OPTIMAL REACTIVE POWER CONTROL OF GRID CONNECTED PHOTOVOLTAIC RESOURCES

by

JOSHUA RYAN TRIMBLE

B.S., Georgia Institute of Technology, 2006

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado Denver
in partial fulfillment
of the requirements for the degree of
Master of Science
Electrical Engineering
2013
This thesis for the Master of Science degree by
Joshua Ryan Trimble
has been approved for the
Electrical Engineering Program
by

Fernando Mancilla-David, Advisor
Titsa Papantoni
Dan Connors

December 12, 2013
ABSTRACT

As more photovoltaic distributed generation resources are installed on distribution power systems, selective control of the inverters connecting the DC power sources presents the opportunity to supply both real and reactive power at the point of common coupling. This thesis presents a simulated distribution system with individually controlled PV resources with the objective of minimizing total system losses while operating at the maximum power point and below the simulated rating of the associated inverters. The control strategy assumes the characteristics of the distribution system are known and solves for the optimal power flow operating point. The ability of each PV source to provide real and reactive power varies instantaneously as irradiance changes, so the operating point for each resource must be constantly recalculated and adjusted. The assumption of known system parameters can be justified in a Smart Grid context, and a solution based on overall system power flow should be considered as a benchmark for any other state estimation or local control approaches.

The form and content of this abstract are approved. I recommend its publication.

Approved: Fernando Mancilla-David
TABLE OF CONTENTS

Figures .............................................................. vi
Tables ................................................................. viii

Chapter

1. Introduction ....................................................... 1
   1.1 Reactive Supply Capabilities and Historical Limitations .... 1
   1.2 Economic Considerations and Operating Modes .............. 2
2. Photovoltaic Array and Inverter Model ................................ 5
   2.1 Model of the PV Cell and Array ................................ 5
   2.2 Model for the Grid Connected PV System .................... 6
   2.3 Voltage Source Inverter Model ................................ 7
3. Power Flow Algorithm ............................................. 10
   3.1 Introduction .................................................. 10
   3.2 Formulation of the Power Flow Equations .................... 10
   3.3 Modified Formulation for Reactive Power Control of PV Resources .... 15
      3.3.1 Formulation of the Power flow equations ............. 16
      3.3.2 Objective Function and System Constraints .......... 18
4. Power System Simulation .......................................... 20
   4.1 Distribution Feeder Model .................................. 20
   4.2 Control System Interface ................................... 22
   4.3 Timed Breaker Logic ....................................... 23
5. Test Cases and Simulation Results ................................ 25
   5.1 Test Case - Constant Reactive Power Output ............... 25
   5.2 Test Case - Variable Irradiance ............................ 26
   5.3 Test Case - Variable Load .................................. 28
   5.4 Measurement of Objective Function .......................... 31
6. Conclusion ......................................................... 33
# References

Appendix

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. MATLAB Reactive Power Control Function</td>
<td>36</td>
</tr>
<tr>
<td>A.1 Main Function - QCTL</td>
<td>36</td>
</tr>
<tr>
<td>A.2 Sub Function - readCF</td>
<td>40</td>
</tr>
<tr>
<td>A.2.1 Sub Function - findk1k2</td>
<td>49</td>
</tr>
<tr>
<td>A.3 Sub Function - Build Ybus</td>
<td>49</td>
</tr>
<tr>
<td>A.4 Sub Function - ObjFun</td>
<td>50</td>
</tr>
<tr>
<td>A.5 Sub Function - NonLinCon</td>
<td>51</td>
</tr>
<tr>
<td>B. Fortran Interface Module to link PSCAD and MATLAB</td>
<td>53</td>
</tr>
</tbody>
</table>
### FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Operating Region of the Voltage Source Inverter</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>Equivalent circuit for (a) a single PV cell and (b) a PV array of an arbitrary size</td>
<td>6</td>
</tr>
<tr>
<td>2.2</td>
<td>Circuit schematic of the power stage for the grid-connected PV system</td>
<td>7</td>
</tr>
<tr>
<td>2.3</td>
<td>Control sources–based dynamic equivalent circuit schematic</td>
<td>7</td>
</tr>
<tr>
<td>2.4</td>
<td>Control sources–based dynamic equivalent circuit schematic</td>
<td>9</td>
</tr>
<tr>
<td>4.1</td>
<td>Feeder Model Part 1</td>
<td>21</td>
</tr>
<tr>
<td>4.2</td>
<td>Feeder Model Part 2</td>
<td>21</td>
</tr>
<tr>
<td>4.3</td>
<td>PSCAD to MATLAB Interface Component</td>
<td>24</td>
</tr>
<tr>
<td>5.1</td>
<td>Real and reactive power output of the Bus3 inverter under high irradiance and zero reactive power command</td>
<td>25</td>
</tr>
<tr>
<td>5.2</td>
<td>Real power output of the Bus5 inverter under low irradiance and zero reactive power command</td>
<td>26</td>
</tr>
<tr>
<td>5.3</td>
<td>Reactive power output of the Bus5 inverter under low irradiance and zero reactive power command</td>
<td>26</td>
</tr>
<tr>
<td>5.4</td>
<td>Real and reactive power output of the Bus3 inverter with steadily increasing irradiance</td>
<td>27</td>
</tr>
<tr>
<td>5.5</td>
<td>Real and reactive power output of the Bus5 inverter with steadily decreasing irradiance</td>
<td>28</td>
</tr>
<tr>
<td>5.6</td>
<td>Reactive power output at Bus5 with steadily decreasing irradiance and reactive power control signal</td>
<td>28</td>
</tr>
<tr>
<td>5.7</td>
<td>Real and reactive loading at Bus4</td>
<td>29</td>
</tr>
<tr>
<td>5.8</td>
<td>Real and reactive loading at Bus6</td>
<td>29</td>
</tr>
</tbody>
</table>
5.9 Reactive output of the inverter at Bus$_3$ ........................................ 30
5.10 Reactive output of the inverter at Bus$_5$ ........................................ 30
5.11 Reactive power supplied by the utility at Bus$_1$ without inverter support 31
5.12 Reactive power supplied by the utility at Bus$_1$ with inverter support ... 32
# TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Summary of known and unknown quantities in the Power Flow problem.</td>
<td>11</td>
</tr>
<tr>
<td>3.2</td>
<td>Summary of known unknown, and monitored quantities in the Power Flow problem modified for PV Control.</td>
<td>18</td>
</tr>
<tr>
<td>4.1</td>
<td>Power Flow Bus Data.</td>
<td>22</td>
</tr>
<tr>
<td>4.2</td>
<td>Power Flow Branch Data.</td>
<td>22</td>
</tr>
<tr>
<td>4.3</td>
<td>Power Flow Generator Data.</td>
<td>23</td>
</tr>
</tbody>
</table>
1. Introduction

This thesis presents a reactive power control strategy for photovoltaic (PV) resources that are connected to a typical distribution feeder through a DC-AC inverter. The aim of the control strategy is to prioritize the transfer of real power, but utilize any unused inverter capacity to supply reactive power for voltage support on a radial distribution feeder. The amount of reactive power to be supplied is based on a power flow solution that assumes the entire network topology is known at every moment in time. While this was not previously practical, advancements in on-line monitoring and utility System Control and Data Acquisition (SCADA) capabilities is narrowing the theoretical gap. Furthermore, a solution based on a known system configuration can serve as a benchmark for the same problem where state estimation techniques are used in lieu of actual quantities.

This chapter discusses the potential operating modes of distributed generation supplied power via an inverter and discusses the economic incentives of both the utility and the power generator.

1.1 Reactive Supply Capabilities and Historical Limitations

In the case of photovoltaic sources, an inverter of some capacity will always be associated with the connection to the utility grid to transform the DC power supplied by the PV array to an AC output that can be connected to the utility system. The complex power rating of the inverter will likely exceed the nameplate rating of the PV system. Several maximum power point algorithms have been developed to adjust the output of the inverter to supply as much real power as possible by considering the V-I curve of the solar array in question. During periods of reduced irradiance due to cloud cover or dark periods of the day, a significant portion of the inverter capacity goes unutilized.
The ability of inverter connected sources to supply reactive power has historically been restricted by utility interconnection guidelines. This was done largely to ensure that the interconnecting generators would not be adding harmonics to the utility grid. Thus, most interconnecting parties have been required to maintain var neutrality at the point of common coupling (PCC). While the work presented in this thesis largely pertains to utility distribution systems, advancements in wind turbines are having a similar effect at the transmission level. The capability of doubly fed induction generators (DFIG’s) to dynamically supply reactive power is giving utilities the option of specifying a non-unity power factor for the injected power to help control system voltages.

Furthermore, the optimal control of any reactive power source would require online adjustment of power levels as the system operating point is constantly changing. FACTS devices on the transmission level have been implemented with real time SCADA supplied system data to optimize operation, but that level of metering and communication has historically not been available at the distribution level. Switched capacitors and inductors are most commonly controlled locally by monitoring vars at the point of common coupling or simply switched when voltage levels exceed a preset bandwidth. With the proliferation of SmartGrid driven initiatives to add SCADA to medium voltage 3 phase feeder networks, smart control of distributed generation is becoming much more commonplace.

1.2 Economic Considerations and Operating Modes

Utilizing the stranded capability of the PV resources to supply reactive power will have a direct economic benefit for the utility. Any excess inverter capacity that supplies reactive power as needed would offset the need for the utility to supply switched or fixed reactive elements on the distribution system, and could potentially save operations of voltage regulating devices such as load tap changers or voltage
regulators as they respond to low voltage conditions as power factor drifts away from unity. The distributed generation provider is typically metered at the point of common coupling strictly on a real power basis and required to maintain unity power factor at all times. While the benefits to the utility presented above have a real cost, it would be difficult if not impossible to attribute the reactive contribution of a single energy provider to avoiding a direct cost if reactive power was being supplied by multiple generators on a given feeder simultaneously. To account for this fact, the control strategy presented in later chapters will give preference to supplying the maximum amount of real power and supply reactive power only with whatever inverter capacity remains. This restriction ensures that the generation provider is compensated for all real power provided without losing any revenue for contributing to the reactive compensation of the system.

Since real and reactive power add in quadrature, the maximum reactive power that the inverter can supply \( Q_{\text{inv}} \) is given by:

\[
Q_{\text{inv}} = \sqrt{S_{\text{inv}}^2 - P_{\text{inv}}^2}
\]

(1.1)

where the \( S_{\text{inv}} \) is the MVA rating of the inverter and \( P_{\text{inv}} \) is the active power being supplied at any given moment.

Figure 1.1 represents the various operating modes of the inverter as a circle with a radius equal to \( S_{\text{inv}} \) [7].

The control strategy presented in this thesis will allow \( P_{\text{inv}} \) to be determined by irradiance, temperature, and a Maximum Power Point Tracking Algorithm (MPPT) while the optimal \( Q_{\text{inv}} \) will be first calculated by a power flow algorithm but limited to the operating regions shown in figure 1.1. The inverter should operate in the left hand plane where \( P_{\text{inv}} < 0 \) under normal operation when power is being supplied to the grid, though some power will be absorbed during the initial energization to charge the DC capacitor. Both positive and negative values of reactive power can be commanded corresponding to a capacitive or inductive effect respectively as needed.
Figure 1.1: Operating Region of the Voltage Source Inverter.

if sufficient inverter capacity is available.
2. Photovoltaic Array and Inverter Model

This chapter presents the modeling approaches for the PV array and inverter system. The final model was abstracted and used to represent independent photovoltaic generators connected to a utility distribution system. The PV array model includes inputs for temperature and irradiance to allow for real world conditions that will change the output power of each individual distributed generator. The final model was realized in PSCAD.

2.1 Model of the PV Cell and Array

Although the circuital model for a single PV cell and its generalization to a number of cells in series is well established in terms of a current source, an anti-parallel diode, a series resistance and shunt resistance [5], the model used in this thesis makes use of a modified circuit that replaces the anti-parallel diode by an external control current source as proposed in [3]. This model was explained in [6], and it is illustrated in Fig. 2.1(a). A chief advantage of this model is that it allows for including an arbitrary number of cells connected in series and/or parallel into a single circuital representation including all details of each cell.

The output current of the PV cell of Fig. 2.1(a) may be expressed as,

\[ i_{pv} = I_{irr} - I_{dio} - I_p \]  \hspace{1cm} (2.1)

where \( I_{irr} \) is the photo current or irradiance current generated when the cell is exposed to sunlight. \( I_{dio} \) is the current flowing through the anti-parallel diode and induces the nonlinear characteristics of the PV cell. \( I_p \) is a shunt current due to the shunt resistor \( R_p \) branch. Substituting relevant expressions for \( I_{dio} \) and \( I_p \),

\[ i_{pv} = I_{irr} - I_0 \left[ \exp \left( \frac{q(v_{pv} + i_{pv}R_S)}{nkT} \right) - 1 \right] - \frac{v_{pv} + i_{pv}R_S}{R_p}, \]  \hspace{1cm} (2.2)
\[ V_I = I_0 \left( \exp \left( \frac{qV}{nkT} - 1 \right) \right) + I_{\text{irr}} \]

\[ V_I = I'_0 \left( \exp \left( \frac{qV_{NS}}{nkT} - 1 \right) \right) + I'_{\text{irr}} \]

**Figure 2.1:** Equivalent circuit for (a) a single PV cell and (b) a PV array of an arbitrary size

where \( q = 1.602 \times 10^{-19} \text{ C} \) is the electron’s electric charge, \( k = 1.3806503 \times 10^{-23} \text{ J/K} \) is the Boltzmann constant, \( T \) is the temperature of the cell, \( I_0 \) is the diode saturation current or cell reverse saturation current, \( n \) is the ideality factor or the ideal constant of the diode, and \( R_S \) and \( R_P \) represent the series and shunt resistance, respectively [5]. This model can be generalized to an arbitrary number of cells connected in series, say \( N_S \), and in parallel, say \( N_P \), to form an array of \( N_S \times N_P \) cells. The generalized model is illustrated in Fig. 2.1(b).

### 2.2 Model for the Grid Connected PV System

The circuit schematic of Fig. 2.2 illustrates the power stage of the single–stage grid-connected PV power plant used in simulation. It includes the cascade connection of a dc capacitor that connects to the output terminals of an arbitrary size PV array; a two–level three–phase voltage source inverter (VSI); a LC–filter in series with a coupling transformer that connects to the ac grid.

As suggested in Fig. 2.2, the overall control is realized through a two–layer strategy. An outer layer tracks the PV array’s MPP utilizing a MPPT algorithm. An
inner loop regulates the VSI in current control mode in order to transfer the power generated by the PV array into the ac grid at an arbitrary power factor. The ability to operate at an arbitrary power factor enables the PV system to dynamically supply reactive power to the grid as commanded by the control described in Chapter 3.

2.3 Voltage Source Inverter Model

The cascaded connection of the dc capacitor, VSI, LC–filter and ac grid is modeled in the \( d - q \) reference frame using dynamic phasors. Fig. 2.3 illustrates the dynamic equivalent circuit in terms of control sources [9, 8, 4].

The state space equations for the system shown in 2.3 are given by (2.3) through (2.6).

\[
\frac{d}{dt} v_{pv} = \frac{1}{C} \left( i_{pv} - \frac{3}{2} \overrightarrow{m} \cdot \overrightarrow{i} \right)
\]  

\[ (2.3) \]
\[ \frac{d}{dt} \vec{i} = -j\omega \vec{i} + \frac{1}{L} (\vec{m}v_{pe} - (R + R_f) \vec{i} - R_f \vec{i}_g - \vec{v}_f) \] (2.4)

\[ \frac{d}{dt} \vec{v}_f = -j\omega \vec{v}_f + \frac{1}{C_f} (\vec{i} - \vec{i}_g) \] (2.5)

\[ \frac{d}{dt} \vec{i}_g = -j\omega \vec{i}_g + \frac{1}{L_g} (\vec{v}_f - (R_g + R_f) \vec{i} + R_f \vec{i}_g - \vec{v}_g) \] (2.6)

In (2.3) through (2.6), the vector quantities relate to their abc counterparts considering a generic vector quantity \( \vec{x} = x_d + jx_q \), with

\[
\begin{bmatrix}
x_d \\
x_q
\end{bmatrix}^T = T_{dq}^{abc} \begin{bmatrix}
x_a \\
x_b \\
x_c
\end{bmatrix}^T,
\] (2.7)

where \( T_{dq}^{abc} \) is the Park transformation. In (2.3), the operator “\( \cdot \)” denotes the dot product between two vectors, yielding a dc value. In (2.3) and (2.4), \( \vec{m} \) is the VSI’s modulating function.

In equations (2.4) through (2.6), \( \omega \) and \( \theta \) are the synchronizing frequency and phase angle, respectively. The synchronizing quantities are measured at the LC–filter’s output and synthesized through a phase–locked–loop (PLL). The PLL computes the grid phase angle by sensing the grid voltage and projecting the corresponding space vector on the \( d \)– and \( q \)–axis of a \( d – q \) rotating frame. This \( d – q \) frame is then rotated in such a way which ensures that the \( d \)–axis of the \( d – q \) frame is aligned with the grid voltage vector, that is, \( v_q = 0 \). In steady–state, the \( d – q \) frame rotational speed equals the grid angular frequency and the extracted angle equals to the grid voltage angle. This angle is used to synchronize the \( d – q \) reference frame for the current control.

The overall control strategy for the VSI is shown in figure 2.4. As observed in the figure, the control is broken down into four blocks: (i) a PLL utilized for
synchronization with the ac grid; (ii) a dc voltage control to track the PV array’s MPP; (iii) a power–to–current transformation; and (iv) a conventional current control to transfer the PV array’s power into the ac grid at an arbitrary power factor.

**Figure 2.4:** Control sources–based dynamic equivalent circuit schematic.

The value of $Q_{ref}$ is specified to compensate for the reactive power consumed by the L-C filter and step-up transformer if strictly real power output is desired. Any reactive power command to be injected by the inverter is added to these compensating levels.
3. Power Flow Algorithm

3.1 Introduction

The basic aim of any power/load flow calculations is to determine the voltage at every bus within a utility system and then analyze the flow of power through the grid. The results can show the required ratings of conductors connecting various generator and can be used to model future system load growth and/or system expansion effects. Load flow studies were originally performed using smaller scale analog models of utility systems called calculator boards where RLC elements were connected in the same topology as the utility grid. Advancements in computing made it possible to digitally simulate and solve the system of equations that represents the actual utility system.

The load flow problem was extended to the optimal power flow (OPF) problem by Carpentier in 1962 [1], which applies constraints to certain parameters (bus voltages or angles for stability purposes, conductor ratings) while minimizing or maximizing a desired objective function such as system losses or generation model costs.

3.2 Formulation of the Power Flow Equations

In order to solve for the voltage magnitude and angle at every bus, it is assumed that the network topology consisting of the points below are known.

1. The total number of busses in the system \( N \) representing the connection points for generators, loads, and switching stations, etc.

2. The transmission topology that connects the various busses of the system including conductor RLC properties.

3. The number of generators and their location within the system.
4. The number and size of loads served by the system and their location within the system.

The busses of the power system are further classified as follows

1. If at least one generator connected to a bus, the bus is designated as a $PV$ bus where the real power and bus voltage magnitude are considered quantities that can be specified by the generator control. The total number of $PV$ busses in the system is designated as $N_{PV}$, and the total number of generator in the system is given by $N_G$.

2. One of the generator busses is arbitrarily identified as a $slack$ bus. The bus voltage is assumed (both magnitude and angle) are assumed to be known. The slack bus voltage angle is considered to be 0 and defines the reference for all other bus voltage angles in the system designated as $\theta_N$. The power provided by the slack bus is modeled to supply system losses only.

3. If no generators are connected to a bus it is identified as a $PQ$ bus and is typically considered a system load. The real and reactive power leaving the system are assumed to be known based on load projection data. The total number of $PQ$ busses in the system is designated as $N_{PQ}$

A summary of the known and unknown quantities in the power system is given in Table 3.1

Table 3.1: Summary of known and unknown quantities in the Power Flow problem.

<table>
<thead>
<tr>
<th></th>
<th>Real Power</th>
<th>Reactive Power</th>
<th>Voltage Magnitude</th>
<th>Voltage Angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>$PQ$ bus</td>
<td>known</td>
<td>known</td>
<td>unknown</td>
<td>unknown</td>
</tr>
<tr>
<td>$PV$ bus</td>
<td>controlled</td>
<td>unknown</td>
<td>controlled</td>
<td>unknown</td>
</tr>
<tr>
<td>$slack$ bus</td>
<td>unknown</td>
<td>unknown</td>
<td>controlled</td>
<td>known</td>
</tr>
</tbody>
</table>
The unknown and controlled quantities of interest are represented by the variable vector $X$.

$$X = \begin{pmatrix} \theta_N - 1 \\ |V_N| \\ P_{NG} \\ Q_{NG} \end{pmatrix} \quad (3.1)$$

The dimensions of $X$ is equal to $2(N + N_G) - 1$. In order to find a unique solution, we will need to write an equal number of equations that include the elements of $X$ and other known quantities. The basis for the system of equations will be the complex power injected at every bus, $S_{bus}$, which includes both a real and imaginary part for each element. $S_{bus}$ can be defined as:

$$S_{bus} = V_{bus}I_{bus}^* \quad (3.2)$$

or rewritten as:

$$S_{bus} = V_{bus}(Y_{bus}V_{bus})^* \quad (3.3)$$

where $Y_{bus}$ is an admittance matrix between all the buses of the system and takes the form:
\[ Y_{bus} = \begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{nn} \end{pmatrix} \]  

(3.4)

\( Y_{bus} \) contains the topology of the system and the electrical properties of each transmission line or branch. \( Y_{bus} \) is of size \( N \times N \) and has the following properties.

- \( Y_{bus} \) is symmetrical.

- The diagonal elements, \( Y_{ii} \), are equal to the sum of the admittances of all elements connected to the \( i \)th node.

- The off-diagonal elements, \( Y_{ij} \), are equal to the negative of the admittance connecting nodes \( i \) and \( j \).

The \( i \)th element of \( S_{bus} \) can be written as

\[ S_{bus_i} = V_{bus_i} \left[ \sum_{j=1}^{n} (y_{ij}V_{bus_j}) \right]^* \quad i, j = 1, 2, \ldots, n \]  

(3.5)

or

\[ S_{bus_i} = \sum_{j=1}^{n} (V_{bus_i}V_{bus_j}^* y_{ij}^*) \quad i, j = 1, 2, \ldots, n \]  

(3.6)

and finally as

\[ S_{bus_i} = \sum_{j=1}^{n} \left[ |V_i||V_j|e^{j\delta_{ij}}(G_{ij} - jB_{ij}) \right] \quad i, j = 1, 2, \ldots, n \]  

(3.7)

where:
\[ Y_{ij} = G_{ij} + jB_{ij} \quad \theta_{ij} = \theta_i - \theta_j \quad i, j = 1, 2, ..., n \quad (3.8) \]

Equation (3.7) gives an expression for the complex power injected at every bus within the system and can be separated into real and imaginary parts by applying Euler’s formula and grouping like terms as show in equations (3.9) through (3.12)

\[ S_{bus_i} = \sum_{j=1}^{n} [||V_i||V_j| \left( \cos \theta_{ij} + j \sin \theta_{ij} \right) \left( G_{ij} - jB_{ij} \right)] \quad i, j = 1, 2, ..., n \quad (3.9) \]

\[ S_{bus_i} = \sum_{j=1}^{n} [||V_i||V_j| \left( G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij} + jG_{ij} \sin \theta_{ij} - jB_{ij} \cos \theta_{ij} \right)] \quad (3.10) \]

\[ P_{bus_i} = \sum_{j=1}^{n} [||V_i||V_j| \left( G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij} \right)] \quad i, j = 1, 2, ..., n \quad (3.11) \]

\[ Q_{bus_i} = \sum_{j=1}^{n} [||V_i||V_j| \left( G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij} \right)] \quad i, j = 1, 2, ..., n \quad (3.12) \]

where:

- \( P_{bus_i} \): The active power injection in bus \( i \).
- \( Q_{bus_i} \): The reactive power injection in bus \( i \).

The final system of equations is written by observing that all complex power into the system must be equal to the complex power out of the system plus losses to preserve energy balance. Since the slack bus accounts for losses, equations for real and reactive power at all other busses can be written as a function of the unknown vector \( X \) as follows:

\[ P_{bus_i} = P_{gi} - P_{Li} = f(V, \theta) \quad Q_{bus_i} = Q_{gi} - Q_{Li} = f(V, \theta) \quad (3.13) \]
where:

\( P_{gi} \): The active power of the generator in the bus \( i \).

\( Q_{gi} \): The reactive power of the generator in the bus \( i \).

\( P_{Li} \): The active power of the load in the bus \( i \).

\( Q_{Li} \): The reactive power of the load in the bus \( i \).

With the formulation of the power flow equations complete, the system can be solved iteratively to determine the load flows in the system. Applying constraints on the variable vector \( X \) pertaining to system stability and capacity will limit the results to a practical operating condition. Typical system constraints include:

1. Limitations on bus voltage magnitude, \( |V_{\text{min}}| \leq |V_i| \leq |V_{\text{max}}| \), to ensure safe and effective power delivery

2. Limitations on voltage angle, \( \theta_{\text{min}} \leq \theta_i \leq \theta_{\text{max}} \), to ensure system stability

3. Limitations on branch power flow to less than the ratings of the circuit, \( S_{ij} \leq S_{ij\text{Rating}} \), to prevent overloading.

The final step of the formulation is to impose an optional objective function to minimize or maximize in order to achieve an optimal power flow from the solution set. In the case of this simulation, the objective will be to minimize system losses. In larger transmission networks, the objective function is usually representative of the cost of dispatching the generators on the system, which, when minimized, results in the dispatching of lowest cost that supplies the network.

### 3.3 Modified Formulation for Reactive Power Control of PV Resources

The formulation of the power flow equations for the work presented in this thesis is a special case of the concepts presented in the previous section, but the fundamental concepts remain the same. This section discusses the differences in the formulation of the power flow equations. The system constrains imposed on the solution and overall
objective function for optimal control are also presented. The final set of equations were programmed into MATLAB, and the complete code can be found in Appendix A.

3.3.1 Formulation of the Power flow equations

As stated previously, the electrical system of interested for the simulation presented in this thesis is a heavily loaded radial distribution feeder with multiple PV distributed generators in operation. The primary objectives of the control algorithm are as follows.

1. Supply real power from the PV generators whenever possible based on instantaneous irradiance and temperature conditions.

2. Supply reactive power only when inverter capacity is available.

3. The amount of reactive power to be supplied will be determined by the optimal power flow solution, but not to exceed the capacity of the inverter.

With these objectives in mind, it is clear that the system bus where the PV generators connect cannot be treated as a traditonal $PV$ bus in the formulation of the power flow equations since the real power injected, $P_{gi}$, depends on temperature and irradiance and cannot be externally commanded. In the special case of a radial distribution feeder, the only true $PV$ bus is the substation bus that feeds the entire system. The vector of unknown quantities defined in equation (3.1) can be modified as shown in (3.14). The dimensions of $X_0$ is consequently reduced to $2(N + 1) + N_G - 1$. The active power injection of every PV generator becomes an input to the power flow algorithm as opposed to a quantity to be solved for.
\[ X_0 = \begin{pmatrix} \theta_N - 1 \\ |V_N| \\ P_{\text{Substation}} \\ Q_{NG} \end{pmatrix} \] (3.14)

It is worth noting that this approach will require the active power output of the PV generator to be monitored constantly and relayed via a SCADA system to enable online computation of the reactive power command. The instantaneous real power output of the PV generator is likely to be available locally as part of the inverter control and monitoring system, but a SCADA link for distributed generators is usually only required for larger installations to address islanding and synchronization concerns.

Another key distinction between the classic power flow formulation and the special case presented is this thesis is the need to respond quickly to changes in load. While load profiles at a transmission level may be very predictable based on historical load profiles, weather, and time of year, distribution feeder loads can vary quickly as commercial and residential customers change electric consumption. In order to calculate the optimal amount of reactive power to command from the PV generators, the real and reactive power consumption at every load bus will need to be monitored and supplied to the control algorithm online. The final list of quantities that will have to be monitored online and supplied to the reactive power controller as inputs is as follows.

1. The real power generated by every PV distributed generator. Assuming there are no power sources other than PV distributed generator and the utility, the total number of inputs will equal \( N_{PV} - 1 \)

2. The real power consumed at every \( PQ \) load bus, which implies \( N_{PQ} \) signals.
3. The reactive power consumed at every $PQ$ load bus, which implies $N_{PQ}$ signals.

As a result of the need to monitor the above mentioned quantities online, table 3.1 can be modified as shown in table 3.2 to re-designate some of the quantities previously considered as known.

**Table 3.2:** Summary of known unknown, and monitored quantities in the Power Flow problem modified for PV Control.

<table>
<thead>
<tr>
<th></th>
<th>Real Power</th>
<th>Reactive Power</th>
<th>Voltage Magnitude</th>
<th>Voltage Angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>$PQ$ bus</td>
<td>monitored</td>
<td>monitored</td>
<td>unknown</td>
<td>unknown</td>
</tr>
<tr>
<td>$PV_{Substation}$ bus</td>
<td>controlled</td>
<td>unknown</td>
<td>controlled</td>
<td>unknown</td>
</tr>
<tr>
<td>$PV_{Inverter}$ bus</td>
<td>monitored</td>
<td>unknown</td>
<td>controlled</td>
<td>unknown</td>
</tr>
<tr>
<td>slack bus</td>
<td>unknown</td>
<td>unknown</td>
<td>controlled</td>
<td>known</td>
</tr>
</tbody>
</table>

Online feeder monitoring is a key aim of SmartGrid topics and is becoming much more prevalent. One of the challenges of real world implementation is the distributed nature of feeder loads as opposed to discrete load busses. Several state estimation techniques for lumped equivalents have been developed, but the aim of this thesis is to provide an optimal benchmark assuming all of the electrical data is known.

### 3.3.2 Objective Function and System Constraints

The objective function chosen for the simulation presented in this thesis is minimization of the system losses. Given that the simulated system is a radial distribution feeder, all complex power is delivered to the system from the utility at the substation feeder exit plus the contribution of the distributed generators. The active power contribution of the distributed generators take priority over the reactive output and is controlled by a MPPT algorithm. Since the utility has to supply all load not served by the distributed generators, minimization of the transmission losses will provide the most cost effective operating point.
The system constrains imposed on the solution to ensure a realistic solution are as follows:

1. Limitations on bus voltage magnitude in per unit, \( 0.95|V_{\text{nominal}}| \leq |V_i| \leq 1.05|V_{\text{nominal}}| \).

2. Limitations on voltage angle, \(-50^\circ \leq \theta_i \leq 50^\circ\)

3. Limitations on the complex power supplied by the inverter, \( S_{\text{inv}} = 500\, \text{kVA} \)

As previously discussed, the inverter’s complex power rating \( S_{\text{inv}} \) is the quadrature addition of the real and reactive power supplied by the inverter. The real power component is controlled strictly by the maximum power point tracking algorithm as a function of temperature and irradiance. The maximum reactive power component is computed according to equation (1.1) where \( S_{\text{inv}} \) is limited to 500 kVAR and \( P_{\text{inv}} \) is read as an input to the controller.
4. Power System Simulation

In order to test the reactive power control algorithm’s ability to respond to online changes in system conditions and commands, a model of a distribution feeder was developed in PSCAD. The PV array and voltage source inverter model presented in Chapter 2 were abstracted as PSCAD components and applied to the simulated distribution feeder as distributed generators. A PSCAD component was developed to pass the online monitored signals to the MATLAB power flow algorithm and also to return the reactive power commands. Finally, some timed breaker logic was added to demonstrate the controller’s ability to deal with changes in system loading.

4.1 Distribution Feeder Model

As previously stated, the scope of the simulation presented in this thesis is a radial medium voltage distribution feeder. The model for the feeder is shown in 4.1 and 4.2. Note that the model has been split into two parts for clarity. Bus\(_3\) is shown in both figures.

The three phase system is assumed to be balanced and does not include lateral ties to other distribution feeders. The substation feeder exit is shown connected to Bus\(_1\), the PV distributed generators are shown connected at Bus\(_3\) and Bus\(_1\). Lumped loads are shown connected between all of the power sources at Bus\(_2\), Bus\(_4\), and at the end of the feeder at Bus\(_6\). A branch impedance between all busses of 0.1 + j0.05 ohms is included as well.

In order to simulate typical utility voltage regulation, the voltage source connected at Bus\(_1\) that simulates the substation exit is modeled as a controlled voltage source. The magnitude of the starting feeder voltage is controlled by the power flow algorithm within a tolerance of +/- 5%, which is the ANSI limit for utility voltage tolerance.
Distribution voltage regulation is typically achieved by the mechanical operation of a load tap changer or external voltage regulator on the secondary side of the substation.
power transformer.

In order to supply the reactive power control algorithm with the static known system parameters, a text file was developed that organizes the system parameters in the IEEE standard format for power flow data [2]. The data of interest for the simulated feeder is shown in tables 4.1 through 4.3.

**Table 4.1:** Power Flow Bus Data.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Voltage (V)</th>
<th>MW Gen</th>
<th>Mvar Gen</th>
<th>MW Load</th>
<th>Mvar Load</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bus1</td>
<td>Slack</td>
<td>12470</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Bus2</td>
<td>PQ</td>
<td>12470</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>1.7</td>
</tr>
<tr>
<td>Bus3</td>
<td>PV</td>
<td>12470</td>
<td>0</td>
<td>0.4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Bus4</td>
<td>PQ</td>
<td>12470</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>1.0</td>
</tr>
<tr>
<td>Bus5</td>
<td>PV</td>
<td>12470</td>
<td>0</td>
<td>0.4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Bus6</td>
<td>PQ</td>
<td>12470</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>1.7</td>
</tr>
</tbody>
</table>

**Table 4.2:** Power Flow Branch Data.

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Resistance (ohms)</th>
<th>Reactance (ohms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bus1</td>
<td>Bus2</td>
<td>0.1</td>
<td>0.05</td>
</tr>
<tr>
<td>Bus2</td>
<td>Bus3</td>
<td>0.1</td>
<td>0.05</td>
</tr>
<tr>
<td>Bus3</td>
<td>Bus4</td>
<td>0.1</td>
<td>0.05</td>
</tr>
<tr>
<td>Bus4</td>
<td>Bus5</td>
<td>0.1</td>
<td>0.05</td>
</tr>
<tr>
<td>Bus5</td>
<td>Bus6</td>
<td>0.1</td>
<td>0.05</td>
</tr>
</tbody>
</table>

**4.2 Control System Interface**

While the PV Array, voltage source inverter, and the feeder itself are all simulated in PSCAD, the power flow equations and iterative solution are performed in MATLAB. A custom component in PSCAD was developed to handle the exchange
of data between the two programs. As outlined in the previous chapter, the following
data has to be monitored online and fed to the controller as an input.

1. Supply real power from the PV generators whenever possible based on instantaneous irradiance and temperature conditions.

2. Supply reactive power only when inverter capacity is available.

3. The amount of reactive power to be supplied will be determined by the optimal power flow solution, but not to exceed the capacity of the inverter.

The output of the MATLAB power flow algorithm is the reactive power commands to each voltage source inverter, which is also passed through the interface component. The graphical representation of the interface as applied to the distribution feeder model is shown in 4.3.

The FORTRAN code that governs the components data transfer is shown in Appendix B. Note that a triggering input is included in the component that governs how frequently MATLAB will execute the control algorithm. For the purposes of the test cases presented in this thesis, a period of 0.25 seconds was selected.

### 4.3 Timed Breaker Logic

In order to demonstrate the controller’s ability to respond appropriately to changes in feeder load, timed breaker logic was added to the PQ busses, *Bus*$_4$ and *Bus*$_6$ as shown in figure 4.2.
The breaker logic is intended to change the load state quickly in between executions of the reactive power control algorithm so that the subsequent reactive power command can be evaluated. The details of each breaker operation will be detailed for each test case in the next chapter.

Figure 4.3: PSCAD to MATLAB Interface Component
5. Test Cases and Simulation Results

The models for the PV array, voltage source inverter, and distribution feeder were modeled in PSCAD and verified by simulation. The behavior of the MATLAB power flow algorithm can also be observed by considering the distribution feeder model, the input parameters that are monitored online, the reactive power commands issued from the controller, and the final system response. Several test cases are presented in this chapter in order of least to most complex.

5.1 Test Case - Constant Reactive Power Output

In order to first verify the simulation of the PV array and voltage source inverter, the first case presented is simply a constant reactive power command. A case of particular interest is operation at unity power factor. A constant reactive power command of zero is supplied to the voltage source inverters connected at Bus$_3$ and Bus$_5$ for the system shown in figures 4.1 and 4.2. Additionally, the PV array at Bus$_3$ will be supplied with a large irradiance value to simulate full sun exposure, while the PV array at Bus$_5$ will be supplied with a minimal irradiance value.

![Figure 5.1: Real and reactive power output of the Bus$_3$ inverter under high irradiance and zero reactive power command](image-url)
Figure 5.2: Real power output of the $Bus_5$ inverter under low irradiance and zero reactive power command

Figure 5.3: Reactive power output of the $Bus_5$ inverter under low irradiance and zero reactive power command

We can observe that the real power output of the PV Generator at $Bus_3$ that was subjected to a high irradiance value is nearly equal to the maximum inverter rating $S_{inv}$ of 500 kVA, while the real power output of the the PV Generator at $Bus_5$ was essentially zero. For both cases, the reactive power output was essentially zero. Both results confirm that the invert model’s real power output is dependant on irradiance, and a desired reactive power value can be successfully commanded from the MATLAB algorithm.

5.2 Test Case - Variable Irradiance

Another test case of interest is the real and reactive power response of the system to steadily increasing or decreasing irradiance while the feeder is heavily loaded. The goal of this simulation is to demonstrate the following:
1. Real power supplied by the PV array is directly proportional to irradiance.

2. The magnitude of reactive power supplied by the PV array is indirectly proportional to irradiance.

3. The magnitude of the reactive power supplied is limited by the inverter rating $S_{inv}$ as expressed in equation (1.1)

4. The system is able to respond to the quantities monitored online and subsequent changing reactive power commands.

The simulations results are presented in figures 5.4 through 5.6

![Figure 5.4: Real and reactive power output of the Bus3 inverter with steadily increasing irradiance](image)

We can see in figure 5.4 the expected response to steadily increasing irradiance, and the quadrature relationship between $P_{inv}$ and $Q_{inv}$. The same effect is even more pronounced in figure 5.5 where real power quickly drops off as the applied irradiance decreases.

The online tracking of input parameters and response to changing reactive power commands is illustrated in 5.6, which plots the reactive power command issued from the MATLAB control algorithm on top of the measured reactive power output.

We can clearly see that the MATLAB interface component is executing the reactive power control algorithm and providing an updated control signal every 0.25 seconds as expected and the subsequent response of the inverter control.
Figure 5.5: Real and reactive power output of the Bus5 inverter with steadily decreasing irradiance

Figure 5.6: Reactive power output at Bus5 with steadily decreasing irradiance and reactive power control signal

5.3 Test Case - Variable Load

The next test case presented is intended to highlight the controller’s ability to respond to changes in load. Timed breaker logic is applied to the loads connected at Bus4 and Bus6 to simulate capacitor switching and abrupt changes in load that can occur on a distribution system due to customer interaction or protective equipment operation. The timed breaker operations are designed to impose the following conditions.

- The initial load at Bus4 is 3 MW and 1.7 Mvar inductive.
- The initial load at Bus6 is 3 MW and 1.7 Mvar inductive.
- At $t = 0.6sec$, the load at Bus6 will quickly change to 3 MW and 1.2 Mvar capacitive to simulate a capacitor being switched on.
• At $t = 1.1sec$, the load at $Bus_4$ will quickly change to 1 MW and 0 MVar to simulate a fuse operation.

The loading at $Bus_4$ and $Bus_6$ over the simulation period are shown in figures 5.7 and 5.8.

**Figure 5.7:** Real and reactive loading at $Bus_4$.

**Figure 5.8:** Real and reactive loading at $Bus_6$.

The irradiance applied to both inverters will be held at practically zero to enable the inverter to supply reactive power almost completely up to the inverter rating $S_{inv} = 500kVA$.

The system response is shown in figures 5.9 and 5.10.

The following observations can be made at each 0.25 second interval when the controller recalculates the reactive power commands:
Figure 5.9: Reactive output of the inverter at Bus3

Figure 5.10: Reactive output of the inverter at Bus5

- At $t = 0.25sec$ and $t = 0.5sec$, both of the inverters at Bus3 and Bus5 are supplying maximum reactive power due to the heavy loading at nearby busses.

- At $t = 0.75sec$, the control algorithm is called for the first time since the load at Bus6 has changed significantly from the previous state. Less reactive power is required to fully compensate the load and line loss in the rest of the feeder from Bus6.

- There is no significant change of state at $t = 1.0sec$.

- At $t = 1.25sec$, the control algorithm is called for the first time since the load at Bus4 has changed significantly from the previous state. The load at Bus4
has dropped to the point that minimal reactive power is required at $Bus_3$, and an excess of reactive power at the of the feeder has caused the inverter at $Bus_5$ to enter the inductive operating range and start absorbing vars.

5.4 Measurement of Objective Function

In order to demonstrate that the objective function is being probably realized, a final test cases is presented. First, if the reactive power commands issued to the inverter are designed to minimize system losses, any load demand for reactive power should be passed directly to the inverter and supplied up to the inverter’s capacity. To illustrate this, consider the same circuit shown in 4.1 and 4.2, without any timed breaker operations and a low value of irradiance applied to every PV system. The intent is to model a heavily loaded figure with the distributed generators able to supply reactive power only. As a baseline, the reactive power supplied by the modeled utility system a $Bus_1$ is shown in figure 5.11.

![Test Case 4 - Unity Power Factor](image)

**Figure 5.11:** Reactive power supplied by the utility at $Bus_1$ without inverter support

When the reactive power control algorithm is enabled, the reactive power supplied by the system is shown in figure 5.12.

As expected, we can observe roughly a 1 Mvar reduction in the amount of reactive power that the utility system has to supply.
Figure 5.12: Reactive power supplied by the utility at $Bus_1$ with inverter support
6. Conclusion

This thesis has presented a comprehensive model of a distribution feeder with multiple independent PV distributed generators, and a control strategy for the associated voltage source inverters that is capable of providing reactive compensation to the utility system without compromising the amount of real power generated by the PV systems. In other words, the control strategy takes advantage of any available inverter capacity to help regulate the system without impacting the privately owned distributed generator.

The challenges in realizing this system in practice is chiefly the required online monitoring and adjustment of real power injection from the various generators and load conditions. Prior to the implementation of cellular communications and the proliferation of fiber optic communication circuits, it would not have been possible to achieve this level of control. Fiber optics are being increasingly installed on utility circuits either as part of the neutral conductor or separately strung on the same overhead poles as the primary conductors as part of the SmartGrid initiative. Even with the advances in communication capabilities on utility systems, the ability to monitor load conditions comprehensively and instantaneously, especially at the distribution level, will likely remain challenging.

The results presented in this thesis could be used as a benchmark for future work on similar systems that assume less than perfect knowledge of the utility system at every moment in time and apply state estimation techniques to extrapolate the unknown quantities from particular data points. Additional functionality could also be added to the reactive power control algorithm to deal with interruptions to the flow of online input parameters when loss of communication occurs. Since the voltage source inverter is capable of generating waveforms above the power frequency, control
could be added to provide active filtering of harmonics introduced by non-linear loads as well. This is readily achievable since the control of the inverter is already performed in the $d-q$ reference frame. Finally, more sophisticated modeling of the distribution feeder itself could lead to more interesting test cases.
REFERENCES


Appendix A. MATLAB Reactive Power Control Function

The C language code for the reactive power flow algorithm developed in MATLAB is presented in this appendix. The code is organized as the main function called by PSCAD during the simulation and also includes the referenced sub functions.

A.1 Main Function - QCTL

%Output QCMD - reactive power commands.
%Input PG - real power output of each inverter,
%Input PL - real power consumed at each PQ Bus
%Input QL - reactive power consumed at each PQ Bus

function [Qcmd, Vs] =Qctl(PG, PL, QL)

%Initial settings

j = sqrt(-1);

S.base.S=1000000;
S.base.Vg=12470;
S.base.Zg=S.base.Vg^2/S.base.S;
S.base.Ig=S.base.S/S.base.Vg/sqrt(3);
S.base.Vi=270;
S.base.Ii=S.base.S/S.base.Vi/sqrt(3);

%
% This initializes S
S.fname=' ';

% This calls the function read_cf
S=read_cf(S);

% save Sfile

% Update input values monitored online.
S.Bus.MWGen(S.Bus.PV) = PG;
S.Bus.MWLoad(S.Bus.PQ(2:length(S.Bus.PQ))) = PL;
S.Bus.MVARLoad(S.Bus.PQ(2:length(S.Bus.PQ))) = QL;

% Build the Y-bus matrix
S=build_ybus(S);

% Variables:
% 1) Angles at every bus (excluding slack)
% 2) Voltages at every bus
% 4) Active Power at substation
% 5) Reactive Power at every generator (PV + substation)

na=S.Bus.n-1;
nv=S.Bus.n;
np=1;
npq=length(S.Gen.Bus);
tetamin=-50;
tetamax=50;
Vmin=.95;
Vmax=1.05;

%lower bound
LB=[tetamin*ones(na,1)*pi/180; Vmin*ones(nv,1); -inf; -inf; -ones(nq-1,1)];
%upper bound
UB=[tetamax*ones(na,1)*pi/180; Vmax*ones(nv,1); inf; inf; ones(nq-1,1)];

%Initial condition
%x0=[zeros(na,1); ones(nv,1); 0.5*(S.Gen.Pmin+S.Gen.Pmax)/100; 0.5*(S.Gen.Qmin+S.Gen.Qmax)/100]
x0=[zeros(na,1); ones(nv,1); 14000000/S.base.S; 1/S.base*S*[3000000; 500000; 500000];

A=[];
b=[];
Aeq=[];
beq=[];

%Defines optimset
OptOpf = optimset('Display','iter','Diagnostics','on','DerivateCheck','on',...}
'LargeScale','off','GradConstr','off','GradObj','off','Hessian','off',...}
'MaxIter', 5000, 'MaxFunEvals', 100000);

[x,fval,exitflag,output,lambda] = fmincon(@ObjFun,x0,A,b,Aeq,beq,LB,UB,@NonLinCon,OptOpf,S);

%compute number of each variable
na=S.Bus.n-1;

nv=S.Bus.n;

np=1;

nq=length(S.Gen.Bus);

%Updates Voltages
Vbus=x(na+1:na+nv); %added
i_av=find(S.Bus.type~=3);

Vbus(i_av)=x(na+1:na+nv);
Vbus(S.Bus.Swing)=Vbus(S.Bus.Swing)*exp(j*0);
Vbus(i_av)=abs(Vbus(i_av)).*exp(j*x(1:na));

Sinj=Vbus.*conj(S.Ybus*Vbus);

Qcmd = x(na+nv+np+2:length(x))';

Vs = x(na+1);

end

A.2 Sub Function - readCF
The intent of this sub function is to build a MATLAB structure with all of the static bus and branch data of the power system needed to build the power flow equations.

```matlab
function S=read_cf(S)

% gets the name of the file
fname='JRTFeeder.cf';
S.fname=fname;

% opens the file for reading
fcf=fopen(fname,'r');

% gets the name of the case
s=fgetl(fcf);
while strcmp(s,''), s=fgetl(fcf); end;
S.CaseName=s;

% reads bus data
while strcmp(s(1:min(3,length(s))),'BUS')~=1, % Find the start of bus data
    s=fgetl(fcf);
end
s=fgetl(fcf);
while strcmp(s,''), s=fgetl(fcf); end;
```

40
S.Bus.number=[];
S.Bus.name=[];
S.Bus.area=[];
S.Bus.zone=[];
S.Bus.type=[];
S.Bus.V=[];
S.Bus.angle=[];
S.Bus.MWLoad=[];
S.Bus.MVARLoad=[];
S.Bus.MWGen=[];
S.Bus.MVARGen=[];

%Extra stuff added
S.Bus.BasekV=[];
S.Bus.DesiredVolt=[];
S.Bus.MVarMax=[];
S.Bus.MVarMin=[];
S.Bus.G=[];
S.Bus.B=[];

while s(1)~='-
    k2=1;
    [k1 k2]=find_k1k2(s,k2);
n_aux=str2num(s(k1:k2));
S.Bus.number(n_aux,1)=n_aux;

[k1 k2]=find_k1k2(s,k2);
S.Bus.name=strvcat(S.Bus.name,s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Bus.area(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Bus.zone(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Bus.type(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Bus.V(n_aux,1)=str2num(s(k1:k2))/S.base.Vg;

[k1 k2]=find_k1k2(s,k2);
S.Bus.angle(n_aux,1)=str2num(s(k1:k2))*pi/180;

[k1 k2]=find_k1k2(s,k2);
S.Bus.MWLoad(n_aux,1)=str2num(s(k1:k2))/S.base.S;

[k1 k2]=find_k1k2(s,k2);
S.Bus.MVARLoad(n_aux,1)=str2num(s(k1:k2))/S.base.S;

[k1 k2]=find_k1k2(s,k2);
S.Bus.MWGen(n_aux,1)=str2num(s(k1:k2))/S.base.S;

[k1 k2]=find_k1k2(s,k2);
S.Bus.MVARGen(n_aux,1)=str2num(s(k1:k2))/S.base.S;

[k1 k2]=find_k1k2(s,k2);
S.Bus.BasekV(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Bus.DesiredVolt(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Bus.MVarMax(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Bus.MVarMin(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Bus.G(n_aux,1)=str2num(s(k1:k2));
\[ [k1 \ k2] = \text{find}_{k1k2}(s, k2); \]
\[
S.\text{Bus.B(n\_aux, 1)} = \text{str2num}(s(k1:k2));
\]
\[
s = \text{fgetl}(fcf);
\]
\[
\text{end}
\]

\% Find indices for various types of buses
\[
S.\text{Bus.PQ} = \text{find}(S.\text{Bus.type} == 0);
\]
\[
S.\text{Bus.PV} = \text{find}(S.\text{Bus.type} == 2);
\]
\[
S.\text{Bus.Swing} = \text{find}(S.\text{Bus.type} == 3);
\]
\[
S.\text{Bus.n} = \text{max}(S.\text{Bus.number});
\]

\% Reads branch data

\[
\text{while strcmp(s(1:min(6, length(s))), 'BRANCH') \neq 1, \% Find the start of bus data}
\]
\[
s = \text{fgetl}(fcf);
\]
\[
\text{end}
\]
\[
s = \text{fgetl}(fcf);
\]
\[
\text{while strcmp(s,'''), s = fgetl(fcf); end;}
\]

\[
S.\text{Branch.From} = [];
\]
\[
S.\text{Branch.To} = [];
\]
\[
S.\text{Branch.area} = [];
\]
\[
S.\text{Branch.zone} = [];
\]
S.Branch.circuit=[];
S.Branch.type=[];
S.Branch.r=[];
S.Branch.x=[];
S.Branch.b=[];
S.Branch.MVA1=[];

n_aux=0;
while s(1)~='-''

n_aux=n_aux+1;
k2=1;

[k1 k2]=find_k1k2(s,k2);
S.Branch.From(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Branch.To(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Branch.area(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Branch.zone(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Branch.circuit(n_aux,1)=str2num(s(k1:k2));
[k1 k2]=find_k1k2(s,k2);
S.Branch.type(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Branch.r(n_aux,1)=str2num(s(k1:k2))/S.base.Zg;

[k1 k2]=find_k1k2(s,k2);
S.Branch.x(n_aux,1)=str2num(s(k1:k2))/S.base.Zg;

[k1 k2]=find_k1k2(s,k2);
S.Branch.b(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Branch.MVA1(n_aux,1)=str2num(s(k1:k2))/S.base.S;

s=fgetl(fcf);
end
S.Branch.n=n_aux;

while strcmp(s(1:min(9,length(s))),'GENERATOR')~=1, % Find the start of generator data
    s=fgetl(fcf);
end
s=fgetl(fcf);
while strcmp(s,''), s=fgetl(fcf); end;
if strcmp(s(1),'('), s=fgetl(fcf); end;
while strcmp(s,''), s=fgetl(fcf); end;

S.Gen.Bus=[];
S.Gen.Pmax=[];
S.Gen.Pmin=[];
S.Gen.Qmax=[];
S.Gen.Qmin=[];
S.Gen.S=[];
S.Gen.a=[];
S.Gen.b=[];
S.Gen.c=[];

n_aux=0;
while s(1)~='-'
    n_aux=n_aux+1;
    k2=1;

    [k1 k2]=find_k1k2(s,k2);
    S.Gen.Bus(n_aux,1)=str2num(s(k1:k2));

    [k1 k2]=find_k1k2(s,k2);
    S.Gen.Pmax(n_aux,1)=str2num(s(k1:k2));

47
[k1 k2]=find_k1k2(s,k2);
S.Gen.Pmin(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Gen.Qmax(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Gen.Qmin(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Gen.S(n_aux,1)=str2num(s(k1:k2))/S.base.S;

[k1 k2]=find_k1k2(s,k2);
S.Gen.a(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Gen.b(n_aux,1)=str2num(s(k1:k2));

[k1 k2]=find_k1k2(s,k2);
S.Gen.c(n_aux,1)=str2num(s(k1:k2));

s=fgetl(fcf);

dend

S.Gen.n=n_aux;
fclose(fcf);

A.2.1 Sub Function - findk1k2

The intent of this helper function is to find the start and stop coordinates of numerical values in a text file in order to avoid problems with excessive white space.

function [k1 k2]=find_k1k2(s,k2);
    k1=k2+1;
    while s(k1)==’ ’, k1=k1+1; end
    k2=k1;
    while s(k2)~=' ' 
        if k2==length(s)
            break
        end
        k2=k2+1;
    end

A.3 Sub Function - Build Ybus

This subfunction builds the sparse \textit{Ybus} Matrix.

function S=build_ybus(S)

\%Build y-bus

\%fills the diagonal elements from shunt compensators
S.Ybus=sparse(diag(S.Bus.G+j*S.Bus.B));

\%fills diagonal elements from transmission lines and transformers
S.Ybus=S.Ybus+sparse(S.Branch.From,S.Branch.From,...
1./(S.Branch.r+j*S.Branch.x)+j*S.Branch.b/2,...
S.Bus.n,S.Bus.n);

S.Ybus=S.Ybus+sparse(S.Branch.To,S.Branch.To, ...
  1./(S.Branch.r+j*S.Branch.x)+j*S.Branch.b/2, ...
  S.Bus.n,S.Bus.n);

%fills off diagonal elements
S.Ybus=S.Ybus+sparse(S.Branch.From,S.Branch.To, ...
  -1./(S.Branch.r+j*S.Branch.x), ...
  S.Bus.n,S.Bus.n);

S.Ybus=S.Ybus+sparse(S.Branch.To,S.Branch.From, ...
  -1./(S.Branch.r+j*S.Branch.x), ...
  S.Bus.n,S.Bus.n);

A.4 Sub Function - ObjFun

This subfunction defines the objective function to be minimized or maximized by MATLAB’s built-in Fmincon function.

function f=ObjFun(x,S)

na=S.Bus.n-1;
nv=S.Bus.n-1;
np=1;
nq=length(S.Gen.Bus);
\[ f = x(na + nv + 1) + \text{sum}(S.\text{Bus}.\text{MWGen}(S.\text{Bus}.PV)) - \text{sum}(S.\text{Bus}.\text{MWLoad}(S.\text{Bus}.PQ)) \]

### A.5 Sub Function - NonLinCon

This subfunction defines the non-linear equality and inequality constraints for use with MATLAB's built-in Fmincon function.

```matlab
function [C, Ceq] = NonLinCon(x, S)

% compute number of each variable
na = S.Bus.n - 1;

nv = S.Bus.n;  \% took out -1
np = 1;
nq = length(S.Gen.Bus);

% Updates Voltages
Vbus = S.Bus.V;
Vbus = x(na + 1:na + nv); \% added
Vbus(S.Bus.Swing) = Vbus(S.Bus.Swing) * exp(j*0);
i_av = find(S.Bus.type ~= 3);
% Vbus(i_av) = x(na + 1:na + nv);
Vbus(i_av) = abs(Vbus(i_av)) .* exp(j * x(1:na));

% Lines' Flow
Sfrom = Vbus(S.Branch.From) .* conj((Vbus(S.Branch.From) - Vbus(S.Branch.To)) ./ (S.Branch.r + j * S.Branch.x) + Vbus(S.Branch.From) .* (j * S.Branch.b/2));

Sto = Vbus(S.Branch.To) .* conj((Vbus(S.Branch.To) - Vbus(S.Branch.From)) ./ (S.Branch.r + j * S.Branch.x) + Vbus(S.Branch.To) .* (j * S.Branch.b/2));
```

51
Sinj=Vbus.*conj(S.Ybus*Vbus);

%update injected power
Pinj=-S.Bus.MWLoad;
Qinj=-S.Bus.MVARLoad;
Pinj(S.Bus.PV)=S.Bus.MWGen(S.Bus.PV);
Pinj(S.Bus.Swing)=x(na+nv+1);
Qinj(S.Gen.Bus)=x(na+nv+np+1:na+nv+np+nq);
Sinj=Pinj+j*Qinj;

Rate=S.Branch.MVA1;

%Inequality Constraints
C=[abs(Sfrom)-Rate; abs(Sto)-Rate; sqrt( real(Sinj(S.Gen.Bus)).^2+imag(Sinj(S.Gen.Bus)).^2)-S.Gen.S];

%Power flow equations
Sv=Vbus.*conj(S.Ybus*Vbus);

%Equality constraint
Ceq=[Pinj-real(Sv); Qinj-imag(Sv)];
Appendix B. Fortran Interface Module to link PSCAD and MATLAB

The following code was developed for the PSCAD component that links the simulation to the MATLAB reactive power control algorithm. The code handles the memory assignments of the various inputs and outputs.

```fortran
#STORAGE REAL:8
#LOCAL INTEGER I_CNT
IF ($TRIG==1) THEN
    DO I_CNT=1,2,1
        STORF(NSTORF+I_CNT-1) = $pg(I_CNT)
    ENDDO

    DO I_CNT=1,2,1
        STORF(NSTORF+2+I_CNT-1) = $pl(I_CNT)
    ENDDO

    DO I_CNT=1,2,1
        STORF(NSTORF+4+I_CNT-1) = $ql(I_CNT)
    ENDDO

    CALL MLAB_INT("$Path", "$Name","R(2) R(2) R(2)","R(2)")

    DO I_CNT=1,2,1
        $Qcmd(I_CNT) = STORF(NSTORF+6+I_CNT-1)
    ENDDO
END IF
```
NSTORF = NSTORF + 8