to provide a generic case for prototyping smart grid dynamics concepts using the DS. In order to make the case widely usable, the bus count was selected to be no more than the 42 bus limit for the free power flow and transient stability software that is provided with [16]. Since it was meant to model PMU time frame system dynamics, the case needed to have sufficient load and generation to provide reasonable frequency response. While several different nominal voltages could have been chosen, 345/138 kV was selected since this could supply the desired load level and is a common voltage mix in the US Midwest. Overall the case load is about 10,000 MW. Transmission line and transformer impedances were set using typical per km values, derived using the techniques from [17], and then assuming an overall system footprint of about 250 km horizontally and 150 km vertically.

There was flexibility in how the load and generation was laid out. Our approach was to have a sufficient, yet not overwhelming, number of generators to allow a mix of dynamic models. The case has 14 individual generators at six substations, with a mix of fossil, hydro and wind generation. Generators are modeled with their dynamic components such as machines, governors, and exciters. Over excitation limiters, and over/under voltage/frequency relays are modeled, to automatically trip generators on excessive values. Load is represented using both static voltage dependent models, with 30% of the load being induction motors having the ability to restart. All transmission lines are modeled with inverse time overcurrent relays.

Although this case could be used in a variety of scenarios, parameter values of the different dynamic models were tuned to provide an interesting response for a scenario meant to mimic a tornado as mentioned in the previous section, moving by a large 345 kV substation (Prairie Substation, located towards the center of the system), and causing the loss of three 345 kV lines and a large generator over the course of 40 seconds. Figure 2.3 shows the system on the brink of a potential cascading blackout, with the first frequency drop caused by the loss of the 500 MW generator at the Prairie Substation, and the second caused by two of the generators at the Oak Substation tripping on over-excitation.

However, the case was designed such that with rapid control actions, using potential smart grid technologies such as fast visualization of PMU data, the blackout can be averted. An example of the results of such control actions is shown in Figure 2.4 which continues the Figure 2.3 simulation, showing the impact of rapidly shedding 690 MW of load. Readers interested in trying this case can download it from [17]. For this case such actions are required to prevent a voltage collapse.

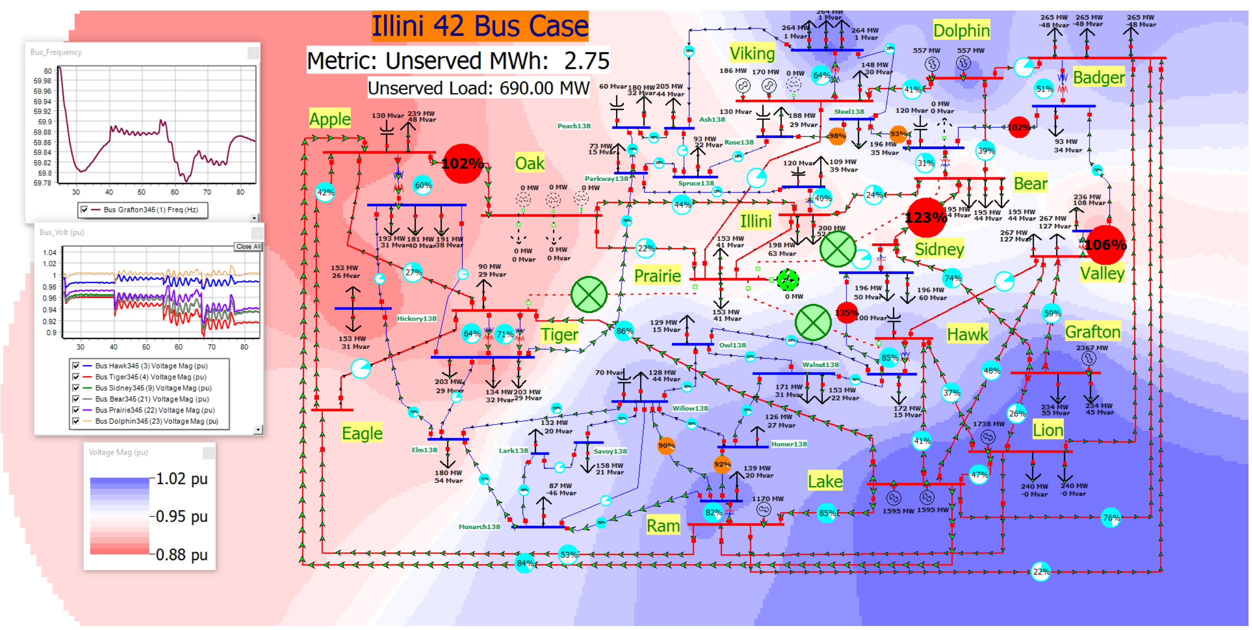

Figure 2.4 42 Bus Case Figure 2.3 Scenario with Load Shedding to Prevent Voltage Collapse

Next, to further facilitate the WSU communication network simulations the case was modified by adding a bus and substation on the left side of the network (bus 43, substation 43, "LoneGens") that contained two generators, shown in Figure 2.5. The system was set to model a common type of remedial action scheme (RAS) in which the loss of one of the two 345 kV transmission lines incident to bus 43 required that one of the generators be opened to avoid losing both generators due to over frequency. Both generators are modeled with over frequency relays that will do an instantaneously trip if their frequency is above 61.8 Hz (1.03 per unit).

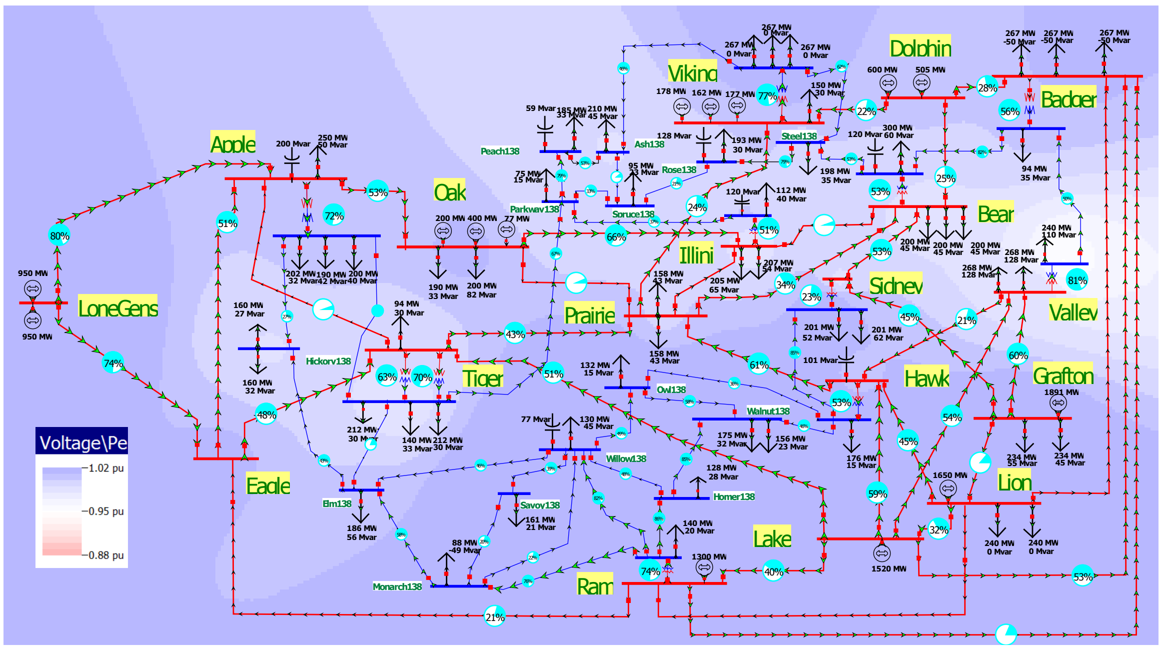

Figure 2.5 Modified 42 Bus Case with Remedial Action Scheme Generation

To illustrate the need for this RAS, Figure 2.6 shows the bus 43 frequency for a fault at time t=0 on the 345 kV line between LoneGens and Eagle, which is cleared by opening the line after three cycles (t=0.05 seconds). The loss of the line causes the generators to rapidly accelerate and their bus voltages to fall, with both eventually tripping at due to over frequency at about 1.3 seconds. Figure 2.7 shows the per unit bus 43 voltage magnitude. The loss of both units can be prevented by utilizing a RAS that rapidly trips one of the units, allowing the other unit to remain connected. This is illustrated in figure 2.8, which shows the bus 43 voltage magnitude if one of the generators is tripped at 0.5 seconds after the fault; delaying much longer would result in the loss of both units. Hence delays in communication can be quite important, an issue that is further explored in the next chapter.

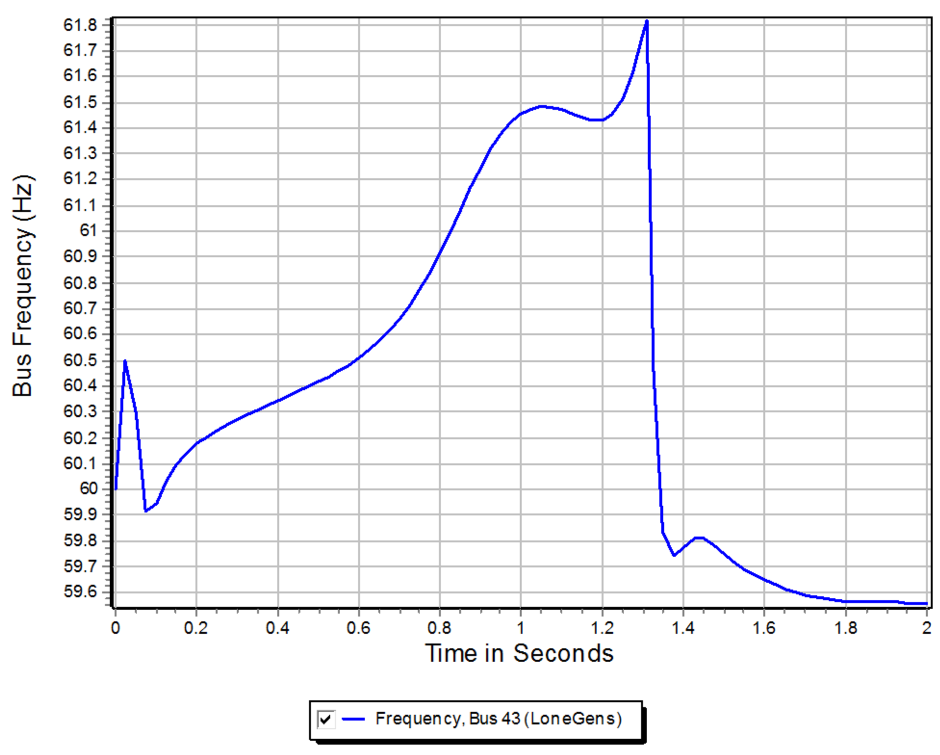

Figure 2.6 Bus 43 Frequency without RAS

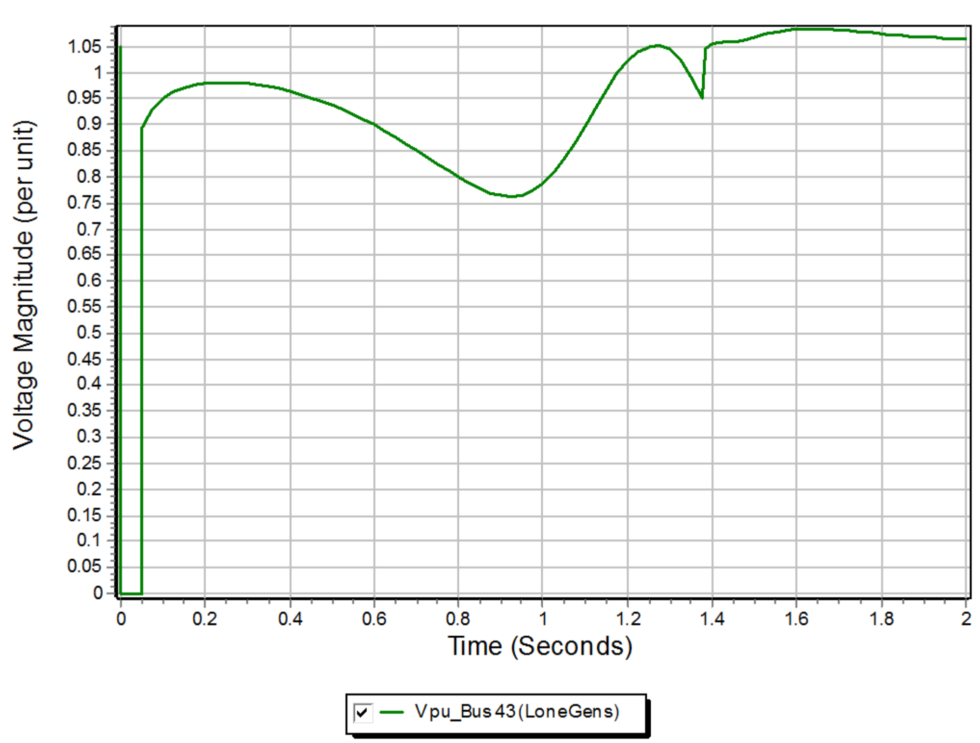

Figure 2.7 Bus 43 Voltage Magnitude without RAS

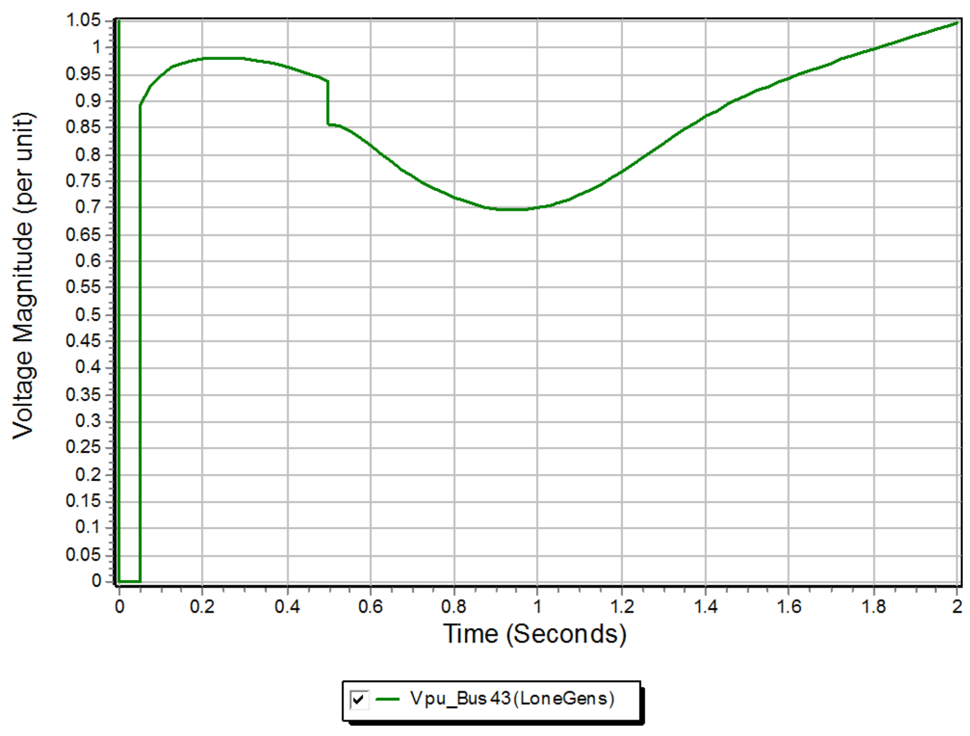

Figure 2.8 Bus 43 Voltage Magnitude With RAS at 0.5 Seconds

2.4 Sparkline Visualization of Time-Varying Information

One of the opportunities, and challenges in bringing PMU information into the EMS is how to effectively the associated time-varying information, which increases the dimensionality of the problem due to the addition of time. While power systems have many different time frames (as indicated in Figure 2.1) for the EMS type visualizations considered here the most important are power flow (many minutes) in which the data would be provided by SCADA, and the PMU time frame introduced previously (seconds to minutes).

Traditionally, time-varying information has been shown operationally using strip-chart recorders, in which a continuous recording is made in real-time of one or more data values; examples are shown in the previous figures. While certainly useful when just a small number of values need to be displayed, they are less useful for displaying the large number of values that are encountered in wide-area visualizations. In commenting on the number of colors do use in displays in which color is used for identification in [18] the author states, “Although color coding is an excelling way to display category information, only a small number of codes can be rapidly perceived; estimates vary between about five and ten codes.” Hence, individual strip-charts such as those shown previously can only show a small number of signals. Another disadvantage is it is difficult to show the geographic location of the associated data.

An alternative, familiar to anyone who has watched weather radar, is to use animation loops for visualizing data variation trends. The use such trend playback in an electrical control room is presented in [19] in which flows and voltage contours over time periods of up to one day can be rapidly visualized, and is what is used in the DS. An advantage of this approach is it can leverage all of the available static time visualizations, such as contours, and hence is familiar to users of these visualizations. Disadvantages include it takes time to run the animation loop so results are not available at a glance, and data trends may not be as easy to comprehend.

Another approach presented in [20] is to embed small strip charts onto existing onelines near the fields of interest. The advantage of this approach is the charts can be shown in their geographic context, but a disadvantage is because of space limitations it would be difficult to show a large number of charts. A solution to this issue is to use what are called "sparklines", defined in [21] as intense, simple, word-sized graphics. The idea of a sparkline is to show the time-variation in a signal using about the same amount of display space as the numeric string showing the value. Thus a sparkline is a graph without axis labels and numbers. Obviously there is a tradeoff between display space, and the amount of information shown. Sparklines show data with a few significant digits, but "the idea is to be approximately right rather than exactly wrong" ([21], pp. 50).

In a power system oneline context the x-axis time-scale could be common for all sparklines (e.g., one hour for SCADA data or perhaps 20 seconds for a transient stability study). The y-axis could also be implicit based on the type of value, for example between 75% and 150% for transmission line flows, or between 0.85 and 1.05 for voltage magnitudes, or 59.8 and 60.2 for frequency. When used online visualization sparklines could also only be shown for values that are trending towards limit violations or dynamically sized to highlight those at or near limits. Because of their small size sparklines could also be embedded in tabular displays, such as showing voltage variation in the column next to the field showing the present voltage value.

As an example, Figure 2.9 takes the Figure 2.4 tornado scenario and replaces the yellow and green 345 kV substation name labels with display objects that combine the substation name with a sparkline showing the variation in the substation 345 kV voltage magnitude over the previous 100 seconds using a y-axis scale of 0.85 to 1.05 pu. A color contour is still used to visualize the present voltage. Now at a glance the time variation in the seventeen 345 kV voltage magnitudes is apparent in a geographic context. Zooming and panning could be used to provide more detail, and additional data dialogs are available by right-clicking on the objects. Further applications of sparklines for power system visualization is an area of ongoing research.

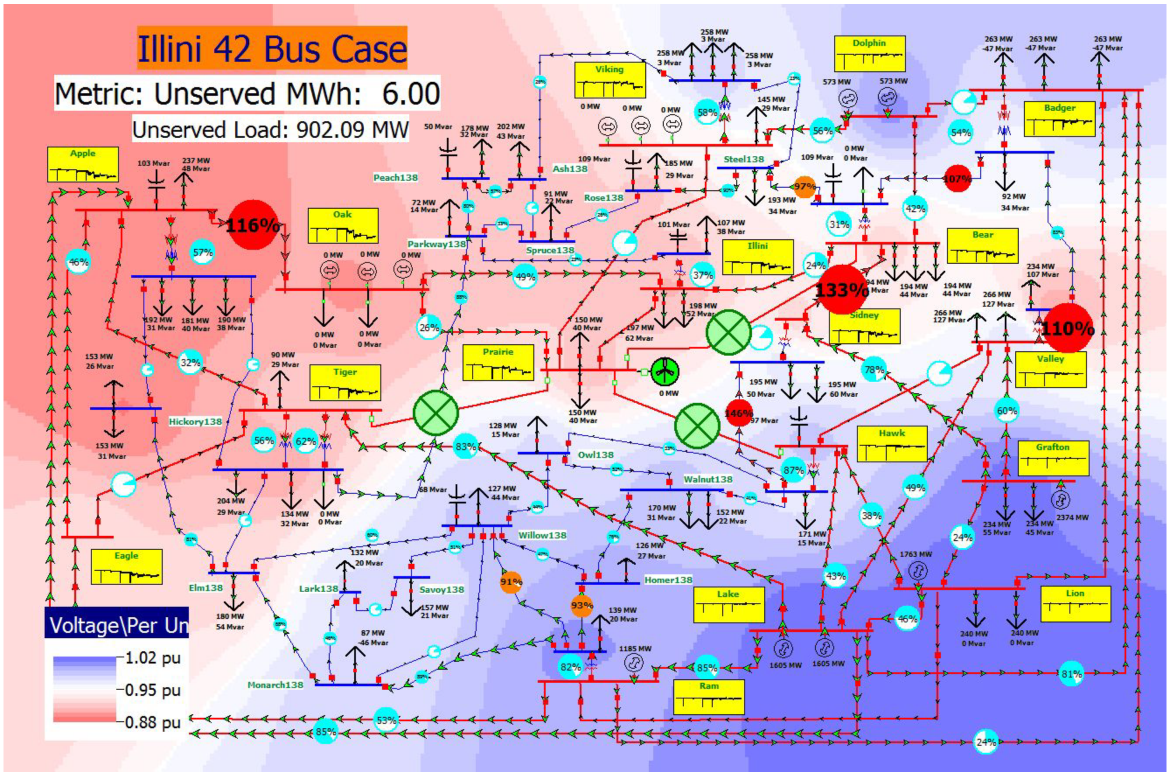

Figure 2.9 Visualization of Figure 2.4 System Using Sparklines

2.5 Data Historian Program Integration

In order to provide an interactive and extensible enviroment for prototyping smart grid communication and control, the DS can export data in C37.118 format, and can receive control commands. This section demonstrates the use of C37.118 format in which the DS is acting as a phasor data concentrator (PDC), exporting the PMU data from its substations to a data historian interface program (Historian) via the use of an application program interface node (API-Node) [22]. The API-Node also acts as the forwarding agent for the Historian. To better provide replication of an actual transmission grid system with attendant issues relating to data transport, the API-Node software is co-located with the DS on a single computer, and data is forwarded via a simulated Wide Area Network (WAN) structure to the machine operating the Historian functions.

Several functions are included within the API-Node structure; these functions improve overall data transport for the simulated WAN and provided supplementary functionality in the event of a network or historian outage. The first function is designed to reduce the overall amount of data necessary for transport over the WAN. This is performed via two different but complementary functions, 1) data stream compression, as seen in many forms of data transport over a constrained bandwidth network, and 2) "swinging-door compression" [22], which uses an exception deviation value for each data point to determine what data is required to recreate the actual value trend while removing redundant information and/or "noise".

The second function provided by the API-Node structure is that of local data storage in the event of loss of network connectivity or unavailability of the Historian. The overall amount of data and the span of time for which all of the data streams may be stored locally is ultimately dependent on two factors; first, the amount of spare disk space made available directly for storage on the API-Node, and second, the maximum rate at which locally stored (or buffered) data can be downloaded to the historian [22], [23].

With this data structure, all data to be downloaded to the Historian must be sent out before the allotted disk space is completely filled. The buffer file must be cleared before normal data transport can resume without loss of data. Otherwise if the file fills completely storing incoming data before all of said data is forwarded, data buffering effectively stops. Any incoming data, from the time the file is filled to the time when the data buffered in that file is forwarded, is effectively 'lost'. For this reason, the amount of space for buffering is often kept lower than could be stored overall; an undesired but necessary decision which must be made to balance recovery from an event with maximum network transport speed [22], [23].

Once the data leaves the DS and API-Node, it traverses the simulated WAN arriving at the Historian. Here the information streams, usually from multiple sources and API-Nodes, converge and are functionally made available to several upstream functions. The first and foremost is the data archiving system. The Historian also performs a second screening of the data (which can accomplish greater reduction of data, but never is less than the 'width' of the exception deviation in units or %-of data span) prior to that data being historized [22]. Ultimately, the data is stored within large data files called archives. The files may be delineated from each other initially by size or timespan, but ultimately it is their timespan which determines where certain data is contained, and which file is 'current', that is actively receiving data from sources designated for storage. The definitive overall span of time which a Historian may contain is only limited by disk storage space [22].

Concurrent to the archival function, the Historian is the process of serving data up to client software requesting information. These clients may be Graphical User Interfaces (GUIs), on-demand reports, as well as other programs such as EMS, Performance Monitors, Maintenance Management Systems, System Predictive Modelers, etc. Indeed most of the 'downstream' systems that perform analysis on the data subsequently return their result information to the Historian for storage along with the 'original' data. With these results also stored in the archives, both initial data and the calculations performed on that data may be presented in reports and on GUIs to improve user understanding of system events and quality of function. Either via the resident Historian GUI or through applications programming events may be played and re-played at varying rates to assist in troubleshooting and system analysis. This capability assists functions such as Root-Cause-Analysis (RCA), event discovery, data correlation, and even training of operations personnel [22].

To enable data transport from DS using C37.118 data interchange to the Historian, an administrator must 'construct' the receiving records, or data points, on the Historian. Many data point types are available to receive data; those most often used include floating-point and integer numerical classes, digital data (where the 0-to-x integers are correlated to a table of alpha-numeric values), and alpha-numeric string values [22]. Actual configuration involves naming the point, defining its data type, range of the data, deviation values for the compression routines, and often assigning the data point to the C37.118 interface. Once the data point has been configured on the Historian it is made available to the various interfaces. The interface software periodically 'queries' the Historian to find if data points have been added to, deleted from, or modified in its set of assigned points. Whether upon initial start-up of the interface, or during normal operation, data points assigned to the interface are loaded and data exchange initiated between the source system and the Historian.

With data exchange in operation, Historian users may access the information by one of the forms mentioned previously. For this discussion, topics are limited to the Historian's resident GUI, with minor mention of the reporting function in a spreadsheet program. In the GUI a visual representation of some portion of the system monitored is constructed, with the capability of showing numerical and digital values, bar-graphs which change in unison with the values, data trend graphs showing values over a period of time, and various active elements which can change a fill-in color based upon the point value of the monitored data [24].

2.6 Conclusions

A core smart grid dynamics simulation (DS) environment is developed utilizing a commercial transient stability package as its simulation engine. This simulation can calculate and output the PMU data that would be produced by the system during a dynamic simulation. An example case is developed using a 42 bus power grid. Some tools to visualize the data and store the data are also developed. A faulted case is simulated that is normally unstable but can be stabilized using a specifically designed remedial action scheme (RAS). In the next section, how the communication delays affect the RAS scheme and the stability of the power grid is discussed.

3 Communication Network Simulations (Washington State University)

3.1 Introduction

Due to the rapid development of Phasor Unit Measurements (PMUs) installed in large power systems, the communication between PMUs and the control center (CC) is becoming more and more important. Imperfect communication links will introduce unexpected delays, which would have impact on power systems operation, stability and control. To meet the strict latency requirements of power system controls, we need to consider the most efficient communication network which have the fastest data transfer. In this chapter, we postulate four different communication networks in the 42 bus system that UIUC generates in the previous chapter. In addition, we need to consider the packet size and sending rate for each substation. The commonly used software package NS3 is used for this network simulation.

NS3 is a discrete-event simulator for Internet systems, which is based on C++ programming language. Five different layers of protocols can be setup completely in the NS3 simulator. Otherwise, NS3 can simulate the queuing in the router, which is important and random in the Internet system. In this chapter, we focus on the constant rate of data sending by PMUs that each substation generates.

In today's PMU technology, phasor measurements can be generated at rates of 10 to 60 or more per second [25]. The smart grid of the future is expected to have PMU data available widely across the grid (network) [26]. It is required that a real time information infrastructure be proposed because of the strict latency requirement for fast controls. Smart grid applications are designed to exploit these high throughput real-time measurements. Most of these applications have a strict latency requirement in the range of 100 milliseconds to 5 seconds [27].

The communication delays actually have four parts: propagation delays, transmission delays, processing delays and queuing delays [28]. Propagation delays are related to the speed of the light and distance between two sides. Transmission delays introduce a new concept of transmission rate. The transmission rate is all about links technology: copper, optical fiber and so on. Each can carry at different transmission rates. Processing delays are easy to understand. When each packet goes through the router (or any terminal), it needs to be checked for the protocol header and the forwarding is decided through which link. This process obviously needs time. NS3, however, doesn't have the notion of processing delay [29]. Fortunately, the processing delay is always small compared to other delays so it can be neglected. These three delays are almost constant. This means if we know the packet size, transmission rate and distance, we can forecast these three delays, even though the transmission rate could have a little bit of variation. However, the last one, queuing delays, is always random depending on other traffic. We can't decide the queuing delays exactly in real life because we never know how much data is in the network at some specified time. Luckily, however, many computer science scholars developed models that can describe the queuing in the network; moreover, NS3 has these models so we don't need to consider how to build these queuing models.

In reference [30] a detailed survey of smart grid applications based on latency and bandwidth requirements has been presented. Latency is a measure of time delay experienced in a communication system. Whereas, bandwidth is the rate of data transfer in bits per second, that can be achieved by a communication resource. According to [30] applications pertaining to power system operation can be classified in the increasing order of their latency requirements as follows: transient stability (<100 milliseconds), small signal stability (<1 sec), state estimation (<1 sec), voltage stability (1-5 sec), post-postmortem analysis of grid disturbances (> few minutes).

Another thing we need to be careful about is the Phasor Data Concentrator (PDC). PDCs are designed to collect the data from several PMUs and are increasingly being installed, one for every few substation. When a certain packet goes through one PDC, the PDC needs to process this packet. This will add an extra delay in the data transfer. Thus this part of the delay should be considered in the smart grid communication research. However, previous research does not consider the PDC processing delay and its impact on the power system performance. That's why we examine the four levels of PDC delay in this report, including 10ms 50ms, 100ms and 500ms. Each length of PDC delay causes different performance of our example system.

Although many communication networks can connect the 42 bus system, it's not enough. It is useless unless we consider the impact of the data delays on the power system performance. The performance of the controls on the transient stability of the 42 bus system is considered towards the end of this chapter.

3.2 Preparation of Communication Network

3.2.1 Network Setup

A computer network actually has five stacks (OSI reference model [31] is not used in our case): application layer, transport layer, network layer, link layer and physical layer. We can setup the first four parts of them in NS3. And we use optical fiber as the physical layer. The following are the key points for each layer.

1) *Application Layer:* In this layer, PMU data need to be sent from each substation. Because of this, we need a standard protocol for PMU, namely C37.118 standard protocol [32]. The characteristics of PMU data is described in [33]. They are sent from Power World DS 30 times per second. We need to decide how much data to be sent and at what rate they are sending. Their final destination for all data is the Control Center (CC). In order to get the delay results, some assumptions are needed to be made: First, PMUs are installed in both end of a line and they measure 3-phase voltages and currents. So in each end, there has 6 phasors, three of them are voltage phasors and the others are current phasors; Second, the PMU sending rate is

constant, namely 30 packets per second. The detailed application data information of the 42 bus system is shown in TABLE 3.1.

TABLE 3.1 INFORMATION OF PACKET NEEDED TO BE SENT IN EACH SUBSTATION

Sub Name	Sub ID	Number of Phasors	Packet Size(bytes)	Sending Rate(kbps)
Badger	Sub1	36	444	106.56
Oak	Sub2	18	222	53.28
Apple	Sub3	30	370	88.8
Sidney	Sub4	18	222	53.28
Hickory	Sub5	12	148	35.52
Elm	Sub6	24	296	71.04
Viking	Sub7	24	296	71.04
Illini	Sub8	24	296	71.04
Hawk	Sub9	42	518	124.32
Bear	Sub10	36	444	106.56
Valley	Sub11	24	296	71.04
Prairie	Sub12	36	444	106.56
Dolphin	Sub14	18	222	53.28
Lake	Sub15	30	444	106.56
Ram	Sub16	42	518	124.32
Tiger	Sub17	36	444	106.56
Eagle	Sub18	24	296	71.04
Owl	Sub19	12	148	35.52
Walnut	Sub20	18	222	53.28
Grafton	Sub21	18	222	53.28
Ash	Sub27	18	222	53.28
Peach	Sub28	12	148	35.52
Spruce	Sub31	18	222	53.28
Parkway	Sub32	24	296	71.04
Steel	Sub33	12	148	35.52
Rose	Sub34	12	148	35.52
Savoy	Sub36	12	148	35.52
Monarch	Sub38	24	296	71.04
Willow	Sub39	36	444	106.56
Homer	Sub41	18	222	53.28
Lion	Sub42	30	370	88.8
LoneGens	Sub43	12	148	35.52

2) Transport Layer: There are two popular protocols in this layer: user datagram protocol (UDP) and transmission control protocol (TCP). UDP has smaller header, and it doesn't have any data protection mechanism [34]. Whereas, TCP is a reliable data transfer protocol. It has a bigger header, some control actions: flow control and congestion control, and some other reliable transfer mechanisms [35]. In this way TCP needs more time to setup the environment, to send data, to control sending behavior compared to UDP. For this reason, UDP is preferably used in NS3.

3) Network Layer: Internet protocol version 4 (IPv4) is used in this layer [36]. In order to improve the processing of this layer and for more safety, IPv6 is proposed recently [37]. However, Ipv6 is more complicated than IPv4 and it has a bigger header than IPv4. Thus the configuration in IPv6 is not an easy task. Moreover, our network for the 42 bus system is somehow different from the real life network. It doesn't have any big data, and it doesn't have any different types of information. It only has PMU data (or SCADA data). Besides we don't consider cyber security in our network model. We assume all network is secure enough to transfer PMU data. Hence IPv4 is enough for the 42 bus system network.

4) Link Layer: In today's network, most use the Ethernet protocol as the link layer. Ethernet protocol is a kind of wire protocol in the network [38]. In the alternative, we have WiFi protocol [39]. Ethernet protocol has MAC address in its header. Besides, it has shared medium transfer standard, which is commonly used in local area networks (LAN).

NS3 can set these up with some coding. The details of the protocols are not necessary to view. On the other hand, NS3 has routing protocols so it can set the packet path automatically. In this case, global routing protocol is used: packets always go the path which has least communication links (routers).

The overview of the process is shown in Figure 3.1.

3.3 Communication Networks for 42 Bus System

As mentioned before, 4 different network architectures are provided in this subchapter to be able to study the minimum delays. The first architecture (network type1) is made up of communication lines that are along the same right-of-way as the transmission lines; the second (network type 2) is divided up into three areas with three sub-control centers [40]; the third (network type 3) has a centralized structure where all substation directly communicate with the control center; and the fourth architecture (network type 4) has a decentralized structure where the substations in each of the three areas communicate directly to each sub-control center.

Before the network simulation, the event scenario for this system is described as follows: the system has a 3-phase ground fault at 2.0 seconds at bus 15. After 0.5 second, the line between bus 43 and 15 opens. The substations at these two buses send fault data to the CC, and when this data go through each substation or PDC, it will have PDC processing delays. The CC will send back to these two buses the trip control signal after receiving the faulted data. Finally, the generator at bus 43 will trip after receiving the control signal in order to keep the system stable. If the trip signal is delayed more than the critical clearing time, the generator will go unstable.

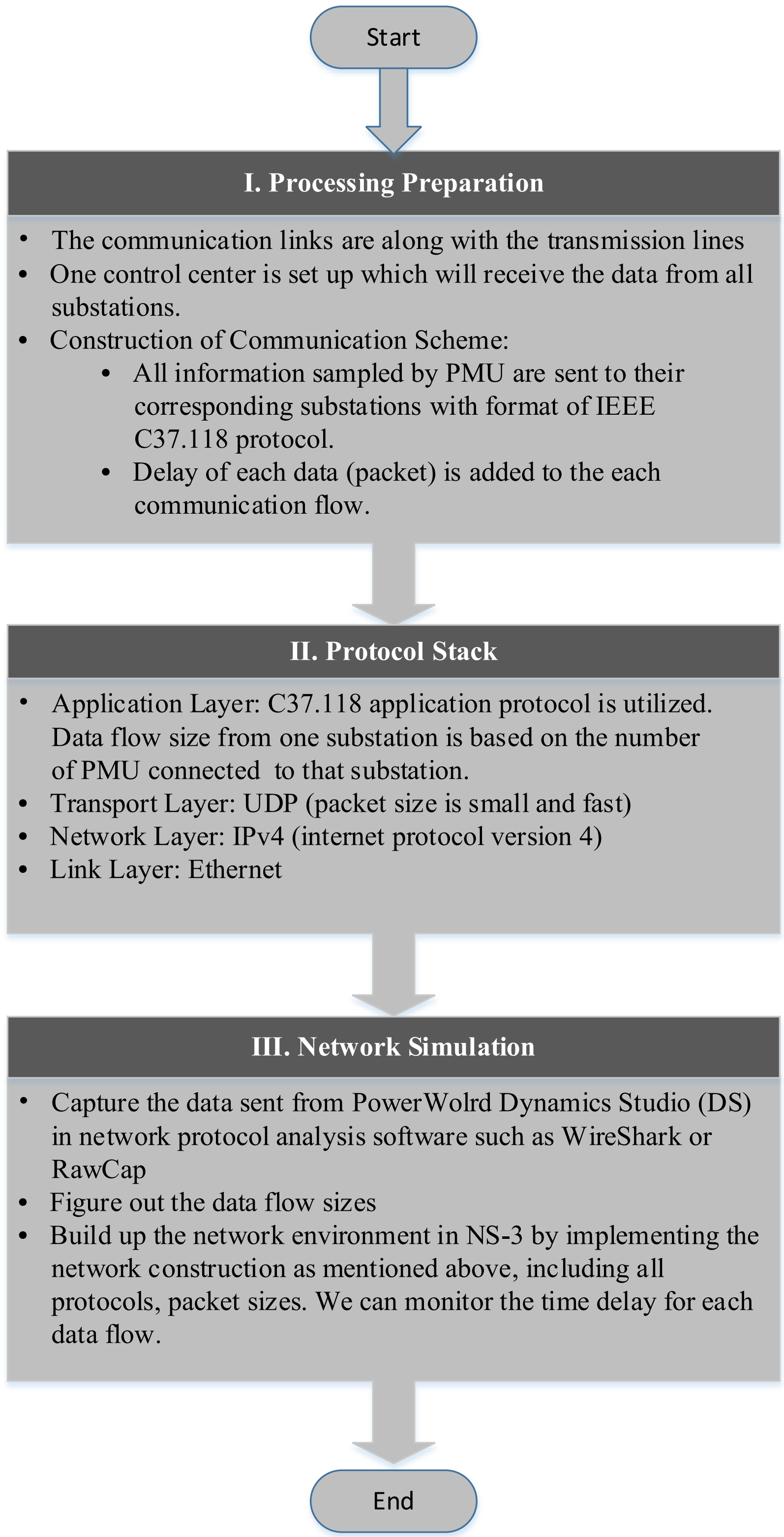

Figure 3.1 Flow Chart for Network Simulation

Because of increasing PMU installments in power systems, PDC delays are becoming bigger part in the communication systems. Thus in this report, we focus on different PDC processing delays including 10ms, 50ms, 100ms and 500ms. Different PDC delays result in different total delays which would have impact on power system transient stability performance. Some of them cause the system to be unstable while some keep system stable under different circumstances (such as bandwidth size and PDC delay). The following parts of the report describe these four network types mentioned above.

3.3.1 Type1: Network along with the Transmission Lines

For this type, one control center is set up. All communication links are along with the transmission lines. The control center is chosen in central location of the 42 bus system due to the more path available for data transfer. In this type, the control center is physically located in the Hawk substation (sub 9). The following figure shows what this type looks like. The control center is not shown in the figure.

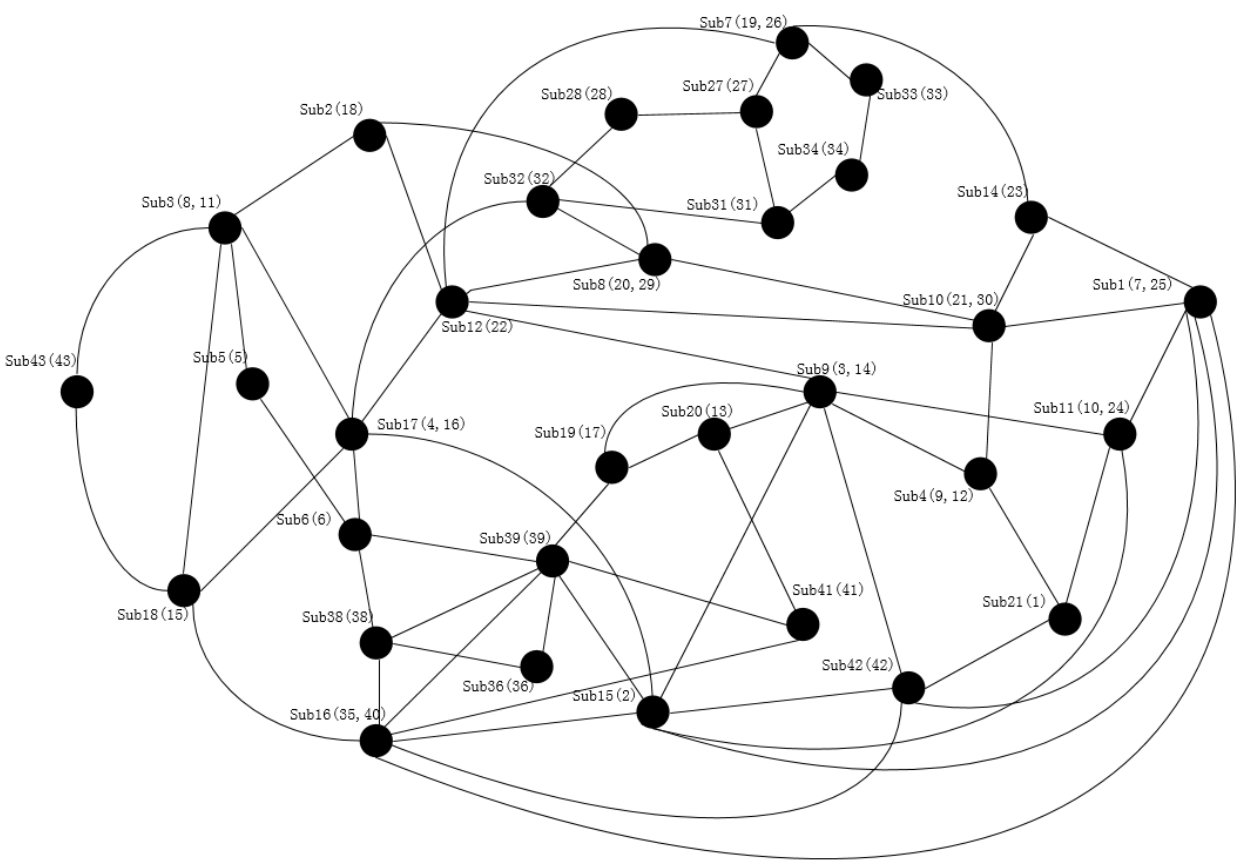

Figure 3.2 Network Along with the Transmission Lines

The total delays from each substation to CC and CC to bus 43 (in order to trip the generator) are shown as follows. It consists of PMU data delay plus control signal back delay. Four curves in this figure denote four PDC processing delays, and the bandwidth consists of 1Mbps, 5Mbps, 10Mbps and 50Mps.

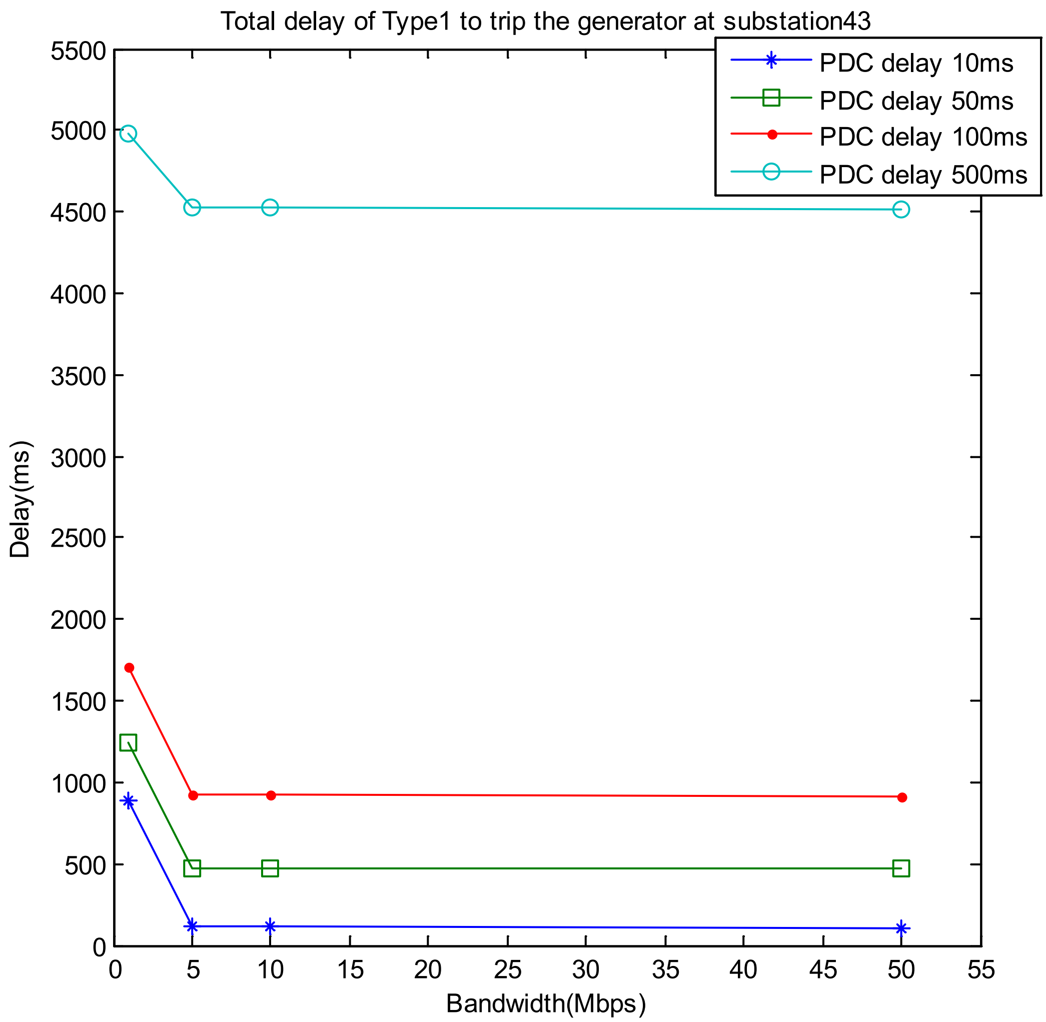

Figure 3.3 Delay to Trip the Generator in Sub43 (Type1)

As we can see with more PDC delay the packet delays are higher. Besides, the delays don't change significantly in bandwidth above 1Mbps. The reason is that between bandwidth of 1Mbps to 50Mbps, queuing delay doesn't play an important role, and that the data in the communication links are not large enough to generate big queuing delay in big bandwidth, like 10Mbps. However, when the bandwidth is below 1Mbps the queuing delay is becoming bigger. Especially at 1Mbps, the total delay are almost queuing delay plus other delays.

The actual impact of delays is described in Section 3.4. One thing we need to keep in mind: communication architectures have nothing to do with the system stability. It always has to do with when we trip the generator. This "when" is determined by the communication architectures, this is what we are trying to figure out. In this way Section 3.4 only needs some important delays that keep system stable, critically stable and unstable.

3.3.2 Type2: Network Divided by Three Areas

In this type of communication structure, we assume that 3 different power utilities control their own power area in the 42 bus system. Communication links of each area are along with their own transmission lines. There are no direct communication connections between substations in different areas. Each area has its own control room, which is called a sub control center. Sub control centers are connected to each other and they are all connected to the higher control center. Its structure is shown below.

In Figure 3.4, as you can see, Sub CC1 is located at Sub18, Sub CC2 is located at Sub9 and Sub CC3 is located at Sub1. They all connected to the control center, which is not shown in the figure, but is physically again in sub 9. The higher control center is responsible for the data coordination with each sub control center.

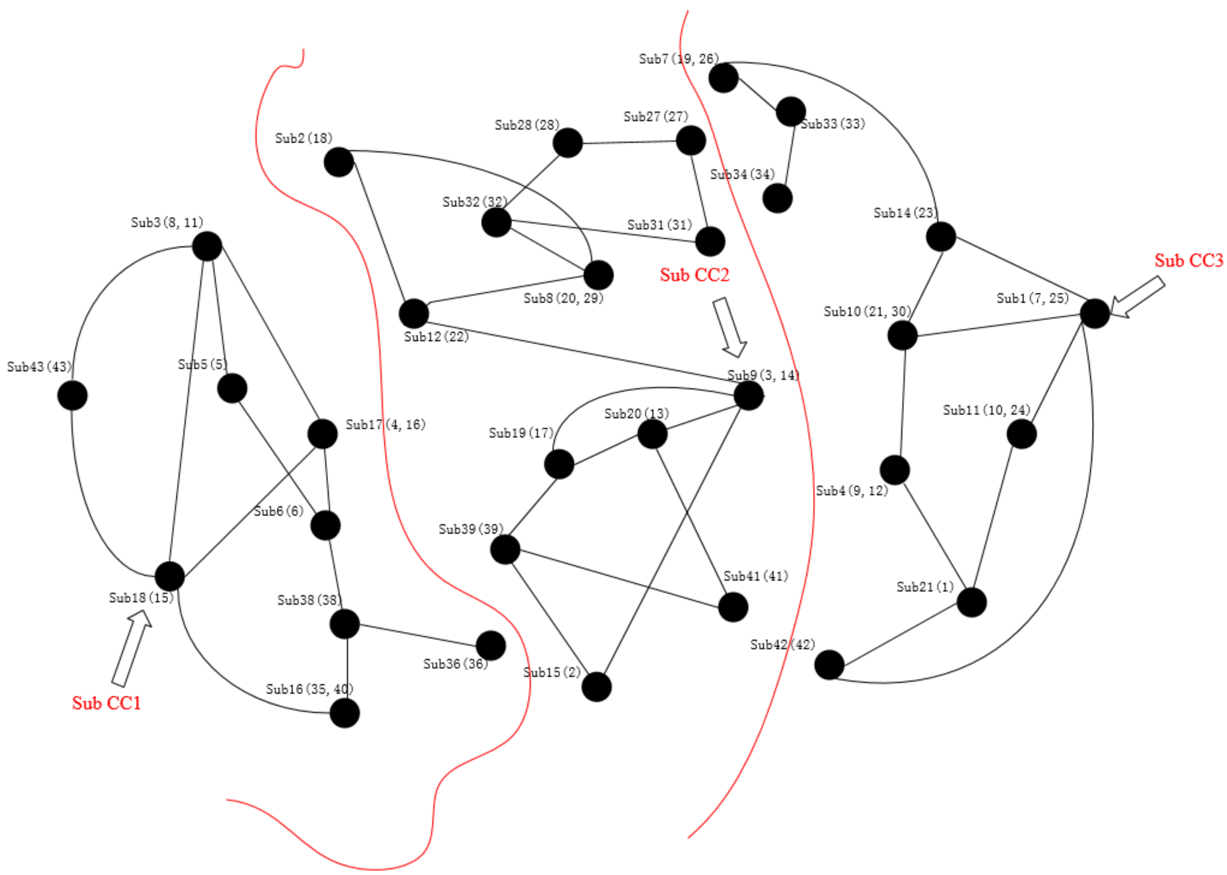

Figure 3.4 Network Divided by Three Areas

Similarly, delay results are shown below.

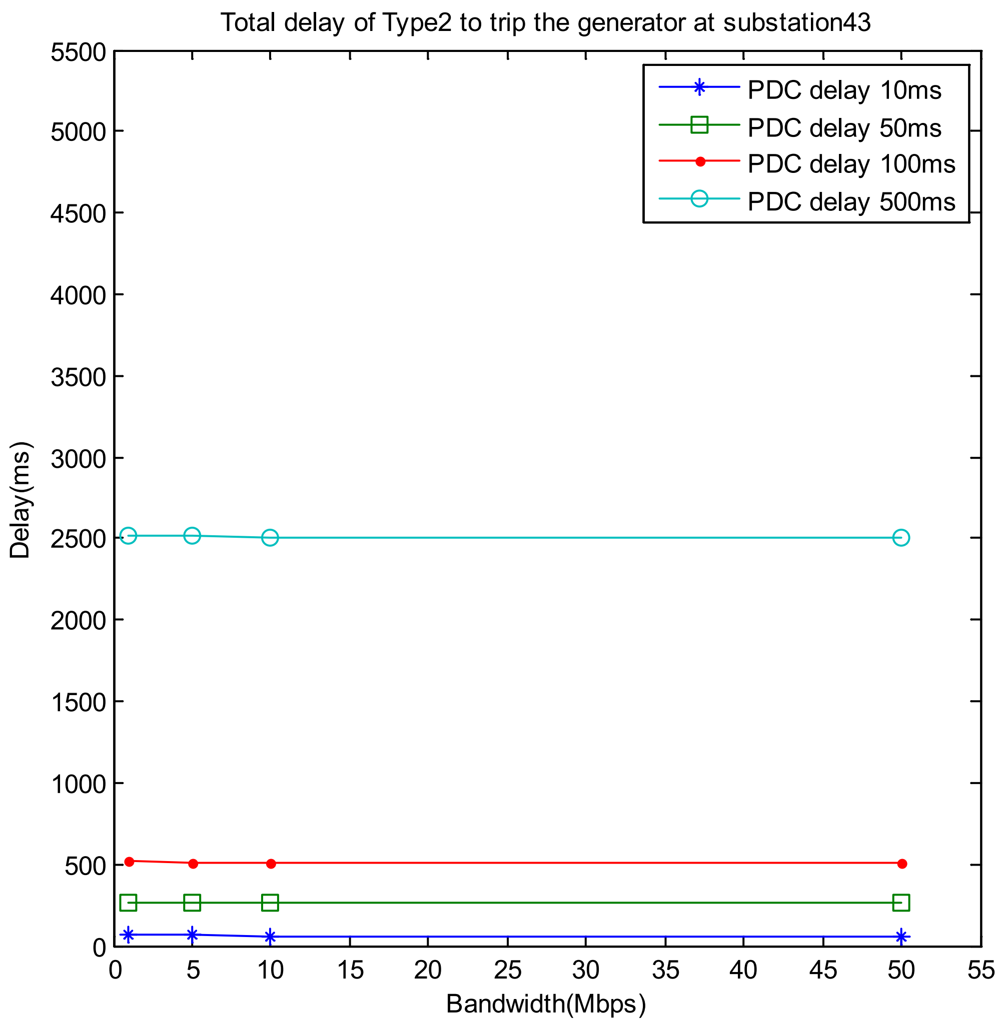

Figure 3.5 Delay to Trip the Generator in Sub43 (Type2)

It is shown in Figure 3.5 that the delays are nearly constant, which are denoted by a line. This means even near 1Mbps the queuing delays don't play a critical role in the communication delays. The maximum delays of this type is about 2500ms in PDC processing delay 500ms, which is nearly half of delays in type 1. The reason for the smaller delays is that the system become smaller (like 3 sub power system), so the data can finish transferring at the faster rate.

3.3.3 Type3: Centralized Structure

In this type, there is only one control center. It's located at Sub 9, as the type 1 does. Each substation connects to the CC directly. Thus each substation sends its data to the Sub 9 PDC and finally through that to the control center. It is clear that in this type each package has 2 PDC processing delays, itself and Sub 9. The following shows what this type looks like.

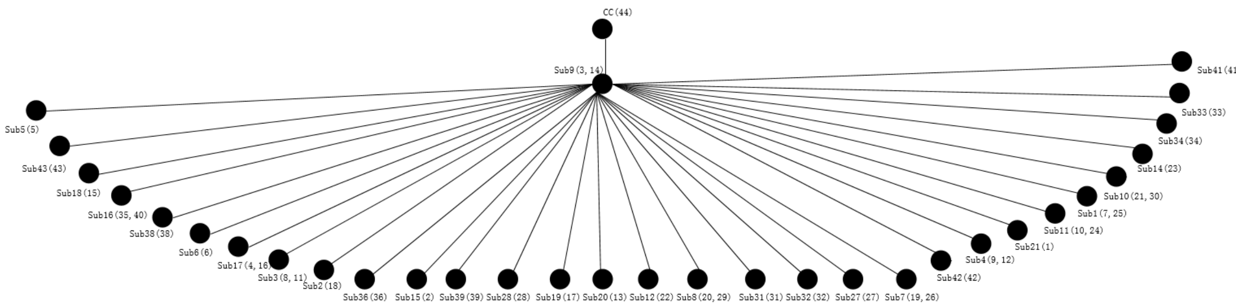

Figure 3.6 Centralized Structure

The total delay is shown below.

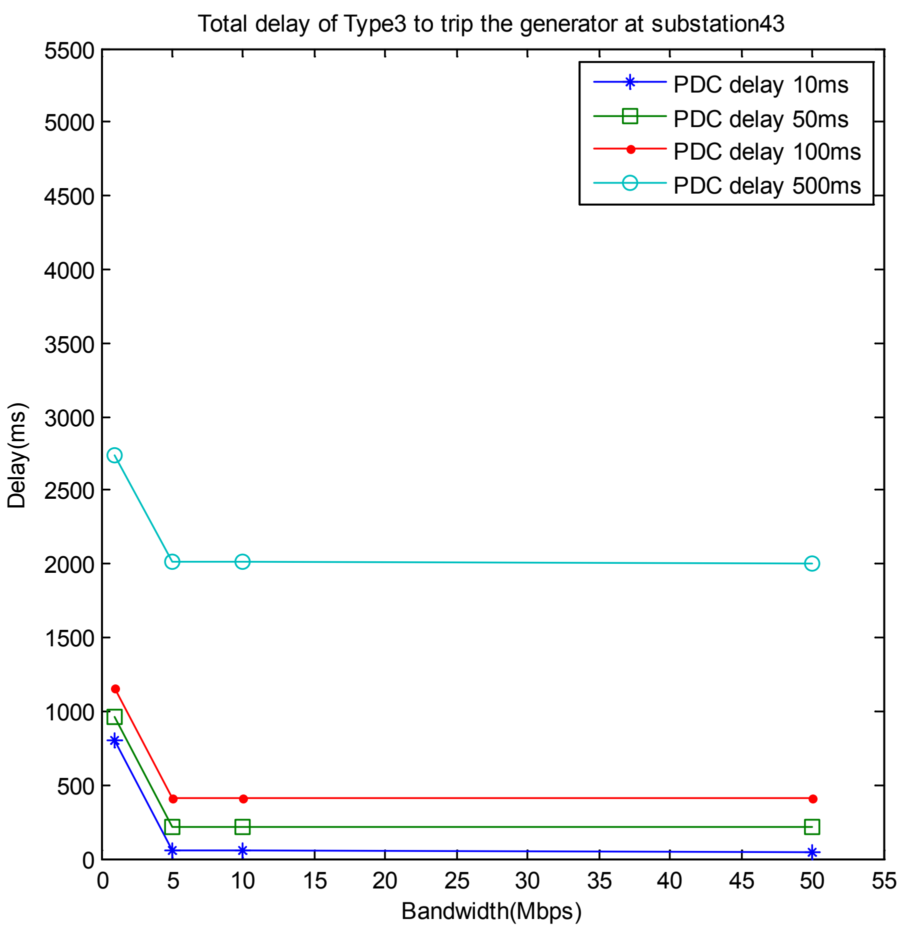

Figure 3.7 Delay to Trip the Generator in Sub43 (Type3)

It is shown that this type doesn't give any big delays as expected. The reason for this situation is that there is a maximum of two hops for the data: each packet has only two of PDC processing delay which means the total delay can't be very large even though there exists 500ms of PDC delay and bigger queuing delay.

This communication architecture is more likely realistic in today's power system. The structure is simple but it will be more costly when the distance of two communication nodes, say, between substation and control center, is far and has no intermediate switches.

3.3.4 Type4: Decentralized Structure

Compared to centralized structure in 42 bus system, this type of structure has 3 Sub CC and 1 CC. 3 Sub CC are connected to each other and they are all connected directly to CC, just like type 2. However, the communication links within each area are not along with the transmission lines; they are connected to Sub CC directly.

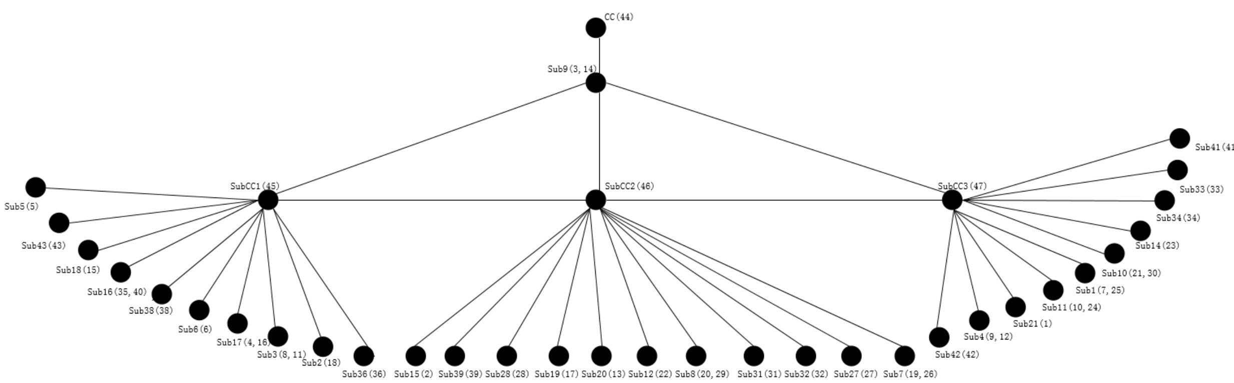

Figure 3.8 Decentralized Structure

The differences from the centralized structure are that it has 3 Sub control center to coordinate the data within its own area, and they can communicate with each other. In this situation, the delay results are shown as following.

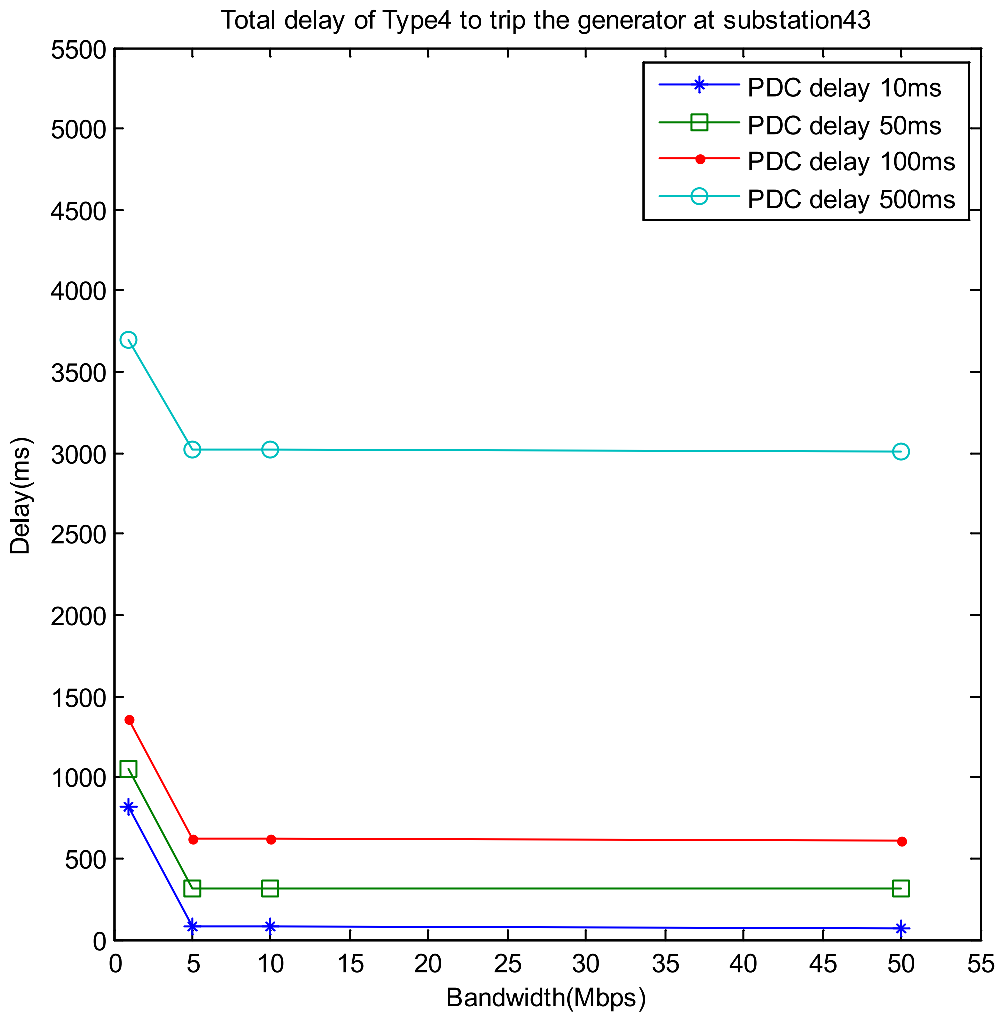

Figure 3.9 Delay to Trip the Generator in Sub43 (Type4)

This type of communication architecture is supposed to have smaller delays since it has smaller power areas. However, the opposite appears to be true. The reason is clear that each packet has to go through more than 2 PDC in Type 3 resulting in the bigger delays.

It is clear that discussions of different communication architectures are useless without any power system performance. Because of this, the next section discusses how the imperfect communication links have impact on power system control performance, especially on transient stability.

3.4 Impact of Imperfect Communication Links on Power System Transient Stability

As mentioned before, in this section only 3 circumstances are considered in power system transient stability: stability, critical stability and instability. The critical delay point to keep the power system stable is around 800ms. This means if total delay of the trip signal is greater than 800ms, then system becomes unstable even when the generator in Sub 43 is tripped. The power system performance is simulated in Power World by UIUC.

This 42 bus system is faulted in the line between Sub 43 and Sub 18 at 0.20s. At this time, the PMU data being sent to the CC has an abrupt change due to the three-phase fault. After 0.05s, the breakers open the line at both ends. And when the control center receives the fault condition data it realizes something wrong in the line between Sub 43 and Sub 18. Thus it sends the control signal back to Sub 43 to trip one of the generators in that substation to keep the system stable. If the generator is tripped in time, the system will be stable again after some oscillations. If the generator is tripped too late, the system will be unstable.

In the first place, we examine the situation (case 1) where the generator is tripped in a very short time (<300ms). This could happen in high bandwidth and low PDC processing delay cases. For example, the power system transient stability performance for a Type 3, 10Mbps network with a 50ms PDC delay is shown as follows. Here the rotor angles of two generators in Sub 43 are presented.

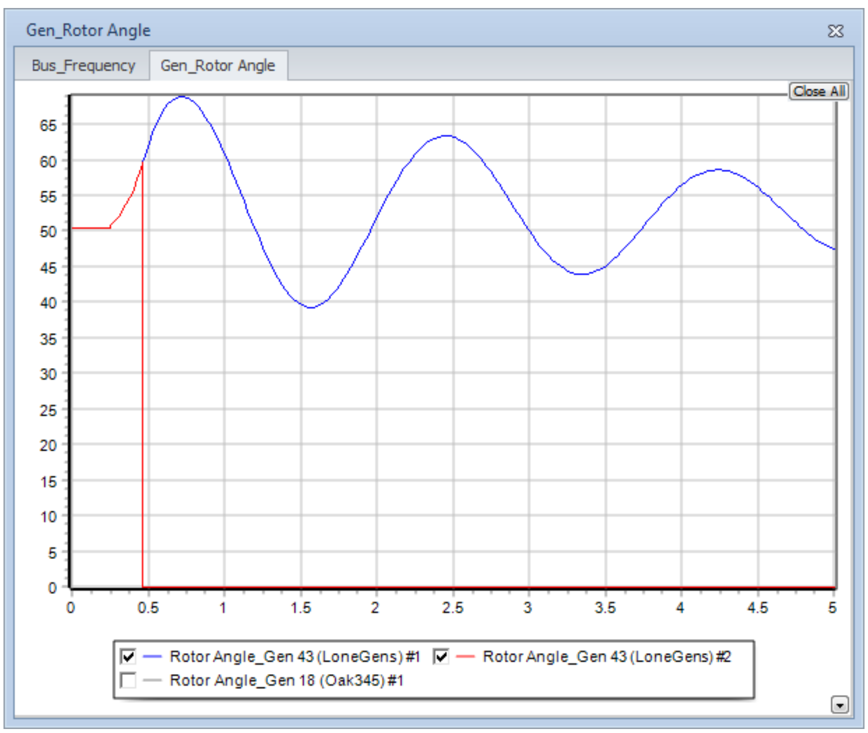

Figure 3.10 Rotor Angles of Generators in Sub 43 (Case 1)

It is shown that the maximum degree for system oscillation is roughly 70 degree. As can be seen, the red line is the generator which we tripped. The other one will be stable again after some time, which is not shown in the figure as it needs more settling time.

Similarly, a critically stable case (case 2) is considered in the next step. As mentioned before, the critical point for stability is around 800ms to trip the generator. This case could happen in a Type 4 with 2Mpbs bandwidth network and a PDC delay 100ms, which produces a round trip delay of 741ms. Here are the rotor angles.

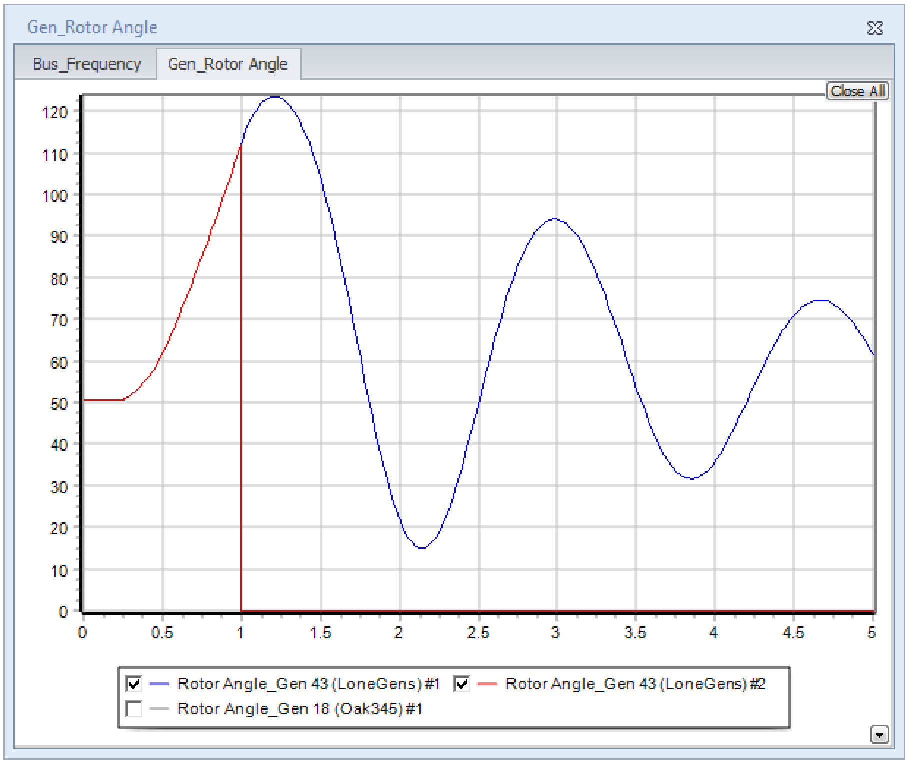

Figure 3.11 Rotor Angles of Generators in Sub 43 (Case 2)

As expected, the maximum degree for oscillation is roughly 122 degree. The reason is that the generator has to wait for a longer time to trip so the system reaches a higher angle separation. Fortunately, the system is still stable in this case.

Finally, the unstable situation (case 3) is shown. Because the signal is slow, so this could mostly happen under low bandwidth and/or big PDC processing delay. Take a Type 2 network with 10Mbps bandwidth and PDC delay of 500ms for instance. The rotor angles are shown below.

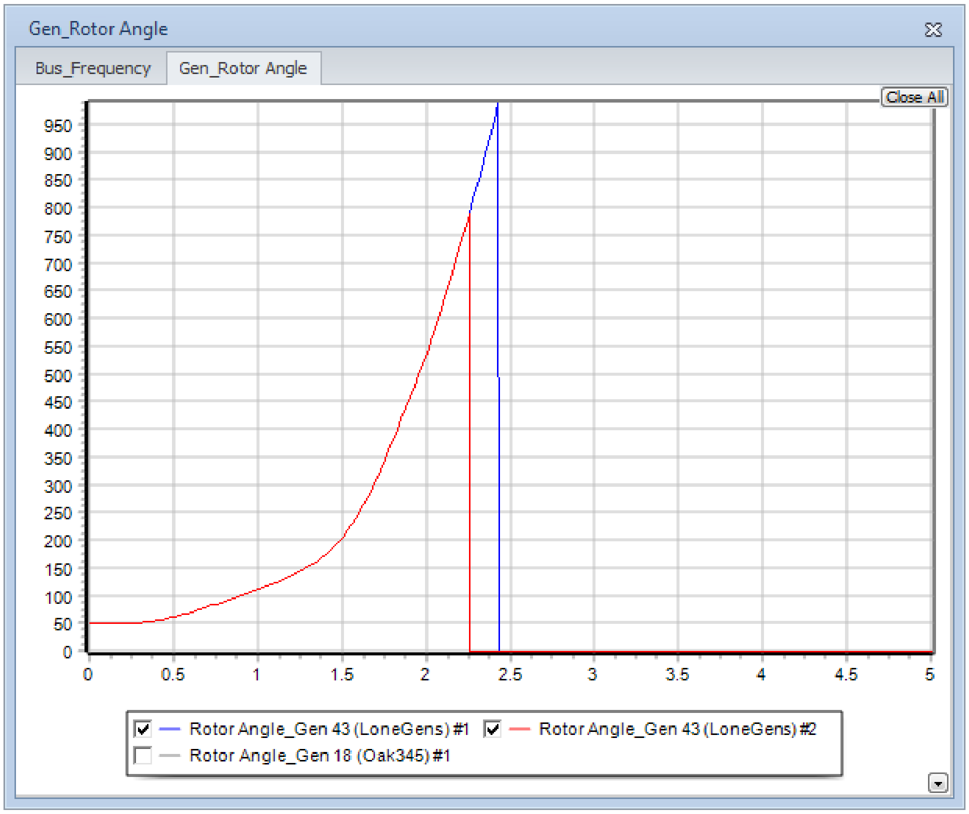

Figure 3.12 Rotor Angles of Generators in Sub 43 (Case 3)

Obviously, the system is unstable and the simulation portion which is greater than 180 degree is pointless in the power system analysis. However, we can conclude that the system is unstable as if the trip signal is not even received by the generator at Sub 43. The case presented here is dangerous since it causes system instability. In designing wide-area controls like this we have to avoid such circumstances in real life where the communication latencies are not guaranteed.

3.5 Conclusions

In the WSU part, 4 different communication architectures are proposed for this 42 bus system. These architectures are tested with channels of four different bandwidths. We also examine four different PDC processing delays to see its impact on the power system transient stability performance. In each type network for this case, only PMU data from Sub 43 to CC is our consideration since one of the generator at Sub 43 needs to be tripped. Obviously, all other wide-area controls will have to be tested the same way to ensure that the communications architecture is robust enough to guarantee the latencies needed for stability.

The cons and pros are shown below for each type:

Type 1: It's a very simple architecture since the communication links are along with the transmission lines. However, some of the links are wasted because the data may not go through them as other paths are always faster. In this way, this type is simple but wasted resources. In 42 bus system case this type might be the worst one.

Type 2: The type 2 network also uses communication links along the transmission paths but they do not go across company boundaries. As in the example, the three companies have their own sub-control centers which receive the data from each area. The Sub CCs are then connected to each other and a higher level CC. This type has better latency than type 1, but just as in type 1 some of the links are wasted.

Type 3: Today's power system communication architecture is like type 3. All data are sent to the control center. This type has only one CC connected Sub 9, so each packet has to wait in Sub 9 especially in the low bandwidth. This leads to more delays whereas it's easy to control the whole power system because of one control center.

Type 4: Similarly as type 2, this type has 3 Sub CC and they are connected to each other. The substations within each area are directly sending data to their own sub control center. This type has more delay than type 3 since it has more hubs (PDC). In this way, the PDC delay has more impact on power system control actions.

The conclusion for this part is as follows: If the PDC processing delay can be kept low, the system can be stable even with the low communication bandwidth (like Type 3 with 1Mbps). However, low bandwidth would be a bad design decision, especially with today's technology, as some of them could lead to instability in the power system due to bigger queuing delay in the low bandwidth. For the high PDC processing delays, say the 500ms cases, all structures we proposed led to instability even for the Type 2 networks which performs best the most cases. The PDC processing delay is becoming the biggest impediment part in the communication path. In these cases, they cannot be neglected as we did before. Thus, minimizing the PDC processing and choosing the most efficient communication architecture are the critical decisions for wide area control.

The next part, Georgia Tech, might use our delay information to do the control strategy.

4 Decentralized Applications (Georgia Institute of Technology)

4.1 Introduction

Previous reports under the Seamless Energy Management Systems (EMS) project have presented assessment of existing technologies, as well as architectures, and requirements for next generation EMS systems. Some of the desirable characteristics of seamless EMS are:

a) PMU based, rather than SCADA based.
b) Cyber-controlled, where the cyber aspects of communication, information and computation are formally considered in the design of the systems.
c) Decentralized, allowing for massive scalability of control and decision-making, and for coordinated interactions of systems at all scales.
d) Natively suitable for High-Performance Computing (HPC) architectures
e) Predictive, incorporating look-ahead capabilities
f) Stochastic, to address growing uncertainty due to integration of renewable and distributed energy resources.

The work described in previous sections of this report addresses dynamic simulation and communication system modeling for next generation EMS, covering major needs of items a) and b) above. In this section, we focus on the third requirement of EMS systems: decentralization. We leverage some of the features provided by cyber-control and the simulation of communication systems as well as the co-simulation of communications and power. We focus on the fast time scales associated with power system control and dynamics. This section provides a short introduction to decentralized power control requirements, co-simulation, and decentralized algorithms for power agreement and frequency regulation. Finally, we present an example of decentralized control including assessment of the effect of communication delays.

4.2 Cyber-Physical Future Grid Reference Model

The traditional control structures of the electric power grid will not be able to reliably handle the imminent deployment of large amounts of renewable energy sources, smart devices and active participants. This deficiency has resulted in a worldwide effort towards realizing the vision of a future grid [56], which proposes to overlay the electric grid with a more extensive communication, information, and computation infrastructure. Unfortunately, most of the early applications under the umbrella of smart grid have been developed in an ad-hoc manner, without an underlying framework. This lack of an underlying framework has resulted in a set of isolated technologies, standards, and applications that are difficult to integrate and extend for the future [41]. In the past, various other complex engineering domains have faced similar problems in their early days. Research communities for those engineering domains overcame this problem by developing reference models for the domain that could enable clear understanding among different stakeholders and inform the development of an integrated set of technologies and standards for that domain [42]. We leverage this idea of a reference model and propose a cyber-physical reference model for the future grid.

The proposed reference model for the future grid is based on service-oriented computing (SOC) paradigm, as this paradigm is uniquely capable of handling the large scale, open nature, and long lifecycle of future grid scenarios. However, traditional SOC paradigms, used in enterprise computing domain through popular technologies such as Web Services, cannot be directly applied to the future grid, since this paradigm is not capable of handling hard real-time aspects. Therefore, we extend the traditional SOC paradigm by introducing resource-aware service deployment and QoS-aware service monitoring phases, e.g. a cyber-physical service-oriented model for the future grid.

On the other hand, the task-based reference model used in typical distributed, real-time systems, such as automotive and avionics, cannot be directly applied to the domain of the electricity grid as this reference model is not capable of handling the large scale, open nature, and long lifecycle of future grid scenarios [43][44][45]. The reference model essentially extends the traditional service-oriented computing paradigm by introducing resource-aware service deployment and QoS-aware service monitoring phases.

According to the reference model for future grid, each scenario is characterized by three elements, shown in Figure 4.1 [46]:

1. An application model that describes the future grid applications to be supported by the system as a set of resource- and QoS-aware service descriptions.
2. A platform model that describes the future grid platform as a set of computing nodes, communication links, sensors, actuators, and power system entities.
3. A set of algorithms that achieve resource-aware service deployment, QoS-aware service discovery, and QoS-aware service monitoring.

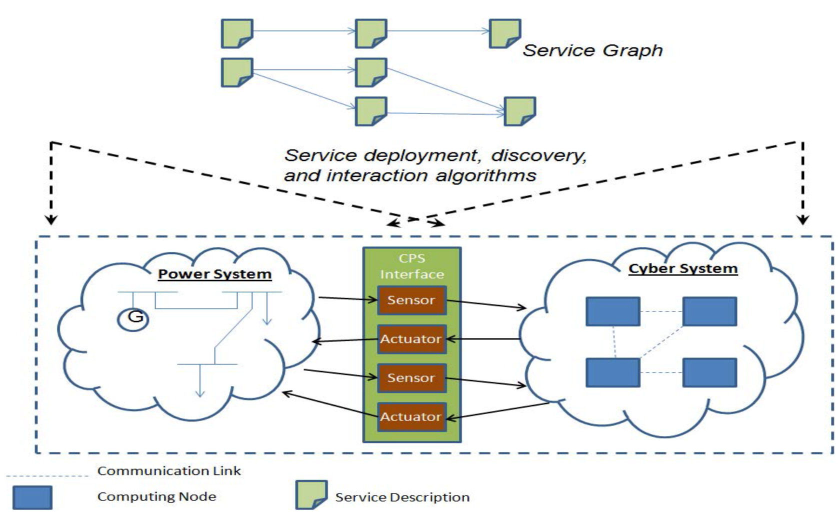

Figure 4.1 Cyber-Physical Service-Oriented Model for the Future Grid

In the platform porting step, a generic, service-based computing platform is ported to all the heterogeneous computing nodes involved in a future grid scenario. In the service modeling step, the future grid application is modeled as a set of services that interact with each other as well as with physical entities through sensing and actuation. In the service implementation phase, the implementation code for the services is developed. Both service modeling and service implementation steps contribute to the development of service descriptions. The service description of a service not only defines the messages that a service exchanges with other services, but it also defines sensing and control actions that the service takes on the co-located physical entities. Moreover, a service description identifies the quality-of-service (QoS) constraints on message exchanges with other services and platform resource requirements of a service. A service description also identifies various modes of operation of a service for various QoS fault scenarios.

In the service deployment phase, all the services are deployed on their associated computing nodes. This leads to the service discovery step, where all the services involved discover their peer services. This step could be performed online or offline depending on the nature of the future grid application. In the service interaction step, services involved in a future grid application interact by sending messages to each other. During the service interaction step, services switch between different modes of operation if QoS faults occur. Finally, through a service update phase, this future grid reference model supports system maintenance and system updates. In the service update phase, services involved in the future grid application are updated. These services again pass through service implementation and service deployment steps. Again, the service update step can be designed to work online or offline depending on the nature of future grid application [47][48][49].

4.3 Cyber-Physical Co-Simulator

Traditionally, control and communication have been different domains with little overlap. This tradition has included control of power systems. The common assumption in power system control design has been that all data communication required by the algorithm can be performed with infinite reliability, precision, and bandwidth. Communication components are hence treated decoupled from the control logic. A similar assumption in communication system is that power and energy are assumed also perfect. In reality, communication constraints such as spectrum, delays, and bandwidth are a real concern in control systems. Another important aspect that has traditionally been included in control design is the assumption that information processing, and data transmission occurs instantaneously. In reality, data arrival times are often delayed and difficult to predict. Moreover, that transmitted data may be corrupted, lost, or disrupted due to noise in the communication channel, congestion, malfunctioning of equipment or protocol, or cyber-intrusion/attack.

As the complexity of the control tasks in power systems become more complex, the effect of the "cyber-layer" limitations becomes much more important. For future grid applications, which may include a large-number of distributed energy resources (DERs) and intelligent devices, the amount of data from sensors can be much larger than the capability of economically justifiable communication links. The communication capacity limitations and therefore the problem of optimal communication network capacity become very relevant when designing controls for the future grid.

It is clear that new tools able to simulate the integrated complexity and couplings of the cyber-physical system are required. The co-simulation tool must include:

a) Cyber aspects: information, computation, and communication
b) Physical electric power aspects
c) Control

Georgia Tech has developed a cyber-physical co-simulator able to model power, communication, and control aspects and their interactions. Currently, NS3 network simulator is used to model the cyber aspects. NS3 nodes are used to encapsulate interactions of decentralized agents. The new NS3 version was assembled by assembled by John Abraham, Brian Swenson, and Umer Tariq and designed to be updated and make aligned with the Linux version. The new Window's Test Runner code was created.

The power aspects of the system are modeled using PowerWorld Simulator v. 18, using the COM object interface. Figure 4.2 shows the architecture of the "cyber-physical co-simulation environment" for the future grid. The input to the simulation environment consists of an XML file that describes the future grid scenario to be simulated, including decision-making participants, power network parameters, power states and events, communication network topology and states, channel and protocol modeling, and communication disturbances. Support for detailed power system dynamics and their controls is also available, hence allowing for a federated co-simulation.

An XML parser reads this XML file, sets up the underlying simulation environment, and starts the simulation. During the simulation run, the output is written into a set of CSV files. At the end of a simulation run, a MATLAB M-file is executed that reads these CSV files and plots the results by using MATLAB graphing capabilities. This XML file is used as an input to the above mentioned simulation infrastructure for simulating the integrated operation of decentralized applications.

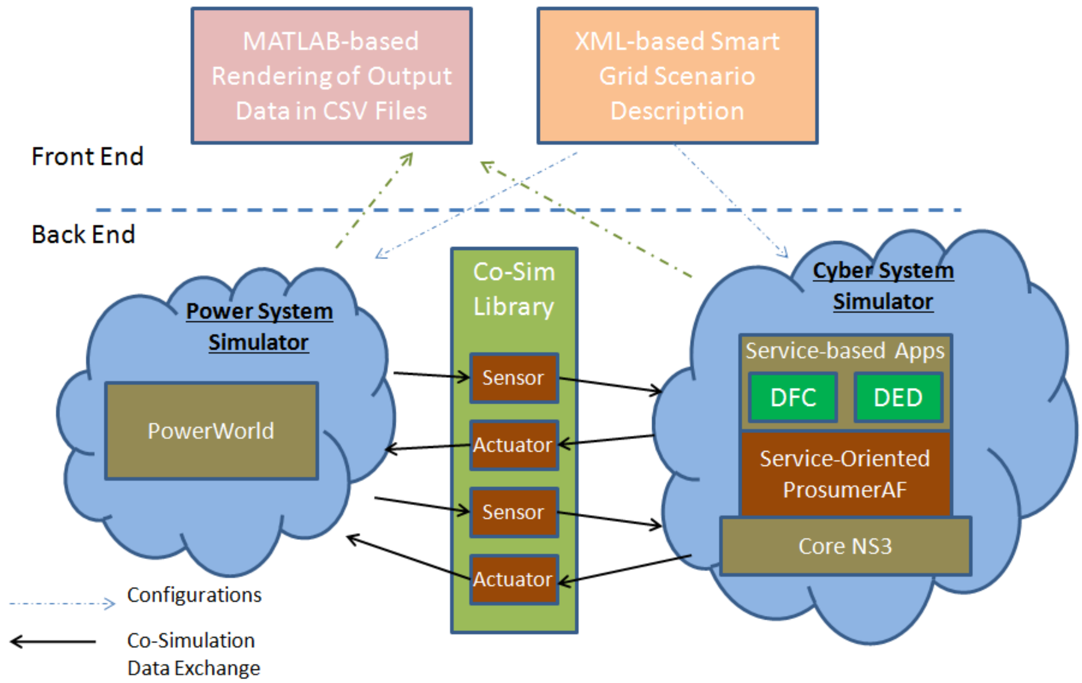

Figure 4.2 Architecture of a Computing Platform Aware Co-Simulation Environment

4.4 Need for Decentralized Control

In recent years, changes in the power industry have been posing challenges to the power grid. As renewable resources drop in cost and approach price parity with fossil power, intermittent sources will become a larger part of total generation. Additionally, power generation will be more distributed, with residential customers more frequently having generation capacity. As this shift occurs, the lines between producer and consumer become less clear leading to a hybrid prosumer. In the near future, any agent on the power grid will be able to have generation capacity, storage capacity, and loads. For a detailed discussion on this trend see [50][51]. From an external perspective, the only difference between a large utility, a residence, or micro-grid, will be scale. Noticing this similarity between different actors on the grid, we choose to look at every agent through the lenses of a new abstraction, the prosumer, which is introduced in Grijalva and Tariq [52]. In fact, the power grid is undergoing a shift from a heterogeneous network where producers and consumers are easily delineated to a homogeneous prosumer network.

The future electricity grid will be composed of billions of smart devices and millions of active energy decision-makers (microgrids, homes, buildings, DER systems, utilities, etc.) Under high penetrations of renewable energy, what happens in a region of the grid affects other regions, which results much higher requirements for coordination. Fundamentally, coordination needs to be faster, tighter, automated, and much more accurate. Because it is impossible that a single, centralized agent controls the overall grid, the control needs to be decentralized, but coordinated across subsystems and scales. This is illustrated in Figure 4.3.

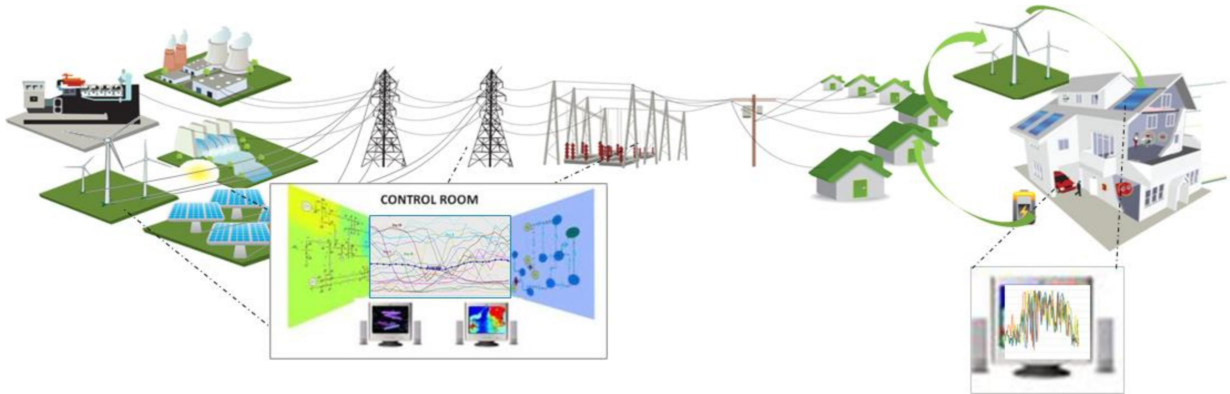

Figure 4.3 Illustration of the Future Grid with Coordination at All Scales

Decentralized control is gaining momentum within the power research community as the need for new control architecture becomes clear. Decentralized control just means that there is a formal recognition of more than one active decision-maker. The following use cases, which are topics of interest to the industry during the last decade, are all examples of decentralized control:

a) Microgrids
b) Demand Response
c) Building Energy Management Systems
d) Home Energy Management Systems
e) Building, Home, Vehicle, X to Grid
f) Transmission/distribution effects
g) Consumer Empowerment
h) Prosumers
i) Imbalance Markets
j) Distribution System Operators (DSO)
k) ISO Seams Issues
l) Wide-Area Control

For objectives of decentralized simulation, we model the power grid as a collection of connected prosumer agents and investigate distributed algorithms which can be used to control and coordinate power. We select system models that allow us to focus on structural implications of solving distributed minimization problems.

The prosumer abstraction allows us to reason about heterogeneous actors on the grid in a unified manner by considering electric power subsystems. Prosumers emerge naturally due to ownership of operational responsibility of portions of the electrical network. Prosumers may range from electric vehicles to interstate bulk power network. They can be organized in a flat, networked manner, or in a hierarchical manner. In this report, we also present algorithmic solutions to the power agreement problem for said prosumer networks. Finally, we present an investigation of how the different solutions to this problem have different structural implications for the information exchange network needed to accompany the physical electric grid. We start with modeling of the power allocation problem as a constrained weighted least squares problem assuming and derive two scalable decentralized controllers, which solves the problem and compare their topological implications on the required information network. The power allocation problem is essentially a distributed optimization problem. The connections between algorithmic choices and the information topology induced by that choice is a key focus of this work. The co-simulator described above is be used to test decentralized control such as decentralized power agreement applications including the effect of imperfect communication and delays.

4.5 Decentralized Power Agreement

Consider a set of k prosumers (arbitrary regions or balancing areas) connected through a transmission grid. We assume that the communication topology mimics the topology of the power network [53]. The power networks is represented by a graph G = (**V**, **E**), where the agents are located at vertices in the set **V** and **E** denotes the set of graph edges connecting the vertices. We assume that each agent i has computed its desired power need d taking into consideration its own load, storage, and generation capabilities. The value of d is positive for a positive export of power, and negative otherwise. The collection of agents' desired power is represented by the vector $\boldsymbol{d}$.

In a physical power network, the power produced by any node must subsequently be consumed by some other node or nodes (agents). Therefore, prosumers cannot produce or consume power in isolation. The actual power which is being produced or consumed by an agent is a determined by the amount of power which is being injected and withdrawn from the transmission/distribution infrastructure represented as power flow along the edges.

Letting **A**(G) be the node to edge incidence matrix for G, and given a flow vector **r**, we define the power vector $\mathbf{p} = \mathbf{Ar}$. The vector r represents the power flows along the edges whose directionality is determined by the incidence matrix **A**. For a power network, the directionality of each edge is arbitrary (non-directed graph). We would like to compute a power flow vector **r** such that the net weighted discrepancy between the power $\mathbf{p} = \mathbf{Ar}$ and desired power **d** is minimized. This problem can be phrased as a least-squares optimization problem,

$$\min_{r} \frac{1}{2}(\mathbf{Ar}-\mathbf{d})^{T}\,\mathbf{W}(\mathbf{Ar}-\mathbf{d}) \quad (1)$$

where **W** is a diagonal, positive definite weight matrix. The interpretation is that we are optimizing over power flows in order to ensure that agents' power needs are satisfied as closely as possible, in a least-squares sense. The weight matrix **W** captures the relative importance of each agent's need in the network. If agent j is a critical, its power needs must be met with more precision. Then the w_{jj} term is made larger. Also, smaller agents may have smaller tolerances and less strong safety mechanisms and cannot handle large power fluctuations. This is also taken into account when assigning weights for the agents.

It is possible to solve for p directly without computing r, as the solution to the minimization problem:

$$\begin{aligned}&\min_{\mathbf{p}} \frac{1}{2}(\mathbf{p}-\mathbf{d})^T \mathbf{W}(\mathbf{p}-\mathbf{d}) \\ &s.t. \quad \mathbf{1}^T \mathbf{p} = 0\end{aligned} \tag{2}$$

The constraint $\mathbf{1}^T \mathbf{p} = 0$ is equivalent to asserting that **p** belongs to range(**A**). The Lagrangian for the above problem is given by:

$$L(\mathbf{p}, v) = \frac{1}{2}(\mathbf{p}-\mathbf{d})^T \mathbf{W}(\mathbf{p}-\mathbf{d}) + v\mathbf{1}^T \mathbf{p} \tag{3}$$

where v is the Lagrange multiplier. Applying optimality conditions we have that:

$$\mathbf{p}^* = \mathbf{d} - v\mathbf{W}^{-1}\mathbf{1}; \qquad v^* = \frac{\mathbf{1}^T \mathbf{d}}{\mathbf{1}^T \mathbf{W}^{-1}\mathbf{1}} \tag{4}$$

We rewrite the equivalent optimization problem as:

$$\begin{aligned}&\min_{\mathbf{p}} \frac{1}{2}\mathbf{p}^T \mathbf{W}\mathbf{p} \\ &s.t. \quad \mathbf{D}^T \mathbf{W}(\mathbf{d}-\mathbf{p}) = 0\end{aligned} \tag{5}$$

with Lagrangian given by:

$$L(\mathbf{p}, \boldsymbol{\varphi}) = \frac{1}{2}\mathbf{p}^T \mathbf{W}\mathbf{p} + \boldsymbol{\varphi}^T \mathbf{W}(\mathbf{d}-\mathbf{p}) \tag{6}$$

We determine the Lagrange dual function as:

$$g(\boldsymbol{\varphi}) = \inf_{\mathbf{p}} L(\mathbf{p}, \boldsymbol{\varphi}) = -\frac{1}{2}\boldsymbol{\varphi}^T \mathbf{D}^T \mathbf{W}\mathbf{D}\boldsymbol{\varphi} + \boldsymbol{\varphi}^T \mathbf{D}^T \mathbf{W}\mathbf{d} \tag{7}$$

where $\mathbf{D}^T \mathbf{W}\mathbf{D}$ is the weighted Laplacian. With this, we obtain the following update law for the flow:

$$\dot{\boldsymbol{\varphi}} = \frac{\partial g(\boldsymbol{\varphi})}{\partial \boldsymbol{\varphi}} = -\mathbf{D}^T \mathbf{W}\mathbf{D}\boldsymbol{\varphi} + \mathbf{D}^T \mathbf{W}\mathbf{d} \tag{8}$$

We can reformulate the problem by using the graph Laplacian as:

$$\begin{aligned}&\min_{\mathbf{p}} \frac{1}{2}\mathbf{p}^T \mathbf{W}\mathbf{p} \\ &s.t. \quad \mathsf{L}\mathbf{W}(\mathbf{d}-\mathbf{p}) = 0\end{aligned} \tag{9}$$

The Lagrangian is given by:

$$L(\mathbf{p},\mathbf{q}) = \frac{1}{2}\mathbf{p}^T\mathbf{W}\mathbf{p} + \mathbf{q}^T \mathsf{L}\,\mathbf{W}(\mathbf{d}-\mathbf{p}) \tag{10}$$

where q are the graph vertices potentials. This implies that each agent needs to keep track of its graph potentials using an update law:

$$\dot{\mathbf{q}} = \mathbf{W}(-\mathsf{L}\,\mathbf{q} + \mathbf{d}) \tag{11}$$

Equation provides a mechanism for decentralized algorithm determination of power agreement using the described dynamics. Figure 4.4 illustrates the rates of convergence for random generator power disagreement initial conditions for various topologies using 50 agents. The rate of convergence is associated with the second eigenvalue of the communication graph Laplacian.

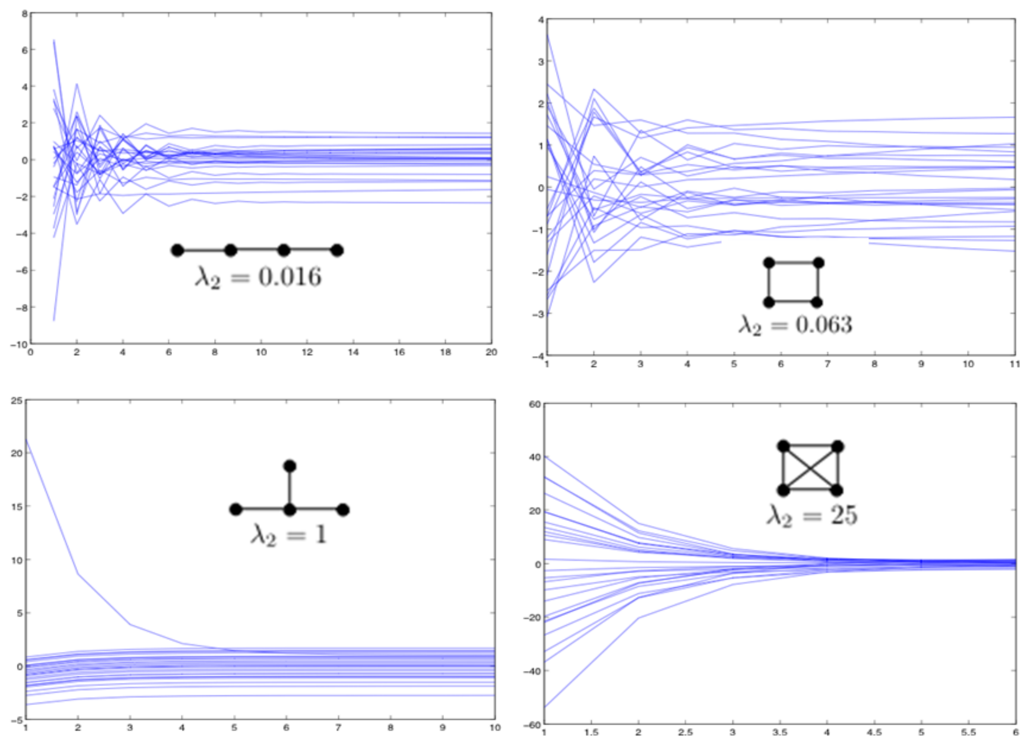

Figure 4.4 Rate of Convergence for Various System Topologies [53]

4.6 Decentralized Frequency Regulation

The state-of-the-art in frequency regulation or secondary frequency control adopts one of two different architecture designs:

a) The current approach, which is unilateral or fully decentralized;

b) Centralized with wide area monitoring and (closed-loop) control systems (WAMCS).

Under the decentralized architecture, each control area measures its local power deviations and deviations on tie-line flows and adjusts internal frequency regulators in response to these deviations. Theoretically, this is similar to decomposing a large-scale optimal control problem into sub-problems and solving each sub-problem separately without considering coupling constraints [54]. The main drawback of the current approach is a lack of coordination between frequency regulators. In order to reduce this problem, different methods have been proposed, such as implementing power system stabilizers (PSS) and/or FACTS devices, to enhance the damping of oscillatory modes [55]. Unfortunately, none of the proposed methods could guarantee frequency stabilization.

The second architecture for frequency regulation relies on WAMCS systems to collect information from different parts of the grid. Although this architecture has the potential to obtain near-optimal control strategies without creating inter-area oscillations, it needs a large-scale centralized control/communication infrastructure and has a single point of failure, which makes the system vulnerable to cyber-attacks. Note that currently WAMCS systems are only monitoring the state of the grid and not providing closed-loop automatic control.

To overcome these challenges, in [56], a distributed framework for frequency regulation of prosumer-based energy systems is proposed. The DFR framework, proposed in [8] and extended in this paper, only requires one-hop communication between prosumers and allows prosumers to obtain optimal control strategies through a consensus-based ADMM (Alternating Direction Method of Multipliers) method. This method has been successfully used for other decentralized power algorithms [57].

Frequency regulation includes bringing frequency deviations to the desired value, 60 or 50 Hz depending on the country, using minimal control effort. This is indeed an optimal control problem whose objective is to drive the power deviations to zero using minimal control effort. Decentralized Frequency Control (DFC) seeks to:

$$\begin{aligned} &\min_{\mathbf{u}} J(\mathbf{x}(t_c),\mathbf{u}) = \min_{\mathbf{u}} \sum_{i \in N} p_i x_i (t_c+1)^2 + r_i u_i^2 \\ &s.t.\;\; x_i(t_c+1) = a_{ii}x_i(t_c) + b_{ii}u_i(t_c) + \sum_{j \in N_i} a_{ij}x_j(t_c) + b_{ij}u_j \end{aligned} \tag{12}$$

Where N is the set of all agents in the network, and **u** and **x** are the control input and state vectors, respectively. P corresponds to the power deviations for each agent. $\mathbf{A} = [a_{ij}]$ and $\mathbf{B} = [b_{ij}]$. Using the Alternate Direction Method of Multipliers (ADMM) [58][59], and One-Step MPC [60][61][62], where we denote by $\mathbf{U}=[u_{ij}]$ the perception of agent i from the control action of its neighbor j, we obtain the following formulation of the DFC problem:

$$\begin{aligned} &\min_{\mathbf{U}} = \sum_{i-1}^{n} \left(p_i \left[\mathbf{A}_i^T X_i + \mathbf{B}_i^T U_i \right]^2 + r_i U_{ii}^2 \right) \\ &s.t.\;\; U_{ij} = U_{jj,} \quad \forall i \in N, j \in N_i \end{aligned} \tag{13}$$

In order to solve the DFC problem, the constraints are augmented in the objective function and the ADMM method is used to produce the augmented Lagrangian function as:

$$\mathrm{L}_{\rho,i}(U_i,U_i^h,\lambda_i^h)=p_i\left[\mathbf{A}_i^T X_i+\mathbf{B}_i^T U_i\right]^2+r_i U_{ii}^2+\lambda_i^{h,T}(U_i-\bar{U}_i^h)+\frac{\rho}{2}\|U_i-\bar{U}_i^h\|_2^2 \tag{14}$$

where ρ >0 is a given penalty factor, and $\bar{U}_i^h$ is a column vector which includes the average control strategy of agent I and that of its neighbors, as:

$$\bar{U}_{ij}^h=\frac{\sum_{l\in N_j\cup\{j\}} U_{lj}^k}{|N_j|+1},\quad \forall j\in N_i\cup\{j\} \tag{14}$$

At teach iteration, each agent solves for its optimal control strategy by solving a self-constrained problem of the form:

$$U_i^{h+1}=\arg\min_{U_i}\mathrm{L}_{\rho,i}(U_i,\bar{U}_i^h,\lambda_i^h) \tag{15}$$

Agents then share their perceptions with their neighbors and continue this process until errors in power deviations and errors in perceptions become smaller than a tolerance value.

4.7 Example: Decentralized Control with Communication Delays

Let us consider an example of Decentralized Power Agreement. Assume a power system partitioned into various regions. Let us assume that region may need to change reference production (net interchange) due to one of the following:

a) Power plant emergency
b) Contingency
c) Variation in Renewables

Each region has the following variables associated with the processing of decentralized power agreement:

a) Desired power (ex-ante)
b) An agreed upon power based on the decentralized algorithm
c) The actual power (ex-post).

By using the decentralized power agreement protocol, we know that after a disturbance, regions can agree on the needed levels of interchange to balance the system in a decentralized manner. But, how fast can the agreement be reached? And how the time needed for the agreement affects the dynamic performance of the system.

Consider the small 42-case in the Figure below. The dynamic behavior of the system is simulated using conventional transient stability models and numerical integration techniques in an interactive environment described in Section 2 of this report. The System is divided in various regions with generation capabilities (prosumers or control areas).

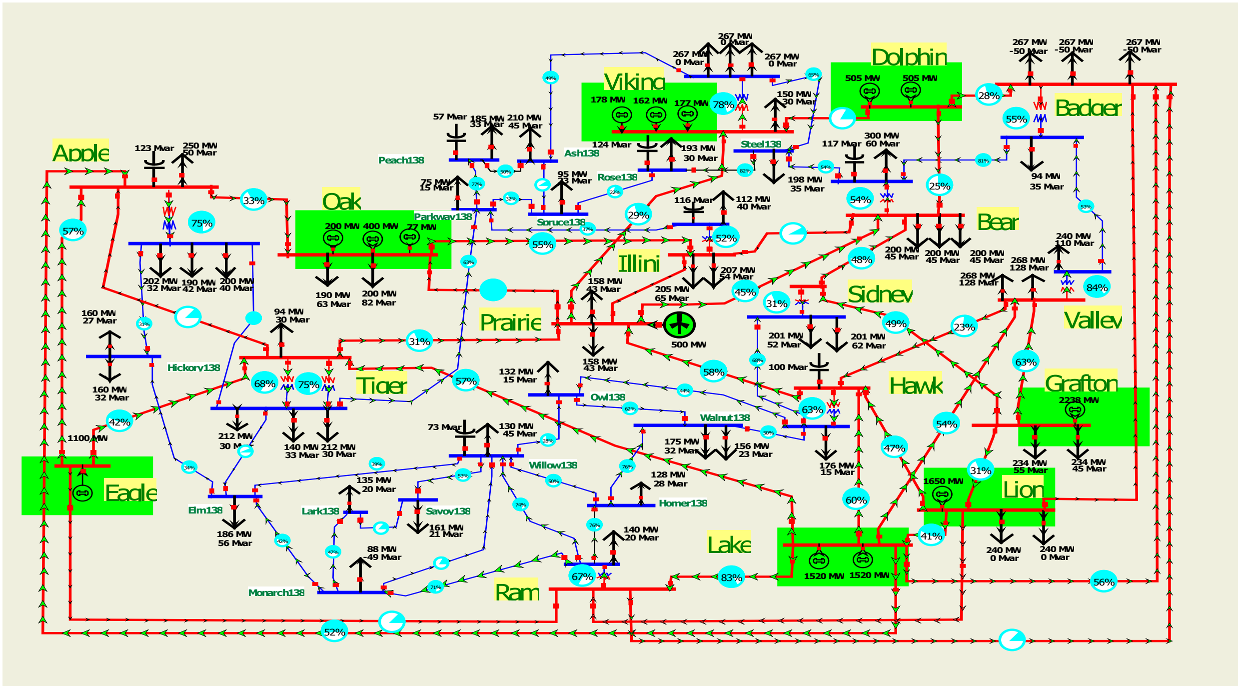

Figure 4.5 Sample System for Decentralized Control Simulation

Figure 4.6 shows the system with the corresponding partitions and communication lines. In order to simulate dynamic performance, we assume that the system suffers from disconnection of a large generator as illustrated in the Figure.

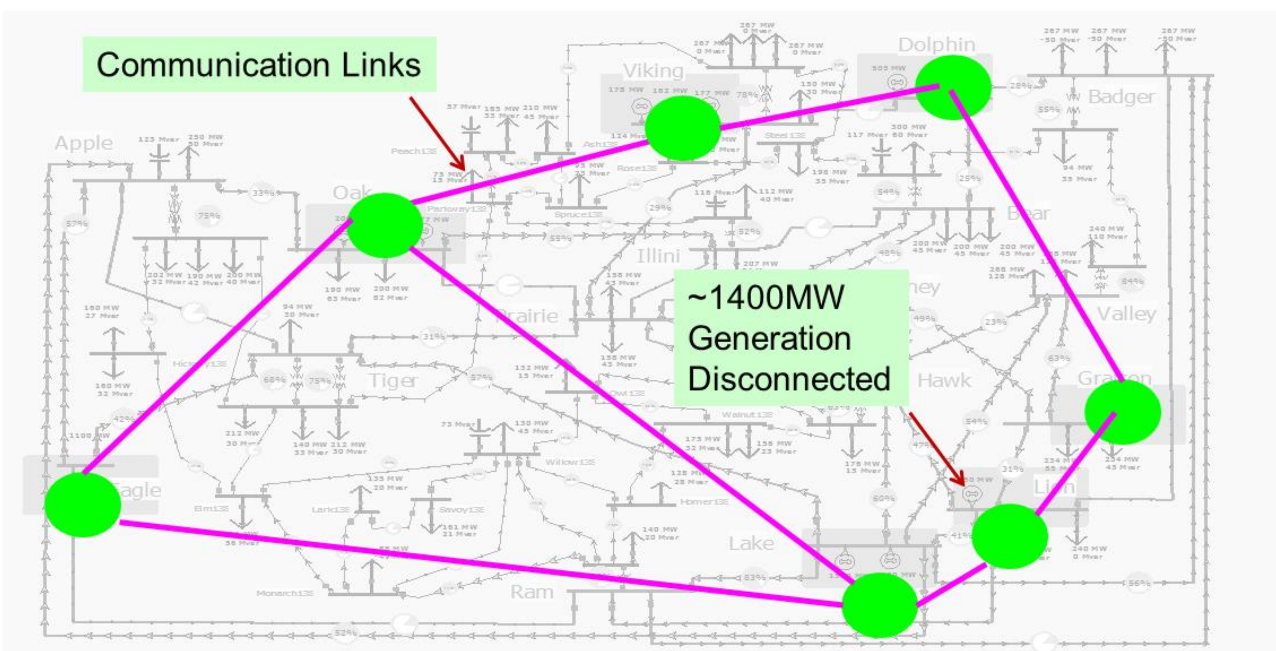

Figure 4.6 Communication Topology of the 42 Bus, 7-Region System

Sequence of Actions

1. At time t = 10 sec a line trips, causing a minor, stable oscillations
2. At time t = 25 seconds, a generator producing 1,400 MW is disconnected in region *i* causing a total system imbalance and inner region imbalance equal toΔPi.
3. As soon as the generator trips, the control agent of region *i* invokes the decentralized power agreement protocol to match the imbalance:

$$\sum_{j=1, j\neq i}^{N} \Delta P_j = \Delta P_i$$

4. Regions reach agreement in a few iterations.
5. Regions take action immediately after agreement has been reached, by ramping up corresponding generating units.
6. Performance of the system depends on communication delays.

Figure 4.7 below illustrates how the various agents reach agreement on power in a decentralized manner. We see that in 10-12 iterations of the algorithm, the regions determine (agree on) the power needed to compensate for the generator trip. It is important to emphasize that these are not actual produced generator power, but rather the evolution of each agents' perception of the needed power change to reach agreement.

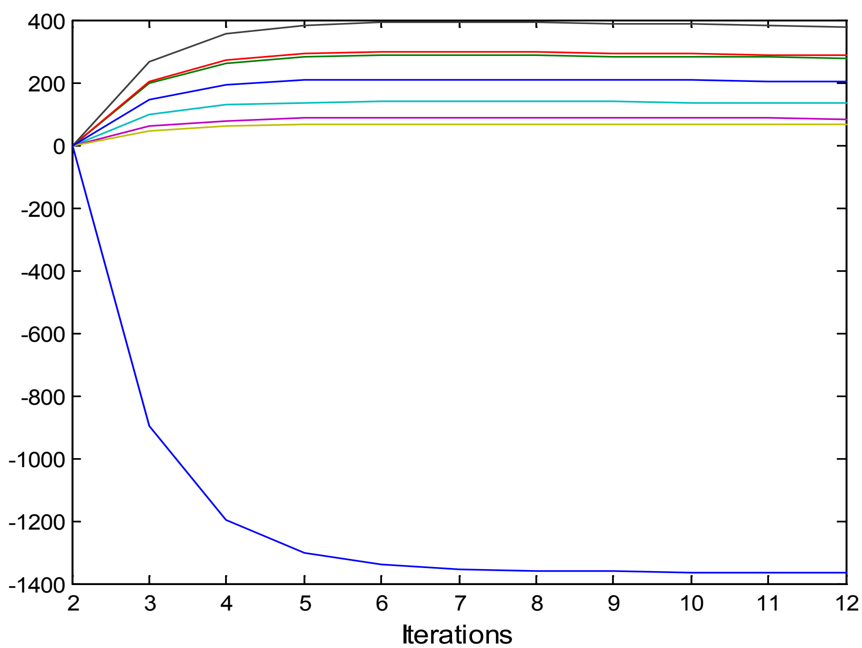

Figure 4.7 Decentralized Power Agreement: Agreement is Reached in 10-12 Iterations

Figure 4.8 below illustrates the transient behavior of the systems. The plots show the initial impact of the line trip, followed by stable oscillation. At t = 25 sec, the generator trips causing a change in the system frequency. In the Base Case, there is no power agreement involved. The generators respond based on their given controllers. The system frequency changes to about 59.1Hz recovering balance converging to a frequency of about 59.48 Hz.

The rest of the lines present the system behavior for the case when power agreement was invoked. The numbers closer to the lines represent the total time to agreement from when the protocol was invoked to when the regions start to initiate control actions. We considered the range from 0.1 seconds to 5 seconds. We note the following:

1. For all cases of decentralized agreement protocol, the frequency tends to recover to nominal. This is expected because the new injections are now not unilateral, but guided by the agreement in power, intended to balance generation with demand.
2. The faster agreement is reached, the less severe the resulting frequency excursion is, resulting in some improvement with up to 5 seconds delay.

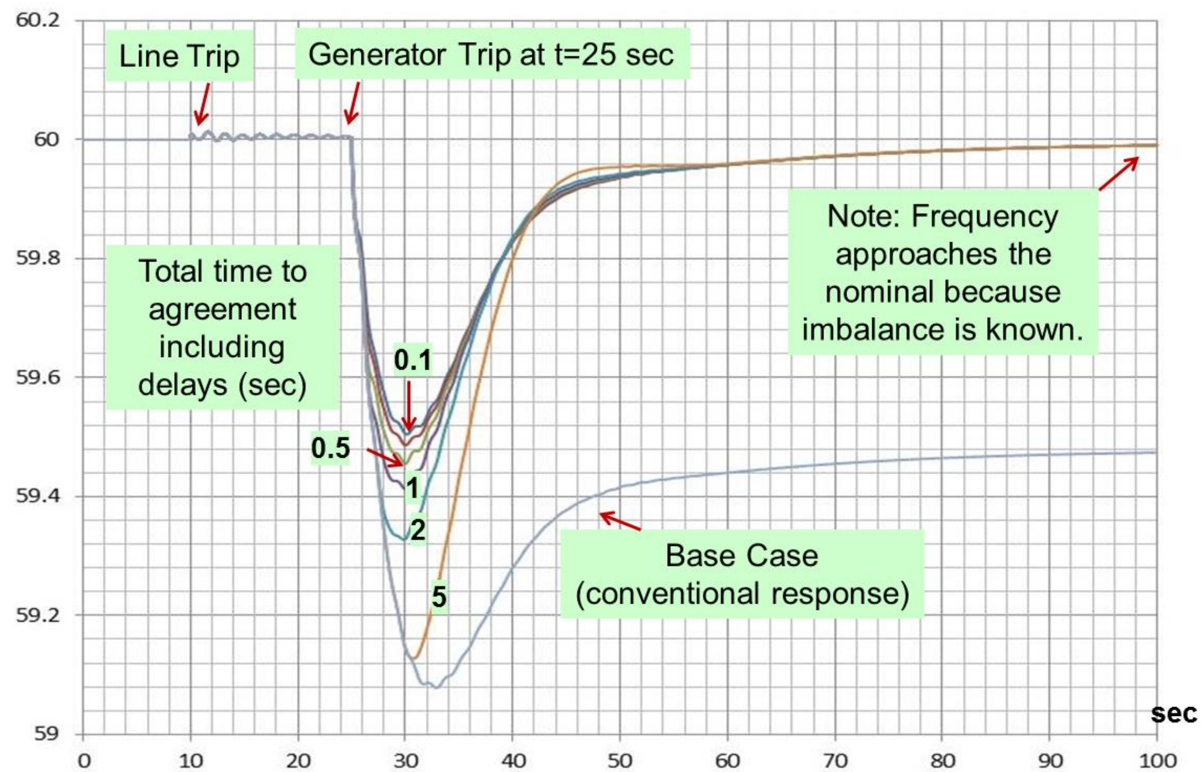

Figure 4.8 Performance of Transient Response

The example of power agreement protocol above is pretty extreme and it was selected to illustrate an application towards the faster scales of control and operation of power system. We see that the algorithm reaches agreement within 10-12 iterations. That means that the agents need to exchange information with their neighbors 10-12 times. Computationally speaking each agent's portion of the computation is not expensive and hence communication will represent the vast majority of the delay time. The above use case of decentralized control and its response could be implemented with current communication technology.

4.8 Conclusions

The growing complexity of control and operation of emerging power systems requires cyber-physical modeling and simulation capability. We have presented a cyber-power co-simulation environment that supports investigation of communication and computation realities and their effects in power system control and operation. This research has described the design of this co-simulator.

The complexity of the future grid involves many emerging decision-makers and a billion smart devices that need to be controlled and operated in a coordinated manner. The control of the future grid will therefore be decentralized. In this section we presented two scalable decentralized power algorithms: power agreement and decentralized frequency control. Communication and computation realities have direct effects on distributed networked control algorithms.

We have developed an example of dynamic behavior of power system under a decentralized power agreement control regime, including effects of delays on communication networks and illustrated some potential benefits.

References

[1] Grijalva, Santiago. "Integrating Real-Time Operations and Planning using Same-Format Power System Models." *Power Engineering Society General Meeting, 2007.* IEEE. 2007.

[2] Grijalva, Santiago, and Anirban Roy. "Automated Handling of Arbitrary Switching Device Topologies in Planning Contingency Analysis: Towards Temporal Interoperability in Network Security Assessment." *Power Systems, IEEE Transactions on* 28.2 (2013): 1523-1530.

[3] Podmore, Robin, and Marck Ramon Robinson. "The role of simulators for smart grid development." *Smart Grid, IEEE Transactions on* 1.2 (2010): 205-212.

[4] Machowski, Jan, Janusz Bialek, and James Richard Bumby. *Power system dynamics and stability*. John Wiley & Sons, 1997.

[5] Podmore, R., J. C. Giri, M. P. Gorenberg, J. P. Britton, and N. M. Peterson. "An advanced dispatcher training simulator." *Power Apparatus and Systems, IEEE Transactions on* 1 (1982): 17-25.

[6] Overbye, Thomas J., Peter W. Sauer, Caroline M. Marzinzik, and George Gross. "A user-friendly simulation program for teaching power system operations." *Power Systems, IEEE Transactions on* 10, no. 4 (1995): 1725-1733.

[7] Zimmerman, Ray D., and Robert J. Thomas. "PowerWeb: A tool for evaluating economic and reliability impacts of electric power market designs." In *Power Systems Conference and Exposition, 2004. IEEE PES*, pp. 1562-1567. IEEE, 2004.

[8] Dommel, Hermann W. "Digital computer solution of electromagnetic transients in single and multiphase." *Networks IEEE* 88.4 (1969): 388-399.

[9] Mets, Kevin, Juan Aparicio Ojea, and Chris Develder. "Combining power and communication network simulation for cost-effective smart grid analysis."*Communications Surveys & Tutorials, IEEE* 16.3 (2014): 1771-1796.

[10] D. Anderson, C. Zhao, C. H. Hauser, V. Venkatasubramanian, D. E. Bakken, A. Bose, "A Virtual Smart Grid," *IEEE Power and Energy Magazine*, Jan/Feb 2012, pp. 49-57.

[11] K. Zhu, S. Deo, A. T. AL-Hammouri, N. Honeth, M. Chenine, D. Babazadeh, L. Nordstrom, "Test Platform for Synchrophasor Based Wide-Area Monitoring and Control Applications," Proc. IEEE PES 2013 General Meeting, July 2013, Vancouver, BC.

[12] D. M. Nicol, C. M. Davis, T. J. Overbye, "A Testbed for Power System Security Evaluation," *International Journal of Information and Computer Security*, vol. 3, number 2, pp. 114-131, 2009.

[13] PowerWorld Transient Stability Add-on, http://www.powerworld.com/products/simulator/add-ons-2/transient-stability.

[14] J. D. Weber and T. J. Overbye, "Voltage contours for power system visualization," *IEEE Trans. on Power Systems*, pp. 404-409, February, 200.

[15] University of Washington Power Systems Test Case Archive, https://www.ee.washington.edu/research/pstca/.

[16] J. D. Glover, M.S. Sarma, T. J. Overbye, *Power System Analysis and Design*, Fifth Edition, Cengage Learning, 2012.
[17] University of Illinois Power Dynamics Scenarios, http://publish.illinois.edu/smartergrid/Power-Dynamics-Scenarios/.
[18] C. Ware, *Information Visualization: Perception for Design*, 2nd Edition, Morgan Kaufmann, Boston, MA, 2004.
[19] R. Klump, W. Wu, G. Dooley, "Displaying Aggregate Data, Interrelated Quantities, and Data Trends in Electric Power Systems," Proc. 36th Hawaii International Conference on System Sciences, Waikaloa, HI, January 2003 .
[20] R. Klump, R.E. Wilson, K.E. Martin, "Visualizing Real-time Security Threats using Hybrid SCADA./PMU Measurement Displays," Proc. 38th Hawaii International Conference on System Sciences, Poipu, HI, January 2005.
[21] E. Tufte, *Beautiful Evidence*, Graphics Press, Cheshire, CT, 2006, ISBN 978-096139217.
[22] OSIsoft, LLC., "PI Data Archive 3.4.390 Introduction to System Management Guide_EN", Available on-line at https://techsupport.osisoft.com/ April 8, 2014.
[23] OSIsoft, LLC., "PI-API Installation Instructions", Available on-line at https://techsupport.osisoft.com/ April, 2015.
[24] OSIsoft, LLC., "PI ProcessBook 2014 User Guide", Available on-line at https://techsupport.osisoft.com/, April, 2014.
[25] Tholomier, D., H. Kang, and B. Cvorovic. "Phasor measurement units: Functionality and applications." *Power Systems Conference, 2009. PSC'09.* IEEE, 2009.
[26] Department of Energy. "The Smart Grid: An Introduction ".
[27] Kansal, Prashant, and Anjan Bose. "Smart grid communication requirements for the high voltage power system." *Power and Energy Society General Meeting, 2011 IEEE*. IEEE, 2011.
[28] Kurose, J., and K. Ross. *Computer Networking: A Top Down Approach, 4e*. Vol. 1. 2012.
[29] Henderson, Tom, et al. "ns-3 tutorial." (2008).
[30] Kansal, Prashant, and Anjan Bose. "Bandwidth and latency requirements for smart transmission grid applications." *Smart Grid, IEEE Transactions on* 3.3 (2012): 1344-1352.
[31] Zimmermann, Hubert. "OSI reference model--The ISO model of architecture for open systems interconnection." *Communications, IEEE Transactions on* 28.4 (1980): 425-432.
[32] Martin, Kenneth E., et al. "IEEE standard for synchrophasors for power systems." *Power Delivery, IEEE Transactions on* 13.1 (1998): 73-77.
[33] Bose, Anjan. "Smart transmission grid applications and their supporting infrastructure." *Smart Grid, IEEE Transactions on* 1.1 (2010): 11-19.
[34] UDP, User Datagram Protocol, and Datagram Sockets. "User Datagram Protocols." *Transport* (2014).
[35] Postel, Jon. "Transmission control protocol." (1981).
[36] Postel, Jon. "Internet protocol." (1981).
[37] Deering, Stephen E. "Internet protocol, version 6 (IPv6) specification." (1998).

[38] IEEE 802.3x, Specification for 802.3 Full Duplex Operation, IEEE Standard 802.3, 1997 Edition.
[39] Rappaport, Theodore S. "Wireless communications: principles and practice". *Vol. 2. New Jersey: Prentice Hall PTR*, 1996.
[40] Wang, Yannan, Pradeep Yemula, and Anjan Bose. "Decentralized communication and control systems for power system operation." *Smart Grid, IEEE Transactions on* 6.2 (2015): 885-893.
[41] Zimmermann, H.: OSI reference model–The ISO model of architecture for open systems interconnection. *IEEE Transactions on Communications* 28(4), 425–432 (1980).
[42] NIST Special Publication 1108R2, NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0. (2012).
[43] Liu, J.W.S.: Real-time systems. Prentice Hall, New Jersey (2000).
[44] Erl, T.: Service-oriented architecture: concepts, technology, and design. Prentice Hall, New Jersey (2005).
[45] Kopetz, H.: Real-time systems: design principles for distributed embedded applications. Springer, New York (2011).
[46] M. U. Tariq, S. Grijalva, M. Wolf, "A Service-Oriented, Cyber-Physical Reference Model for Smart Grid", 2014.
[47] Conejo, A., Morales, J., Baringo, L.: Real-time demand response model. IEEE Transactions on Smart Grid 1(3), 236–242 (2010).
[48] Khaitan, S.K., McCalley, J.D.: Cyber physical system approach for design of power grids: a survey. In: Proceedings of the IEEE Power and Energy Society General Meeting, pp. 21–25 (2013).
[49] Khaitan, S.K., McCalley, J.D.: Design techniques and applications of cyber physical systems: a survey. *IEEE Systems Journal* PP (99), 1–16 (2014).
[50] S. Grijalva, M. Costley, and N. Ainsworth, "Prosumer-based control architecture for the future electricity grid," in Proc. 2011 IEEE Int. Conf. Control Applications (CCA), 2011, pp. 43–48, IEEE.
[51] Ipakchi, A. and Albuyeh, F. "Grid of the Future", *Power and Energy Magazine*, IEEE, 7(2), 52-62, 2009.
[52] S. Grijalva and M. Tariq, "Prosumer-based smart grid architecture enables a flat, sustainable electricity industry," in Proc. 2011 IEEE PES Innovative Smart Grid Technologies (ISGT), Jan. 2011, pp. 1–6.
[53] Ramachandran, Costello, Kingston, Grijalva, Egerstedt. Distributed Power Allocation in Prosumer Networks, NecSys, 2012.
[54] Kundur, P.: Power system stability and control. McGraw-Hill, New York (1994).
[55] M. Ilic and X. Liu, "A modeling and control framework for operating large-scale electric power systems under present and newly evolving competitive industry structures," *Math. Probl. Eng.*, vol. 1, no. 4, pp. 317–340, 1995.

[56] M. Nazari, Z. Costello, J. Feizollahi, S. Grijalva, M. Egerstedt, "Distributed Frequency Control of Prosumer-Based Electric Energy Systems," Power Systems, *IEEE Transactions on Power Systems*, 29(6), pp. 2934-2942, Nov. 2014.
[57] M. Feizollahi, S. Ahmed, M. Costley, S. Grijalva, "Large-Scale Decentralized Unit Commitment" *Elsevier International Journal of Electrical Power and Energy Systems*, April, 2015.
[58] M. Rotkowitz and S. Lall, "A characterization of convex problems in decentralized control," *IEEE Trans. Autom. Control*, vol. 51, no. 2, pp. 274–286, 2006.
[59] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," Found. Trends Mach. Learn., vol. 3, pp. 1–122, Jan. 2011.
[60] G. Droge and M. Egerstedt, "Adaptive time horizon optimization in model predictive control," in Proc. Amer. Control Conf. (ACC), 2011, 2011, pp. 1843–1848.
[61] C. R. P. S. D. Q. Mayne and J. B. Rawlings, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–814, 2000.
[62] A. Venkat, I. Hiskens, J. Rawlings, and S. Wright, "Distributed MPC strategies with application to power system automatic generation control," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 6, pp. 1192–1206, 2008.