Spring 7-7-2017

Impact Analysis of Increased Dispatchable Resources on a Utility Feeder in OpenDSS

Crystal Eppinger
Portland State University

Impact Analysis of Increased Dispatchable Resources on a Utility Feeder in OpenDSS

by

Crystal Eppinger

A thesis submitted in partial fulfillment of the
requirements for the degree of

Master of Science
in
Electrical and Computer Engineering

Thesis Committee:
Robert Bass, Chair
Jonathan Bird
Mark Faust

Portland State University
2017

Abstract

Oregon utilities are replacing their portfolios of traditional fossil fuel generation with renewable generating sources. Stepping away from carbon-producing energy will leave a deficit of on-demand power, resulting in decreased reliability. To overcome these technical challenges, utilities must maximize the use of their present dispatchable resources. One such resource is the Portland General Electric (PGE) Dispatchable Standby Generation Program (DSG), which is an aggregated 105 MWs of distributed generation (DG). These resources are brought on-line when there is a critical need for power. Resources are added to the program if a transfer trip scheme is in place or a modeling study reveals that the feeder load is at least three times the generator capacity. If the load-to-capacity ratio were lower, more assets could be added to the DSG program.

To investigate the impacts of lowering the DG load-to-capacity ratio on existing distribution feeders, we use Open-Source Distribution System Simulator (OpenDSS). We modeled the Oxford Rural feeder by converting a utility CYME database to instantiation files using several MATLAB programs. A MATLAB control program varies the load-to-capacity ratio of the OpenDSS feeder model and monitors the generator behavior immediately following a fault. We analyzed the results to determine the ideal load-to-capacity ratio that prevents unintentional islanding. The results show that the instantaneous (50) relay element settings

dictate both the minimum load-to-capacity ratio and the maximum DG capacity. The present three-to-one ratio is very conservative and can be reduced.

Additional dispatchable resources include a five MW battery-inverter system currently used as grid-back up. The battery is grid-tied to a 12.4 kV feeder making it an ideal candidate for conservation voltage reduction (CVR). Using the same feeder model, we investigated the effects of lowering the system voltage to the allowable minimum using injections of reactive power. A lower system voltage reduces the load at peak times. Conversely, increasing the voltage prevents generation conflicts. To determine the benefit of CVR by VAr-injection on the Oxford Rural feeder, we created a MATLAB optimization program to output the optimal feeder voltage for reduced system power. We use a Simulink feedback model to determine the appropriate reactive power needed to achieve the voltage change. We analyze the system model to reveal that the feeder is ideal for CVR but the system capacity must be increased to achieve the maximum power reduction.

Dedication

This thesis is dedicated to Emily Barrett and Linda Rankin. You two ladies make Wonder Woman look like a real slouch.

Acknowledgements

This thesis wouldn't be possible without the efforts of Dr. Bob Bass. You have created an educational atmosphere that has allowed me and so many others to flourish. You have shown me that being an active, loving parent and a successful engineer are not mutually exclusive. You are a champion for women in the sciences. I am so grateful for every educational opportunity you have afforded me. Thank you.

I would like to thank Kevin Whitener and PGE for their generosity and support. Kevin, you are a utility inspiration. I hope you continue to ask tough questions in the industry and allow us at PSU to investigate them.

My little family, Nathan and Nico, you have shown unwavering support in my relentless pursuit of an education. You bring so much silliness and dimension to my life and work. I love you both with all of my heart.

My Mom and Dad taught me to be unapologetic in the pursuit of my dreams. You told me I could do anything, and for once, I listened to you.

Annie, Michael and Tylor, you are the dream team. Thank you for being my people.

Contents

List of Tables

List of Figures

1 Introduction

Oregon's Renewable Portfolio Standard (RPS) requires utilities to meet 50% of demand with renewable energy by 2040 [6]. According to its 2016 Integrated Resource Plan (IRP), Portland General Electric (PGE) will meet this standard by investing in new, renewable generation sources and closing Oregon's only coal-fired plant [7]. As of 2009, the 550 MW Boardman coal plant was responsible for 15% of PGE's energy production. Switching from coal-fired generation to renewable resources will result in an increase in stochastic supply and a deficit of dispatchable power. This combination of consequences makes it difficult to provide affordable, reliable power.

A possible solution is the strategic use of presently available dispatchable resources. One such resource, is the PGE Dispatchable Standby Generation (DSG) program. PGE has acquired 105 MWs of DSG by partnering with local businesses with back-up generators [8]. This generating capacity counts toward their non-spinning reserves, a resource that is not connected to the system but can be brought on-line within a ten minute window [9]. PGE does not use the distributed generation (DG) resources unless a transfer trip scheme is in place that will cause the generator breaker to trip in the event of a feeder fault, or a modeling study reveals that the load of the feeder is at least three times the generator capacity. This long standing protection ratio is thought to ensure that the feeder load is so large it will trip the generator breaker and prevent unintentional islanding. This approach utilizes already existing GenOnSys control software (K. Whitener, personal communication, July 2, 2015), which eliminates the costly installation of transfer-trip scheme.

Additional dispatchable resources include a 5 MW, lithium-ion Battery Inverter System (BIS) at the Salem Smart Power Center (SSPC), located south of Portland, in Salem, Oregon. The battery is a remnant of the Pacific Northwest Smart Grid Demonstration Project but presently serves as grid back-up [10]. Since the BIS is grid-tied to a 12.4 kilovolt (kV) feeder, it can be utilized for conservation voltage reduction (CVR). This is the process of running a system at its minimum voltage without violating the American National Standard Institute for Electric Power Systems and Equipment (ANSI C84.1) voltage ratings, which states that service voltage must fall between +/- 5% of the system nominal voltage [11]. Both customer and utility equipment are designed to work within these ranges. If the voltage falls outside the ANSI voltage rating the utility can be fined and equipment could suffer damage. To ensure that the load voltages do not drop below the minimum, the head voltage at the substation is generally maintained at approximately 4% greater than the system nominal voltage. However, the head node voltage can be decreased to a minimal value without violating the ANSI standard. Conservation voltage reduction can be achieved with VAr injection; utilizing the support function of the BIS inverter to inject reactive power is injected into the system. These injections increase or decrease the system voltage. When the voltage is decreased at peak times, the power consumption is reduced. Conversely, the system voltage can be increased when the demand is low. Increasing the load accommodates must-run scenarios, e.g. non-dispatchable resources such as hydro, wind or solar generation.

The BIS "creates" reactive power by changing the phase lag or lead of the injection current with respect to the bus voltage. The system's apparent power is affected by the change in reactive power. However, the real power remains the same. This allows the battery to be used to alleviate the load in times of generation shortage or alleviate excess supply in times of overabundant generation.

1.0.1 Problem Statement

PGE requested that the Portland State University (PSU) Power Group investigate the impacts of lowering the DG load to capacity ratio (LCR) on existing distribution feeders. Lowering the LCR increases the availability of the dispatchable resource and prevents the costly installation of a transfer trip protection scheme. Additionally, we examined the real power savings resulting from CVR by VAr-injection on the Oxford Rural feeder. The power savings must be between three and five percent to warrant the installation of an active voltage controller that autonomously performs CVR.

1.0.2 Details about Problem Statement

To investigate the effects of increased dispatchable resource use on PGE's Oxford Rural feeder, we used OpenDSS, an open-source distribution system simulator as our modeling software. We converted PGE's CYME model of the Oxford Rural feeder to instantiation files using several MATLAB programs. A MATLAB control program varies the load to capacity ratio of the OpenDSS feeder model and monitors the generator behavior immediately following a fault. We analyzed the results to determine the ideal load to capacity ratio to prevent unintentional islanding.

Using the same OpenDSS model, we modified the loads, using ZIP coefficients to monitor the changing relationship between power and voltage. To capture the dynamics of the BIS, we collected power and voltage data at the SSPC and modeled the system in Simulink. Lastly, we created an optimization loop in MATLAB to achieve the optimal system voltage that minimizes the power of the load at peak times.

1.0.3 Related Research

1.0.3.1 Distribution Network

The distribution network is the last stage of the standard three phase power delivery process. It begins at the substation, where transmission or sub-transmission voltages are stepped down to a medium voltage, which varies from 4120 V to 33 kV, depending on utility requirements. A substation contains transformers, breakers and protective relaying schemes and may also have voltage support equipment like capacitors and voltage regulations. Several feeders radiate from the substation. Overhead and underground lines branch off of the main feeder, resembling an arterial system. If the feeder is long it may have capacitor banks or voltage regulators at some distance from the substation to prevent voltage from dropping below the ANSI standard. The arterial system terminates at a load, where the voltage is stepped down once more, in preparation for consumer delivery. Figure 1.1 displays a typical distribution network.

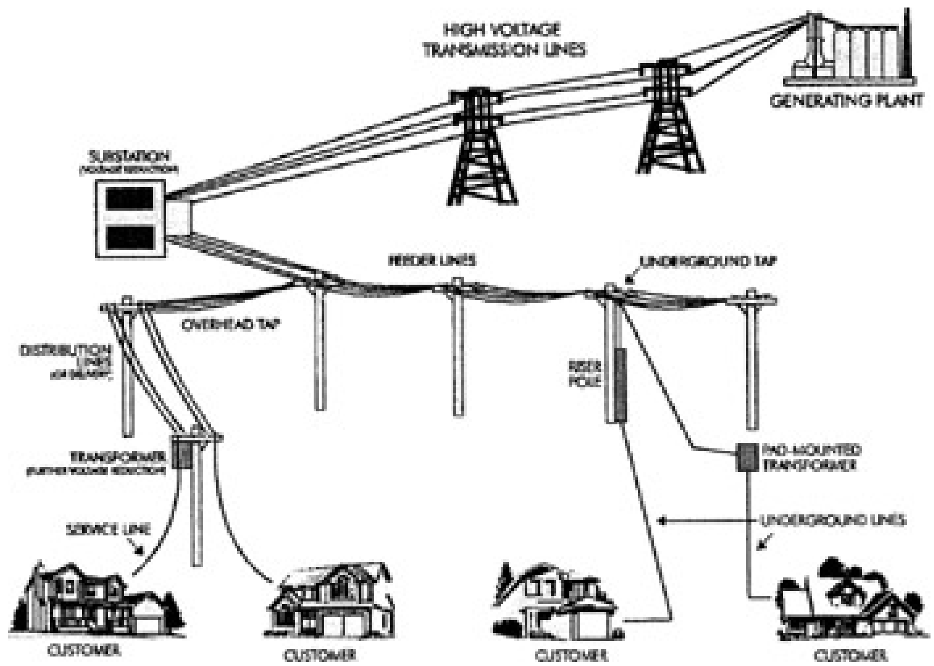

Figure 1.1: Distribution Circuit [2]

1.0.3.2 Distributed Generation

A conventional distribution system has a single generating point and terminates at a load. A system with integrated DG is referred to as an active distribution system due to bi-directional power flow. The many potential benefits of an active distribution system include increased reliability, peak power consumption reduction and reactive power compensation. The primary disadvantage of DG is the concern for islanding, which occurs when the feeder substation encounters a fault and the generator continues to power the load. In such circumstance, the standby generator and the grid become significantly out of phase. If the recloser operates and energizes the feeder, it could result in damages as catastrophic as a complete shearing of the generator shaft [12]. Protection schemes, such as transfer trip, aim to prevent islanding, but the high cost of installation can far outweigh the benefits. As a result, DG adoption by utilities has historically been low [13].

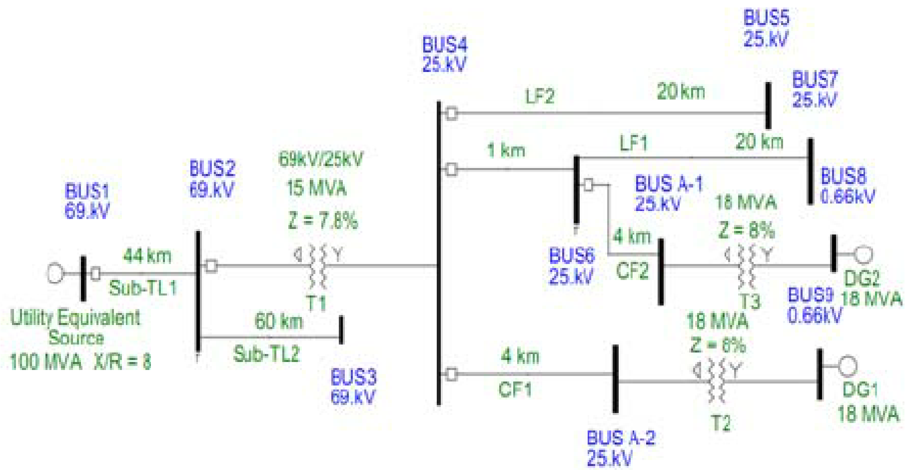

Figure 1.2: One Line diagram with Distributed Generation [2]

1.0.3.3 Transfer Trip

Many utilities require a transfer trip (TT) protective relaying scheme for DG interconnection. The scheme ensures that in the event of a fault or equipment failure at a substation, a lockout relay will trip all of the breakers connected to it. One of the contacts from the lockout relay will initiate a TT signal to a remote breaker at the DG asset [14], which ensures that the DG will not unintentionally island. The communications and equipment cost between $50,000 and $100,000. This high cost can render some projects infeasible. PGE will not bring DG on-line unless it has a TT scheme or if a feeder modeling study reveals that the load is more than three times greater than the generator capacity. This feeder modeling study uses available GenOnSys equipment, which reduces the cost of interconnect projects.

1.0.3.4 Distribution Feeder Modeling Software

Due to the inherent imbalance of the distribution network, the software choices for modeling utility feeders are limited. PGE uses CYME as their distribution analysis software. A customized package is tailored to fit the needs of each utility. Licensed, proprietary software promises security and reliability but limits availability due to the high cost. To increase availability and promote standardization, the Department of Energy (DOE) encouraged the development of an open-source software model platform as part of its grid modernization effort [15]. The Electric Power Research Institute (EPRI) purchased and maintains an open-source distribution system simulator (OpenDSS) for distributed generation integration research in 2008. Among the open-source software available at the time of this project, OpenDSS was able to analyze unbalanced loads and conduct dynamic analysis. Additionally, the program receives commands from scripts, files or an external driver, such as MATLAB. This functionality allows us to create a component database using text files of instantiation statements. Using the OpenDSS script we created the Oxford-Rural feeder model using a series of redirect statements to the component database. MATLAB drove the circuit, creating a custom solution mode and allowing for dynamic modification and recording of the results.

1.0.3.5 CYME to OpenDSS Converter

At the start of the research project, EPRI did not offer a CYME-to-OpenDSS conversion program. EPRI now offers Python 2.7 or Excel VBA converters upon request to utility customers. EPRI does not have a universal converter. One of many converters may be used depending on the version of CYME software the utility uses. In addition, the converter itself must be modified to account for the custom tailoring of the CYME program.

1.0.3.6 Load Modeling

The real power consumption of a load varies with changing voltage depending upon whether it is a constant impedance (Z), constant current (I) or constant power (P) load. Most loads behave like a mixture of the three. ZIP coefficients can be used to create models of loads that predict the power response in response to a changing voltage.

The ZIP coefficients used in this project are found in the 2014 paper, "Experimental Determination of the ZIP Coefficients for Modern Residential, Commercial, and Industrial Loads" by Bokhari, et al. [1]. To obtain the coefficients, they conducted load composition field surveys for residential, commercial and industrial customers. Collected data from voltage variation tests on modern devices was fit to ZIP coefficients using constrained optimization methods. The coefficients were tested against real composite loads to determine the accuracy.

The load composition of a residential customer is predictable. The ZIP coefficients can be stratified based on the power consumption. Commercial customers loads are more complex. For instance, a laundromat load has drastically different behavior than a retail store. The commercial loads are classified first by consumption and then divided into subclasses by the type of business. Industrial customers have a separate class.

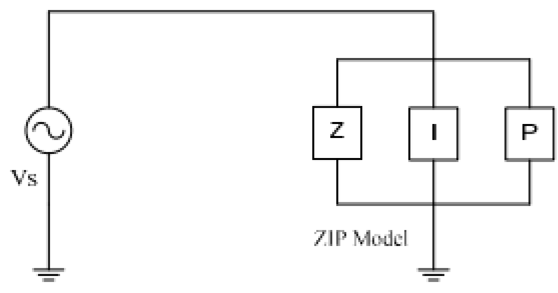

Figure 1.3: ZIP Load Model [1]

1.0.3.7 Conservation Voltage Reduction and VAr control

Conservation voltage reduction generally refers to changes in distribution equipment and operations to reduce line losses and peak loads. It is a reduction in the system voltage to a minimum value that still maintains customer service voltage within +/- 5% of the nominal voltage. Volt/VAr optimization is an advanced form of CVR that includes VAr control. This can reduce both capacity needs and energy use. Traditional CVR equipment includes fixed capacitors, voltage regulators and load tap changers (LTC) as seen in Figure 1.4.

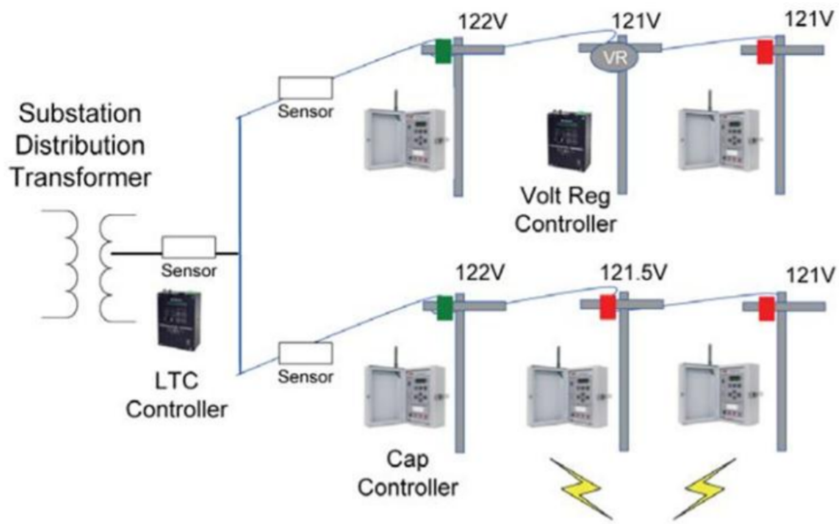

Figure 1.4: Traditional CVR Equipment Diagram [3]

Using the BIS for CVR does not affect the kWh storage. As can be seen in Figure 1.5 changing the power factor angle will change the apparent and the reactive power but the real power remains the same. The battery has tremendous value as a frequency responsive spinning reserve, a generating capacity that can respond within ten seconds [16]. Using the BIS for volt-VAr optimization is considered economically viable because it can function as grid back-up and participate in CVR at the same time.

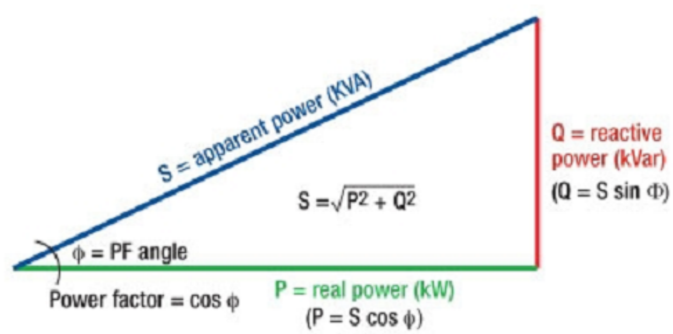

Figure 1.5: Power Triangle[4]

1.0.3.8 Model based Volt-VAr Optimization

The primary purpose of Volt-VAr control is to maintain a voltage within the standard ANSI range. Advanced volt-VAr objectives are peak electrical demand reduction through CVR. Several options for volt-VAr control are available, such as the "Standalone" method that uses voltage regulators and LTC controls as depicted in Figure 1.4.

Our approach uses a MATLAB optimization program that determines the optimal voltage based on the load. A Simulink BIS plant model calculates the reactive power injection to achieve the desired voltage. For investigative purposes we use the Oxford Rural OpenDSS feeder model. Figure 1.6 displays the sequence of events for the model based approach. This approach increases the value of existing assets, such as the BIS. The changes to the system and implementation at the SSPC would be minimal when compared to the "Standalone" method.

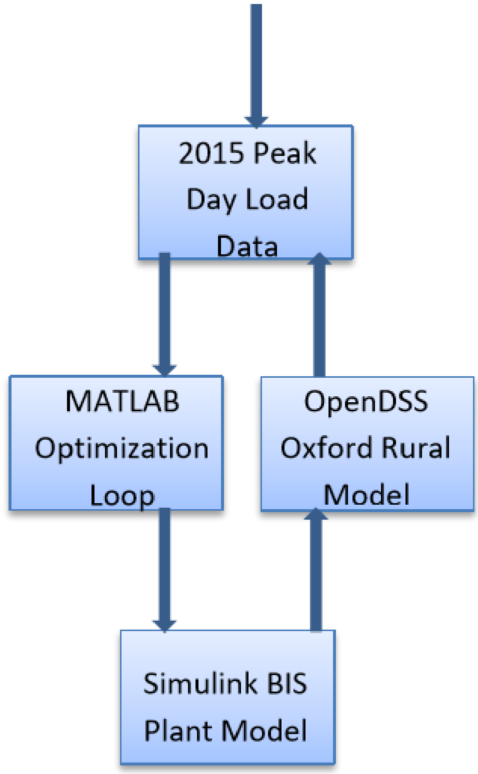

Figure 1.6: Volt/VAr Optimization Approach

1.0.3.9 Lagrange Multipliers

The Lagrange multipliers method is a mathematical tool used in optimization to find the maximum or minimum of an objective function with equality constraints [17].

$$\begin{cases} min_{x,y}\ f(x,y) \\ \text{subject to} \\ g(x,y) = c \end{cases} \tag{1.1}$$

The Lagrange multiplier λ, is multiplied by the constraints and added or subtracted from the objective function to form the Lagrangian.

$$L(x, y, \lambda) = f(x, y) - \lambda(g(x, y) - c) \tag{1.2}$$

We take the partial derivative of the Lagrange with respect to x, y and λ to form a system of equations, which is set to zero and solved. The resulting solution is the minimizer to the objective function subject to the constraints. The variable λ indicates the sensitivity of the objective function to a changing constraint. This is particularly interesting because it indicates the likelihood of an ANSI violation at each node as we move along the feeder.

2.1 Feeder Modeling

To investigate the impacts of lowering the DG load to capacity (LCR) ratio on existing distribution feeders, we built an OpenDSS model of the Oxford Rural feeder. The BIS is grid-tied to the 12.47 kV, Oxford-Rural feeder. Additionally, it has two generator sites that participate in the Distributed Generation Standby Program.

2.1.1 CYMDIST data

The CYMDIST data for the Oxford Rural model is spread across two ACCESS databases and ninety tables. The network database holds information specific to each feeder; for example, the overhead line conductor type and the number of phases. These network parameters are linked to the equipment database that holds manufacturing specifications such as conductor resistance and geometric mean radius (GMR). To develop the algorithms for component instantiation, we created a database map for each device. Figure 2.1 displays the database relationship for the overhead lines.

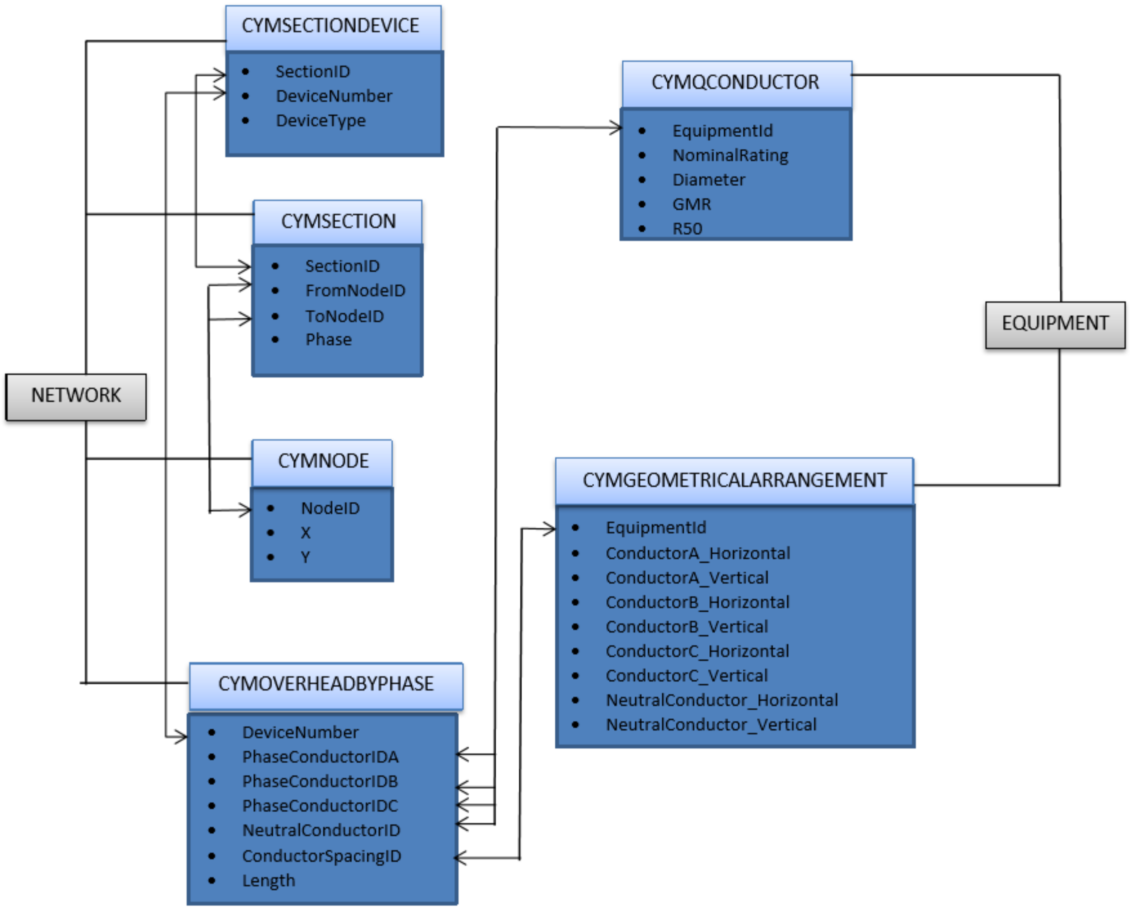

Figure 2.1: CYME Database Relationship Diagram for Overhead Lines

2.1.2 OpenDSS

OpenDSS is a distribution system simulator capable of handling unbalanced systems [5]. It was purchased by EPRI, initially for distributed generation analysis. We chose OpenDSS because it is an open-source program capable of conducting a dynamics study on distributed generation. It can be used alone or in conjunction with an external driver such as MATLAB.

The primary OpenDSS structure is represented by Figure 2.2.

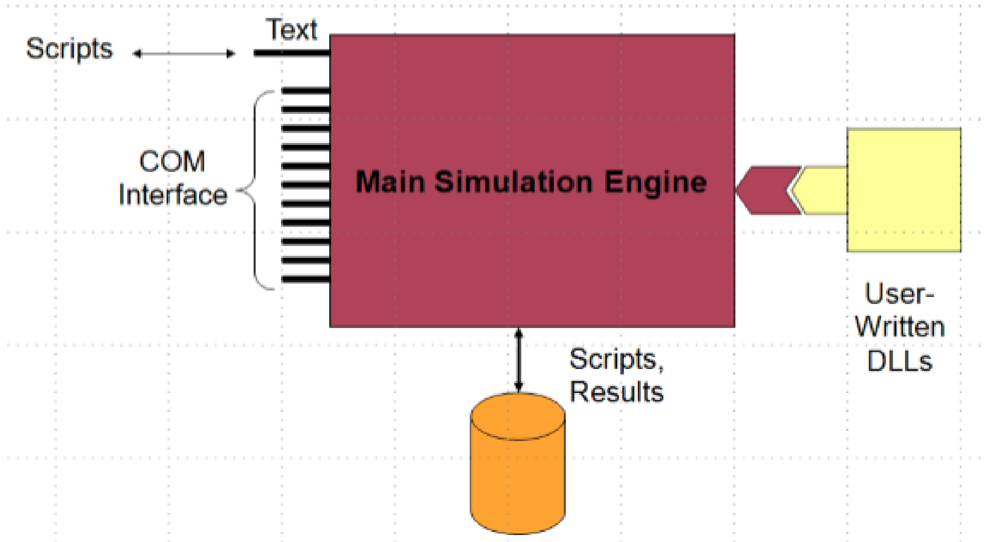

Figure 2.2: OpenDSS Structure [5]

Dynamic solution mode is utilized for the study of the effects of DG on the Oxford Rural feeder. In *dynamic* mode the generator is modeled as a simple swing equation as opposed to a negative load. The generator model used is the standard OpenDSS element with parameter modifications to best simulate standard PGE generators. The mathematical characteristics of the generator model in *dynamic* mode are described as:

Derivative Calculation:

$$\frac{dv}{dt} = \frac{P_{shaft} - P_{term} - Dv}{M} \tag{2.1}$$

$$\frac{d\theta}{dt} = v \tag{2.2}$$

Where,

$$v = shaft\ speed\ relative\ to\ synchronous\ speed$$

$$\theta = shaft,\ or\ power\ angle\ (relative\ to\ synchronous\ reference\ frame)$$

$$P_{term} = terminal\ power\ out$$

$$D = power\ damping\ coefficient$$

$$M = inertia\ coefficient$$

Integration

$$\theta_{n+1} = \theta_n + \frac{\Delta t}{2}\left[\left.\frac{d\theta}{dt}\right|_n + \left.\frac{d\theta}{dt}\right|_{n+1}\right] \tag{2.3}$$

$\Delta t = time\ step\ size$

2.1.3 Oxford Rural Distribution model

OpenDSS functions are carried out from text based commands through the COM interface, text files or execution of a script. We created a component database of the Oxford Rural feeder model in text files that contain instantiation statements. OpenDSS instantiation syntax statements take the form:

$$New\ Object.Name\ Parameter1 = Value1\ Parameter2 = Value2\ ...$$

To develop the algorithms for the component instantiation statements, we compared the database map for each CYME distribution system component to the OpenDSS device parameters. In some cases the conversion was a one-to-one database mapping. In the case of the overhead and underground lines, it required a more detailed conversion process. Figure 2.3 displays the CYME parameters used to develop the OpenDSS overhead line components.

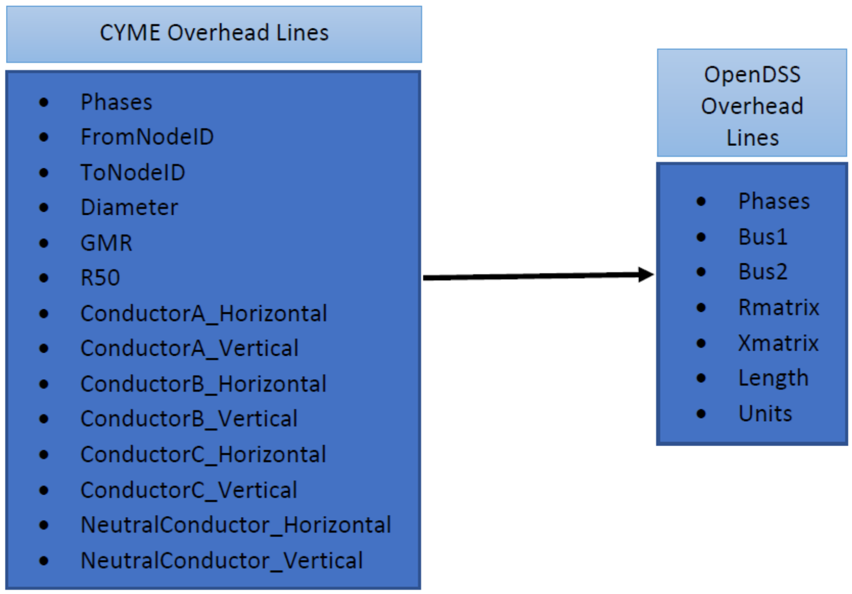

Figure 2.3: CYME to OpenDSS Overhead Line Conversion

Figure 2.4 displays the Oxford Rural Model in OpenDSS.

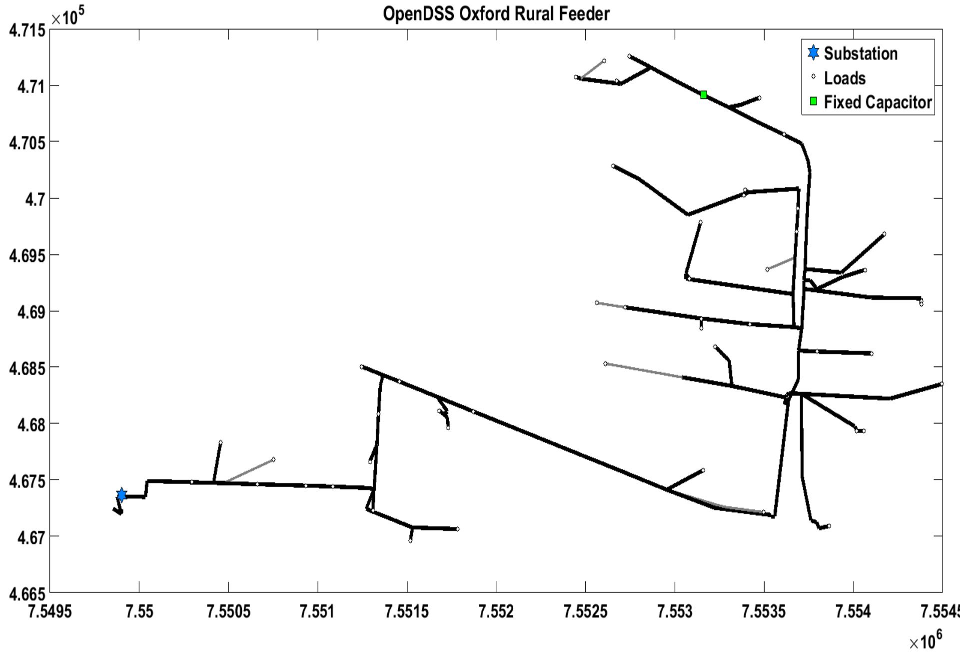

Figure 2.4: OpenDSS Oxford Rural Feeder Model

2.1.3.1 Oxford Rural Feeder Model Validation

The OpenDSS model is not a replica of the utility CYME model, nor is it intended to be. It is a system model of the Oxford Rural feeder, created from utility database information. For this reason, the OpenDSS model is validated against system behavior, not the likeness to its CYME counterpart. We evaluated the model behavior by solving the circuit in five minute increments for 24 hours using the loadshape from the peak day in 2015. At each solution step the program queried the voltage at every load and the current in all feeder lines. A line is considered overloaded if the current exceeds the rated value per the utility database information. A voltage violation is defined by falling outside of the ANSI standard of +/- 5% of the system base voltage. A load voltage violation or line overload indicates the model is an inaccurate representation of the system. The validation program shows that neither the voltage nor currents exceeds their respective limits for the peak 2015 day. Appendix A.0.1 contains the line current and load voltages of the 2015 peak hour.

Figure 2.5 shows a snapshot of the per unit voltage of the model at the maximum load. The line voltage decreases as the distance from the substation increases. This is the expected system behavior and is due to line losses. All per unit voltages fall within the ANSI standard of +/-5% of the system base voltage.

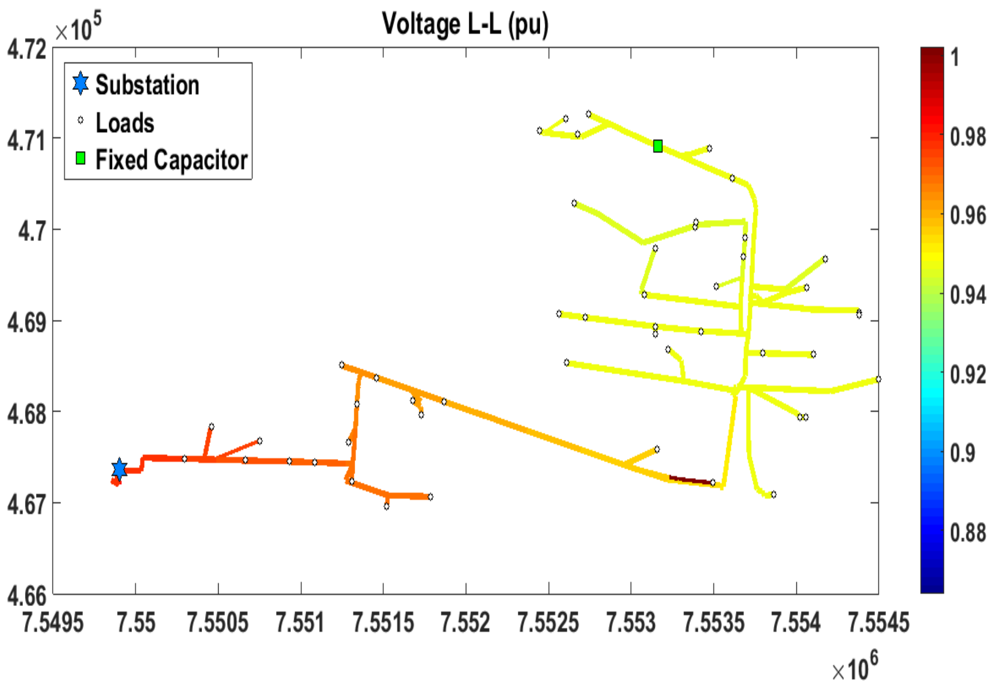

Figure 2.5: OpenDSS Oxford Rural Feeder Per Unit Voltage

Figure 2.6 shows the snapshot solution of the ratio of the system current to the nominal cable rating, at the maximum load. While the loading varies, the lines do not exceed rated current.

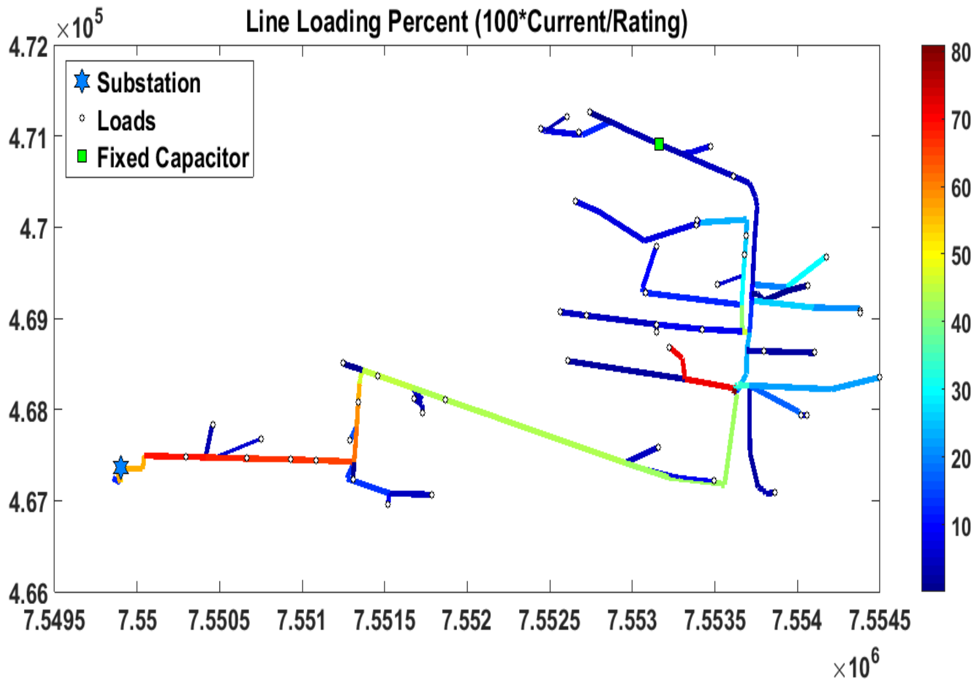

Figure 2.6: OpenDSS Oxford Rural Feeder Line Loading

2.1.4 Loadshape

We created the load multiplier using the PGE Oxford Rural 2015 hourly load data, which we interpolated to create a 1440 point curve. We used one minute resolution to increase the likelihood of meeting the load to capacity ratio. The curve data are from the peak day load, on October 21st, 2015. The loadshape multiplier for the peak day in 2015 is found in Appendix A.0.2.

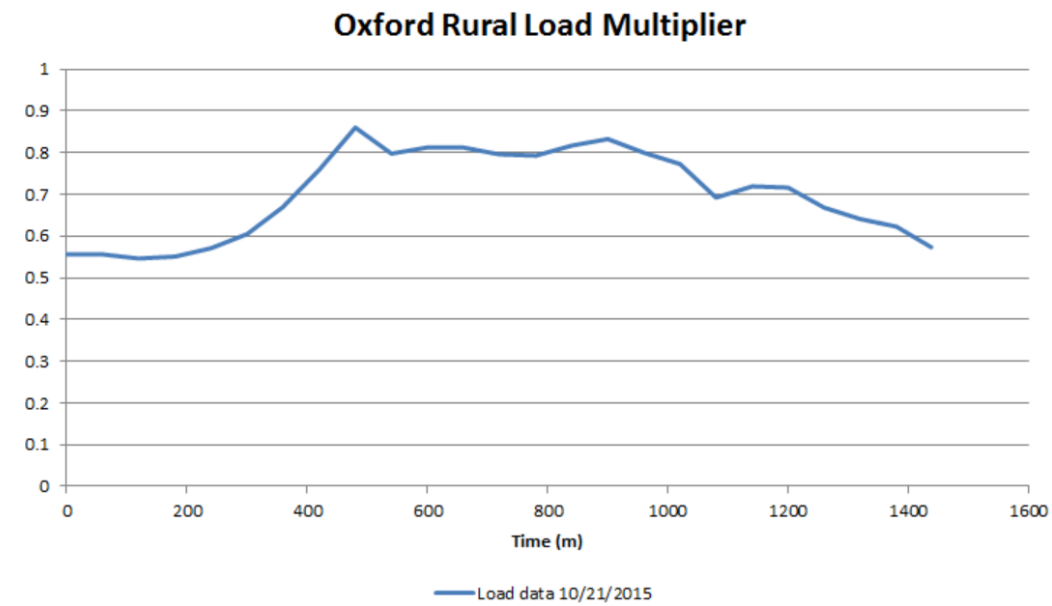

Figure 2.7: Load Multiplier Curve

2.1.5 Relay Curve

A relay object in OpenDSS is represented as a curve constructed from a series of time and current values. We used the "extremely inverse" U4 time-overcurrent relay curve as a basis for all generator relay protection. We used current and time values from the SEL U4 curve with a time dial setting of 1. We varied the instantaneous (50) setting from 2.4 to 2.7 times the rated current.

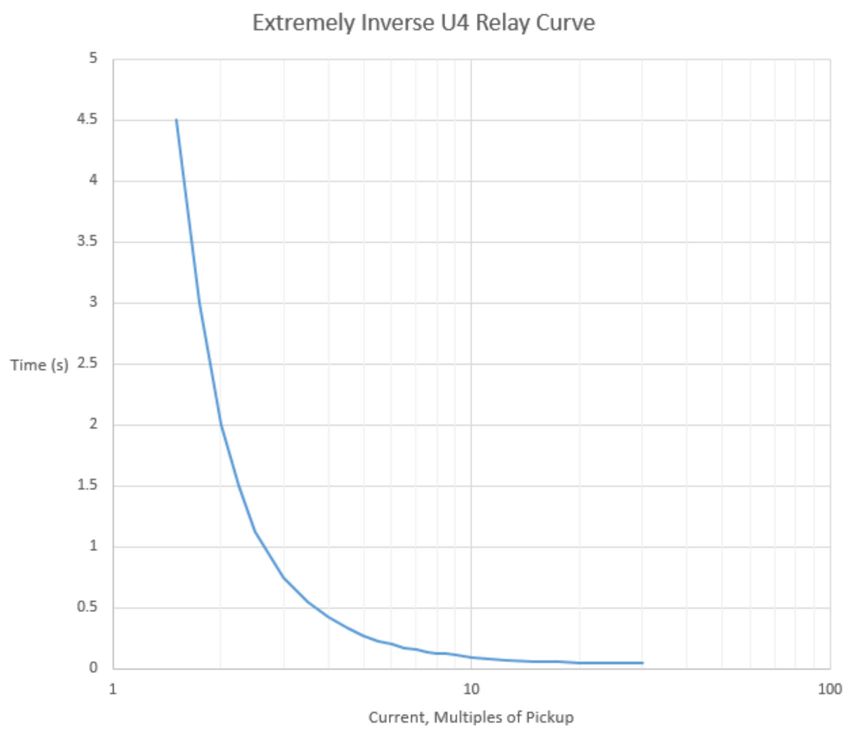

Figure 2.8: Extremely Inverse Curve U4

2.2 MATLAB Dynamic Study

We used MATLAB to drive OpenDSS for the dynamic study of the Oxford Rural feeder. We built a loadshape from the PGE 2015 load data, featuring the peak day: October 21, 2015. A 24 hour day is simulated in time steps of one minute, equaling 1440 data points. The generator model capacity varies from 250 kW to 5000 kW in steps of 250 kW. The load to capacity ratio (LCR) decrements from three to one in steps of 0.10.

At each one minute time step, the load multiplier updates and the circuit solves in *time* mode. The substation current and voltage exports to a CSV file. The control program calculates the apparent power from the exported values. If the generator capacity (kVA) is found to be equal to or greater than the load to capacity ratio multiplied by the feeder load, the generator enables and the circuit solves in *time* mode for one minute, until the generator reaches its rated current. At which point, the solution mode switches to *dynamic* mode and the circuit solves again. Changing the solution mode converts the generator from a negative load to a Thevenin equivalent model governed by a simple swing equation, as referenced in Equation 2.1 and Equation 2.2. The substation relay opens to simulate a fault. The circuit solves every one millisecond for a maximum of five seconds. At each solution step, the program queries the generator relay to see if it has activated. If the generator relay is active, the program records the time, disables the generator and closes the substation relay. Figure 2.9 displays the dynamic study flow chart. The MATLAB control program is found in Appendix A.0.3.

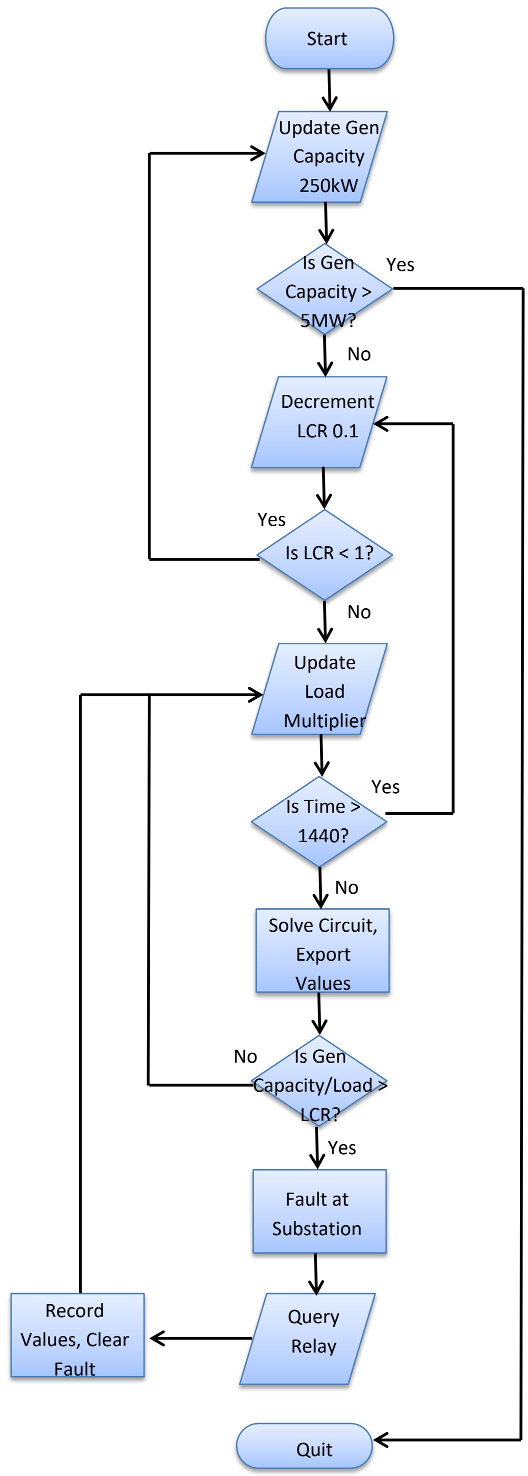

Figure 2.9: Dynamic Study Flow Chart

2.2.1 Resource Availability MATLAB program

The resource availability program monitors and records the substation power for the 2015 peak day to evaluate the potential peak shaving associated with a decreasing load-to-capacity ratio. The generator power capacity is fixed at 2250 kW. The instantaneous (50) element is set at 2.7 x *rated* for the generator located at customer site 1. The program reads the substation power in five minute intervals and calculates the load to capacity ratio. If the LCR is met, the generator enables and activates. The program decrements the LCR from three to one in increments of 0.1. Figure 2.10 displays the resource availability program flow chart.

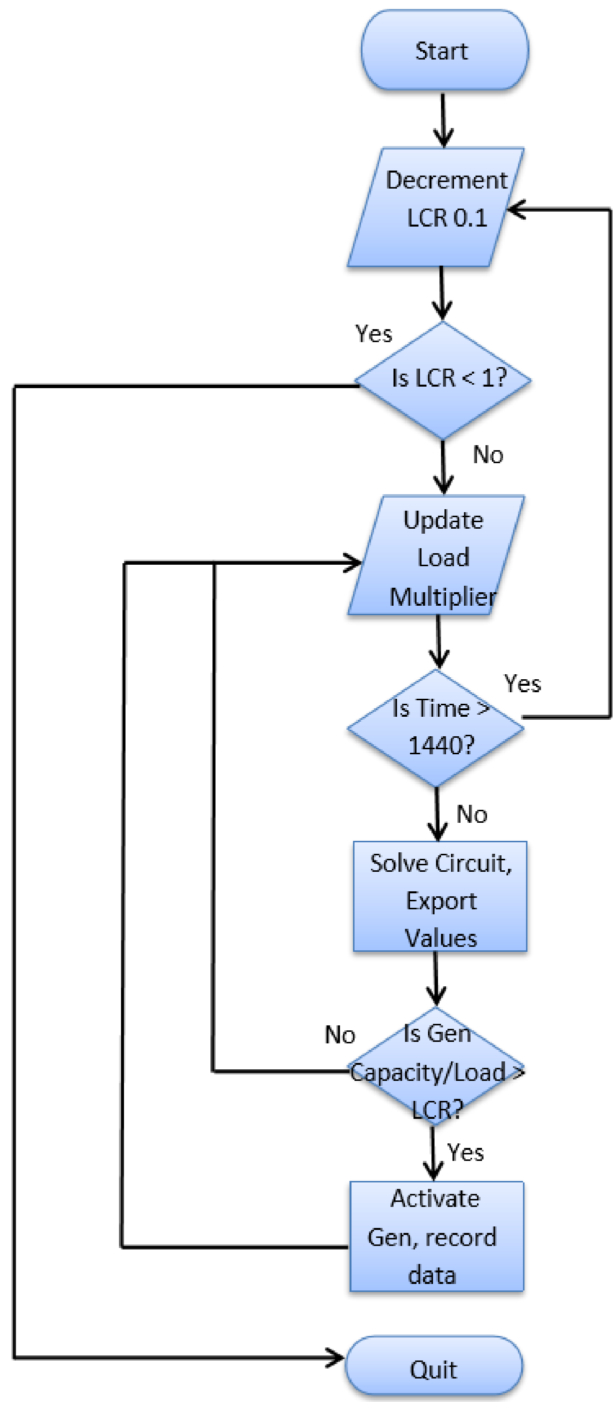

Figure 2.10: Resource Availability Program Flow Chart

2.3 CVR by VAr-injection Study

We examine the real power savings resulting from CVR by VAr-injection on the Oxford Rural feeder to determine the worth of installing an autonomous control program at the substation. The OpenDSS Oxford Rural model loads are modified to account for the varying relationship between power and voltage. We built a battery inverter system feedback model to determine the reactive power injections necessary to raise or lower the voltage. An optimization routine using either the interior-point method or the Lagrange multipliers method determines the optimal voltage. An autonomous control program determines the power savings for the peak 2015 day.

2.3.1 Load Modeling

Each load is assigned a ZIP coefficient value based on the customer class and kilowatt hour (kWh) consumption. In practice, the coefficient that best models a commercial business depends on the type of equipment that is used. Due to privacy concerns, this information is unavailable to us. In an effort to maintain diversity among the commercial classifications, we stratified the commercial loads based on consumption, similar to that of residential loads. Table 3.1 displays how we applied ZIP coefficients based on consumer class and consumption. The ZIP coefficients build upon the work of Bokhari, et al [1].

Customer Class	Low(kW)	High(kW)	Zp	Ip	Pp	Zq	Iq	Pq
Residential 1	0	1948	1.5	-2.31	1.81	7.41	-11.97	5.55
Residential 2	1948	2897	1.57	-2.48	1.91	9.28	-15.29	7.01
Residential 3	2897	3897	1.56	-2.49	1.93	10.1	-16.75	7.65
Residential 4	3897	5239	1.31	-1.94	1.63	9.2	-15.27	7.07
Residential 5	5239	7741	0.96	-1.17	1.21	6.28	-10.16	4.88
Residential 6	7741	20000	1.18	-1.64	1.47	8.29	-13.67	6.38
Small Commercial 1	0	1000	0.27	-0.33	1.06	5.48	-9.7	5.22
Small Commercial 2	1000	5000	0.69	0.04	0.27	1.82	-2.24	1.43
Small Commercial 3	5000	15000	0.77	-0.84	1.07	8.09	-13.65	6.56
Small Commercial 4	15000	50000	0.55	0.24	0.21	0.55	-0.09	0.54
Large Commercial 1	50000	150000	0.4	-0.41	1.01	4.43	-7.98	4.56
Large Commercial 2	150000	250000	0.76	-0.52	0.76	6.92	-11.75	5.83
Industrial	250000	500000	1.21	-1.61	1.41	4.35	-7.08	3.72

Table 2.1: ZIP Coefficient Parameters [1]

2.3.1.1 Model Sensitivity to ZIP Coefficient Parameters

We analyzed the sensitivity of the model voltage and currents to changes in the ZIP parameters by varying the coefficients +/- 10% and solving the circuit for the peak day in 2015. Table 2.2 displays the average, minimum and maximum change in line current. The relationship shows a proportionality constant of nearly -1 for both the 10% increase and decrease. The current did not exceed the rated value in any of the feeder model lines as a result of the ZIP coefficient changes.

Table 2.2: ZIP Coefficient Sensitivity Analysis for Line Currents

Percent Change	-10%	+10%
$\Delta_{Average}$	9.02	-9.23
$\Delta_{Maximum}$	10.76	-10.8
$\Delta_{Minimum}$	-0.79	0.71

Table 2.3 shows the average, minimum and maximum change in per unit load voltages. The voltage is significantly less sensitive to changes in the ZIP parameters than the current.

The system load voltages did not exceed +/-5% of the system base voltage as the load modeling parameters were varied.

Table 2.3: ZIP Coefficient Sensitivity Analysis for Load Voltages

Percent Change	**-10%**	**+10%**
$\Delta_{Average}$	-0.28	0.28
$\Delta_{Maximum}$	-0.04	0.87
$\Delta_{Minimum}$	-0.86	0.04

2.3.2 Plant Model

We captured the relationship between reactive power and feeder voltage on March 13, 2016 at the Salem Smart Power Center (SSPC). The inverter system injected reactive power at the feeder head node and we recorded the resulting Oxford Rural feeder voltage. Figure 2.11 and Figure 2.12 display the interpolated data. The largest reactive power step of 4000 kVAr results in a voltage change of 2.3%.

Injection (kVAr)	**Volt Min (V)**	**Volt Max (V)**	**Volt Change (V)**	**Volt Change (%)**
100	12813	12842	29	0.23
500	12762	12832	70	0.56
1000	12669	12761	92	0.73
2000	12636	12802	166	1.31
4000	12651	12943	292	2.31

Table 2.4: Voltage Change as a Function of kVAr Injection

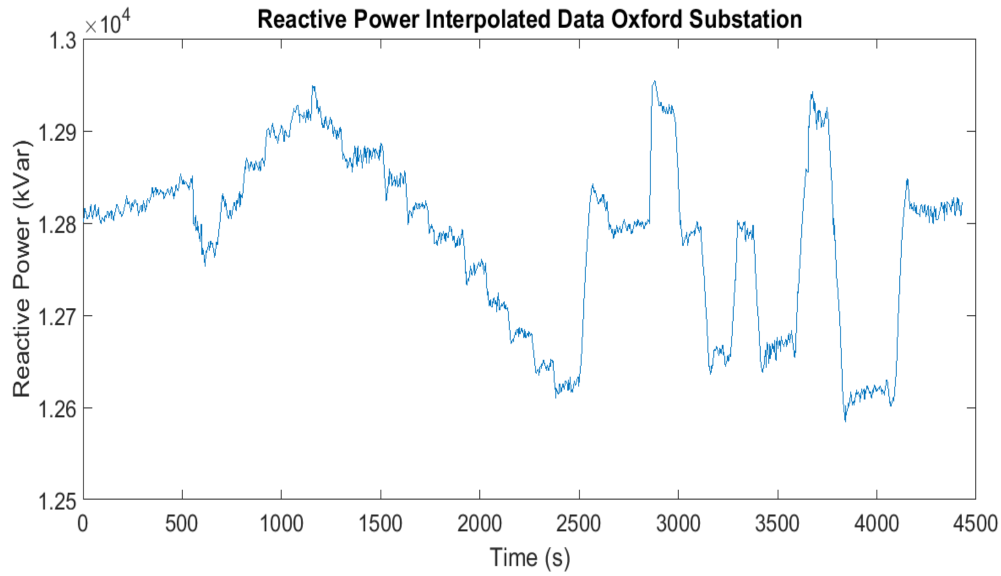

Figure 2.11: Battery Inverter System Reactive Power Injections

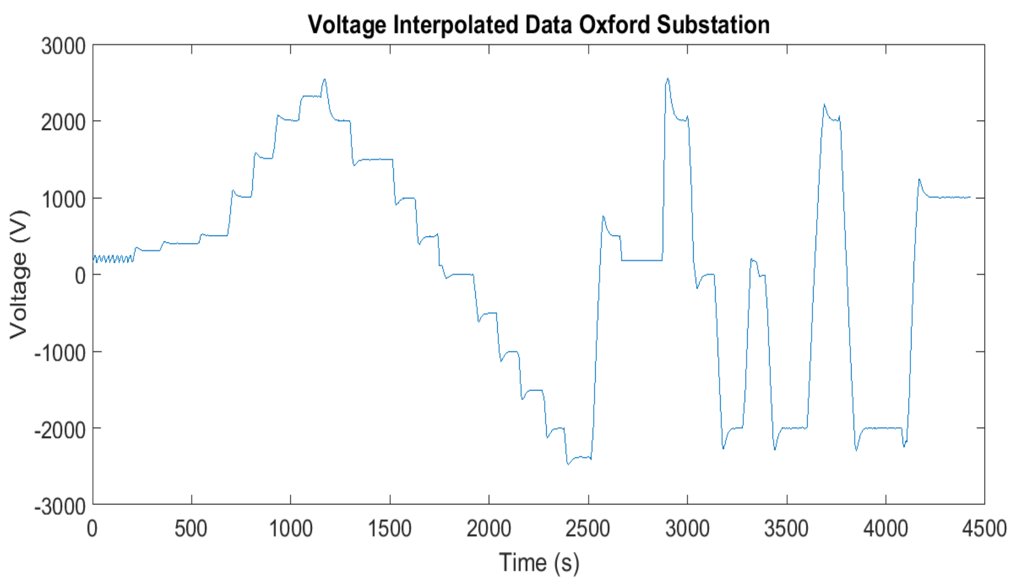

Figure 2.12: Oxford Rural Feeder Voltage Response Data

We used the MATLAB System Identification toolbox to create the transfer functions based on the interpolated data. The system dynamics depend upon the amount of kVAr injected. The larger the injection, the larger the overshoot and the longer the settling time.

To retain the system dynamics, we created five different feedback system models. Table 2.4 summarizes the selected data used to build the plant models in Simulink. The feedback models were verified against the collected data. We confirmed the system stability by plotting the frequency response and confirming the real eigenvalues were positive numbers. Figures 2.13 through Figure 2.17 display the frequency response of each plant model.

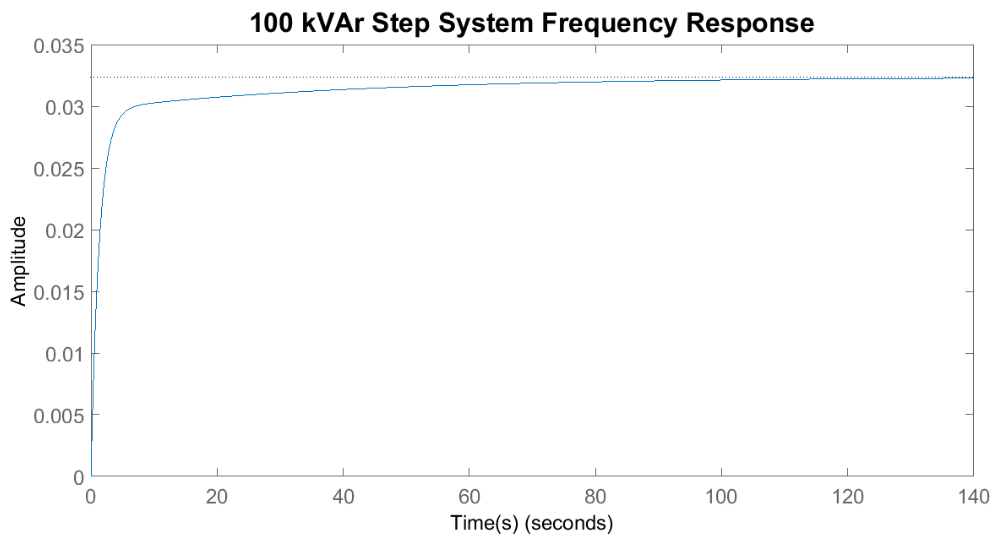

Figure 2.13: Plant Model Response - 100 kVAr

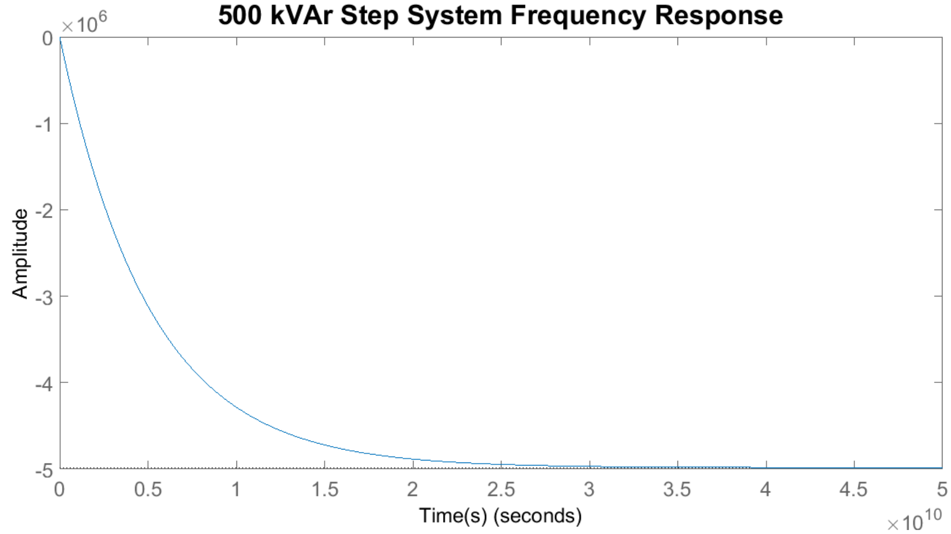

Figure 2.14: Plant Model Response - 500 kVAr

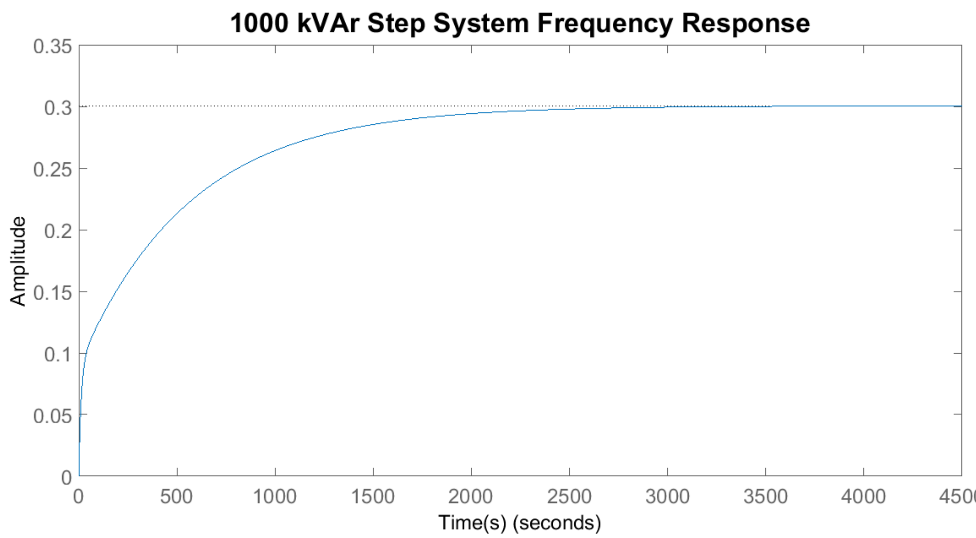

Figure 2.15: Plant Model Response - 1000 kVAr

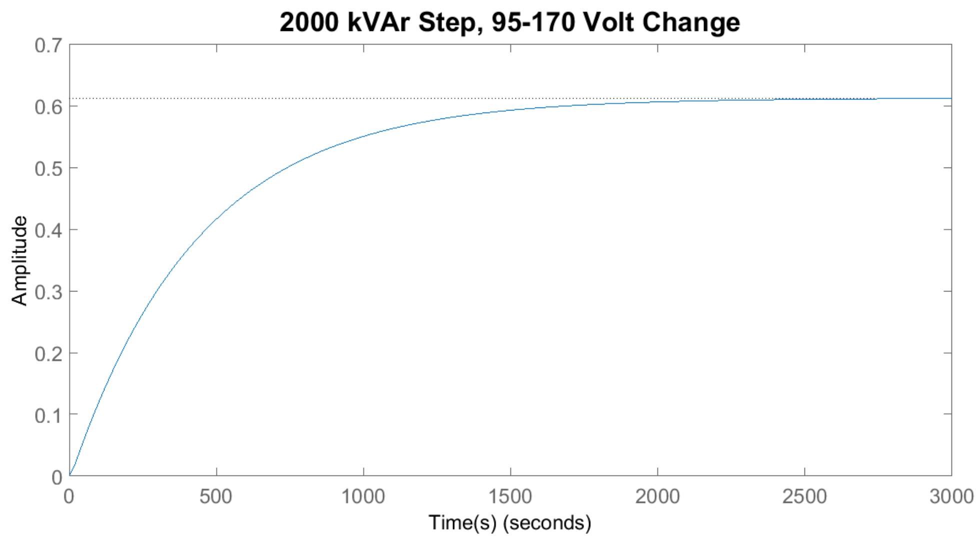

Figure 2.16: Plant Model Response - 2000 kVAr

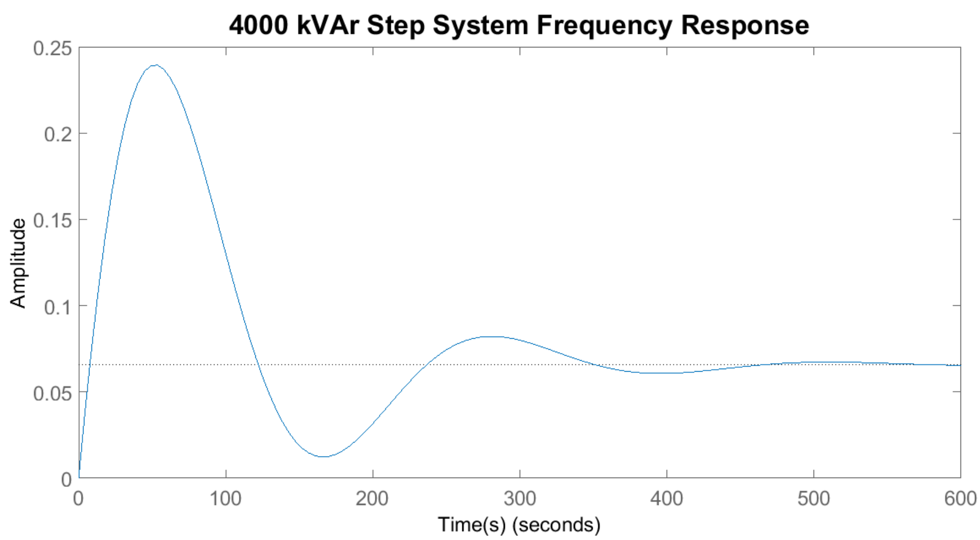

Figure 2.17: Plant Model Response - 4000 kVAr

The BIS is grid-tied to the Oxford Rural feeder. Therefore, the system response data is influenced by the connection to the load. We created a multiplier to ensure continuity between the plant model response and that of the OpenDSS Oxford Rural feeder. Figure 3.9 displays the relationship between reactive power and voltage for the two systems.

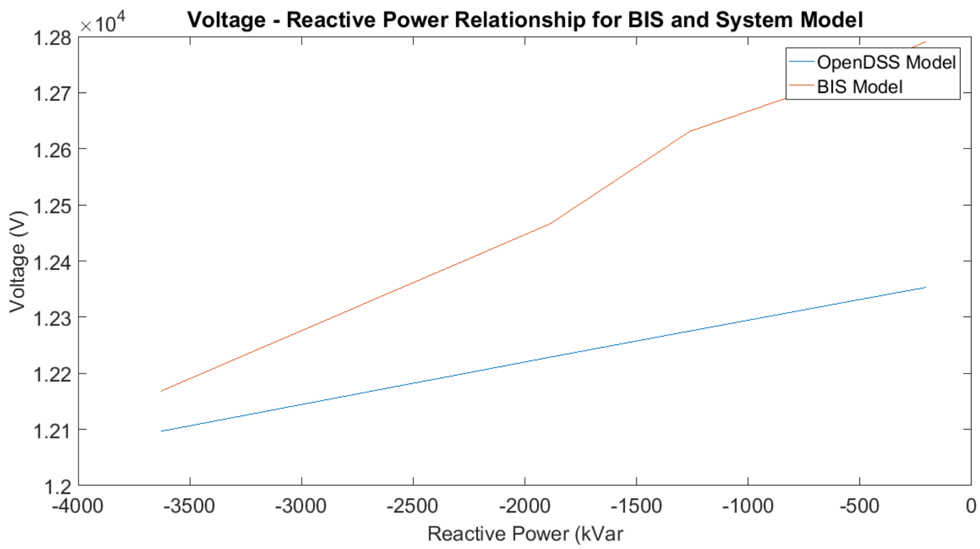

Figure 2.18: Voltage and Reactive Power Relationship for the BIS and OpenDSS Model

The MATLAB polyfit function determined the linear relationship for each respective system. Figure 2.19 shows the block diagram relationship for the voltage and reactive power systems.

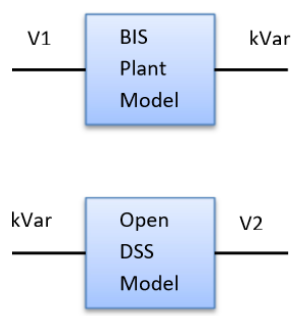

Figure 2.19: System Block Diagrams

The BIS and Oxford Rural feeder system can be described by Equation 2.4 :

$$V_1 X = kVAr \tag{2.4}$$

Where,

$$V_1 = [v_{opt}, 1]$$

$$X = [5.40, -6.93e+04]^T$$

$$v_{opt} = \text{Optimized Voltage}$$

$$kVAr = \text{BIS Plant Model Output}$$

The OpenDSS model of the BIS and the Oxford Rural feeder can be described by Equation 2.5:

$$kVArY = V_2 \tag{2.5}$$

Where,

$$V_2 = [v_{out}, 1]$$

$$Y = [0.075, 1.24e+05]^T$$

$$v_{out} = \text{System Voltage}$$

The output voltage of the OpenDSS model V_2, as a function of the optimized voltage V_1, can be described by Equation 2.6.

$$V_2 = \frac{\frac{(V1-Y(1,2)}{Y(1,1)} - X(1,2)}{X(1,1)} \tag{2.6}$$

The volt-VAr optimization program transforms the output voltage of the optimization routine using the relationship between V_1 and V_2. The revised voltage acts as an input to the BIS feedback model. When the resulting reactive power is injected into the OpenDSS model it creates a system voltage equivalent to the original optimized value.

2.3.3 Cost Equation Development

We developed the relationship between power and voltage by sweeping the Oxford Rural OpenDSS model from the minimum to the maximum allowable ANSI voltage. At each solution step the circuit solves, the head voltage is decremented and the power and voltage exports to a CSV file. We used the curve fitting tool in MATLAB to determine the coefficients of the linear relationship for the head feeder node and all system loads. Figure 2.20 displays the voltage and power relationship at the feeder node for the OpenDSS model. Equation 2.7 describes the linear relationship.

$$P(v) = 0.198v + 1603.840 \tag{2.7}$$

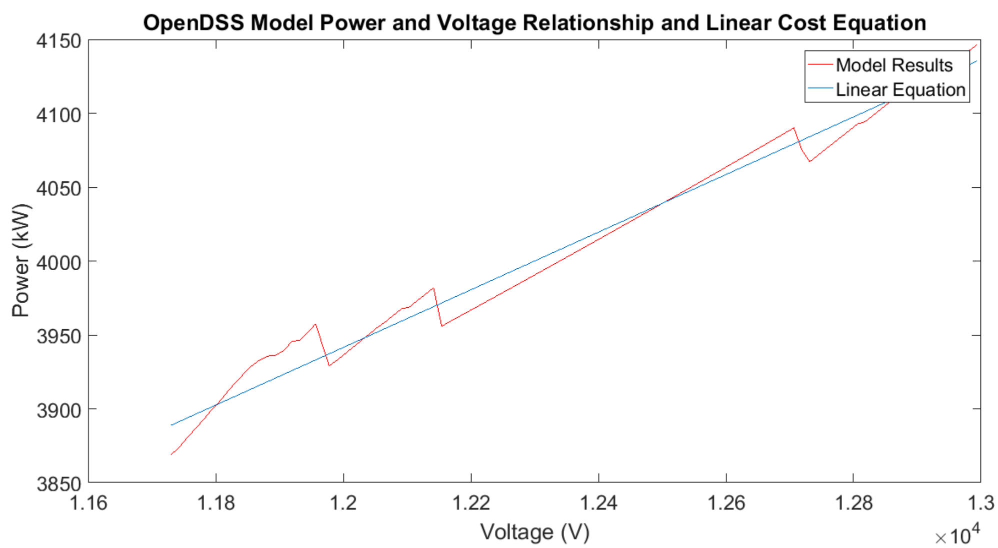

Figure 2.20: OpenDSS Model Power and Voltage and Cost Equation

2.3.4 MATLAB Optimization Routine

The CVR by VAr-injection study is a constrained optimization problem. The routine must minimize the system power while maintaining a voltage at each load which falls within the

ANSI standard range.

$$\begin{cases} min_v \; P(v) \\ \text{subject to} \\ 0.95(V_{base}) <= v <= 1.05(V_{base}) \\ 0.95(V_{base}) <= V <= 1.05(V_{base}) \end{cases} \tag{2.8}$$

Where,

$$P(v) = 0.198v + 1603.840$$

$$v = substation\ voltage\ , v \epsilon \Re^{1}$$

$$V = load\ voltages\ , V \epsilon \Re^{137x1}$$

$$V_{base} = 12470$$

Two different methods are used to obtain the ideal system voltage, the interior-point algorithm from the built-in MATLAB function ‘fmincon’ and the Lagrange mutlipliers method. Since the constrained, optimization problem is relatively simple, using the built-in MATLAB function would generally suffice. However, the information we receive from the Lagrange Multipliers method informs us of any nodes which are particularly sensitive to the voltage constraints, indicating that the system may benefit from monitoring or additional equipment to ensure the voltage does not drop below the ANSI standards.

2.3.5 Fmincon

The built-in MATLAB function, fmincon finds the minimum of an objective function within a defined upper and lower bound, subject to the constraints, $Ax = b$. The upper and lower bounds are defined by the ANSI voltage standards. We create the constraint system of equations by determining a linear relationship between the load voltage at each node as a function of the substation voltage. Each load node voltage equation is set to less than the maximum allowable voltage or greater than the minimum voltage. The equation is then rearranged to comply with MATLAB standards. The remaining set of equations form the constraint matrix A and vector b.

$$V_i(v) = c_i v + d_i$$

$$V_i(v) \geq 0.95(V_{base})$$

$$V_i(v) \leq 1.05(V_{base})$$

Becomes,

$$\begin{aligned} A(i,1)x &\leq b(i,1) \\ A(i+1,1)x &\leq b(i+1,1) \end{aligned} \tag{2.9}$$

Where,

$$A(i,1) = c_i$$

$$A(i+1,1) = -c_i$$

$$b(i,1) = 1.05(V_{base}) - d_i$$

$$b(i+1,1) = -0.95(V_{base}) + d_i$$

The MATLAB code used to create the linear constraint system is found in Appendix B.0.1 Fmincon.

2.3.6 Lagrange Multipliers

The method of Lagrange Multipliers is used to convert a constrained optimization problem into an unconstrained optimization problem. Slack variables convert the inequality constraints to equality constraints. The Lagrangian function, L(x) is formed by subtracting the constraints from the objective function. The system of equations is formed by taking the gradient with respect to each variable, including the Lagrange Multipliers. The system of equations is set to zero and the resulting vector is the minimized solution. The values of λ, the Lagrange multipliers, indicate the sensitivity of the constraints to the objective function.

$$L(v, \lambda_1, \lambda_2, \lambda_3, \lambda_4, s, t, S, T) = P(v) - \lambda_1(c1) - \lambda_2(c2) - \lambda_3(c3) - \lambda_4(c4)$$

Where,

$$P(v) = 0.198v + 1603.840$$

$$c1 = -v + 0.95(V_{base}) + s^2$$

$$c2 = v - 1.05(V_{base}) + t^2$$

$$c3 = -V + 0.95(V_{base}) + S.^2$$

$$c4 = V - 1.05(V_{base}) + T.^2$$

And,

$$\nabla_{L(v,\lambda_1,\lambda_2,\lambda_3,\lambda_4,s,t,S,T)v} = 0.198 + \lambda_1 - \lambda_2 + \lambda_3 \frac{\partial V}{\partial v} - \lambda_4 \frac{\partial V}{\partial v}$$

$$\nabla_{L(v,\lambda_1,\lambda_2,\lambda_3,\lambda_4,s,t,S,T)\lambda_1,\lambda_2,\lambda_3,\lambda_4} = \begin{vmatrix} v - 0.95(V_{base}) - s^2 \\ v + 1.05(V_{base}) - t^2 \\ V - 0.95(V_{base}) - S.^2 \\ -V + 1.05(V_{base}) - T.^2 \end{vmatrix}$$

$$\nabla_{L(v,\lambda_1,\lambda_2,\lambda_3,\lambda_4,s,t,S,T)s,t,S,T} = \begin{vmatrix} -\lambda_1(2s) \\ -\lambda_2(2t) \\ -\lambda_3(2S) \\ -\lambda_4(2T) \end{vmatrix}$$

The MATLAB code used to create the linear constraint system is found in Appendix B.0.2 Lagrange Multipliers.

2.3.7 Model based Volt-VAr Optimization Algorithm

The autonomous control program is capable of CVR and feeder smoothing. The CVR program calculates the voltage that minimizes the system power and determines the corresponding kVAr value that will achieve the optimal voltage. The feeder smoothing profile option minimizes or maximizes the system power depending upon the load consumption and renewable energy generation. If the renewable energy generation is lower than the consumption, the routine will minimize the voltage and decrease the system power. If the generation is greater than the consumption, the routine calculates the maximum allowable voltage. Once the target voltage set point is found the routine retrieves the reactive power necessary to achieve the optimized voltage. Figure 2.21 displays autonomous control program flow chart. The optimization routine MATLAB code is found in Appendix B.0.3.

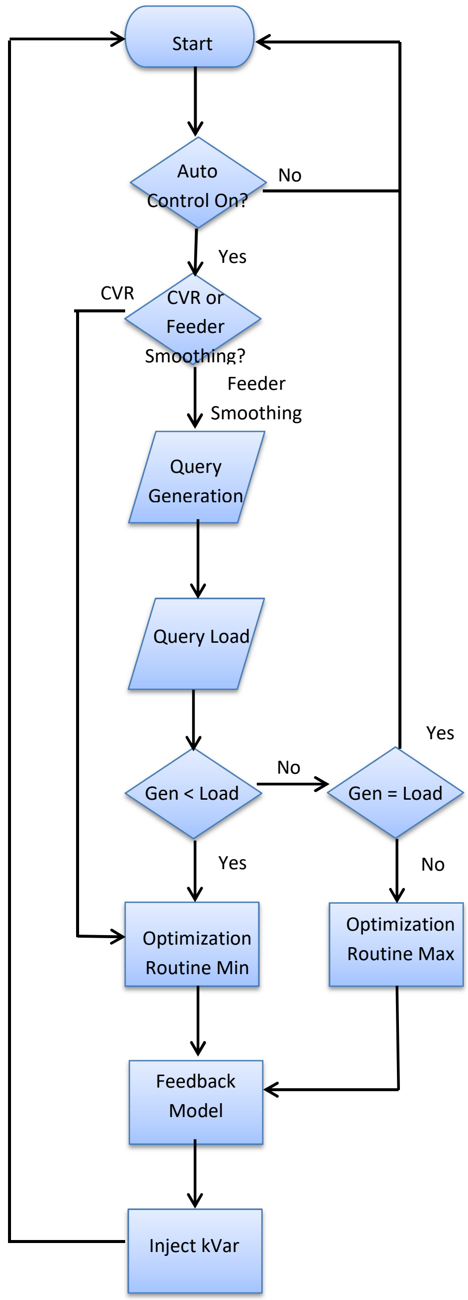

Figure 2.21: Autonomous Control Program Flow Chart

3.0.1 MATLAB Dynamic Study

The results of the MATLAB control program for both DG customer sites show that the instantaneous (50) setting of the generator protection relay dictates the maximum load to capacity ratio (LCR).

The power consumption of the feeder is measured at the substation. When the DG is not active, the power measured at the substation, P_{Sub}, and the load power, P_{Load}, are equivalent.

$$P_{Sub} = P_{Load} - P_{Gen} \tag{3.1}$$

$$LCR = \frac{P_{Sub}}{P_{Gen_{Rated}}} \tag{3.2}$$

When the LCR is defined, as in Equation 3.2, the DG is made available as a resource. If the DG is brought on-line, the power measured at the substation reduces by an amount equal to the generator capacity. Once the generator turns on, the ratio of the load to the generator capacity, P_{Gen} may be expressed in terms of P_{Load} and $P_{GenRated}$ as:

$$
\begin{aligned}
LCR &= \frac{P_{Sub}}{P_{Gen_{Rated}}} \\
&= \frac{P_{Load} - P_{Gen_{Rated}}}{P_{Gen_{Rated}}} \\
&= \frac{P_{Load}}{P_{Gen_{Rated}}} - 1
\end{aligned} \tag{3.3}
$$

The generator protection settings are determined by the ratio of the load power to the generator power. Once the ratio of the load to the generator exceeds the instantaneous (50) element setting, GENOC50, the relay will activate.

$$
GENOC50 = \frac{P_{Load_{Max}}}{P_{Gen_{Rated}}} \tag{3.4}
$$

Substituting Equation 3.4 into Equation 3.3 reveals the minimum ratio that ensures the instantaneous (50) protection element will activate as expected is:

$$
\begin{aligned}
LCR_{Min} &= \frac{P_{Load_{Max}}}{P_{Gen_{Rated}}} - 1 \\
&= GENOC50 - 1
\end{aligned} \tag{3.5}
$$

Additionally, the maximum DG capacity on a feeder that guarantees the instantaneous (50) element will operate is:

$$
P_{Gen_{Max}} = \frac{P_{Load_{Max}}}{LCR_{Min} + 1} \tag{3.6}
$$

3.0.1.1 DG Customer Site 1

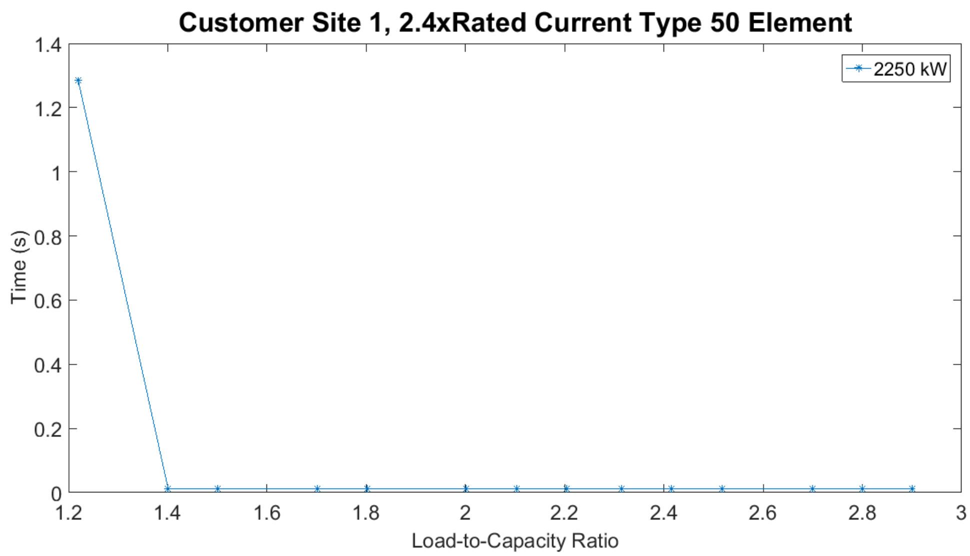

Figure 3.1: DG Site 1, 50/51 Element Operating Times, 2.4xRated

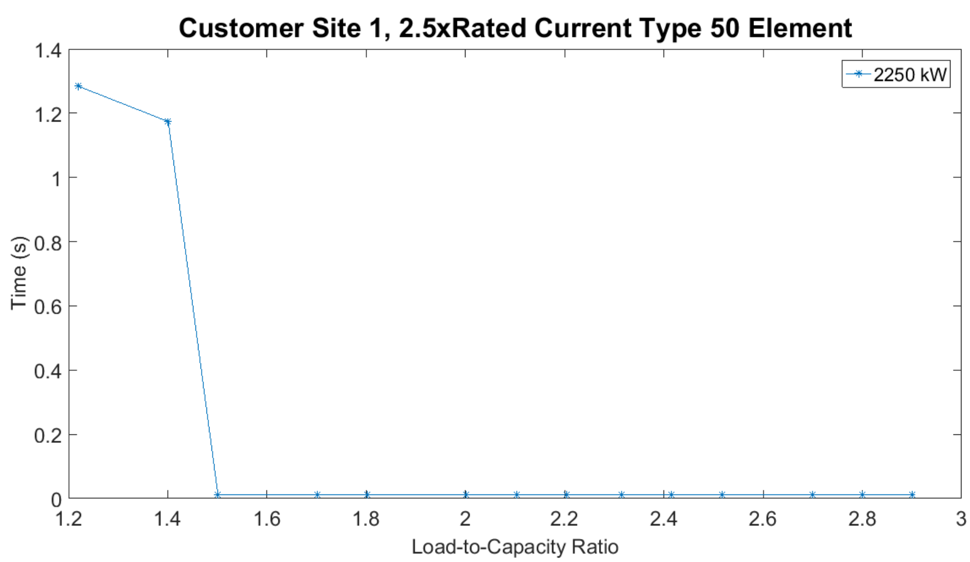

Figure 3.2: DG Site 1, 50/51 Element Operating Times, 2.5xRated

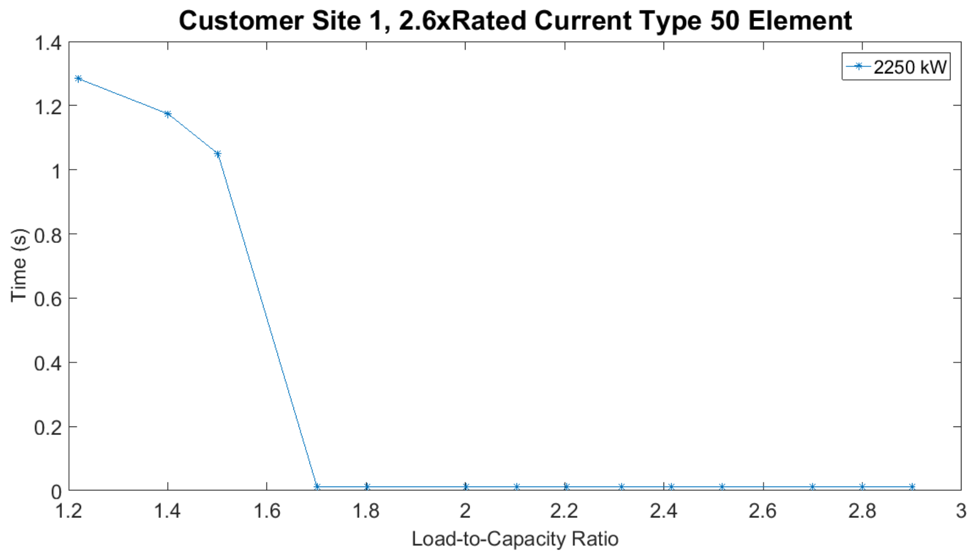

Figure 3.3: DG Site 1, 50/51 Element Operating Times, 2.6xRated

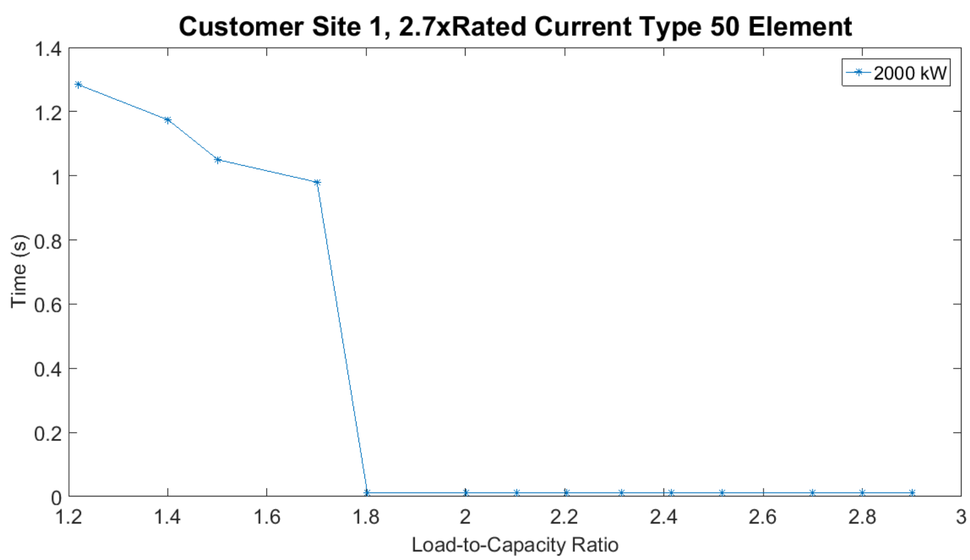

Figure 3.4: DG Site 1, 50/51 Element Operating Times, 2.7xRated

Figure 3.1 through Figure 3.4 display the generator 50/51 protection element operating times. The instantaneous (50) element activates whenever the generator capacity exceeds the maximum as calculated in Equation 3.6 and the ratio drops below the minimum as calculated in Equation 3.5. Thus, the relay operates as expected, until the ratio falls below:

$$LCR < GENOC50 - 1$$

In the case of the Figure 2.8, it appears as though the results deviate from the trend. However, the program is written to search for a target window surrounding the load to capacity ratio. Throughout the 24 hour period the load did not reach 1.6 +/- 0.09 times the generator capacity. While there is not a data point for this ratio, the relay behaved as though governed by the calculations above. The instantaneous (50) element did not activate after the calculated ratio fell below the minimum.

3.0.1.2 DG Customer Site 2

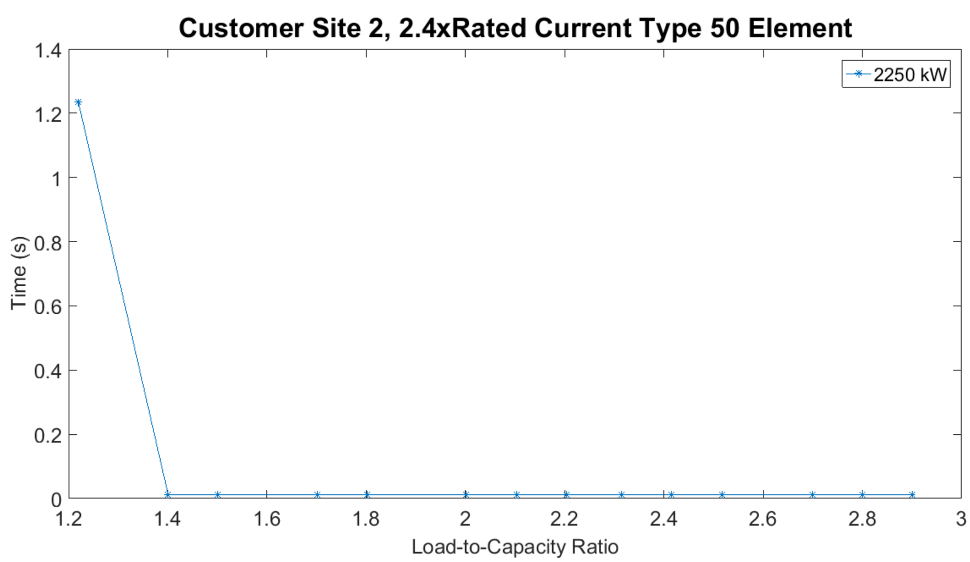

Figure 3.5: DG Site 2, 50/51 Element Operating Times, 2.4xRated

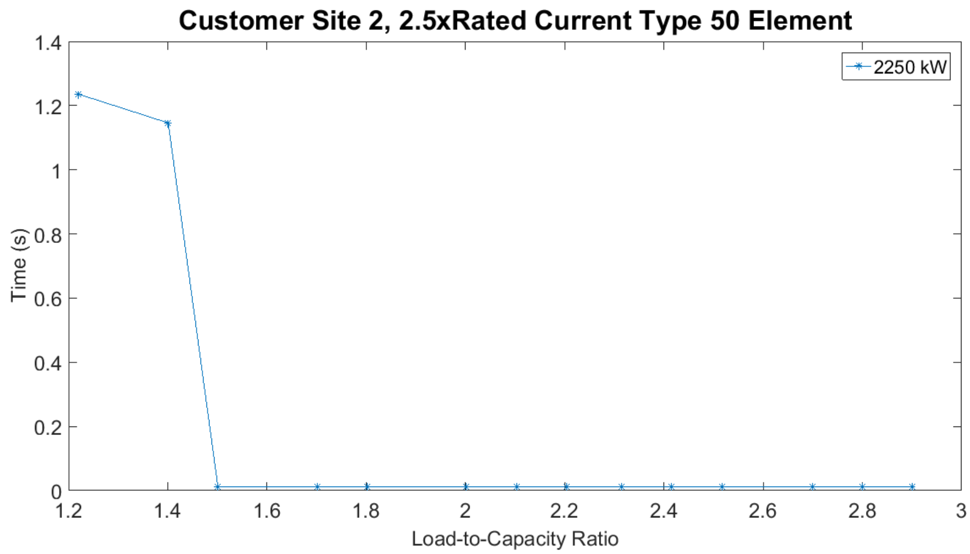

Figure 3.6: DG Site 2, 50/51 Element Operating Times, 2.5xRated

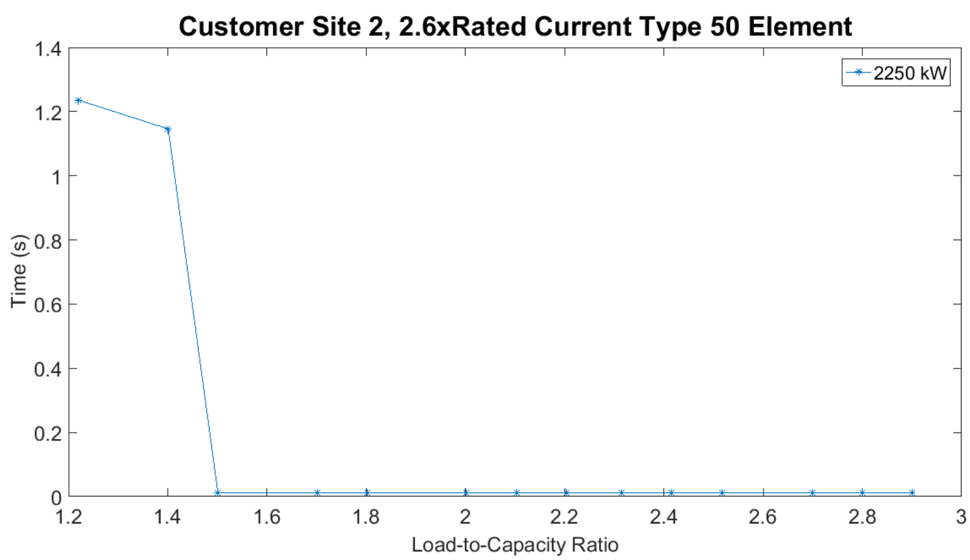

Figure 3.7: DG Site 2, 50/51 Element Operating Times, 2.6xRated

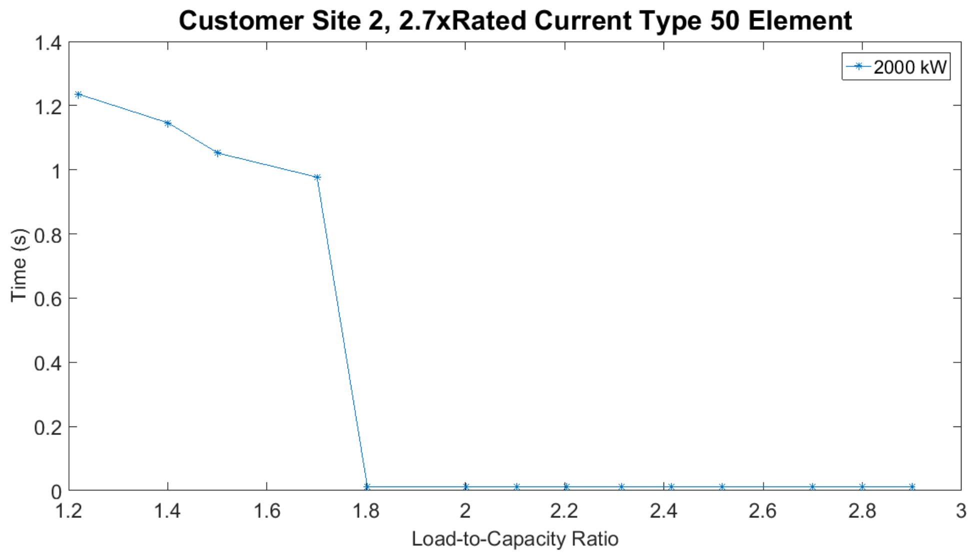

Figure 3.8: DG Site 2, 50/51 Element Operating Times, 2.7xRated

Results for customer site 2 are very similar to customer site 1. The non-instantaneous (51) element times differ, but the instantaneous element reacts as governed by equations 3.5 and 3.6.

3.0.1.3 Resource Availability and Load to Capacity Ratio

Figure 3.9 displays the results of the MATLAB resource availability program for the 2015 peak day. The present load to capacity ratio is only met during peak loads.

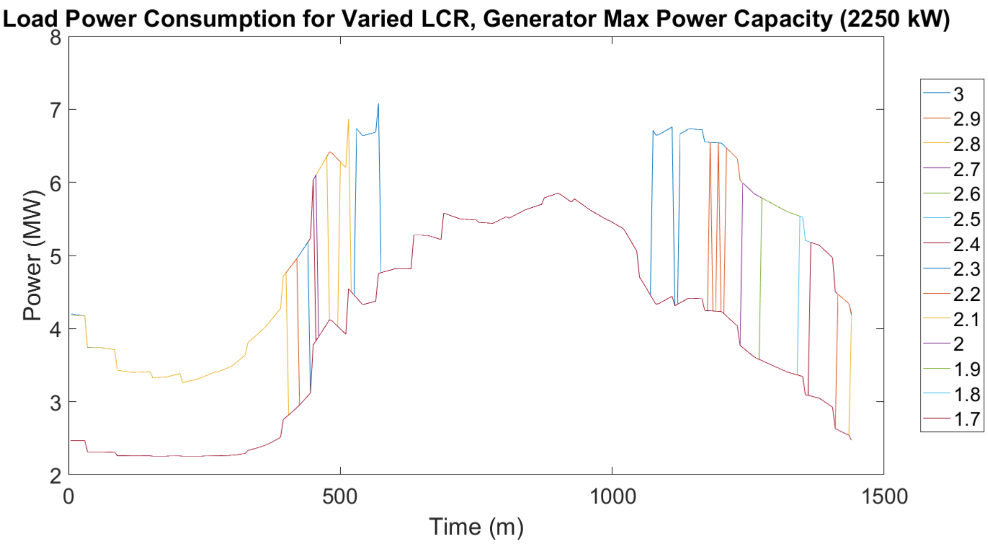

Figure 3.9: Load Consumption, LCR Varied from 3.0 to 1.7

Table 1 displays how the resource availability increases for a decreasing load to capacity ratio on the peak day of 2015.

Table 3.1: Distributed Generation Resource Availability for 2015 Peak Day

LCR	Resource Availability
3.0	36%
2.9	46%
2.8	51%
2.7	54%
2.6	57%
2.5	62%
2.4	63%
2.3	67%
2.2	69%
2.1	72%
2.0	100%
1.9	100%
1.8	100%
1.7	100%

3.0.2 Model based Volt-VAr Optimization

3.0.2.1 CVR by Volt-VAr Injection

Traditional CVR methods result in a power savings of 4% at peak load. The CVR by VAr-injection method reduces the peak load by only 3%. The BIS cannot output the maximum system reactive power without jeopardizing its function as a frequency responsive spinning reserve. This restriction reduces the maximum voltage differential, thus preventing the system from ever reaching the optimized voltage. While the CVR by VAr-injection method does not result in the maximum power reduction, it still meets the PGE prerequisite criteria of 3 to 5% power savings. Figure 3.10 compares the power savings using the volt-VAr injection CVR approach and traditional methods.

Table 3.2: Power Savings using Volt-VAr Injection and Traditional CVR Methods

CVR Method	Max Peak Reduction
VAr-Injection	3%
Traditional	4%

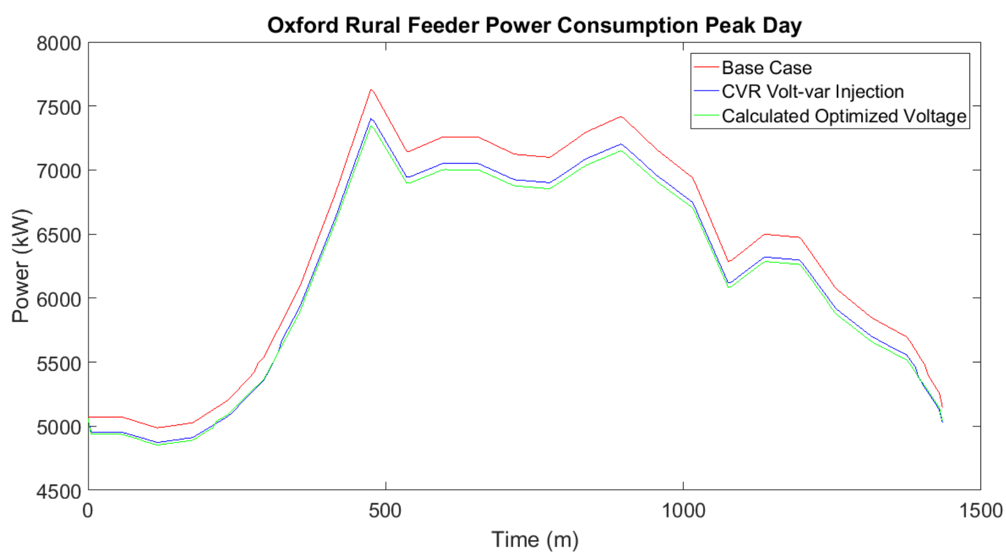

Figure 3.10: Minimized Power using CVR and VVO

3.0.2.2 Smooth Feeder Profile

The smooth feeder profile algorithm resulted in a peak shaving of 3% and an increase in power consumption during non-peak hours by 1%. PGE can overcommit to renewable resources to avoid shortages that result from the unreliability of a renewable supply. However, curtailing renewable resources at a time of overgeneration can be technologically difficult. According to the 2016 IRP, PGE has more flexibility to ramp up to meet demand than it does to ramp down. Installing the control program at the SSPC not only meets the PGE peak shavings requirement, it has the additional cost benefit of reducing renewable curtailment events. Figure 3.11 displays the optimized power profile achieved with the smooth feeder algorithm.

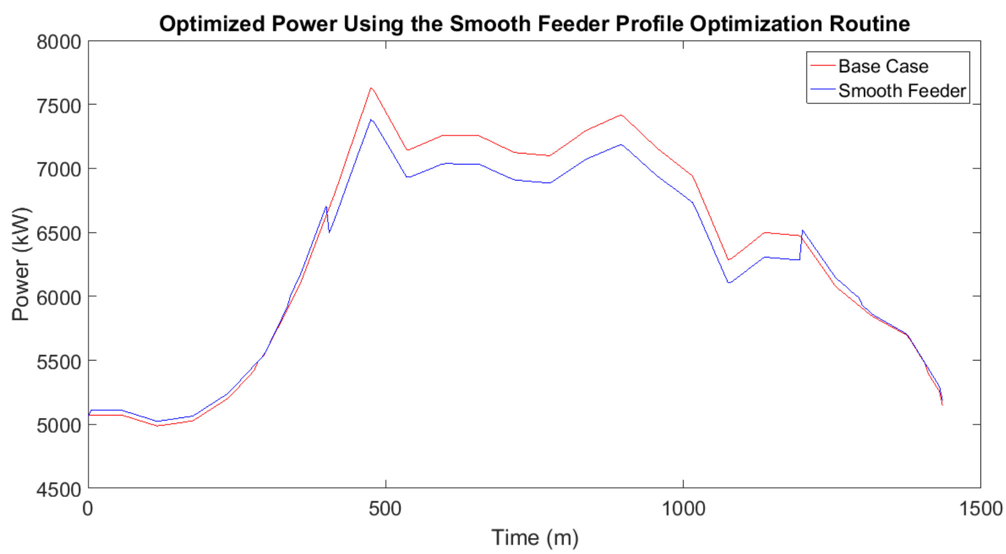

Figure 3.11: Optimized Power Using the Smooth Feeder Profile Algorithm

3.0.2.3 Lagrange Multipliers vs. Interior-Point Method

The interior-point optimization algorithm achieved the minimal voltage value in less processing time than the Lagrangian Multipliers method. However, the Lagrange multipliers algorithm reveals important information about the sensitivity of the constraints to the objective function. High λ values indicate which load nodes are most likely to violate the ANSI regulations.

Table 3.3: Power Savings using Lagrange Multipliers and Interior Point Algorithms

Algorithm	Max Peak Reduction	Max Increase
Lagrange Multipliers	3%	1%
Interior-Point	3%	1%

To visualize the lagrange multipliers, we sorted the λ values by node location. Since a feeder is not a straight line there are breaks in the ascending order that indicate a discontinuity in the neighboring node location. This is displayed in Figure 3.12 by the large jumps in lambda values. The multipliers are clustered around one, a small number, which indicates the solution is a minimum.

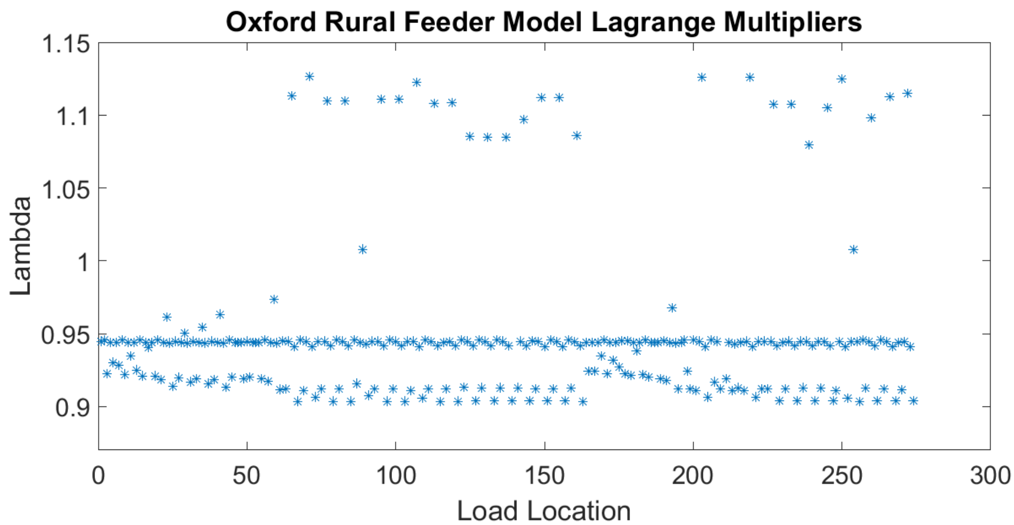

Figure 3.12: Lagrange Multipliers

The DG dynamic study should be conducted on additional feeders. Testing the program on an IEEE feeder model will confirm the results can be extended beyond PGE feeders.

The OpenDSS model must be validated against system data. Voltage readings from the substation to the "end of line" are needed to confirm the losses are equivalent. The actual system "end of line" voltage may be higher than the model. This would result in a lower optimized system voltage. Being able to run the system at a lower voltage is a benefit to PGE, as it results in a greater cost savings.

The excess reactive power in the system is dissipated in the form of heat. This raises concern for the thermal limits of the system. The cables must be sized to adequately handle the reactive power injections without becoming damaged. The feeder must be analyzed to confirm that the large reactive power injections will not affect the system negatively.

The system does not run at the minimum allowable voltage because the BIS is not large enough to achieve the desired voltage change. The potential power savings at the absolute minimum voltage could exceed the increased capacity costs.

5 Conclusion

5.0.0.1 MATLAB Dynamic Study

Presently, fifteen percent of PGE energy qualifies as renewables per Senate Bill 1547. To reach the qualifying 50% RPS by 2040, they plan to bring 515 MW of new wind resources on-line and eliminate their use of coal. This will result in a stochastic resource portfolio that requires PGE to strategically use their dispatchable energy resources, such as the aggregate 105MWs of DG participating in the Dispatchable Standby Generation program.

DSG is projected to grow into the 2030s. To increase the utility of this DSG program, the load to capacity ratio must be reduced, allowing more of these resource to be brought on-line. We analyzed the impact of increased DG on the Oxford Rural feeder and it revealed that the 50/51 protection element settings of the generator dictate both the maximum distributed generation capacity and the minimum load to capacity ratio.

- Currently, the load must be three times greater than the DG capacity for distributed generation to be brought on-line. The results of the study show that the three-to-one ratio is conservative.

- The dynamic study revealed that the generator instantaneous (50) relay will continue to provide protection as long as the load to capacity ratio meets the criteria:

$$LCR < GENOC50 - 1$$

Once the load-to-capacity ratio falls below this value, the generator is only protected by the time over-current (51) relay.

The maximum distributed generation capacity allowed on the feeder without concern for unintentional islanding is:

$$P_{Gen_{Max}} = \frac{P_{Load}}{LCR_{Min} + 1}$$

Decreasing the load to capacity ratio increases the availability of the distributed generation resources. For a LCR of 2.7 the resource availability increases by 18% percentage points.

5.0.0.2 Model Based Volt-VAr Optimization

Preliminary cost benefit analysis of two PGE, pilot CVR programs show that the power savings benefits outweigh the installation cost of CVR equipment [7].

The power savings criteria to warrant the installation of a control system at the SSPC is between three and five percent of the peak load. We examined the real power reduction that results from CVR by VAr-injection on the Oxford Rural feeder. The model based volt-VAr optimization study revealed that the autonomous control program will result in a peak power savings that meets the PGE criteria for project approval.

- The CVR by VAr-injection method resulted in a peak power savings of 3.0% and a maximum power reduction of 3.3%.

- Adding a smooth feeder profile algorithm maintains the peak shaving results while reducing renewable curtailment events by increasing the power consumption at times of overgeneration.

Bibliography

[1] Abdullah Bokhari, Ali Alkan, Rasim Dogan, Marc Diaz-Aguilo, Franscisco de Leon, Dariusz Czarkowski, Zican Zabar, Leo Birenbaum, Anthony Noel, and Resk Ebrahem Uosef. Experimental determination of the zip coefficients for modern residential, commercial, and industrial loads. *Power Delivery, IEEE Transactions on*, 29(3):1372–1381, June 2014.

[2] James E. Mack. Chapter 5: Distribution Circuits. Web, April 2017. http://www.globalspec.com/reference/67873/203279/chapter-5-distribution-circuits.

[3] Ameren. Conservation Voltage Reduction. Web, January 2017. http://ilsagfiles.org/SAG-files/Meeting-Materials/2017/January-24-2017/AIC-MES-2013-Presentation.pdf.

[4] Stack Exchange. What does it mean for reactive power to be delivered / consumed? Web, April 2017. https://electronics.stackexchange.com/questions/169957/what-does-it-mean-for-reactive-power-to-be-delivered-consumed.

[5] Roger C. Dugan. The Open Distribution System Simulator (OpenDSS). Web, March 2016. http://download2.nust.na/pub4/sourceforge/e/project/el/electricdss/OpenDSS/OpenDSSManual.pdf.

[6] Oregon Department of Energy. Renewable Portfolio Standard. Web, April 2017. https://www.oregon.gov/energy/energy-oregon/Pages/Renewable-Portfolio-Standard.aspx.

[7] Portland General Electric. PGE 2016 Integrated Resource Plan. Web, November 2016. https://www.portlandgeneral.com/our-company/energy-strategy/resource-planning/integrated-resource-planning.

[8] Portland General Electric. Dispatchable Generation. Web, April 2017. https://www.portlandgeneral.com/business/get-paid-to-help-meet-demand/dispatchable-standby-generation.

[9] Energy Storage Association. Spinning Reserve. Web, April 2017. http://energystorage.org/energy-storage/technology-applications/spinning-reserve.

[10] Portland General Electric. Salem Smart Power Center. Web, April 2017. https://www.portlandgeneral.com/our-company/energy-strategy/smart-grid/salem-smart-power-center.

[11] National Electric Manufacturers Association. American National Standard For Electric Power Systems and Equipment— Voltage Ratings (60 Hertz) n. Web, December 2006. https://www.nema.org/Standards/ComplimentaryDocuments/Contents-and-Scope-ANSI-C84-1-2011.pdf.

[12] Mark T. Osborn. Putting Standby Generators to Work on Grid Support. Web, March 2001. http://www.power-eng.com/articles/print/volume-105/issue-3/features/putting-standby-generators-to-work-on-grid-support.html.

[13] U.S. Department of Energy. The Potential Benefits of Distributed Generation and Rate-Related Issues That May Impede Their Expansion. Web, February 2007. https://www.ferc.gov/legal/fed-sta/exp-study.pdf.

[14] Power Systems Engineering. Direct Transfer Trip Scheme. Web, May 2011. http://peguru.com/2011/04/direct-transfer-trip-scheme/.

[15] U.S. Department of Energy. DOE Grid Modernization Laboratory Consortium Awards. Web, April 2017. https://energy.gov/under-secretary-science-and-energy/doe-grid-modernization-laboratory-consortium-gmlc-awards.

[16] NERC. Glossary of Terms Used in NERC Reliability Standards. Web, April 2017. http://www.nerc.com/files/glossary-of-terms.pdf.

[17] Jorge Nocedal and Stehen J. Wright. *Numerical Optimization*. Springer, New York, New York, 2006.

Appendix A: MATLAB Dynamic Study

A.0.1 Oxford Rural Feeder Model Validation Results

Table A.1: OpenDSS Oxford Rural Feeder Model 2015 Peak Load Voltage (p.u.)

Oxford Rural Feeder Model Peak Load Voltage (p.u.)					
Name	**Voltage(pu)**	**Name**	**Voltage(pu)**	**Name**	**Voltage(pu)**
764_112_5_b	1.017	26750_50_a	0.998	270_750_c	0.973
764_112_5_c	0.99	26750_50_b	1.033	505_0_000001_a	0.999
30621_300_a	0.999	26750_50_c	0.975	505_0_000001_b	1.032
30621_300_b	1.032	1469_500_a	0.999	505_0_000001_c	0.973
30621_300_c	0.972	1469_500_b	1.032	4574_37_5_a	0.999
5530_75_a	1.003	1469_500_c	0.973	4574_37_5_b	1.032
5530_75_b	1.008	29924_10_a	0.999	4574_37_5_c	0.972
5530_75_c	1.001	29924_10_b	1.031	40439_15_a	1.002
30463_15_a	0.999	29924_10_c	0.974	40439_15_b	1.015
30463_15_b	1.032	963_300_a	0.999	40439_15_c	0.992
30463_15_c	0.973	963_300_b	1.032	786_500_a	1.002
38175_25_a	0.998	963_300_c	0.973	786_500_b	1.012
38175_25_b	1.033	865_300_a	0.999	786_500_c	0.995
38175_25_c	0.975	865_300_b	1.031	1104_112_5_a	0.999
91542_25_a	1.003	865_300_c	0.974	1104_112_5_b	1.032
91542_25_b	1.009	846_300_a	0.999	1104_112_5_c	0.973

Oxford Rural Feeder Model Peak Load Voltage (p.u.)					
Name	**Voltage(pu)**	**Name**	**Voltage(pu)**	**Name**	**Voltage(pu)**
91542_25_c	0.999	846_300_b	1.031	30381_150_a	0.998
66323_15_a	1.004	846_300_c	0.974	30381_150_b	1.033
66323_15_b	1.004	329_225_a	0.999	30381_150_c	0.975
66323_15_c	1.006	329_225_b	1.032	12421_75_a	1.002
285_1000_a	0.999	329_225_c	0.973	12421_75_b	1.017
285_1000_b	1.033	975_150_a	1.001	12421_75_c	0.99
285_1000_c	0.973	975_150_b	1.023	6547_75_a	1.002
4517_37_5_a	1.003	975_150_c	0.983	6547_75_b	1.014
4517_37_5_b	1.01	187_225_a	0.999	6547_75_c	0.993
4517_37_5_c	0.998	187_225_b	1.032	8535_50_a	1.002
31502_2000_a	0.999	187_225_c	0.972	8535_50_b	1.016
31502_2000_b	1.03	498_0_000001_a	1.003	8535_50_c	0.991
31502_2000_c	0.974	498_0_000001_b	1.011	94459_25_a	0.998
718_300_a	0.999	498_0_000001_c	0.997	94459_25_b	1.033
718_300_b	1.032	5270_15_a	0.999	94459_25_c	0.974
718_300_c	0.972	5270_15_b	1.032	35724_50_a	0.999
4519_37_5_a	1.004	5270_15_c	0.973	35724_50_b	1.032
4519_37_5_b	1.006	270_750_a	0.999	35724_50_c	0.973
4519_37_5_c	1.003	270_750_b	1.031	98576_25_a	0.999
98576_25_b	1.031	44999_25_a	0.999	663_150_c	0.997
98576_25_c	0.974	44999_25_b	1.032	1048_75_a	0.998
6642_100_a	1.003	44999_25_c	0.972	1048_75_b	1.033
6642_100_b	1.011	8680_75_a	0.999	1048_75_c	0.975

Oxford Rural Feeder Model Peak Load Voltage (p.u.)					
Name	**Voltage(pu)**	**Name**	**Voltage(pu)**	**Name**	**Voltage(pu)**
6642_100_c	0.997	8680_75_b	1.031	62592_50_a	1.003
17434_25_a	0.999	8680_75_c	0.974	82033_25_a	0.998
17434_25_b	1.032	36151_10_a	1.002	46599_15_a	0.999
17434_25_c	0.973	36151_10_b	1.018	5771_75_b	1.023
31785_75_a	1.004	36151_10_c	0.989	74463_25_a	0.999
31785_75_b	1.006	663_150_a	1.003	65709_50_c	0.973
31785_75_c	1.003	663_150_b	1.011		

Table A.2: OpenDSS Oxford Rural Feeder Model 2015 Peak Line Current (p.u.)

Oxford Rural Feeder Model Peak Line Current (% of Rated)					
Name	**Loading**	**Name**	**Loading**	**Name**	**Loading**
10209308	11%	1103543	4%	1090687	13%
10209311	20%	1103544	8%	1090688	12%
10209313	20%	11912805	1%	1101098	1%
10209314	20%	11912806	1%	1101172	61%
10209315	20%	11912807	1%	1101174	60%
10209321	20%	11912808	1%	1101176	30%
10209326	1%	12629136	64%	1101274	2%
10482389	6%	12629137	2%	1103553	1%
10482391	3%	12629151	4%	12279297	3%
1090662	14%	12629152	4%	1101144	68%
1090663	4%	12629153	4%	1101249	26%
1090664	2%	12629154	4%	1101250	46%

Oxford Rural Feeder Model Peak Line Current (% of Rated)					
Name	**Loading**	**Name**	**Loading**	**Name**	**Loading**
1090665	3%	12629155	8%	1101275	28%
1090666	9%	12629160	20%	1101276	28%
1090701	11%	12629161	38%	1101277	28%
1090702	4%	12629163	15%	1101278	27%
1090739	3%	12629164	15%	1101279	56%
1090740	3%	12629166	15%	1101280	21%
1090744	4%	12743299	3%	1101281	21%
1090745	5%	12743301	3%	1103515	18%
1090762	5%	12969344	1%	1103516	13%
1101085	6%	12969345	1%	1103531	1%
1101086	6%	12969346	3%	12686936	21%
1101090	4%	12969353	12%	1090761	11%
1101091	2%	12974264	64%	1101143	32%
1101102	7%	12974265	64%	1101171	31%
1103486	3%	13008528	1%	1103561	20%
1103487	7%	13008529	1%	1103563	20%
1103488	2%	12629168	1%	12279296	3%
1103489	1%	1103559	4%	13008527	61%
1103491	1%	1103560	4%	10193003	11%
1103493	2%	12222913	2%	1101267	1%
1103511	10%	12743300	1%	1101268	1%
1103517	8%	12930298	4%	1101269	21%
1103519	10%	13063619	2%	1101270	20%

Oxford Rural Feeder Model Peak Line Current (% of Rated)					
Name	**Loading**	**Name**	**Loading**	**Name**	**Loading**
1103520	10%	1103499	2%	1103537	20%
1103523	1%	1090686	13%	1103538	20%
1103539	20%	12969342	21%	1101251	26%
1103562	20%	10192996	50%	1103555	4%
1103566	20%	10192997	50%	1103556	4%
1090722	1%	10192998	29%	1103557	2%
1090723	1%	10193001	21%	12969343	2%
1090724	1%	10193002	9%	13270058	2%
1090725	1%	1090719	9%	13437513	26%
1090726	2%	1090720	9%	13437514	26%
1090727	2%	1090729	9%	13437515	46%
1101272	20%	1101145	62%	13437516	26%
1103536	20%	1101173	60%	13454751	9%
1090728	9%	1101175	60%	13536475	1%

A.0.2 Load Multiplier

Table A.3: PGE 2015 Peak Day Load Multiplier

PGE 2015 Peak Day Load Multiplier					
Time(m)	**Multiplier**	**Time(m)**	**Multiplier**	**Time(m)**	**Multiplier**
4	0.646	484	0.995	964	0.928
9	0.646	489	0.989	969	0.925
14	0.646	494	0.983	974	0.923

PGE 2015 Peak Day Load Multiplier A.3					
Time(m)	**Multiplier**	**Time(m)**	**Multiplier**	**Time(m)**	**Multiplier**
19	0.646	499	0.977	979	0.920
24	0.646	504	0.971	984	0.917
29	0.646	509	0.965	989	0.915
34	0.646	514	0.959	994	0.912
39	0.646	519	0.953	999	0.910
44	0.646	524	0.947	1004	0.907
49	0.646	529	0.941	1009	0.904
54	0.646	534	0.935	1014	0.902
59	0.646	539	0.929	1019	0.899
64	0.646	544	0.929	1024	0.892
69	0.645	549	0.930	1029	0.884
74	0.644	554	0.931	1034	0.877
79	0.643	559	0.933	1039	0.869
84	0.642	564	0.934	1044	0.861
89	0.641	569	0.936	1049	0.853
94	0.640	574	0.937	1054	0.845
99	0.639	579	0.939	1059	0.838
104	0.638	584	0.940	1064	0.830
109	0.637	589	0.942	1069	0.822
114	0.636	594	0.943	1074	0.814
119	0.635	599	0.944	1079	0.806
124	0.635	604	0.945	1084	0.807
129	0.636	609	0.945	1089	0.810

PGE 2015 Peak Day Load Multiplier A.3					
Time(m)	**Multiplier**	**Time(m)**	**Multiplier**	**Time(m)**	**Multiplier**
134	0.636	614	0.945	1094	0.812
139	0.637	619	0.945	1099	0.815
144	0.637	624	0.945	1104	0.817
149	0.638	629	0.945	1109	0.820
154	0.638	634	0.945	1114	0.823
159	0.639	639	0.944	1119	0.825
164	0.639	644	0.944	1124	0.828
169	0.639	649	0.944	1129	0.830
174	0.640	654	0.944	1134	0.833
179	0.640	659	0.944	1139	0.836
184	0.642	664	0.943	1144	0.836
189	0.644	669	0.941	1149	0.836
194	0.646	674	0.940	1154	0.835
199	0.648	679	0.938	1159	0.835
204	0.650	684	0.937	1164	0.835
209	0.652	689	0.935	1169	0.834
214	0.654	694	0.933	1174	0.834
219	0.656	699	0.932	1179	0.834
224	0.658	704	0.930	1184	0.834
229	0.660	709	0.929	1189	0.833
234	0.662	714	0.927	1194	0.833
239	0.664	719	0.926	1199	0.833
244	0.667	724	0.925	1204	0.829

PGE 2015 Peak Day Load Multiplier A.3					
Time(m)	**Multiplier**	**Time(m)**	**Multiplier**	**Time(m)**	**Multiplier**
249	0.670	729	0.925	1209	0.824
254	0.673	734	0.924	1214	0.820
259	0.676	739	0.924	1219	0.815
264	0.680	744	0.924	1224	0.810
269	0.683	749	0.923	1229	0.806
274	0.686	754	0.923	1234	0.801
279	0.690	759	0.923	1239	0.796
284	0.693	764	0.923	1244	0.792
289	0.696	769	0.922	1249	0.787
294	0.699	774	0.922	1254	0.782
299	0.703	779	0.922	1259	0.778
304	0.708	784	0.924	1264	0.775
309	0.715	789	0.926	1269	0.772
314	0.721	794	0.928	1274	0.769
319	0.727	799	0.931	1279	0.767
324	0.734	804	0.933	1284	0.764
329	0.740	809	0.935	1289	0.761
334	0.746	814	0.938	1294	0.759
339	0.753	819	0.940	1299	0.756
344	0.759	824	0.942	1304	0.754
349	0.765	829	0.945	1309	0.751
354	0.772	834	0.947	1314	0.748
359	0.778	839	0.949	1319	0.746

PGE 2015 Peak Day Load Multiplier A.3					
Time(m)	**Multiplier**	**Time(m)**	**Multiplier**	**Time(m)**	**Multiplier**
364	0.786	844	0.951	1324	0.744
369	0.795	849	0.953	1329	0.742
374	0.803	854	0.954	1334	0.740
379	0.812	859	0.956	1339	0.738
384	0.821	864	0.957	1344	0.737
389	0.829	869	0.959	1349	0.735
394	0.838	874	0.960	1354	0.733
399	0.846	879	0.962	1359	0.731
404	0.855	884	0.963	1364	0.730
409	0.864	889	0.965	1369	0.728
414	0.872	894	0.966	1374	0.726
419	0.881	899	0.968	1379	0.724
424	0.890	904	0.965	1384	0.720
429	0.900	909	0.962	1389	0.715
434	0.910	914	0.959	1394	0.710
439	0.920	919	0.956	1399	0.705
444	0.929	924	0.953	1404	0.700
449	0.939	929	0.950	1409	0.695
454	0.949	934	0.946	1414	0.690
459	0.959	939	0.943	1419	0.685
464	0.969	944	0.940	1424	0.680
469	0.978	949	0.937	1429	0.675
474	0.988	954	0.934	1434	0.670

PGE 2015 Peak Day Load Multiplier A.3					
Time(m)	**Multiplier**	**Time(m)**	**Multiplier**	**Time(m)**	**Multiplier**
479	0.998	959	0.931	1439	0.666

A.0.3 MATLAB Dynamic Study

```
%Crystal Eppinger

%GRA, Portland State University

%June 2016


%Control program which uses OpenDSS and Oxford

%Rural Feeder model,data from CYMDIST courtesy

%of PGE.

%Runs through a 24 hr day with a resolution of 1

%min, total 1440 points.

%At each minute the current and voltage at the

%sub is exported and the magnitude of the apparent

%power is calculated. If the apparent power is

%ratio*apparent power greater than or equal to

%the generator capacity the generator is turned

%on. When the generator has been on for a minute

%the system injects a fault at the sub and queries

%the relay every 1ms to see if it has opened.

%If it has it clears the fault and records the time.


clc
```

```
clear all

%

cap = 250;

%Start generator capacity at 250kW and step up

%in 250kW increments

PF = 0.8; %Generator PF

gencap = cap/PF; %Generator capacity in KVA

gen_name = 'DG_1'; %Generator name

cap_run = 5000/250;

step = 250; %Increment value for generator capacity

a = 5; %Row where Line.oxfo_r135 is in currents

%array, it seems to move around a bit, so start %there

b = 4; %Column where I1 is in currents array

c = 5; %Column where I1 angle is in currents array

d = 4; %Column where V1 is in voltages array

e = 5; %Column where V1 angle is in voltages array

f = 157;

%Row where the generator line current may start,

%it seems to move around a bit

i = 1; %Row variable that represents 24 hr loop

points = 1440; %Number of points in the loadshape file

fault_length = 5000; %Length that the fault occurs at the sub

start = 3.000; %Starting generator to capacity ratio

finish = 0.999; %Ending generator to capacity ratio

ratio = start; %Generator/capacity ratio

incr = 0.1;
```

```
%Value to decrease the generator to capacity ratio
k = 1; %third dimension array variable
gen_time_on = 0.001; %Sampling rate for when the
%fault occurs to find out when the Recloser opens
reg_int = 1; % 1 minute intervals used to calculate
%the current to see if the generator can come on
true = 'true';
volt = 277; %Line to neutral voltage at the generator
filename1 =
'C:\Users\Crystal\Desktop\DG_site_2_Phase\
Oxford_Rural_EXP_ElemCurrents.csv';
%File where OpenDSS exports currents
filename2 =
'C:\Users\Crystal\Desktop\DG_site_2_Phase\
Oxford_Rural_EXP_ElemVoltages.csv';
%File where OpenDSS exports voltages
m = 2; %Row variable for "reached" array

%Label columns of reached array
reached{1,1} = 'Off time';
reached{1,2} = 'Line I at fault';
reached{1,3} = 'Time';
reached{1,4} = 'Current prior to sub fault';
reached{1,5} = 'Generator Capacity (kW)';
reached{1,6} = 'Ratio';
reached{1,7} = 'S prior to sub fault';
```

```
reached{1,8} = 'Calculated Ratio';
reached{1,9} = 'Phase Relay Setting';


line_name_gen = '"Line.gen1"';
line_name_fault = '"Line.oxfo_r135"';
filename3 =
'C:\Users\Crystal\Desktop\DG_site_2_Phase
\LoadShape_Rural.csv';
%Loadshape file name
filename4 =
'C:\Users\Crystal\Desktop\DG_site_2_Phase
\Oxford_Rural_Mon_DG_1_i.csv';
loadmult = csvread(filename3);
h = 1; %Variable to count the number of faults occurred
turn_on = 0;
%Variable that dictates whether the generator turns
%on or not


%Initialize OpenDSS
% Instantiate the OpenDSS Object
Obj = actxserver('OpenDSSEngine.DSS');
DSSStart = Obj.Start(0);
DSSText = Obj.Text;
DSSCircuit = Obj.ActiveCircuit;
DSSSolution = DSSCircuit.Solution;
```

```
% Compile the circuit
DSSText.command = 'Compile C:\Users\Crystal\Desktop\DG_site_2_Phase\
Rural_Main_Site_2_Phase2.dss';
DSSText.command =
'Set mode=daily controlmode=time'; %The circuit must be first solved in
%time mode

for l = 1:cap_run

    %Calculate the ratio, capacity and the phaseinst
    %Recloser setting
    ratio = start; %Generator/capacity ratio, start
    %at 500 and step up to 5000
    gencap = cap/0.8; %Generator capacity in KVA

    phase_i = (gencap*10^3/3)/volt; %Calculate the generator current
    %for the relay setting
%     phase_inst = phase_i*(1.5); %Calculate the instantaneous relay
%setting
    one_time_through = 1; %Variable
    %which notes whether the program
    %is on its first run for a
    %specified capacity
    power = strcat('[',num2str(gencap),',',
    num2str(gencap),']');
```

```
    DSSText.Command = ['Generator.'
    num2str(gen_name) '.kW='
    num2str(cap)]; %Change the
    %generator capacity from the COM
    %server
    DSSText.Command = ['Recloser.Recloser_gen1.
    Phasetrip=' num2str(phase_i)];
    %Change the Recloser setting fo
    %the generator
%     DSSText.Command = ['Recloser.Recloser_gen1.Phaseinst='
%num2str(phase_inst)];
%Change the instantaneous Recloser setting for the generator

    %Vary the ratio from 3 to 1 for
    %all capacity values
    while(ratio >= finish) %Varies
    %the ratio from 3 to 1 in steps
    %of 0.1
        DSSText.Command = '? Generator.DG_1.Enabled';
        %Query the generator
        %to see if it's enabled
        gen_status = DSSText.result;
        %Save true/false in variable

        fault = 0; %Variable which signifies whether a fault
        %has already occurred in a 24 hr period
```

```
n = 1; %Reset row variable for on array
i = 1; %Reset row variable for 24 hr loop
g = 1; %Loadshape multiplier row variable
%Every minute poll the current and check to see if the
%load:capacity ratio is met
enabled = strcmp(gen_status,true);
%Check to see if the gen is enabled
DSSText.command = 'Set mode=daily controlmode=time';
%Set mode to time

while(i <= points) %Runs through 1440 data points
    %If the generator is on, mimic a fault at the sub
    %and see if the Recloser trips

    %If the generator is on and a fault has not occurred
    %in the 24hr period
    if((enabled == 1) && (fault==0))
        fault = 1;
        %So fault will not occur in 24 hr period
        time = 0; %Time variable for relay
        reached{m,3} = i;
        %Time (min) the fault occurred
        reached{m,4} = line_I(n-1,1,k);
        %Gen current right before the fault occurred
        reached{m,5} = cap; %Gen capacity
        reached{m,6} = ratio; %Ratio variable
```

```
reached{m,7} = S(n-2,1,k);
%Apparent power at the sub
S_mag_comp = reached{m,7};
%Use this to compare to the next ratio
reached{m,8} = on(n-2,1,k);
%Actual calculated ratio
ratio_comp = reached{m,8};
reached{m,9} = phase_i;
%Phase current relay setting

%Reset variables so gen wont come on again
%in the 24 hr period
one_time_through = 0;
turn_on = 0;

%Change to dynamic mode to allow for gen to be
%modeled by simple swing equation instead of a
%negative load
DSSText.Command = 'Set mode=dynamic';
DSSText.Command= 'Recloser.Recloser_SUB.Action=open';
%Open the relay at the sub

%Fault occurs for a maximum of fault_length*gen_time_on
for j=1:fault_length
    DSSText.Command= ['Solve stepsize=
    ' num2str(gen_time_on) 's  number=1'];
```

```
    %Solve the circuit for 0.01s
    time = time+gen_time_on;
    %Keep track of the time to find out when the
    %generator relay opens

    %Query to see if the Recloser is open
    DSSCircuit = Obj.ActiveCircuit;
    %Point it to the circuit
    circ=DSSCircuit.CktElements('Line.Gen1');
    %Here the Cktelements handler is called
    DSSCircuit.SetActiveElement('Line.Gen1');
    %Here the line is set as active element
    check=circ.IsOpen(1,0);
    %Here we ask if the line is open

    %If the Recloser is open record all of the
    %values
    if(check == 1)
        reached{m,1} = time;
        break
    end
end

%Export monitor
DSSText.command = 'Export Monitors DG_1_i';
%Open element currents and voltages file
```

```
fileID = fopen(filename4);
dgi_temp = textscan(fileID,
'%s %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s
%s','Delimiter',',');

fclose('all');
reached{m,2} = dgi_temp{1,11}{2,1};



m = m+1; %Increment the reached array
DSSText.Command=
'Recloser.Recloser_Sub.Action=close';
%Close the sub Recloser

DSSText.Command=
'Recloser.Recloser_Gen1.Action=close';
%Close the gen Recloser

DSSText.Command =
'Generator.DG_1.Enabled=false';
%Disable the generator

enabled = false; %Set enabled to false
hour = floor(i/60); %Reset the hour
secs = mod(i,60)*60; %Reset the seconds
DSSText.Command =
```

```
        ['Set time=(' num2str(hour),
        num2str(secs) ')'];
        DSSText.command =
        'Set mode=daily controlmode=time';
        %Reset the mode to time

    else
        %Solve the circuit,
        %export the currents and voltages,
        %do the math to figure out if the generator should
        %be turned on
        multiplier = loadmult(g,2);
        %Set the load multiplier
        DSSText.Command= ['Set loadmult=
        ' num2str(multiplier)];
        %Change the load multiplier
        DSSText.Command= ['Solve stepsize=
        ' num2str(reg_int) 'm  number=1'];
        g = g+1;
        i = i+1;

        %Export currents and voltages
        DSSText.command = 'Export ElemCurrents';
        DSSText.command = 'Export ElemVoltages';

        %Open element currents and voltages file
```

```
fileID = fopen(filename1);
currents = textscan(fileID,
'%s %s %s %s %s %s %s %s %s %s %s %s %s %s %s',
'Delimiter',',');

fileID = fopen(filename2);
voltages = textscan(fileID,
'%s %s %s %s %s %s %s %s %s %s %s %s %s %s %s',
'Delimiter',',');

fclose('all');

%Find the line current which seems to move around,
%start at 158 and look for a match
match1 = 0;
f = 157;

while(match1 == 0)
    line_name_match_gen = currents{1,1}{f,1};
    match1 =
    strcmp(line_name_match_gen,line_name_gen);
    if(match1 == 1)
        line_I(n,1,k) =
        str2double(currents{1,b}{f,1});
        %Record the generator current
        break
```

```
    end
    f = f+1;
end;
%Calculate the feeder load (S) from the voltages
%and currents at line 1157v2 which is at the head
%of the Oxford Rural feeder

match2 = 0;
a = 5;

while(match2 == 0)
    line_name_match_fault =
    currents{1,1}{a,1};
    match2 =
    strcmp(line_name_match_fault,line_name_fault);
    if(match2 == 1)
        I1_sub_mag(n,1,k) =
        str2double(currents{1,b}{a,1});
        rho = I1_sub_mag(n,1,k);
        I1_sub_ang(n,1,k) =
        str2double(currents{1,c}{a,1});
        theta = I1_sub_ang(n,1,k);
        [I1x,I1y] = pol2cart(theta,rho);
        Z = I1x + 1j*I1y;
        I1_conj = conj(Z);
        I1_conj_mag = real(I1_conj);
```

```
        %Calculate the voltage
        V1_sub_mag(n,1,k) =
        str2double(voltages{1,d}{a,1});
        V1_sub_ang(n,1,k) =
        str2double(voltages{1,e}{a,1});

        % S = 3*(Vphase)(Iphase*)
        S_mag =
        abs(3*I1_conj_mag*V1_sub_mag(n,1,k));
        S(n,1,k) = S_mag;
        on(n,1,k) = S_mag/(gencap*10^3);
        %Ratio of feeder load to gen capacity
        n = n+1;
        break
    end
    a = a+1;
end

%Check to see if the ratio is met - done in
%steps for troubleshooting purposes ,
%could be changed to one if statement

%If the designated ratio is less or equal to the
%calculated ratio
%If the calculated ratio is no more than 0.1 the
%designated ratio - to ensure I get data for
```

```
%2.9, 2.8, etc.

if(on(n-1,1,k) >= ratio)

    if(one_time_through == 1 && fault == 0)

        turn_on = 1;

    elseif(one_time_through == 0 && fault == 0)

        if(ratio_comp - incr >= on(n-1,1,k))

            if(S_mag_comp > S_mag)

                turn_on = 1;

            else

                turn_on = 0;

            end

        else

            turn_on = 0;

        end

    end

else

    turn_on = 0;

end



if(turn_on == 1)

    DSSText.Command = '? Generator.DG_1.Enabled';

    %Query the generator

    gen_status = DSSText.result;

    %Store the true/false result in gen_status

    enabled = strcmp(gen_status,true);
```

```
%If the gen is off, turn it on and update the
%status
if(enabled == 0)
    DSSText.Command =
    'Generator.DG_1.Enabled=true'; %Enable gen
end
multiplier = loadmult(g,2);
%Set the load multiplier
DSSText.Command=
['Set loadmult=' num2str(multiplier)];
DSSText.Command=
['Solve stepsize=' num2str(reg_int) 'm  number=1'];
%Solve for 1 min
g = g+1;
i = i+1;
DSSText.Command =
'? Generator.DG_1.Enabled';
%Query the generator
gen_status = DSSText.result;
%Store the true/false result in gen_status
enabled = strcmp(gen_status,true);

%Export currents and voltages
DSSText.command = 'Export ElemCurrents';
DSSText.command = 'Export ElemVoltages';
```

```
%Open element currents and voltages file
fileID = fopen(filename1);
currents = textscan(fileID,
'%s %s %s %s %s %s %s %s %s %s %s %s %s %s %s',
'Delimiter',',');

fileID = fopen(filename2);
voltages = textscan(fileID,
'%s %s %s %s %s %s %s %s %s %s %s %s %s %s %s',
'Delimiter',',');

fclose('all');

%Find the line current which seems to move around,
%start at 158 and look for a match
match1 = 0;
f = 157;

while(match1 == 0)
    line_name_match_gen =
    currents{1,1}{f,1};
    match1 =
    strcmp(line_name_match_gen,line_name_gen);
    if(match1 == 1)
        line_I(n,1,k) =
```

```
        str2double(currents{1,b}{f,1});
        %Record the generator current
        break
    end
    f = f+1;
end;
%Calculate the feeder load (S) from the
%voltages and currents at line 1157v2 which
%is at the head of the Oxford Rural feeder

%Calculate the complex conjugate of the current
match2 = 0;
a = 5;

while(match2 == 0)
    line_name_match_fault =
    currents{1,1}{a,1};

    match2 =
    strcmp(line_name_match_fault,line_name_fault);

    if(match2 == 1)
        I1_sub_mag(n,1,k) =
        str2double(currents{1,b}{a,1});

        rho = I1_sub_mag(n,1,k);
```

```
            I1_sub_ang(n,1,k) =

            str2double(currents{1,c}{a,1});


            theta = I1_sub_ang(n,1,k);

            [I1x,I1y] = pol2cart(theta,rho);

            Z = I1x + 1j*I1y;

            I1_conj = conj(Z);

            I1_conj_mag = real(I1_conj);

            %Calculate the voltage

            V1_sub_mag(n,1,k) =

            str2double(voltages{1,d}{a,1});


            V1_sub_ang(n,1,k) =

            str2double(voltages{1,e}{a,1});


            % S = 3*(Vphase)(Iphase*)

            S_mag =

            abs(3*I1_conj_mag*V1_sub_mag(n,1,k));

            S(n,1,k) = S_mag;

            on(n,1,k) = S_mag/(gencap*10^3);

            %Ratio of feeder load to gen capacity

            n = n+1;

            break

        end

        a = a+1;

    end
```

```
        else
            %Query the gen status, if it's on, turn it off,
            %if it's off, keep it off
            DSSText.Command = '? Generator.DG_1.Enabled';
            %Query the generator
            gen_status = DSSText.result;
            enabled = strcmp(gen_status,true);

            if(enabled == 1)
                DSSText.Command =
                'Generator.DG_1.Enabled=false';
                %Disable gen

                enabled = false;
                %Manually set enabled variable
            end

        end

    end

end

ratio = ratio - incr; %Decrement the ratio
```

```
        disp(ratio)
        k = k+1; %Increase the third dimension of the array
        %which keeps the ratio, S, current, etc.
    end

    %Store it all in a really big array, unpack in a structure
    %for easier plotting
    arrays{l,:} = {reached I1_sub_mag on S V1_sub_mag};
    %Array of array names to rename with the
    %generator capacity value included

    cap = cap + step; %Increment generator capacity
    disp(cap)

end
%
filename =
'\\khensu\Home06\eppinger\Desktop\DG_site_2_Inst\
Reached_site2_inst.xlsx';
xlswrite(filename,reached)
```

Appendix B: Model Based Volt-Var Injection

B.0.1 Fmincon

```
%Open substation power/voltage equation
filename6 = '\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916
\PVsub.csv';
PVsub = csvread(filename6);

%Open vector of load voltage/power equations matrix
filename8 = '\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916
\PV_volt.csv';
PV_volt = csvread(filename8);
[rv,cv] = size(PV_volt);

vbase = 12.47*10^3;

%Develop Psub and V_volt - relationship between load voltages
%and sub voltage
j = 1;
for i=1:rv
    A(j,1) = PV_volt(i,1);
    A(j+1,1) = -PV_volt(i,1);
    b(j,1) = 1.05*vbase - PV_volt(i,2);
```

```
    b(j+1,1) = -0.95*vbase + PV_volt(i,2);
    j = j+2;
end
```

B.0.2 Lagrange Multipliers

```
%Open substation power/voltage equation
filename6 = '\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916
\PVsub.csv';
PVsub = csvread(filename6);

%Open vector of load voltage/power equations matrix
filename8 = '\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916
\PV_volt.csv';
PV_volt = csvread(filename8);
[rv,cv] = size(PV_volt);

vbase = 12.47*10^3;
%Create symbolic values
v = sym('v'); %voltage at the sub
s = sym('s'); %constraint for v > 0.95
t = sym('t'); %constraint for v < 1.05
S = transpose(sym('S',[1 rv])); %Constraints for V > 0.95
T = transpose(sym('T',[1 rv])); %Constraints for V < 1.05
%Lagrangian multipliers
lambda1 = sym('lamba1');
lambda2 = sym('lambda2');
```

```
lambda3 = transpose(sym('lambda3',[rv 1]));

lambda4 = transpose(sym('lambda4',[rv 1]));


%Develop Pload, Ploss and Psub

for i=1:rv

    V(i,1) = PV_volt(i,1)*v + PV_volt(i,2);

end

Psub(1,1) = PVsub(1,1)*v + PVsub(1,2);


%Develop system vector

sysvec=

[v;s;t;S;T;lambda1;lambda2;transpose(lambda3);transpose(lambda4)];


%Create auxilliary function

fx = Psub; %Circuit power based on voltage at the substation

c1 = -v + 0.95*vbase + s^2;

%Ensure that the voltage does not go lower than 0.95 of the base

c2 = v - 1.05*vbase + t^2;

c3 = -V + 0.95*vbase + S.^2;

c4 = V - 1.05*vbase + T.^2;


%Lagrangian equation

Lx = fx - lambda1*c1 - lambda2*c2 - lambda3*c3 - lambda4*c4;


[rtemp,ctemp] = size(sysvec);
```

```
for i=1:rtemp
    syseq(i,1) = diff(Lx,sysvec(i,1));
end

for i=1:rtemp
    G1 = 'F(';
    G2 = num2str(i);
    G3 = ') = ';
    G4 = char(syseq(i,1));
    G{i,1} = strcat(G1,G2,G3,G4,';');

end

x = sym('x',[rtemp,1]);
k = rtemp;

for i=1:rtemp
    oldstring = char(sysvec(k,1));
    newstring = strcat('x(',num2str(k),')');
    G = strrep(G,oldstring,newstring);
    k = k-1;
end
```

B.0.3 Optimization Routine

```
tic;
% method = 'Lagrangian';
```

```
method = 'fmincon';

pbis = [5.398975831437784,-6.930390115403396e+04];

pmodel = [14.020877211237167,-1.801643045218248e+05];

filename1 =

'\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916

\Rural_Main_Site_1_Inst.dss';

%Oxfor Rural OpenDSS file

filename7 =

'\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916

\A.csv';

A = csvread(filename7);

filename5 = '\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916

\b.csv';

b = csvread(filename5);

[ra,ca] = size(b);

filename2 =

'\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916

\LoadShape_Rural_5min.csv';

%Loadshape file name

loadshape = csvread(filename2);

filename3 =

'\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916\

Oxford_Rural_Mon_line_1157v2.csv';

%Load filename for export to check voltage

filename4 =

'\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916\
```

```
Oxford_Rural_Mon_line_1157v22.csv';
%Load filename for export to check power
filename5 =
'\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916\Solararray.csv';
solgenprof = csvread(filename5);
filename6 = '\\khensu\Home06\eppinger\Desktop\Rural_BESS_92916
\windarray.csv';
windgenprof = csvread(filename6);
% Initialize variables
numminsday = 24*(60/5);
% number of 5 min intervals throughout peak day
numminsyear = 365*24*(60/5);
%number of 5 min intervals throughout peak year
%Declare variables
sys = 549;
%Size of the Lagrangian system for solving optimal voltage at the sub
g = 1;
%Load multiplier row variable
reg_int = 5;
%Minute solution intervals
feedpowmax = 7.569218159*10^3; %(kW)
Pavg = 2.182100232*10^3;
solpowmax = 1*10^3; %1 MW solar array
windpowmax = 1.5*10^3; %1.5 MW wind farm
Pgen = 2250; %Distributed generation on feeder (kW)
LCR = 2.7; %Load to capacity ratio
```

```
vbase = 12.47*10^3;
x0 = vbase;
optionsfmin =
optimoptions(@fmincon,'StepTolerance',1*10^-15,'ConstraintTolerance',1);
optionsfsolve =
optimoptions(@fsolve,'MaxFunctionEvaluations',549*100,
'MaxIterations',1000);
lb = 0.95*vbase;
ub = 1.05*vbase;

% Start up OpenDSS
global DSSStartOK;
global DSSObj;
global DSSText;

[DSSStartOK, DSSObj, DSSText] = DSSStartup;
DSSText.Command =
'Compile \\khensu\Home06\eppinger\Desktop\Rural_BESS_92916
\Rural_Main_Site_1_Inst.dss'; %Compile the circuit
DSSText.command = 'Set mode=daily controlmode=time';
%Set the solution mode and control mode

for i=1:numminsday %Run loop for entire day
    timerVal(i,1) = tic;
    multiplier = loadshape(g,2); %Set multiplier from imported array
    solmult = solgenprof(g,2); %Solar multiplier
```

```
windmult = windgenprof(g,2);

%Calculate the current power and voltage so the voltage can be
%optimized
[ Pnet,g,DG_on,voltage,Pload ] =
Calcpowerneeds(i,multiplier,reg_int,g,filename3,filename4,LCR,
Pgen,solmult,windmult,solpowmax,windpowmax);
Pfinal(i,1) = Pload;
Vfinal(i,1) = voltage/vbase;
%Get the optimized voltage
x =
Get_Optim_volt(Pnet,Pavg,method,vbase,A,b,lb,ub,optionsfmin,
optionsfsolve,sys,multiplier,i);

%Find actual voltage to input to the system
Vout = (x(1,1)*pmodel(1,1) + pmodel(1,2) - pbis(1,2))/pbis(1,1);

%Ideal voltage to minimize or maximize the power
idvolt(i,1) = Vout;
volt_diff = idvolt(i,1) - voltage;
%Find the difference of the current and the optimal voltage

%Choose appropriate fb system based on the voltage difference
fbfile = choose_fb_system(volt_diff);

%Open up the feedback model and get the input kvar to inject
```

```
    %into the system
    open_system(fbfile); %Open up the Simulink model
    simin.time = 0;
    simin.signals.dimensions = 1;
    simin.signals.values =double(idvolt(i,1));
    simout = sim(fbfile,'SimulationMode','normal');
    Results = simout.get('simout');
    Kvar_point = get( Results, 'data');
    [kr,kc] = size(Kvar_point);
    batt_kva = Kvar_point(kr);
    %Negative reactive power is used for consumption
    if(volt_diff < 0)
        batt_kva = -1*batt_kva;
    end
     if(batt_kva > 4000)
         batt_kva = 4000;
     elseif(batt_kva < -4000)
         batt_kva = -4000;
     end

    kva(i,1) = batt_kva;
    %Set kVA
    turn_on = strcat('Generator.Batt.kvar=', num2str(batt_kva));
    DSSText.Command = num2str(turn_on); %Set the kvar
    timerVal(i,2) = toc;
end
```

```
function [ Pnet,g,DG_on,voltage,Pload] =
Calcpowerneeds(i,multiplier,reg_int,g,filename3,filename4,LCR,
Pgen,solmult,windmult,solpowmax,windpowmax)

%Start up the DSS

global DSSText;

    %Calculate optimal voltage to accomodate renewables
    Pmustrun = solmult*solpowmax + windmult*windpowmax;

    DSSText.Command=

    ['Set loadmult=' num2str(multiplier)];
    %Change the global load multiplier
    DSSText.Command=
    ['Solve stepsize=' num2str(reg_int) 'm  number=1'];
    %Solve for 1 minute
    g = g+1; %Increment the load array pointer
    %Export monitor voltage
    DSSText.command = 'Export Monitors line_1157v2';

    %Open element currents and voltages file
```

```
    voltage_temp = csvread(filename3,1,0);

    voltage = (voltage_temp(i,3))*sqrt(3);


    %Export monitor power

    DSSText.command = 'Export Monitors line_1157v22';

    %Open element powers file

    power_temp = csvread(filename4,1,0);

    [rp,cp] = size(power_temp);

    jjj = rp;

    iii = 3;

    Pload = 0;

    for kk=1:3

        Pload = Pload + power_temp(jjj,iii);

        iii = iii+2;

    end


    Pnet = Pload - Pmustrun;

    if(Pnet >= LCR*Pgen)

        DG_on = 1;

        %Turn on DG, solve the circuit and recalculate

%         DSSText.Command = '? Generator.DG_2.Enabled';

%Query the generator


    else

        DG_on = 0;

%         %Turn on DG, solve the circuit and recalculate
```

```
%         DSSText.Command =
%'? Generator.DG_2.Enabled=false'; %Query the generator
    end


function [ x ] =
Get_Optim_volt(Pnet,Pavg,method,vbase,A,b,lb,ub,
optionsfmin,optionsfsolve,
sys,multiplier)
nonlcon = [];
Aeq = [];
beq = [];
if(Pnet > Pavg && multiplier > 0.5) %If the power needs to be reduced
    if(strcmp(method,'fmincon')==1);
        x0 = 0.97*vbase; %Start at the minimum
        %Optimization loop to reduce voltage - minimize power
        x = fmincon(@Minpower,x0,A,b,Aeq,beq,lb,ub,nonlcon,
        optionsfmin);
        %Solve for optimal voltage
    elseif(strcmp(method,'Lagrangian')==1)
        x0 = ones(sys,1);
        x0(1,1) = 0.97*vbase;
        x = fsolve(@Ideal_voltage_min,x0,optionsfsolve);
        %Solve for optimal voltage
    end
elseif(Pnet < Pavg && multiplier < 0.5)
%If there is max generation, shut off DG,
```

```
%accomodate must-run and maximize voltage
    if(strcmp(method,'fmincon')==1);
        x0 = 1.05*vbase; %Start at the minimum
        %Optimization loop to maximize power
        x = fmincon(@Maxpower,x0,A,b,Aeq,beq,lb,ub,nonlcon,optionsfmin);
        %Solve for optimal voltage
    elseif(strcmp(method,'Lagrangian')==1)
        x0 = ones(sys,1);
        x0(1,1) = 1.05*vbase;
        x = fsolve(@Ideal_voltage_max,x0,optionsfsolve);
        %Solve for optimal voltage
    end
end

end

function [ fbfile ] = choose_fb_system(volt_diff)

sysfile100 = '\\khensu\Home06\eppinger\Desktop\Research\FB_System
\sys_100_f.slx';
sysfile500 = '\\khensu\Home06\eppinger\Desktop\Research\FB_System
\sys_500_f.slx';
sysfile1000 = '\\khensu\Home06\eppinger\Desktop\Research\FB_System
\sys_1000_f.slx';
sysfile2000 = '\\khensu\Home06\eppinger\Desktop\Research\FB_System
\sys_2000_f.slx';
```

```
sysfile4000 = '\\khensu\Home06\eppinger\Desktop\Research\FB_System
\sys_4000_f.slx';

volt_temp = abs(volt_diff);

%Choose feedback system
if(volt_temp < 30)
    fbfile = sysfile100;
elseif(volt_temp >= 30 && volt_temp < 71)
    fbfile = sysfile500;
elseif(volt_temp >= 71 && volt_temp < 95)
    fbfile = sysfile1000;
elseif(volt_temp >= 95 && volt_temp < 170)
    fbfile = sysfile2000;
elseif(volt_temp >= 170)
    fbfile = sysfile4000;
end

end
```