Graduate School ETD Form 9
(Revised 12/07)

# PURDUE UNIVERSITY
## GRADUATE SCHOOL
## Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Diana C. Vasquez

Entitled EXPANSION OF DYNAMIC SIMULATION MODEL FOR A DISTRIBUTED GENERATOR UNINTENTIONAL ISLANDING DETECTION SCHEME

For the degree of Master of Science in Electrical and Computer Engineering

Is approved by the final examining committee:

Chair

Steven Rovnyak

Maher Rizkalla

Yaobin Chen

To the best of my knowledge and as understood by the student in the *Research Integrity and Copyright Disclaimer (Graduate School Form 20)*, this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Approved by Major Professor(s): Steven Rovnyak

Approved by: Brian King 7/14/2010
Head of the Graduate Program Date

Graduate School Form 20
(Revised 1/10)

# PURDUE UNIVERSITY
## GRADUATE SCHOOL

## Research Integrity and Copyright Disclaimer

Title of Thesis/Dissertation:
EXPANSION OF DYNAMIC SIMULATION MODEL FOR A DISTRIBUTED GENERATOR
UNINTENTIONAL ISLANDING DETECTION SCHEME

For the degree of Master of Science in Electrical and Computer Engineering

I certify that in the preparation of this thesis, I have observed the provisions of *Purdue University Teaching, Research, and Outreach Policy on Research Misconduct (VIII.3.1)*, October 1, 2008.*

Further, I certify that this work is free of plagiarism and all materials appearing in this thesis/dissertation have been properly quoted and attributed.

I certify that all copyrighted material incorporated into this thesis/dissertation is in compliance with the United States' copyright law and that I have received written permission from the copyright owners for my use of their work, which is beyond the scope of the law. I agree to indemnify and save harmless Purdue University from any and all claims that may be asserted or that may arise from any copyright violation.

Diana C. Vasquez
Printed Name and Signature of Candidate

7/14/2010
Date (month/day/year)

*Located at http://www.purdue.edu/policies/pages/teach_res_outreach/viii_3_1.html

# EXPANSION OF DYNAMIC SIMULATION MODEL FOR A DISTRIBUTED GENERATOR UNINTENTIONAL ISLANDING DETECTION SCHEME

A Thesis

Submitted to the Faculty

of

Purdue University

by

Diana C. Vasquez

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

August 2010

Purdue University

Indianapolis, Indiana

Para mis mejores amigos: mis padres.

# ACKNOWLEDGMENTS

I would like to gratefully acknowledge my thesis advisor, Dr. Steven Rovnyak, for his guidance, assistance and supervision during the course of this research and thesis work. Dr. Rovnyak shared his knowledge and expertise with me as well as his time and attention to every detail. He was always available to answer my questions, he was open to new ideas, and he was willing to set time aside to ensure that I had a clear understanding of every simulation outcome and was successfully completing every goal, for which I will always be thankful.

I would also like to thank my advisory committee members, Dr. Yaobin Chen, and Dr. Maher Rizkalla for their time and support during the completion of this thesis, and also Dr. Yong Sheng for his insight during the earlier stages of this thesis work.

I would like to extend my special thanks to the Purdue School of Engineering and Technology, IUPUI, specially to Terri Talbert-Hatch in the office of Student Services, Paula Jenkins in the office of Development and External Relations, and Dr. Yaobin Chen, Department Chair of the Electrical and Computer Engineering Department. All of them kept me in mind for every scholarship opportunity available and played a great role in the financial aspect of my thesis work.

I would also like to thank all the faculty and staff who facilitated my thesis work specially to Valerie Lim Diemer and Sherrie Tucker who kindly assisted me in formatting this thesis. Finally I express my deep gratitude to my parents for their immense sacrifice, support and encouragement in every stage in my life, my family and friends, and my fellow classmates.

## TABLE OF CONTENTS

Page

# LIST OF FIGURES

Figure Page

# ABBREVIATIONS

| | |
|---|---|
| DG | Distributed Generator |
| DGSC | Distributed Generation System Controller |
| UIC | Utility Interface Controller |
| EPS | Electric Power System |
| PCC | Point of Common Coupling |
| DFT | Discrete Fourier Transform |

# ABSTRACT

Vasquez, Diana C. M.S.E.C.E., Purdue University, August 2010. Expansion of Dynamic Simulation Model for a Distributed Generator Unintentional Islanding Detection Scheme. Major Professor: Steven Rovnyak.

The interconnection of distributed resources requires specific voltage regulation, monitoring, protective relaying, power quality, and islanding detection. For this reason IEEE established standard IEEE 1547 that ensures the compliance with such requirements and it will help formulate technical specifications for grid interconnection with Distributed Generator (DG) resources. In search of meeting the IEEE 1547 standard requirement of detecting unintentional islanded operation, there has been ongoing research to develop anti-islanding methods that can detect the different changes that can occur when the grid is disconnected. A team of Electrical Engineering faculty at Indiana University Purdue University Indianapolis has worked previously on testing a DG unintentional Islanding Detection Scheme. This scheme uses an active anti-islanding method in which a small 1 Hz perturbation signal is added into the DG system and it helps detect when the grid is disconnected. The scheme uses the premise that a frequency deviation caused by perturbation to the system is smaller when the grid is connected than when it is in an island.

In an initial dynamic simulation model for the islanding detection scheme, a two-machine microgrid system is used to explore frequency and voltage responses when the grid is disconnected. In this thesis, the two-machine microgrid is expanded to a ten-machine system so it can be shown that the frequency deviation caused by a perturbation signal is much smaller when the grid is connected even for a larger DG network. The 1 Hz component of the DG electrical frequency in a multiple machine microgrid system is also calculated in this thesis. This project was conducted in different stages. First, it was necessary to calculate the steady state power flow and electric power of a three-machine system and update the two-machine MATLAB program with the necessary changes. After making the changes, it was necessary to simulate the system and adjust the inertia of the machine that represents the grid to ensure that the simulation output was close in magnitude to previous testing results. When the three-machine system was successfully generated, a brand new program was created so a multiple machine system could be simulated. Then the multiple machine program was used to simulate and experiment with up to a ten-machine system. Finally a program to calculate the 1 Hz component of the DG electrical frequency was generated and used to show that the magnitude squared of the 1 Hz component is inversely proportional to the number of machines connected to the system. These last findings will later help set the threshold for islanding detection appropriately for different numbers of DG.

# 1. INTRODUCTION

## 1.1 Problem Statement

The IEEE standard for interconnection of distributed resources or IEEE 1547-2003 defines the term island as ”A condition in which a portion of an Area Electric Power System (EPS) is energized solely by one or more local EPS through the associated point of common coupling (PCC) while that portion of the Area EPS is electrically separated from the rest of the Area EPS.” [1] The standard is very specific with the requirements for unintentional islanding detection. It requires the detection of unintentional islanded operations within a two second window while allowing the use any detection method that can be proven to pass the IEEE Standard Conformance Test for Equipment Interconnecting Distributed Resources with Electric Power Systems 1547.1 [2]. This test simulates the most difficult scenarios for islanding detection.

To extend the work done previously in which a 1Hz perturbation signal is introduced in a single DG to detect changes occurring at the time the grid is disconnected [3], it is now important to increase the number of DGs in the system and to show that the perturbation method can still be used. An important part of this investigation is the calculation and analysis of the 1 Hz content of the DG electrical frequency. The calculation of the 1 Hz component for different numbers of DGs will later allow the calculation of the threshold for islanding detection for a system of three, four, or more machines.

## 1.2 Previous Work

Islanding detection is an area of research where different methods have been proposed [4–8]. Active and passive methods are the two current categories being used.

Active methods introduce a perturbation signal into the system that helps detect when the grid is disconnected. With the adoption of the IEEE 1547 standard and the possibility of developing and commercializing a universally accepted distributed generation system controller (DGSC), I Power Energy Systems, LLC, and the authors of [3] developed a DGSC to satisfy the IEEE 1547 standard. This work was supported by grants from the Indiana 21st Century Research and Technology Fund.

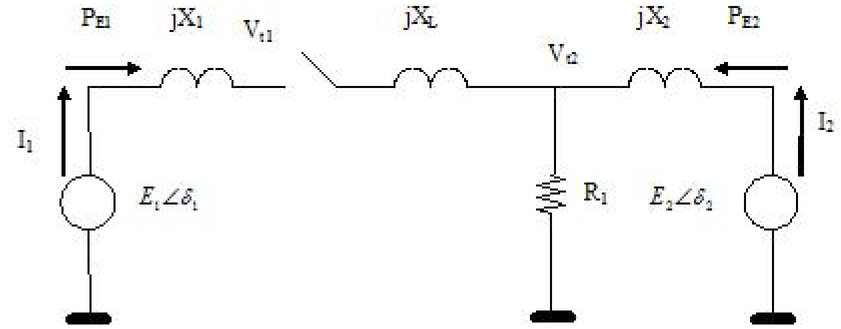


Fig. 1.1.: A one-line diagram of the two-machine system.

In previous work, a simple two-machine microgrid system was simulated to study the frequency and voltage responses after a sudden change of load. For the two-machine simulation, two identical synchronous machines were used to represent the two DG units with typical parameters. However, both machines possessed different control strategies. One of the machines acted as an isochronous generator whose governor was required to maintain constant frequency. The second DG acted as a base load generator with a constant active power.

Figure 1.1 shows the two-machine system used in the initial simulation model where $E_1$ and $E_2$ represent the DG's internal voltages; $X_1$ and $X_2$ represents the DG's transient reactance; $V_{t1}$ and $V_{t2}$ represent the DG's terminal voltage; $R_1$ represents the load; and $X_L$ represents the link reactance between the two DGs. $P_{E1}$ and $P_{E2}$ are the electric power flows corresponding to each of the DGs. Using this system model, $P_{E1}$ and $P_{E2}$ were calculated as a function of the DG internal voltages. The system was then simulated based on the classic electromechanical dynamic equations discussed in Chapter 2 of this thesis.

Figure 1.2 shows the test setup for unintentional islanding detection of a synchronous DG. The test setup shows a switch S1 that is supposed to serve as the disconnection point between the DG and the utility. B0 and B1 are circuit breakers housed in the DG unit and at the point of common coupling (PCC) between the premises containing the DG and the electric utility. The area EPS shown in the figure ”means any source capable of maintaining an island within the voltage and frequency requirements” [1].

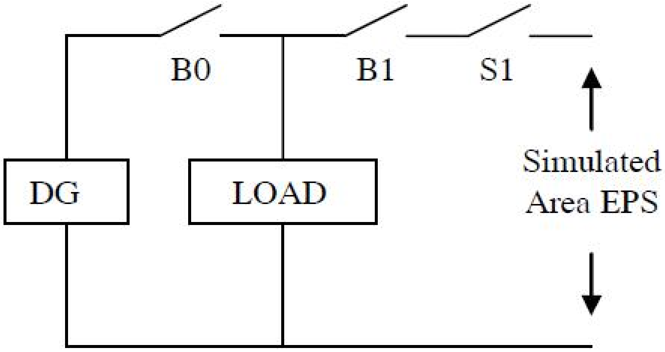


Fig. 1.2.: Unintentional islanding test configuration for synchronous generators.

When comparing Figure 1.1 and Figure 1.2 one can see a great similarities such as the presence of a DG on the left hand side which represents the grid, a load ($R_1$ in Figure 1.1) and an area EPS represented in Figure 1.1 by the second DG on the right side. In order to simulate the grid characteristics in the system it was necessary to increase the inertia of DG 1. When S1 is closed both machines are coupled, the frequency of DG 2 (right hand side) follows the grid and it is hardly affected by the perturbation. When the switch is open the perturbation will have a higher impact on DG 2.

After testing the system based on the set up described above at the I Power testing facility, the frequency measurements obtained are shown in Figure 1.3. The graph in Figure 1.3. shows how the perturbation signal (1Hz signal superimposed to the DG unit's throttle during entire measurement window) produced a larger variation on the DGs and the frequency change after the DG and its matching load became an island at around four seconds.

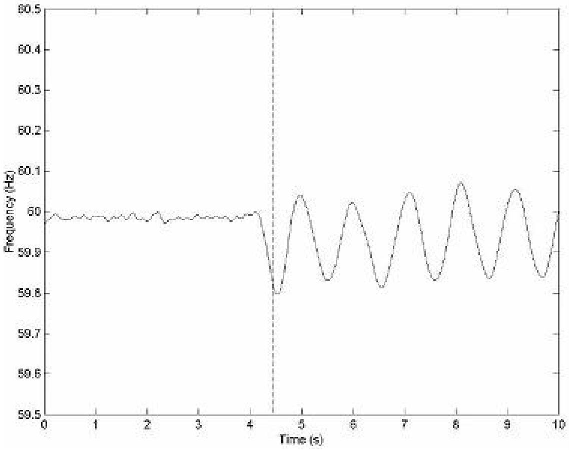


Fig. 1.3.: Frequency measurements during an islanding detection test with DG output of approximately 20kW + 0 kVAR.

The results of the computer simulations shown in Figure 1.4. reflect the similarities between the two set ups. The simulations also showed that the perturbation signal caused a very small frequency deviation when the grid is connected compared to when the DG is in an island.

### 1.3 Objectives

The main objectives of this project are: 1. To extend the work of Rovnyak et al. [3] on unintentional islanding detection schemes by simulating a multiple machine system; 2. Calculate the 1Hz content of the DG electrical frequency after simulating a multiple machine system; and 3. To verify the initial premise that a frequency deviation caused by perturbation to the system is smaller when the grid is connected than when the DG is in an island. After the completion of this last objective it can be determined how much increasing the number of machines decreases the magnitude

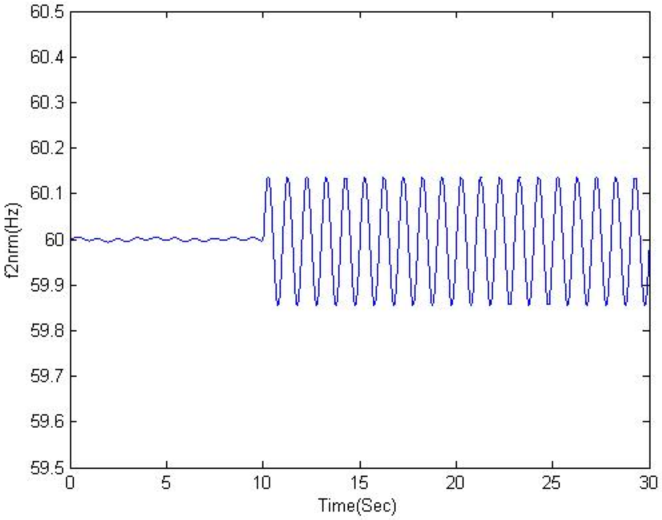


Fig. 1.4.: Frequency results of two machine simulation from previous work.

of the 1 Hz component. It can also help establish where to set the threshold that determines whether the DG system is in an island.

## 1.4 About This Thesis

To accomplish the objectives mentioned previously, it was necessary to first calculate the power flow and of a three-machine system and update the two-machine algorithm with necessary changes. Then the four-machine system was successfully simulated. Later a new program was generated so an N-machine system could also be simulated for any value of N.

Throughout the process there were different calculations involved such as calculating the admittance matrix that allows one to analyze the system power flow and finding the resistance of the load in the DG island. Finally, an algorithm to calculate the 1Hz component of the DG electrical frequency was generated. A total of eight different DG systems were generated with successful results. For each different system ($N = 4, 5, ..., 8$) the program automatically produced the value of the 1Hz component after islanding. The values of the magnitude squared of the 1Hz component was plotted against the number of machines. A polynomial curve was fitted to the resulting data as a function of the number of DGs. The findings of this last stage of the project can help determine how to change the threshold that decides if the grid is connected or disconnected for different number of machines in the system.

# 2. DYNAMIC SIMULATION MODEL FOR A THREE MACHINE SYSTEM

## 2.1 Base Model for a Two Machine System

The simulation process for a three machine system started from previous work by Rovnyak et al. [3] in which the two machine system shown in Figure 1.1 was simulated. The following subsections show the node admittance matrix and electric power calculations for the two machine system as well as the electromechanical dynamic equations and discrete equations used to calculate the frequency and voltage output of the system for a 30 second simulation.

### 2.1.1 Steady State Power Flow and Electric Power Calculation

In order to simulate the system shown in Figure 1.1 it was necessary to calculate the the network admittance matrix and the electric power. The following steps indicate the procedures to calculate electric powers $P_{G1}$ and $P_{G2}$ using the nodal admittance matrix:

- Construct the admittance matrix equations to show the relations between voltage variables and current variables. The dimensions of the admittance matrix depends on the size of the system. The $4 \times 4$ admittance matrix below corresponds to the two machine system in Figure 1.1.

$$\begin{bmatrix} \hat{I}_1 \\ \hat{I}_2 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{jX_1} & 0 & -\frac{1}{jX_1} & 0 \\ 0 & \frac{1}{jX_2} & 0 & -\frac{1}{jX_2} \\ -\frac{1}{jX_1} & 0 & \frac{1}{R_1} + \frac{1}{jX_1} + \frac{1}{jX_L} & -\frac{1}{jX_L} \\ 0 & -\frac{1}{jX_2} & -\frac{1}{jX_L} & \frac{1}{jX_2} + \frac{1}{jX_L} \end{bmatrix} \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \\ \hat{V}_{t1} \\ \hat{V}_{t2} \end{bmatrix} \tag{2.1}$$

$$\begin{bmatrix} \hat{I}_1 \\ \hat{I}_2 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \\ \hat{V}_{t1} \\ \hat{V}_{t2} \end{bmatrix}$$

where,

$$A = \begin{bmatrix} \frac{1}{jX_1} & 0 \\ 0 & \frac{1}{jX_2} \end{bmatrix}, B = \begin{bmatrix} -\frac{1}{jX_1} & 0 \\ 0 & -\frac{}{jX_2} \end{bmatrix} \tag{2.2}$$

$$C = \begin{bmatrix} -\frac{1}{jX_1} & 0 \\ 0 & -\frac{}{jX_2} \end{bmatrix}, D = \begin{bmatrix} \frac{1}{R_1} + \frac{1}{jX_1} + \frac{1}{jX_L} & -\frac{1}{jX_L} \\ -\frac{1}{jX_L} & \frac{1}{jX_2} + \frac{1}{jX_L} \end{bmatrix}$$

- Construct the reduced admittance equations as follows:

$$I_G = \begin{bmatrix} \hat{I}_1 \\ \hat{I}_2 \end{bmatrix} = (A - BD^{-1}C) \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \end{bmatrix} \text{ where, } \ E_G = \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \end{bmatrix}$$

then,

$$I_G = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} E_G = Y E_G \tag{2.3}$$

The $2 \times 2$ matrix is the reduced admittance matrix.

- Calculate the electric powers from the following equations:

$$P_G = \begin{bmatrix} P_{G1} \\ P_{G2} \end{bmatrix} = \begin{bmatrix} \hat{E}_1 \hat{I}_1^* \\ \hat{E}_2 \hat{I}_2^* \end{bmatrix} \tag{2.4}$$

$\hat{I}^*$ is the conjugate of $\hat{I}$.

The equation used to calculate terminal voltages are:

$$V_t = \begin{bmatrix} \hat{V}_{t1} \\ \hat{V}_{t2} \end{bmatrix} = E_G - \begin{bmatrix} jX_1 \hat{I}_1 \\ jX_2 \hat{I}_2 \end{bmatrix} \tag{2.5}$$

A change of load requires updating the admittance matrix to reflect the change. New power flows and terminal voltages are calculated using the same steps mentioned above with the new admittance matrix.

### 2.1.2 Electromechanical Dynamic Equations

The dynamics of the two machine system are simulated based on the following equations for a detailed generator model. Equation 2.6 reflects the presence of the perturbation signal in machine two that helps determine whether the system is in an island. The quantities $I_G$, $P_G$, and $V_t$ for the continuous time case are calculated as shown in Equations 2.3 through 2.5. So the electromechanical dynamic equations for machines 1 and 2 are the following:

For machine 1:

$$\begin{aligned}
\frac{d}{dt}\Delta\delta_1 &= \Delta\omega_1\omega_0 \\
\frac{d}{dt}P_{M1} &= -K\Delta\omega_1 \\
2H_1\frac{d}{dt}\Delta\omega_1 &= P_{M1} - P_{G1} - D_1\Delta\omega_1 \\
E_{f1} &= K_A[V_{1ref} - |\hat{V_{t1}}| + K_D\Delta\omega_1] \\
\frac{d}{dt}E_1 &= \frac{1}{T_3}[K_3(E_{f1} - K_4\Delta\delta_1) - E_1] \\
\hat{E_1} &= E_1 e^{j\Delta\delta_1}
\end{aligned}$$

For machine 2:

$$\begin{aligned}
\frac{d}{dt}\Delta\delta_2 &= \Delta\omega_2\omega 0 \\
\frac{d}{dt}P_{M2} &= -K\Delta\omega_2 \\
P_{G02} &= 0.01P_{G02}(0)sin(2\pi t) \\
2H_2\frac{d}{dt}\Delta\omega_2 &= P_{M2} - P_{G2} - D_2\Delta\omega_2 + P_{G02} \qquad (2.6) \\
E_{f2} &= K_A[V_{2ref} - |\hat{V_{t2}}| + K_D\Delta\omega_2] \\
E_2 &= \frac{1}{T_3}[K_3(E_{f2} - K_4\Delta\delta_2) - E_2] \\
\hat{E_2} &= E_2 e^{j\Delta\delta_2} \qquad (2.7)
\end{aligned}$$

$\omega_0$ is the machine rotor speed reference that is equal to $120\pi$ rad/sec. H is the inertia constant. The load damping constant D is defined as $\frac{\Delta P}{\Delta f}$. In other words, it shows how much the load drops if the frequency is reduced.

### 2.1.3 Discrete Equations

The quantities $I_G$, $P_G$, and $V_t$ for the discrete time case are calculated as shown in Equations 2.3 through 2.5.The following are the discrete equations used for the two machine system dynamic simulations:

For machine 1:

$$\begin{aligned}
\Delta\delta_1(k+1) &= \Delta\delta_1(k) + \Delta\omega_1(k).\omega_0 \cdot \Delta t \\
P_{M1}(k+1) &= P_{M1}(k) - K \cdot \Delta\omega_1(k) \cdot \Delta t \\
\Delta\omega_1(k+1) &= \Delta\omega_1(k) + \frac{P_{M1}(k) - P_{G1}(k) - D_1\Delta\omega_1(k)}{2H_1} \cdot \Delta t \\
E_{f1}(k+1) &= K_A[V_{1ref} - |\hat{V_{t1}}(k)| + K_D \cdot \Delta\omega_1(k)] \\
E_1(k+1) &= E_1(k) + \frac{1}{T_3}[K_3(E_{f1}(k) - K_4\Delta\delta_1(k)) - E_1(k)] \cdot \Delta t \\
\hat{E_1}(k+1) &= E_1(k)e^{j\Delta\delta_1(k)}
\end{aligned}$$

For machine 2:

$$\begin{aligned}
\Delta\delta_2(k+1) &= \Delta\delta_2(k) + \Delta\omega_2(k).\omega_0 \cdot \Delta t \\
P_{M2}(k+1) &= P_{M2}(k) \\
\Delta\omega_2(k+1) &= \Delta\omega_2(k) + \frac{P_{M2}(k) - P_{G2}(k) - D_2\Delta\omega_2(k) + 0.01P_{G02}(0)sin(2\pi\Delta t)}{2H_2} \cdot \Delta t \\
E_{f2}(k+1) &= K_A[V_{2ref} - |\hat{V_{t2}}(k)| + K_D \cdot \Delta\omega_2(k)] \\
E_2(k+1) &= E_2(k) + \frac{1}{T_3}[K_3(E_{f2}(k) - K_4\Delta\delta_2(k)) - E_2(k)] \cdot \Delta t \\
\hat{E_2}(k+1) &= E_2(k)e^{j\Delta\delta_2(k)}
\end{aligned}$$

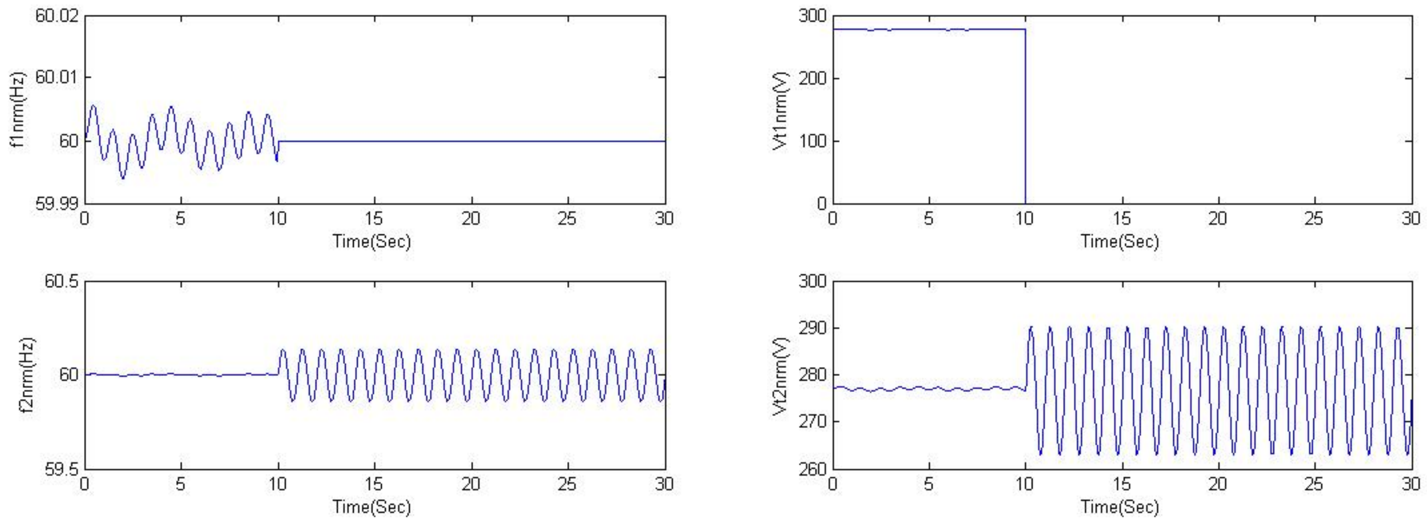
The initial values of each variable are derived from the system steady state which corresponds to the initial results of Equation 2.2. The initial values for $E_1$ and $E_2$ are calculated using Equation 2.7 with the initial values of $\Delta\delta_1$ and $\Delta\delta_2$.

$P_{M1}$ and $P_{M2}$ are set equal to $P_{G1}$ and $P_{G2}$ respectively and the values of $E_{f1}$ and $E_{f2}$ are calculated using the values of $E_1$ and $E_2$ divided by constant K3. The following are the initial values used for the simulation:

$$\begin{aligned}
\delta_1(1) &= 0 \\
P_{G1}(1) &= 0.333 \\
P_{M1}(1) &= 0.333 \\
\Delta\omega_1(1) &= 0 \\
V_{t1}(1) &= 1 \\
E_{f1}(1) &= 3.33 \\
E_1(1) &= 1 \\
\delta_2(1) &= 0 \\
P_{G2}(1) &= 0.166 \\
P_{M2}(1) &= 0.166 \\
\Delta\omega_2(1) &= 0 \\
V_{t2}(1) &= 1 \\
E_{f2}(1) &= 3.33 \\
E_2(1) &= 1
\end{aligned}$$

### 2.1.4 Two Machine Model Simulation Results

After simulating the two machine system model the frequency and voltage results showed how the perturbation signal hardly affects DG 2 before the change in load. However, when the grid is disconnected at 10 seconds the perturbation has a greater impact in the frequency of DG 2. The frequency and voltage results are shown in Figure 2.1.



(a) Frequency Results

(b) Voltage Results

Fig. 2.1.: Two machine system simulation results.

## 2.2 Three Machine System Model

After successfully simulating the two machine system, a new MATLAB program code was generated to simulate the three machine system shown in Figure 2.3. The changes in the calculations are shown in the following subsections.

### 2.2.1 Steady State Power Flow and Electric Power Calculation

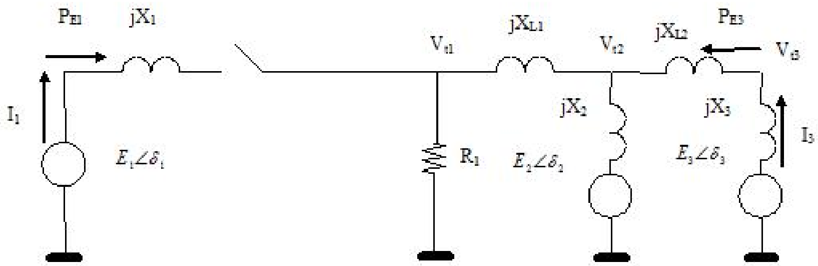


Fig. 2.2.: A one-line diagram of the three-machine system.

In order to simulate the system shown in Figure 2.3 it was necessary to calculate the steady state power flow and the electric power. The calculation procedures are similar to the procedures for the two machine system; however, the admittance matrix needs to be recalculated and new variables need to be added in order to calculate $P_{G3}$. The following steps reflect the changes:

- The dimensions of the admittance matrix for the three machine system is $6 \times 6$. The admittance matrix below corresponds to the three machine system.

$$\begin{bmatrix} \frac{1}{jX_1} & 0 & 0 & -\frac{1}{jX_1} & 0 & 0 \\ 0 & \frac{1}{jX_2} & 0 & 0 & -\frac{1}{jX_2} & 0 \\ 0 & 0 & \frac{1}{jX_3} & 0 & 0 & -\frac{1}{jX_3} \\ -\frac{1}{jX_1} & 0 & 0 & \frac{1}{R_1} + \frac{1}{jX_1} + \frac{1}{jX_{L1}} & -\frac{1}{jX_{L_1}} & 0 \\ 0 & -\frac{1}{jX_2} & 0 & -\frac{1}{jX_{L1}} & \frac{1}{jX_2} + \frac{1}{jX_{L_1}} + \frac{1}{jX_{L_2}} & -\frac{1}{jX_{L_2}} \\ 0 & 0 & -\frac{1}{jX_3} & 0 & -\frac{1}{jX_{L_2}} & \frac{1}{jX_{L_2}} + \frac{1}{jX_3} \end{bmatrix}$$

where,

$$A = \begin{bmatrix} \frac{1}{jX_1} & 0 & 0 \\ 0 & \frac{1}{jX_2} & 0 \\ 0 & 0 & \frac{1}{jX_3} \end{bmatrix}, B = \begin{bmatrix} -\frac{1}{jX_1} & 0 & 0 \\ 0 & -\frac{1}{jX_2} & 0 \\ 0 & 0 & -\frac{1}{jX_3} \end{bmatrix}$$

$$C = \begin{bmatrix} -\frac{1}{jX_1} & 0 & 0 \\ 0 & -\frac{1}{jX_2} & 0 \\ 0 & 0 & -\frac{1}{jX_3} \end{bmatrix},$$

$$D = \begin{bmatrix} \frac{1}{R_1} + \frac{1}{jX_1} + \frac{1}{jX_{L_1}} & -\frac{1}{jX_{L_1}} & 0 \\ -\frac{1}{jX_{L1}} & \frac{1}{jX_2} + \frac{1}{jX_{L_1}} + \frac{1}{jX_{L_2}} & -\frac{1}{jX_{L_2}} \\ 0 & -\frac{1}{jX_{L_2}} & \frac{1}{jX_{L_2}} + \frac{1}{jX_3} \end{bmatrix}$$

- Construct the reduced admittance equations adding one more variable to each array representing the current, and voltage for the third branch of the system as follows:

$$I_G = \begin{bmatrix} \hat{I}_1 \\ \hat{I}_2 \\ \hat{I}_3 \end{bmatrix} = (A - BD^{-1}C) \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \\ \hat{E}_3 \end{bmatrix} \text{ where, } E_G = \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \\ \hat{E}_3 \end{bmatrix}$$

then,

$$I_G = \begin{bmatrix} y_{11} & y_{12} & y_{13} \\ y_{21} & y_{22} & y_{23} \\ y_{31} & y_{32} & y_{33} \end{bmatrix} E_G = YE_G \tag{2.8}$$

where the 3 × 3 matrix is the reduced admittance matrix.

- Calculate the electric powers by adding the third variable for the power flow corresponding to the third machine DG 3 as follows:

$$P_G = \begin{bmatrix} P_{G1} \\ P_{G2} \\ P_{G3} \end{bmatrix} = \begin{bmatrix} \hat{E}_1 \hat{I}_1^* \\ \hat{E}_2 \hat{I}_2^* \\ \hat{E}_3 \hat{I}_3^* \end{bmatrix} \tag{2.9}$$

The equations used to calculate terminal voltages are:

$$V_t = \begin{bmatrix} \hat{V_{t1}} \\ \hat{V_{t2}} \\ \hat{V_{t3}} \end{bmatrix} = E_G - \begin{bmatrix} jX_1\hat{I_1}^* \\ jX_2\hat{I_2}^* \\ jX_3\hat{I_3}^* \end{bmatrix} \tag{2.10}$$

A sudden change of load requires to update matrix A to reflect the change. New power flows and terminal voltages are calculated using the same steps mentioned above.

### 2.2.2 Electromechanical Dynamic Equations

The dynamics of the three machine system is simulated using the equations for machine 1 and 2 shown in Section 2.1.2. and a third set of equations (show below) is added to the simulation.

For machine 3:

$$\begin{aligned}
\frac{d}{dt}\Delta\delta_3 &= \Delta\omega_3\omega_0 \\
\frac{d}{dt}P_{M3} &= -K\Delta\omega_3 \\
2H_3\frac{d}{dt}\Delta\omega_3 &= P_{M3} - P_{G3} - D_3\Delta\omega_3 \\
E_{f3} &= K_A[V_{3ref} - |\hat{V_{t3}}| + K_D\Delta\omega_3] \\
\frac{d}{dt}E_3 &= \frac{1}{T_3}[K_3(E_{f3} - K_4\Delta\delta_3) - E_3] \\
\hat{E_3} &= E_3e^{j\Delta\delta_3}
\end{aligned}$$

### 2.2.3 Discrete Equations

The discrete equations for the three machine system includes the calculation of $E_3$ and the values for $E_1$ and $E_2$ used for the two machine system which remain the same. The quantities $I_G$, $P_G$, and $V_t$ for the discrete time case are calculated as shown in Equations 2.8 through 2.10. The set of equations shown below corresponds to machine

3 and these equations are added when simulating the three machine system to the equations for machine 1 and machine 2 shown in Section 2.1.3. So,

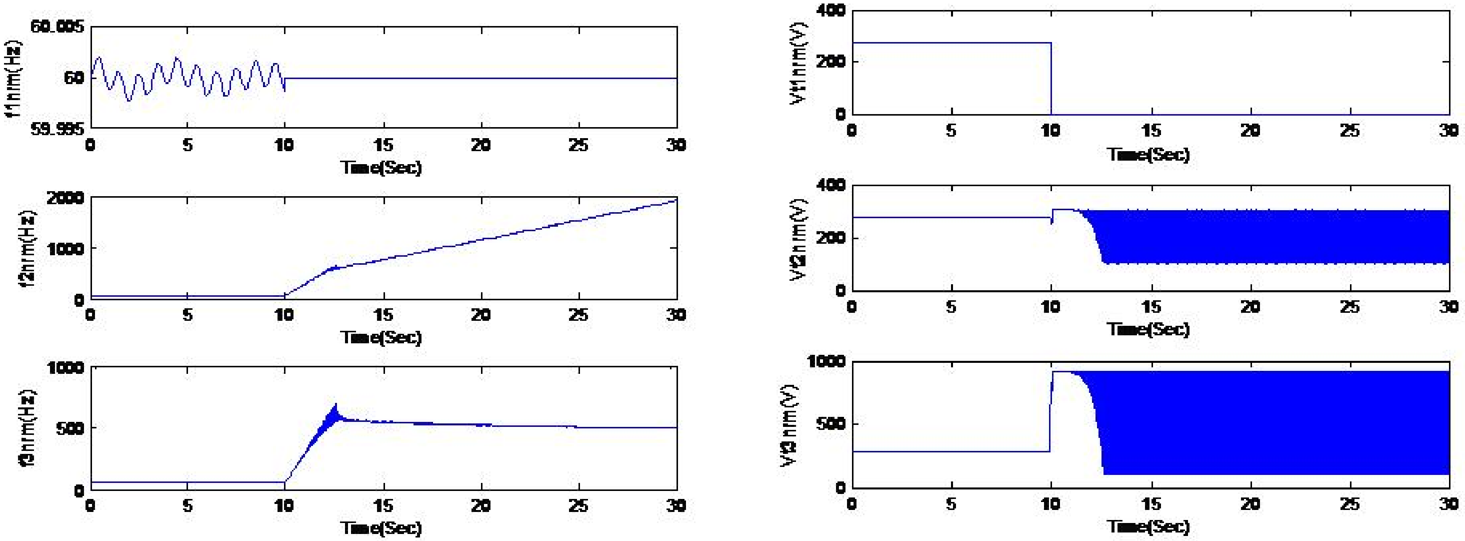
For machine 3:

$$\begin{aligned}
\Delta\delta_3(k+1) &= \Delta\delta_3(k) + \Delta\omega_3(k).\omega_0 \cdot \Delta t \\
P_{M3}(k+1) &= P_{M3}(k) - K \cdot \Delta\omega_3(k) \cdot \Delta t \\
\Delta\omega_3(k+1) &= \Delta\omega_3(k) + \frac{P_{M3}(k) - P_{G3}(k) - D_3\Delta\omega_3(k)}{2H_3} \cdot \Delta t \\
E_{f3}(k+1) &= K_A[V_{3ref} - |\hat{V_{t3}}(k)| + K_D \cdot \Delta\omega_3(k)] \\
E_3(k+1) &= E_3(k) + \frac{1}{T_3}[K_3(E_{f3}(k) - K_4\Delta\delta_1(k)) - E_3(k)] \cdot \Delta t \\
\hat{E}_3(k+1) &= E_3(k)e^{j\Delta\delta_3(k)}
\end{aligned}$$

The initial values used in the three machine simulation for machine 1 and 2 remain the same and a third set of initial values corresponding to machine three are added and equal to the initial values of machine 1 and 2.

### 2.2.4 Three Machine System Simulation Results Before Calculating Load Resistance in DG Island

The results of the three machine simulations are shown in Figure 2.3. Figure 2.3.(a) shows how the frequency of machine two and three become unstable after disconnecting them from machine one (the grid). Similarly, Figure 2.3.(b) shows the voltage output which also becomes unstable after ten seconds. After these results it was suggested to calculate a load resistance corresponding to the DG power output. This means the DG has a matched load and together they are disconnected from the grid. The matched load is shown as $R_{new}$ in Figure 2.4.



(a) Frequency Results

(b) Voltage Results

Fig. 2.3.: Three machine system simulation results without calculating matching load.

### 2.2.5 Calculations of Load Resistance in DG Island

From Figure 2.4. two parallel resistance can be identified: $R_0$ and $R_{new}$. With this new set up, $R_0$ would absorb $P_1$ and $R_{new}$ would absorb $P_2$ and $P_3$. The parallel combination of $R_0$ and $R_{new}$ equals $R_1$ in Figure 2.2:

$$R_1 = R_{new} \parallel R_0 \tag{2.11}$$

where,

$$R_0 = \frac{V_{t_1}^2}{P_1} \tag{2.12}$$

Then,

$$\begin{aligned} \frac{V_{t_1}^2}{P_1} \parallel R_{new} &= R_1 && (2.13)\\ \frac{1}{R_1} &= \frac{P_1}{V_{t_1}^2} + \frac{1}{R_{new}} && (2.14)\\ \frac{1}{R_{new}} &= \frac{1}{R_1} - \frac{P_1}{V_{t_1}^2} && (2.15)\\ R_{new} &= \frac{R_1}{1 - \frac{R_1 P_1}{V_{t_1}^2}} && (2.16) \end{aligned}$$

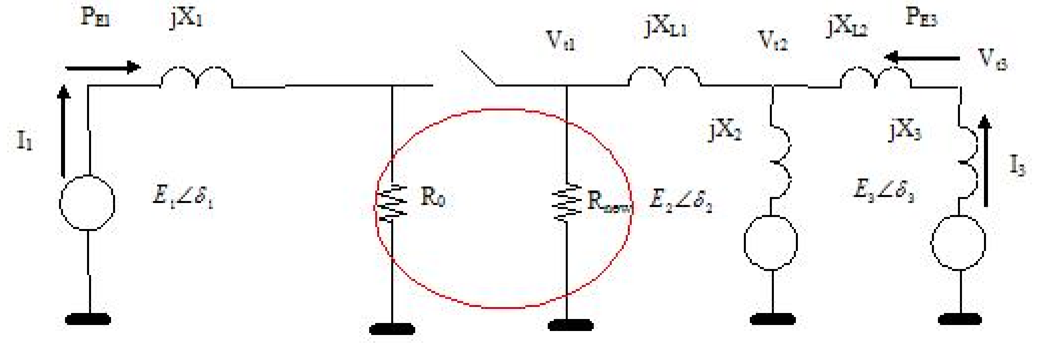
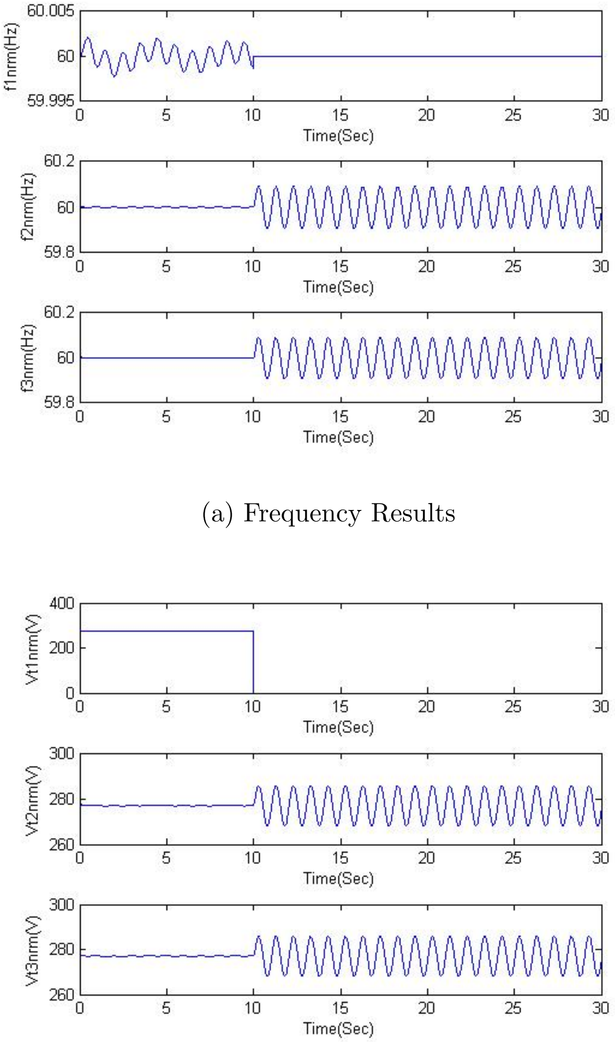


Fig. 2.4.: A one line diagram of a three machine system including matching load ($R_{new}$).

### 2.2.6 Three Machine System Simulation Results After Calculating Load Resistance in DG Island

With the new load value, new results were obtained (See Figure 2.5). The new resistance/load value allowed the system to stabilize and show the same behavior of previous results with the two machine system.



(a) Frequency Results

(b) Voltage Results

Fig. 2.5.: Three machine system simulation results after calculating matching load.

# 3. DYNAMIC SIMULATION MODEL FOR A MULTIPLE MACHINE SYSTEM

Before creating a new MATLAB program to simulate a multiple machine system it was necessary to model and simulate a four machine system. This four machine model helped determine a pattern to update the admittance matrix systematically (or in a loop) to facilitate programming the MATLAB code that simulates a three or more machine system. A similar process to the process shown in Chapter 2 was followed for the four machine system.

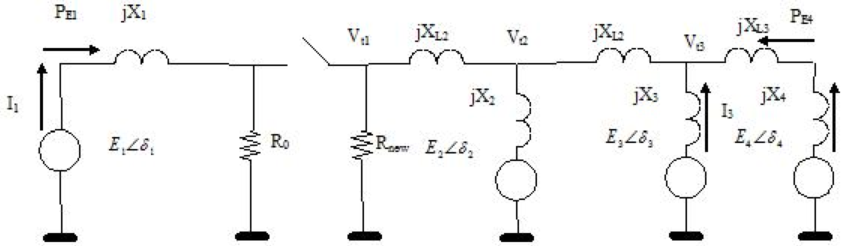


Fig. 3.1.: A one-line diagram for the four-machine system.

## 3.1 Steady State Power Flow and Electric Power Calculation (Four machines)

The dimensions of the admittance matrix for the four machine system is 8 × 8. The admittance matrix below corresponds to the four machine system.

$$\begin{bmatrix} \frac{1}{jX_1} & 0 & 0 & 0 & -\frac{1}{jX_1} & 0 & 0 & 0 \\ 0 & \frac{1}{jX_2} & 0 & 0 & 0 & -\frac{1}{jX_2} & 0 & 0 \\ 0 & 0 & \frac{1}{jX_3} & 0 & 0 & 0 & -\frac{1}{jX_3} & 0 \\ 0 & 0 & 0 & \frac{1}{jX_4} & 0 & 0 & 0 & -\frac{1}{jX_3} \\ -\frac{1}{jX_1} & 0 & 0 & 0 & \frac{1}{R_1} + \frac{1}{jX_1} + \frac{1}{jX_{L1}} & -\frac{1}{jX_{L_1}} & 0 & 0 \\ 0 & -\frac{1}{jX_2} & 0 & 0 & -\frac{1}{jX_{L1}} & \frac{1}{jX_2} + \frac{1}{jX_{L_1}} + \frac{1}{jX_{L_2}} & -\frac{1}{jX_{L_2}} & 0 \\ 0 & 0 & -\frac{1}{jX_3} & 0 & 0 & -\frac{1}{jX_{L_2}} & \frac{1}{jX_{L_2}} + \frac{1}{jX_3} + \frac{1}{jX_{L_3}} & \frac{1}{jX_{L_3}} \\ 0 & 0 & 0 & -\frac{1}{jX_4} & 0 & 0 & -\frac{1}{jX_{L_3}} & \frac{1}{jX_{L_3}} + \frac{1}{jX_4} \end{bmatrix}$$

where,

$$A = \begin{bmatrix} \frac{1}{jX_1} & 0 & 0 & 0 \\ 0 & \frac{1}{jX_2} & 0 & 0 \\ 0 & 0 & \frac{1}{jX_3} & 0 \\ 0 & 0 & 0 & \frac{1}{jX_4} \end{bmatrix}, B = \begin{bmatrix} -\frac{1}{jX_1} & 0 & 0 & 0 \\ 0 & -\frac{1}{jX_2} & 0 & 0 \\ 0 & 0 & -\frac{1}{jX_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{jX_4} \end{bmatrix}$$

$$C = \begin{bmatrix} -\frac{1}{jX_1} & 0 & 0 & 0 \\ 0 & -\frac{1}{jX_2} & 0 & 0 \\ 0 & 0 & -\frac{1}{jX_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{jX_4} \end{bmatrix}, D = \begin{bmatrix} \frac{1}{R_1} + \frac{1}{jX_1} + \frac{1}{jX_{L_1}} & -\frac{1}{jX_{L_1}} & 0 & 0 \\ -\frac{1}{jX_{L1}} & \frac{1}{jX_2} + \frac{1}{jX_{L_1}} + \frac{1}{jX_{L_2}} & -\frac{1}{jX_{L_2}} & 0 \\ 0 & -\frac{1}{jX_{L_2}} & \frac{1}{jX_{L_2}} + \frac{1}{jX_3} + \frac{1}{jX_{L_3}} & \\ 0 & 0 & -\frac{1}{jX_{L_3}} & \frac{1}{jX_{L_3}} + \frac{1}{jX_4} \end{bmatrix}$$

The reduced admittance equations adding one more variable to each array representing the current and voltage for the fourth branch of the system as follows:

$$I_G = \begin{bmatrix} \hat{I}_1 \\ \hat{I}_2 \\ \hat{I}_3 \\ \hat{I}_4 \end{bmatrix} = (A - BD^{-1}C) \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \\ \hat{E}_3 \\ \hat{E}_4 \end{bmatrix} \text{ where, } E_G = \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \\ \hat{E}_3 \\ \hat{E}_4 \end{bmatrix}$$

then,

$$I_G = \begin{bmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{bmatrix} E_G = Y E_G \tag{3.1}$$

where the 4 × 4 matrix is the reduced admittance matrix.

The electric powers by adding the third variable for the power flow corresponding to the forth machine DG 4 as follows:

$$P_G = \begin{bmatrix} P_{G1} \\ P_{G2} \\ P_{G3} \\ P_{G4} \end{bmatrix} = \begin{bmatrix} \hat{E}_1 \hat{I}_1^* \\ \hat{E}_2 \hat{I}_2^* \\ \hat{E}_3 \hat{I}_3^* \\ \hat{E}_4 \hat{I}_4^* \end{bmatrix} \tag{3.2}$$

The equations used to calculate terminal voltages are:

$$V_t = \begin{bmatrix} \hat{V}_{t1} \\ \hat{V}_{t2} \\ \hat{V}_{t3} \\ \hat{V}_{t4} \end{bmatrix} = E_G - \begin{bmatrix} jX_1 \hat{I_1}^* \\ jX_2 \hat{I_2}^* \\ jX_3 \hat{I_3}^* \\ jX_4 \hat{I_4}^* \end{bmatrix} \quad (18) \tag{3.3}$$

### 3.1.1 Electromechanical Dynamic Equations (Four machines)

The dynamics of the three machine system is simulated using the equations for machine 1 and 2 shown in Section 2.1.2., the set of equations shown in Section 2.2.2, and a forth set of equations (show below) added to the simulation.

For machine 4:

$$\begin{aligned}
\frac{d}{dt}\Delta\delta_4 &= \Delta\omega_4\omega_0 \\
\frac{d}{dt}P_{M4} &= -K\Delta\omega_4 \\
2H_4\frac{d}{dt}\Delta\omega_4 &= P_{M4} - P_{G4} - D_4\Delta\omega_4 \\
E_{f4} &= K_A[V_{4ref} - |\hat{V_{t4}}| + K_D\Delta\omega_4] \\
\frac{d}{dt}E_4 &= \frac{1}{T_4}[K_4(E_{f4} - K_4\Delta\delta_4) - E_4] \\
\hat{E}_4 &= E_4 e^{j\Delta\delta_4}
\end{aligned}$$

### 3.1.2 Discrete Equations (Four machines)

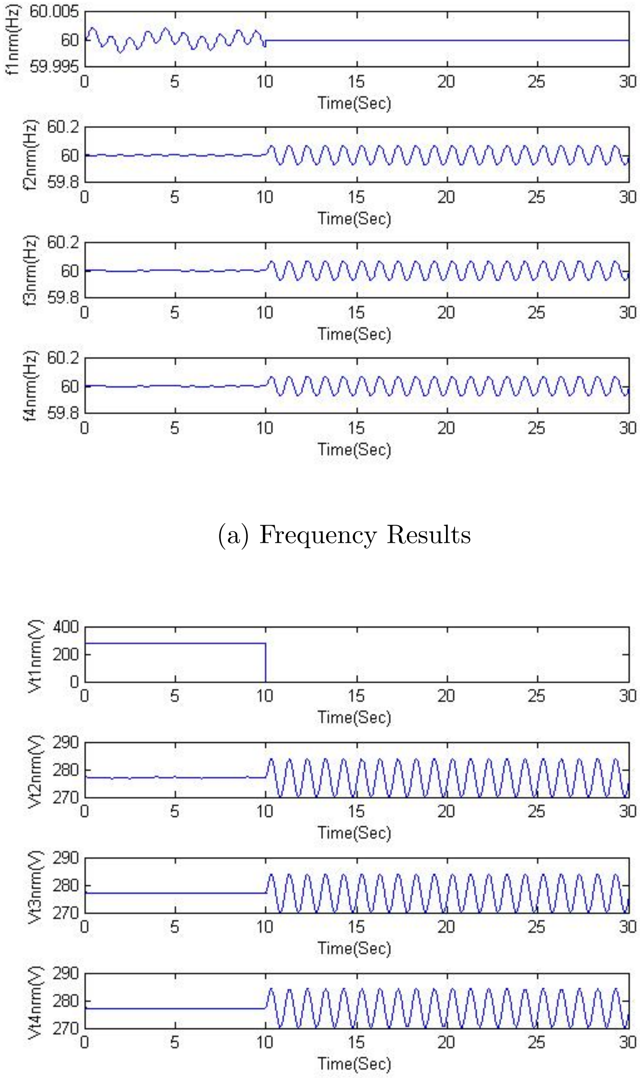
The discrete equations for the four machine system includes the calculation of $E_4$ and the values for $E_1$, $E_2$, and $E_2$ used for the three machine system which remain the same. The quantities $I_G$, $P_G$, and $V_t$ for the discrete time case are calculated as shown in Equations 3.1 through 3.3. The set of equations shown below corresponds to machine 4 and these equations are added when simulating the four machine system to the equations for machine 1 and machine 2 shown in Section 2.1.3., and machine 3 shown in Section 2.2.3. So,

For machine 4:

$$\begin{aligned}
\Delta\delta_4(k+1) &= \Delta\delta_4(k) + \Delta\omega_4(k).\omega_0 \cdot \Delta t \\
P_{M4}(k+1) &= P_{M4}(k) - K \cdot \Delta\omega_4(k) \cdot \Delta t \\
\Delta\omega_4(k+1) &= \Delta\omega_4(k) + \frac{P_{M4}(k) - P_{G4}(k) - D_4\Delta\omega_4(k)}{2H_4} \cdot \Delta t \\
E_{f4}(k+1) &= K_A[V_{4ref} - |\hat{V_{t4}}(k)| + K_D \cdot \Delta\omega_4(k)] \\
E_4(k+1) &= E_4(k) + \frac{1}{T_3}[K_3(E_{f4}(k) - K_4\Delta\delta_1(k)) - E_4(k)] \cdot \Delta t \\
\hat{E}_4(k+1) &= E_4(k)e^{j\Delta\delta_4(k)}
\end{aligned}$$

### 3.1.3 Four Machine System Simulation Results

Figure 3.2 shows the frequency and voltage results for the four machine system simulation respectively. The four machine simulation was obtained after recalculating the load resistance similar to the three machine chase.



(a) Frequency Results

(b) Voltage Results

Fig. 3.2.: Four machine system simulation results.

## 3.2 Program Development for Multiple Machine Simulations

After successfully simulating the two, three, and four machine systems, it was necessary to store all the initial values for the reactance (machine and link reactance)and inertia in array form as shown below:

$$X = \begin{bmatrix} j0.1 \\ \vdots \end{bmatrix} \quad XL = \begin{bmatrix} j0.1 \\ \vdots \end{bmatrix} \quad H = \begin{bmatrix} 50 \\ 0.1 \\ \vdots \end{bmatrix}$$

where X is the array of machine reactances, XL is the array of link reactances, and H is the array of inertias. The size of the arrays equal the number of machines in the system.

In order to update the admittance matrix (See Equations 2.1 and 2.2) the following patterns were identified and applied in the MATLAB program that simulates systems of multiple machines:

1. The size of submatrix A equals $N \times N$ where N is the machine system size $(n = 2, 3, ..., 8)$
   - A is a diagonal matrix.
   - The elements of the diagonal of A correspond to the admittance values of the machines as shown below:

$$\begin{bmatrix} \frac{1}{jX(1)} & 0 & \cdots & 0 \\ 0 & \frac{1}{jX(2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{jX(N)} \end{bmatrix}$$

2. Submatrices B and C are equal diagonal matrices.

3. Submatrices B and C are opposite and equal to matrix A.

4. Matrix D is tridiagonal matrix (Banded matrix) of the form:

$$\begin{bmatrix} d_{11} & d_{12} & 0 & 0 & \cdots & 0 & 0 \\ d_{21} & d_{22} & d_{23} & \ddots & \ddots & 0 & 0 \\ 0 & d_{32} & d_{33} & \ddots & \ddots & d_{N-2,N-1} & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & d_{N-1,N-1} & d_{N-1,N} \\ 0 & 0 & \cdots & \cdots & \cdots & d_{N,N-1} & d_{N,N} \end{bmatrix}$$

5. Elements (1,1), (N,N), (N,N-1), and (N-1,N) of matrix D can be calculated as follows:

$$\begin{aligned} D(1,1) &= \frac{1}{R_1} + \frac{1}{X(1)} + \frac{1}{XL(1)} \\ D(N,N) &= \frac{1}{XL(N-1)} + \frac{1}{X(N)} \\ D(N,N-1) &= \frac{-1}{XL(N-1)} \\ D(N-1,N) &= \frac{-1}{XL(N-1)} \end{aligned}$$

6. Elements of the diagonals of D except D(1,1) and D(N,N) are calculated using the algorithm shown below:

```
for $n = 2$ to N-1 do
    $D(n,n) = \frac{1}{X(n)} + \frac{1}{XL(n)} + \frac{1}{XL(n-1)}$
    $D(n,n-1) = \frac{-1}{XL(n-1)}$
    $D(n-1,n) = \frac{-1}{XL(n-1)}$
end for
```

### 3.2.1 Electromechanical Dynamic Equations for an N Machine System

Once the admittance matrix is updated, the reduced admittance equations are used in the same way they are used when simulating three and four machine systems to calculate current, terminal voltage and power flow (Refer to Equations 2.3 through

2.5). The multiple machine system is simulated using the equations shown below for $N > 2$. The equations for machines $n = 1$ and $n = 2$ are the same as before. Machine $n = 1$ represents the grid, and machine $n = 2$ has the perturbation signal.

For machine n:

$$\begin{aligned}
\frac{d}{dt}\Delta\delta_n &= \Delta\omega_n\omega_0 \\
\frac{d}{dt}P_{Mn} &= -K\Delta\omega_n \\
2H_n\frac{d}{dt}\Delta\omega_n &= P_{Mn} - P_{Gn} - D_n\Delta\omega_n \\
E_{fn} &= K_A[V_{nref} - |\hat{V_{tn}}| + K_D\Delta\omega_n] \\
\frac{d}{dt}E_n &= \frac{1}{T_n}[K_n(E_{fn} - K_n\Delta\delta_n) - E_n] \\
\hat{E_n} &= E_n e^{j\Delta\delta_n}
\end{aligned}$$

### 3.2.2 Discrete Equations for an N Machine System

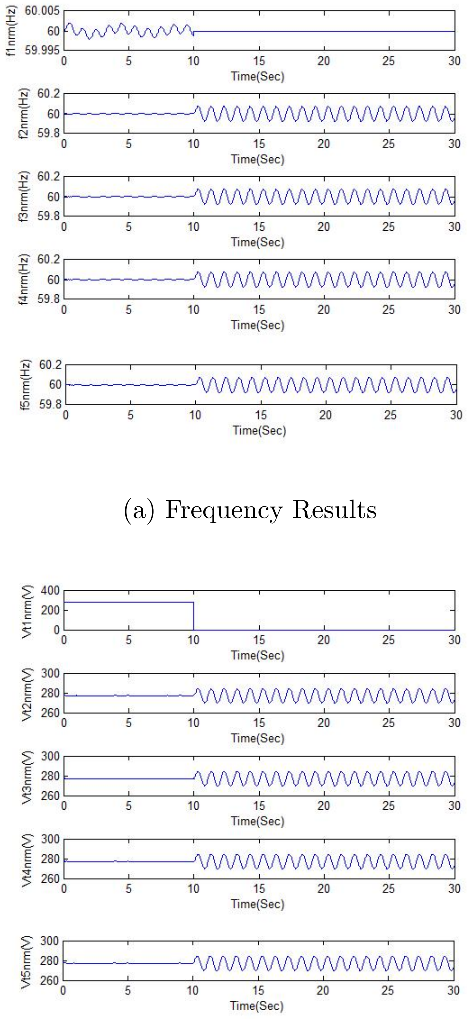
The discrete equations for the N machine system include the following for $n > 2$. The quantities $I_G$, $P_G$, and $V_t$ for the discrete time case are calculated as shown in Equations 3.1 through 3.3.

For machine n:

$$\begin{aligned}
\Delta\delta_n(k+1) &= \Delta\delta_n(k) + \Delta\omega_n(k).\omega_0 \cdot \Delta t \\
P_{Mn}(k+1) &= P_{Mn}(k) - K \cdot \Delta\omega_n(k) \cdot \Delta t \\
\Delta\omega_n(k+1) &= \Delta\omega_n(k) + \frac{P_{Mn}(k) - P_{Gn}(k) - D_n\Delta\omega_n(k)}{2H_n} \cdot \Delta t \\
E_{fn}(k+1) &= K_A[V_{nref} - |\hat{V_{tn}}(k)| + K_D \cdot \Delta\omega_n(k)] \\
E_n(k+1) &= E_n(k) + \frac{1}{T_3}[K_3(E_{fn}(k) - K_n\Delta\delta_1(k)) - E_n(k)] \cdot \Delta t \\
\hat{E_n}(k+1) &= E_n(k)e^{j\Delta\delta_n(k)}
\end{aligned}$$

### 3.2.3 Multiple Machine Systems Simulation Results

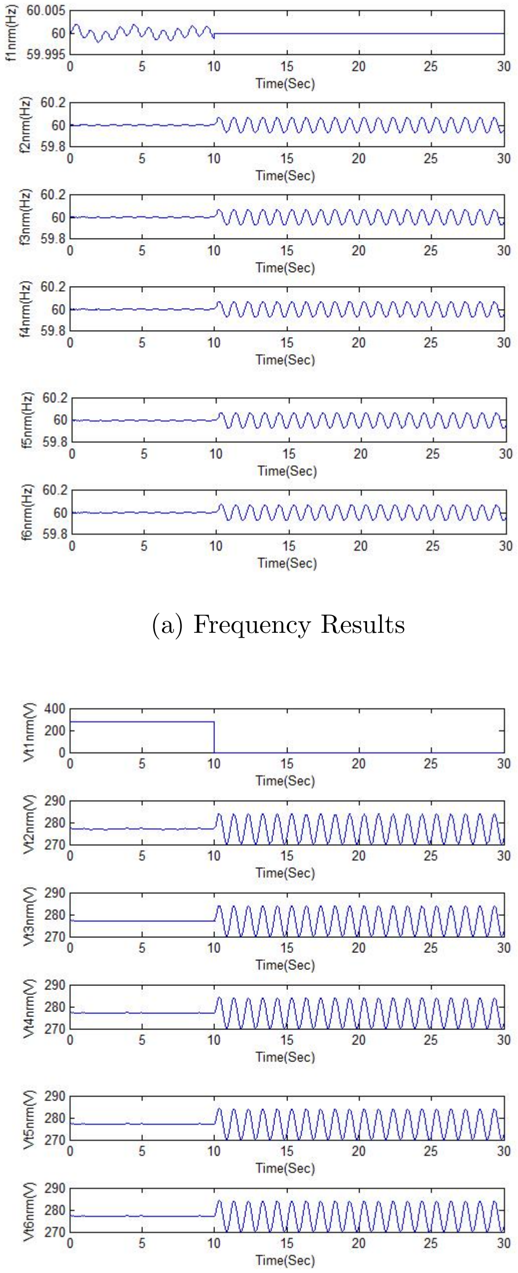
As shown in Figures 3.3 through 3.5 the five, six, and seven machine system results are consistent with the results obtained previously after simulating two, three and four machines. The 1 Hz component must be calculated only before instability develops. Instability can be seen toward the end of the simulations with seven machines.



(a) Frequency Results

(b) Voltage Results

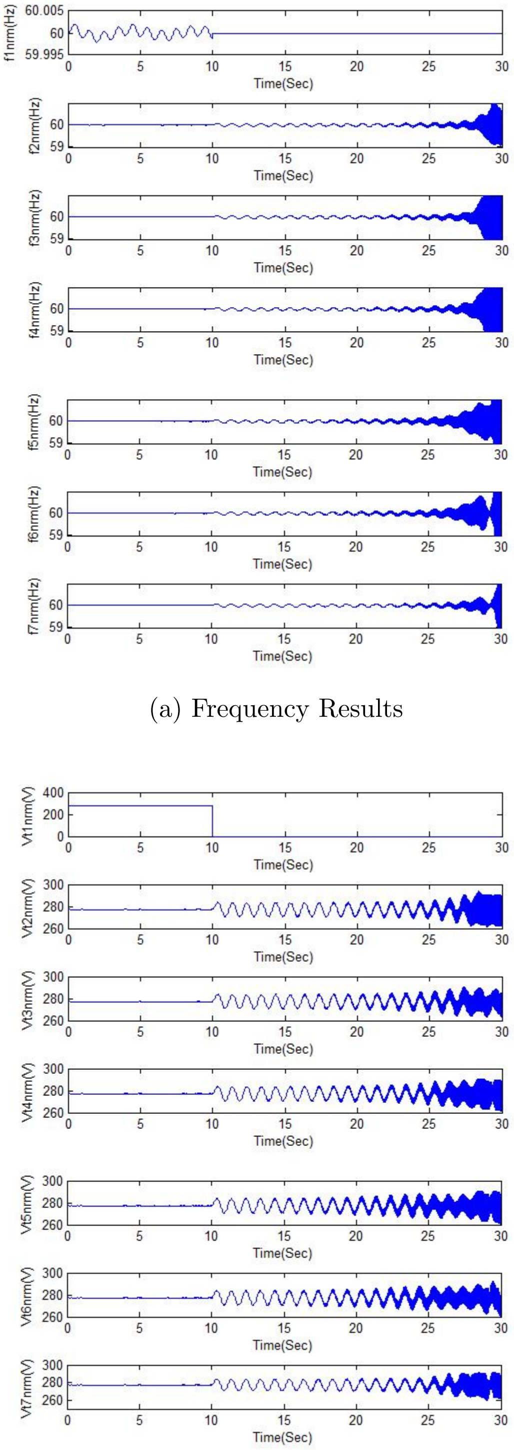
Fig. 3.3.: Five machine system simulation results.



(a) Frequency Results

(b) Voltage Results

Fig. 3.4.: Six machine system simulation results.



(a) Frequency Results

(b) Voltage Results

Fig. 3.5.: Seven machine system simulation results.

# 4. CALCULATION OF 1HZ COMPONENT OF DG FREQUENCY IN MULTIPLE MACHINE SYSTEMS

One it was verified that a frequency deviation caused by perturbation to the system of machines (up to seven) is smaller when the grid is connected than when the DG is in an island, it was determined how much the number of DG machines added to the system decreases the magnitude of the 1Hz component. This calculation can help establish where to set the threshold that determines whether the DG system is in an island.

## 4.1 DG Frequency Sampling

The analysis of the 1 Hz component of the DG frequency is based on the principle of Discrete Fourier Transform (DFT). As described in [3] a 36-point DFT is used to sample the DG frequency at a rate of 36 samples per second. If $v(t)$ is a vector containing the 36 most recent samples of frequency measurements at time t, then as in [3] $v_C$, $v_S$, and $v_{60}$ can be defined as:

$$v_C = \begin{bmatrix} \cos(1\pi/18) \\ \cos(2\pi/18) \\ \vdots \\ \cos(36\pi/18) \end{bmatrix}, v_S = \begin{bmatrix} \sin(1\pi/18) \\ \sin(2\pi/18) \\ \vdots \\ \sin(36\pi/18) \end{bmatrix}, v_{60} = \begin{bmatrix} 60 \\ 60 \\ \vdots \\ 60 \end{bmatrix}$$

The magnitude squared of the 1 Hz component can be calculated [3]:

$$Y_S(t) = (2/36) < v(t) - v_{60}, v_S > \tag{4.1}$$

$$Y_C(t) = (2/36) < v(t) - v_{60}, v_C > \tag{4.2}$$

$< a, b >$ is the inner product of the vectors $a$ and $b$. Then the squared magnitude of the 1Hz component is:

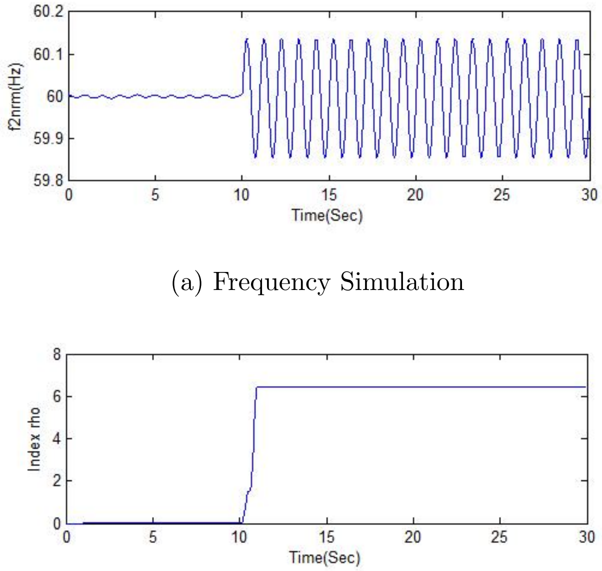
$$|Y|^2 = Y_C^2(t) + Y_S^2(t) \tag{4.3}$$

As in [3] the calculations for the index rho ($\rho$) is given by:

$$\rho = 324|Y|^2 \tag{4.4}$$

## 4.2 1Hz Component Calculation Results

Figures 4.1 through 4.3 correspond to the frequency simulation and rho index calculation of the 2, 4, and 7 machine systems respectively.

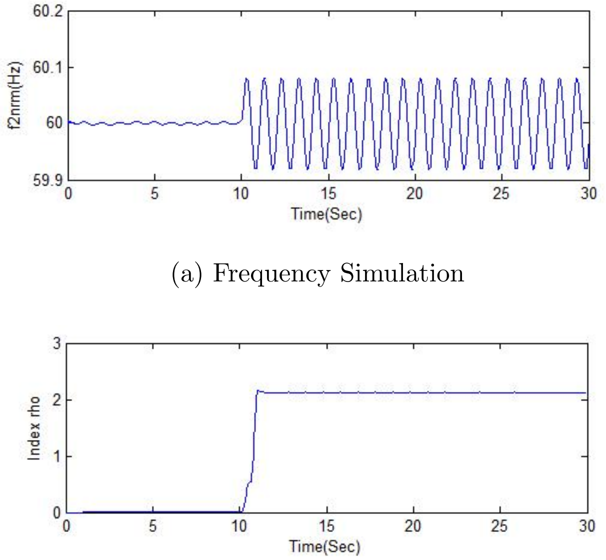


(a) Frequency Simulation

(b) Index $\rho$ calculated from Frequency Simulations

Fig. 4.1.: Two machine system 1Hz component calculation results.

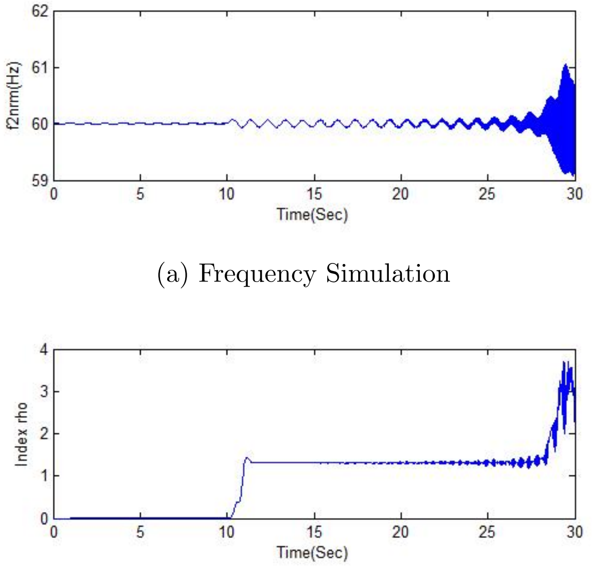
Table 4.1 summarizes the results of the 1Hz calculations for each different system size. It also shows a column that shows the values of the perturbation strength $PG_2(0)$ for each different system size. The normalized perturbation value is given by dividing the $\rho$ index by the perturbation strength.



(a) Frequency Simulation

(b) Index $\rho$ calculated from Frequency Simulations

Fig. 4.2.: Four machine system 1Hz component calculation results.



(a) Frequency Simulation

(b) Index $\rho$ calculated from Frequency Simulations

Fig. 4.3.: Seven machine system 1Hz component calculation results.

Table 4.1: 1Hz calculation results and perturbation strength for systems with different number of machines.

| No. Of DG | Rho Index | Perturbation Strength | Normalized Perturbation |
|---|---|---|---|
| 1 | 6.423983848 | 0.166481687 | 38.58672965 |
| 2 | 2.688302929 | 0.124878049 | 21.5274258 |
| 3 | 2.125266184 | 0.118933675 | 17.86933928 |
| 4 | 1.819456429 | 0.118069018 | 15.41010889 |
| 5 | 1.552119466 | 0.117942922 | 13.15992038 |
| 6 | 1.26955663 | 0.117924526 | 10.76584042 |

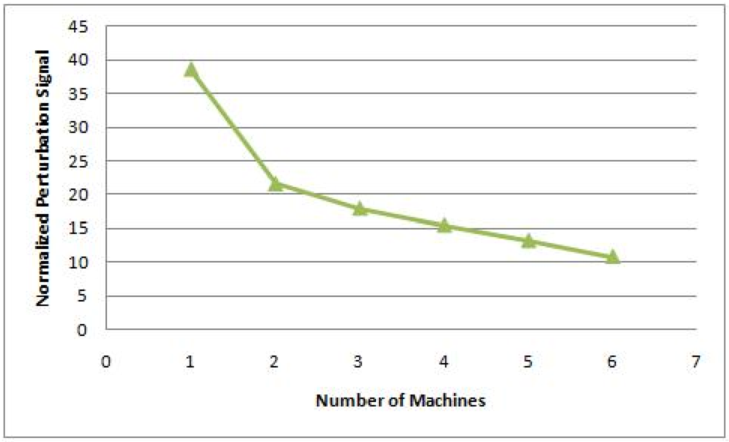


Fig. 4.4.: Normalized Perturbation vs. Number of Machines.

Figure 4.4 shows a the curve of the normalized perturbation vs. the number of machines in the system.

# 5. CONCLUSIONS AND RECOMMENDATIONS

The perturbation method for islanding detection has been tested and shown to be effective for one to six DG in parallel. In every simulation, one machine represents the grid, and the remaining machines represent DG. Therefore the seven machine system in this thesis represents six parallel DG and the grid. When adding the sixth DG, the simulations show instability. Using simulations it was verified for up to six DG in parallel that a frequency deviation caused by perturbation to the system is smaller when the grid is connected than when the DG is in an island. Therefore the proposed island detection method will probably work for up to six DG in parallel.

## 5.1 Future Work

The next step of this research is to test practically the results obtained in the simulations and show compliance with IEEE Standard 1547.1 [2]. To complete this task it is important to consider two design parameters involved: the strength of the perturbation signal, and the threshold value. Instead of having a constant perturbation strength and a threshold that varies with the number of machines, it would be possible to always use the same threshold and vary the perturbation strength depending on the number of DG. It seems simpler to always have the same perturbation strength and vary the threshold.

# LIST OF REFERENCES

# LIST OF REFERENCES

[1] "*IEEE Standard for Interconnectig Distributed Resources with Electric Power Systems*," *IEEE Standard 1547-2003*, July 2003.

[2] "*IEEE Standard Conformance Test Procedures for Equipment Interconnectig Distributed Resources with Electric Power Systems*," *IEEE Standard 1547.1-2005*, July 2005.

[3] S. M. Rovnyak, S. Koskie, Y. Sheng, V. S. Rajput, and X. Hu, "Laboratory testing of a distributed generator unintentional islanding detection scheme," *Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century*, 2008.

[4] P.L.Villeneuve, "Concerns generated by islanding," *IEEE Power and Energy Magazine*, vol. 2, pp. 49–53, May/June 2004.

[5] F.-S. Pai and S.-J.Huang, "A detecion algorithm for islanding-prevention of dispersed consumer-owned storage and generating units," *IEEE Transactions on Energy Conversion*, vol. 16, pp. 346–351, December 2001.

[6] S.-I. Jang and K.-H.Kim, "An islanding detection method for distributed generations using voltage unbalance and total harmonic distortion of current," *IEEE Transactions on Power Delivery*, vol. 19, pp. 745–752, April 2004.

[7] Z.Ye, A. Kolwalkar, Y. Zhang, P. Du, and R. Walling, "Evaluation of anti-islading schemes based on nondetection zone concept," *IEEE Transactions on Power Electronics*, vol. 19, pp. 1171–1176, September 2004.

[8] K. El-Arroudi, G. Joos, I. Kamwa, and D. T. McGillis, "Intelligent-based approach to islanding detection in distributed generation," *IEEE Transactions on Power Delivery*, vol. 22, pp. 828–835, April 2007.

# APPENDICES

## Appendix A: MATLAB Code Two Machine System Simulation

```
% Simulate a two-machine system dynamics
function [f2nrm,time] =m2smls2(x)
%     clear all;
%     format long;
%     clc;
% Machine 1 represents one DG unit:
H1=x;              % Inertia
X1=j*0.1;           % Internal reactance
% Machine 2 represents the other DG unit
H2=0.1;               % Inertia
X2=j*0.1;           % Internal reactance
% Loads
R1=2;            % Represents loads
% Link between machine 1 and machine 2
XL=j*0.1;           % Link reactance
% Construct admittance matrix
A=[1/X1 0; 0 1/X2];
B=[-1/X1 0; 0 -1/X2];
C=[-1/X1 0; 0 -1/X2];
D=[1/R1+1/X1+1/XL -1/XL; -1/XL 1/X2+1/XL];
% Construct reduced admittance matrix
y=A-B*inv(D)*C;
% Calculate power
% EG0=[cos(-1.433*pi/180)+j*sin(-1.433*pi/180); 1];
% Initial E1 and E2
EG0=[cos(0*pi/180)+j*sin(0*pi/180); 1];
IG0=y*EG0;
```

```
Pe0=real(EG0.*conj(IG0));
% Calculate terminal voltage
Vt0=abs(EG0-[X1;X2].*IG0);
% Define step size
dt=0.0001;
% Define simulation length
t=30;
% Define fault occuring time
t1=10;
%Calculate size of arrays
s=round(t/dt);
% Declear arrays
time=zeros(s,1);
delta1=zeros(s,1);
w1=zeros(s,1);
Pe1=zeros(s,1);
Pm1=zeros(s,1);
Vt1=zeros(s,1);
Ef1=zeros(s,1);
E1=zeros(s,1);
delta2=zeros(s,1);
w2=zeros(s,1);
Pe2=zeros(s,1);
Pm2=zeros(s,1);
Vt2=zeros(s,1);
Ef2=zeros(s,1);
E2=zeros(s,1);
% Parameter D is the load damping constant.
% D is defined as how many percent the load will drop if
% the frequency is reduced by 1 percent.
% D is very small for this case, so 0 is used.
D1=0.0;
D2=0.0;
% Parameters
R=0.00;
K=200;
T3=2.4;
K3=0.3;
K4=1.4;
KD=21;%3.5;
% Define nominal angular velocity
w0=120*pi;
% Get initial values
delta1(1)=angle(EG0(1));
w1(1)=0;
Pe1(1)=Pe0(1);
Pm1(1)=Pe0(1);
Vt1(1)=Vt0(1);
E1(1)=abs(EG0(1));
Ef1(1)=E1(1)/K3;
delta2(1)=angle(EG0(2));
w2(1)=0;
Pe2(1)=Pe0(2);
Pm2(1)=Pe0(2);
Vt2(1)=Vt0(2);
E2(1)=abs(EG0(2));
Ef2(1)=E2(1)/K3;
% References
```

```
P1ref=Pe0(1);
P2ref=Pe0(2);
V1ref=1;
V2ref=1;
% Calculated parameters
KA1=E1(1)/(V1ref-Vt0(1))/K3;
KA2=E2(1)/(V2ref-Vt0(2))/K3;
% Solve equations
% Normal operation for 1 second
for k=2:round(t1/dt)
    time(k)=dt*(k-1);
    EG=[E1(k-1)*exp(j*delta1(k-1)); E2(k-1)*exp(j*delta2(k-1))];
    IG=y*EG;
    Pe=real(EG.*conj(IG));
    Vt=abs(EG-[X1;X2].*IG);
    delta1(k)=delta1(k-1)+w1(k-1)*w0*dt;
    Pe1(k)=Pe(1);
    Pm1(k)=Pm1(k-1)+K*(R*P1ref-R*Pm1(k-1)-w1(k-1))*dt;
    %Pm1(k-1)-K*w1(k-1)*dt;
    if Pm1(k) > 1
        Pm1(k) = 1;
    elseif Pm1(k) < 0
        Pm1(k) = 0;
    end
    w1(k)=w1(k-1)+(Pm1(k-1)-Pe1(k-1)-D1*w1(k-1))/2/H1*dt;
    Vt1(k)=Vt(1);
    Ef1(k)=KA1*(V1ref-Vt1(k-1)+KD*w1(k-1));
    E1(k)=E1(k-1)+1/T3*(K3*(Ef1(k-1)-K4*delta1(k-1))-E1(k-1))*dt;
    if E1(k) > 1.1
```

```
        E1(k) = 1.1;
    elseif E1(k) < 0.9
        E1(k)=0.9;
    end
    delta2(k)=delta2(k-1)+w2(k-1)*w0*dt;
    Pe2(k)=Pe(2);
    Pm2(k)=Pm2(k-1);%+K*(R*P2ref-R*Pm2(k-1)-w2(k-1))*dt;
    if Pm2(k) > 1
        Pm2(k) = 1;
    elseif Pm2(k) < 0
        Pm2(k) = 0;
    end
    w2(k)=w2(k-1)+(Pm2(k-1)-Pe2(k-1)-D2*w2(k-1)+
    0.1*P2ref*sin(2*pi*k*dt))/2/H2*dt;

    Vt2(k)=Vt(2);
    Ef2(k)=KA2*(V2ref-Vt2(k-1)+KD*w2(k-1));
    E2(k)=E2(k-1)+1/T3*(K3*(Ef2(k-1)-K4*delta2(k-1))-E2(k-1))*dt;
    if E2(k) > 1.1
        E2(k) = 1.1;
    elseif E2(k) < 0.9
        E2(k)=0.9;
    end
end

%Load shedding

R0=Vt1(1)^2/Pe1(1);
R_new=R0*R1/(R0-R1);
```

```
 % Construct new admittance matrix
A=1/X1;
B=-1/X1;
C=-1/X1;
D=1/R_new+1/XL;
% Construct reduced admittance matrix
y=A-B*inv(D)*C;
% Loss of machine 1 at 10 s
for k=round(t1/dt)+1:s
    time(k)=dt*(k-1);
    EG=[E2(k-1)*exp(j*delta2(k-1))];
    IG=y*EG;
    Pe=real(EG.*conj(IG));
    Vt=abs(EG-X2.*IG);
    delta2(k)=delta2(k-1)+w2(k-1)*w0*dt;
    Pe2(k)=Pe(1);
    Pm2(k)=Pm2(k-1);%+K*(R*P2ref-R*Pm2(k-1)-w2(k-1))*dt;
    if Pm2(k) > 1
        Pm2(k) = 1;
    elseif Pm2(k) < 0
        Pm2(k) = 0;
    end
    w2(k)=w2(k-1)+(Pm2(k-1)-Pe2(k-1)-D2*w2(k-1)+
    0.1*P2ref*sin(2*pi*k*dt))/2/H2*dt;

    Vt2(k)=Vt(1);
    Ef2(k)=KA2*(V2ref-Vt2(k-1)+KD*w2(k-1));
    E2(k)=E2(k-1)+1/T3*(K3*(Ef2(k-1)-K4*delta2(k-1))-E2(k-1))*dt;
    if E2(k) > 1.1
        E2(k) = 1.1;
    elseif E2(k) < 0.9
        E2(k)=0.9;
    end
end
% Calculate frequencies
f1nrm=(w1+1)*w0/2/pi;
Vt1nrm=Vt1*480/sqrt(3);
f2nrm=(w2+1)*w0/2/pi;
Vt2nrm=Vt2*480/sqrt(3);

%
%     % Plot frequencies
%     figure;
%     subplot(2,1,1);
%     plot(time, f1nrm);
% %      xlim([0 30]);
% %      ylim([59.995 60.005]);
%     xlabel('Time(Sec)');
%     ylabel('f1nrm(Hz)');
%     subplot(2,1,2);
%     plot(time, f2nrm);
%     xlim([0 30]);
%     ylim([59.5 60.5]);
%     xlabel('Time(Sec)');
%     ylabel('f2nrm(Hz)');
% %     subplot(3,1,3);
% %     plot(time, f3nrm);
% % %     xlim([0 4]);
% % %     ylim([59.997 60.003]);
% %     xlabel(’Time(Sec)’);
% %     ylabel(’f3nrm(Hz)’);
%     %========================================
%     figure;
%     subplot(2,1,1);
%     plot(time, Vt1nrm);
% %    xlim([0 30]);
% %    ylim([260 300]);
%     xlabel(’Time(Sec)’);
%     ylabel(’Vt1nrm(V)’);
%     subplot(2,1,2);
%     plot(time, Vt2nrm);
% %    xlim([0 30]);
% %    ylim([260 300]);
%     xlabel(’Time(Sec)’);
%     ylabel(’Vt2nrm(V)’);
% %     subplot(3,1,3);
% %     plot(time, Vt3nrm);
% % %    xlim([0 10]);
% % %    ylim([0 400]);
% %     xlabel(’Time(Sec)’);
% %     ylabel(’Vt3nrm(V)’);
```

## Appendix B: MATLAB Code Three Machine System Simulation

```
 % Simulate a three-machine system dynamics loss of machine 1 at 10sec
clear all;
format long;
clc;
% Machine 1 represents one DG unit:
H1=50;              % Inertia
X1=j*0.1;           % Internal reactance
% Machine 2 represents the second DG unit
H2=0.1;               % Inertia
X2=j*0.1;           % Internal reactance
% Machine 3 represents the third DG unit
H3=0.1;               % Inertia
X3=j*0.1;           % Internal reactance
% Loads
R1=2;            % Represents loads
% Link between machine 1 and machine 2
XL1=j*0.1;           % Link reactance
% Link between machine 2 and machine 3
XL2=j*0.1;           % Link reactance
% Construct admittance matrix
A=[1/X1 0 0; 0 1/X2 0; 0 0 1/X3];
B=[-1/X1 0 0; 0 -1/X2 0;0 0 -1/X3];
C=[-1/X1 0 0; 0 -1/X2 0;0 0 -1/X3];
D=[1/R1+1/X1+1/XL1 -1/XL1 0; -1/XL1 1/X2+1/XL1+1/XL2 -1/XL2;
0 -1/XL2 1/XL2+1/X3];

% Construct reduced admittance matrix
```

```
y=A-B*inv(D)*C;

% Calculate power

% EG0=[cos(-1.433*pi/180)+j*sin(-1.433*pi/180); 1];

% Initial E1 and E2

EG0=[cos(0*pi/180)+j*sin(0*pi/180); 1; 1];

IG0=y*EG0;

Pe0=real(EG0.*conj(IG0));

% Calculate terminal voltage

Vt0=abs(EG0-[X1;X2;X3].*IG0);

% Define step size

dt=0.0001;

% Define simulation length

t=30;

% Define fault occuring time

t1=10;

%Calculate size of arrays

s=round(t/dt);

% Declear arrays

time=zeros(s,1);

delta1=zeros(s,1);

w1=zeros(s,1);

Pe1=zeros(s,1);

Pm1=zeros(s,1);

Vt1=zeros(s,1);

Ef1=zeros(s,1);

E1=zeros(s,1);

delta2=zeros(s,1);

w2=zeros(s,1);

Pe2=zeros(s,1);
```

```
Pm2=zeros(s,1);
Vt2=zeros(s,1);
Ef2=zeros(s,1);
E2=zeros(s,1);
delta3=zeros(s,1);
w3=zeros(s,1);
Pe3=zeros(s,1);
Pm3=zeros(s,1);
Vt3=zeros(s,1);
Ef3=zeros(s,1);
E3=zeros(s,1);
% Parameter D is the load damping constant.
% D is defined as how many percent the load will drop if
% the frequency is reduced by 1 percent.
% D is very small for this case, so 0 is used.
D1=0.0;
D2=0.0;
D3=0.0;
% Parameters
R=0.00;
K=200;
T3=2.4;
K3=0.3;
K4=1.4;
KD=21;%3.5;
% Define nominal angular velocity
w0=120*pi;
% Get initial values
delta1(1)=angle(EG0(1));
```

```
w1(1)=0;
Pe1(1)=Pe0(1);
Pm1(1)=Pe0(1);
Vt1(1)=Vt0(1);
E1(1)=abs(EG0(1));
Ef1(1)=E1(1)/K3;
delta2(1)=angle(EG0(2));
w2(1)=0;
Pe2(1)=Pe0(2);
Pm2(1)=Pe0(2);
Vt2(1)=Vt0(2);
E2(1)=abs(EG0(2));
Ef2(1)=E2(1)/K3;
delta1(3)=angle(EG0(3));
w3(1)=0;
Pe3(1)=Pe0(3);
Pm3(1)=Pe0(3);
Vt3(1)=Vt0(3);
E3(1)=abs(EG0(3));
Ef3(1)=(E3(1)/K3);
% References
P1ref=Pe0(1);
P2ref=Pe0(2);
P3ref=Pe0(3);
V1ref=1.001;
V2ref=V1ref;
V3ref=V1ref;
% Calculated parameters
KA1=E1(1)/(V1ref-Vt0(1))/K3;
```

```
KA2=E2(1)/(V2ref-Vt0(2))/K3;
KA3=E3(1)/(V3ref-Vt0(3))/K3;
% Solve equations
    % Normal operation for 10 seconds
for k=2:round(t1/dt)
    time(k)=dt*(k-1);
    EG=[E1(k-1)*exp(j*delta1(k-1)); E2(k-1)*exp(j*delta2(k-1));
    E3(k-1)*exp(j*delta3(k-1))];

    IG=y*EG;
    Pe=real(EG.*conj(IG));
    Vt=abs(EG-[X1;X2;X3].*IG);
    delta1(k)=delta1(k-1)+w1(k-1)*w0*dt;
    Pe1(k)=Pe(1);
    Pm1(k)=Pm1(k-1)+K*(R*P1ref-R*Pm1(k-1)-w1(k-1))*dt
    if Pm1(k) > 1
        Pm1(k) = 1;
    elseif Pm1(k) < 0
        Pm1(k) = 0;
    end
    w1(k)=w1(k-1)+(Pm1(k-1)-Pe1(k-1)-D1*w1(k-1))/2/H1*dt;
    Vt1(k)=Vt(1);
    Ef1(k)=KA1*(V1ref-Vt1(k-1)+KD*w1(k-1));
    E1(k)=E1(k-1)+1/T3*(K3*(Ef1(k-1)-K4*delta1(k-1))-E1(k-1))*dt;
    if E1(k) > 1.1
        E1(k) = 1.1;
    elseif E1(k) < 0.9
        E1(k)=0.9;
    end
```

```
delta2(k)=delta2(k-1)+w2(k-1)*w0*dt;
Pe2(k)=Pe(2);
Pm2(k)=Pm2(k-1);%+K*(R*P2ref-R*Pm2(k-1)-w2(k-1))*dt;
if Pm2(k) > 1
    Pm2(k) = 1;
elseif Pm2(k) < 0
    Pm2(k) = 0;
end
w2(k)=w2(k-1)+(Pm2(k-1)-Pe2(k-1)-D2*w2
(k-1)+0.1*P2ref*sin(2*pi*k*dt))/2/H2*dt;

Vt2(k)=Vt(2);
Ef2(k)=KA2*(V2ref-Vt2(k-1)+KD*w2(k-1));
E2(k)=E2(k-1)+1/T3*(K3*(Ef2(k-1)-K4*delta2(k-1))-E2(k-1))*dt;
if E2(k) > 1.1
    E2(k) = 1.1;
elseif E2(k) < 0.9
    E2(k)=0.9;
end
delta3(k)=delta3(k-1)+w3(k-1)*w0*dt;
Pe3(k)=Pe(3);
Pm3(k)=Pm3(k-1);
if Pm3(k) > 1
    Pm3(k) = 1;
elseif Pm3(k) < 0
    Pm3(k) = 0;
end
w3(k)=w3(k-1)+(Pm3(k-1)-Pe3(k-1)-D3*w3(k-1))/2/H3*dt;
Vt3(k)=Vt(3);
```

```
    Ef3(k)=KA3*(V3ref-Vt3(k-1)+KD*w3(k-1));
    E3(k)=E3(k-1)+1/T3*(K3*(Ef3(k-1)-K4*delta3(k-1))-E3(k-1))*dt;
    if E3(k) > 1.1
        E3(k) = 1.1;
    elseif E3(k) < 0.9
        E3(k)=0.9;
    end
end
%Load shedding

R0=Vt1(1)^2/Pe1(1);
R_new=R0*R1/(R0-R1);

% Construct new admittance matrix
A=[1/X2 0 ;0 1/X3];
B=[-1/X2 0; 0 -1/X3];
C=[-1/X2 0;0 -1/X3];
D=[1/(R_new+XL1)+1/XL2+1/X2 -1/XL2;-1/XL2 1/XL2+1/X3];
% Construct reduced admittance matrix
y=A-B*inv(D)*C;

% Loss of machine 1 at 10 sec
for k=round(t1/dt)+1:s
    time(k)=dt*(k-1);
    EG=[E2(k-1)*exp(j*delta2(k-1));E3(k-1)*exp(j*delta3(k-1))];
    IG=y*EG;
    Pe=real(EG.*conj(IG));
    Vt=abs(EG-[X2;X3].*IG);
    delta2(k)=delta2(k-1)+w2(k-1)*w0*dt;
```

```
Pe2(k)=Pe(1);
Pm2(k)=Pm2(k-1);%+K*(R*P2ref-R*Pm2(k-1)-w2(k-1))*dt;
if Pm2(k) > 1
    Pm2(k) = 1;
elseif Pm2(k) < 0
    Pm2(k) = 0;
end
w2(k)=w2(k-1)+(Pm2(k-1)-Pe2(k-1)-D2*w2
(k-1)+0.1*P2ref*sin(2*pi*k*dt))/2/H2*dt;

Vt2(k)=Vt(1);
Ef2(k)=KA2*(V2ref-Vt2(k-1)+KD*w2(k-1));
E2(k)=E2(k-1)+1/T3*(K3*(Ef2(k-1)-K4*delta2(k-1))-E2(k-1))*dt;
if E2(k) > 1.1
    E2(k) = 1.1;
elseif E2(k) < 0.9
    E2(k)=0.9;
end
delta3(k)=delta3(k-1)+w3(k-1)*w0*dt;
Pe3(k)=Pe(2);
Pm3(k)=Pm3(k-1);
if Pm3(k) > 1
    Pm3(k) = 1;
elseif Pm3(k) < 0
    Pm3(k) = 0;
end
w3(k)=w3(k-1)+(Pm3(k-1)-Pe3(k-1)-D3*w3(k-1))/2/H3*dt;
Vt3(k)=Vt(2);
Ef3(k)=KA3*(V3ref-Vt3(k-1)+KD*w3(k-1));
```

```
        E3(k)=E3(k-1)+1/T3*(K3*(Ef3(k-1)-K4*delta3(k-1))-E3(k-1))*dt;
        if E3(k) > 1.1
            E3(k) = 1.1;
        elseif E3(k) < 0.9
            E3(k)=0.9;
        end
    end

    f1nrm=(w1+1)*w0/2/pi;
    Vt1nrm=Vt1*480/sqrt(3);
    f2nrm=(w2+1)*w0/2/pi;
    Vt2nrm=Vt2*480/sqrt(3);
    f3nrm=(w3+1)*w0/2/pi;
    Vt3nrm=Vt3*480/sqrt(3);
   % Plot frequencies
    figure;
    subplot(3,1,1);
    plot(time, f1nrm);
%     xlim([0 4]);
%     ylim([59.997 60.003]);
    xlabel('Time(Sec)');
    ylabel('f1nrm(Hz)');
    subplot(3,1,2);
    plot(time, f2nrm);
%     xlim([0 4]);
%     ylim([59.997 60.003]);
    xlabel('Time(Sec)');
    ylabel('f2nrm(Hz)');
    subplot(3,1,3);
    plot(time, f3nrm);
%     xlim([0 4]);
%     ylim([59.997 60.003]);
    xlabel('Time(Sec)');
    ylabel('f3nrm(Hz)');
    %=====================================
    figure;
    subplot(3,1,1);
    plot(time, Vt1nrm);
%    xlim([0 10]);
%    ylim([0 400]);
    xlabel('Time(Sec)');
    ylabel('Vt1nrm(V)');
    subplot(3,1,2);
    plot(time, Vt2nrm);
%    xlim([0 10]);
%    ylim([0 400]);
    xlabel('Time(Sec)');
    ylabel('Vt2nrm(V)');
    subplot(3,1,3);
    plot(time, Vt3nrm);
%    xlim([0 10]);
%    ylim([0 400]);
    xlabel('Time(Sec)');
    ylabel('Vt3nrm(V)');
```

## Appendix C: MATLAB Code Multiple Machine System Simulation

```
      % Simulate a multiple-machine system dynamics loss of
  % machine 1 at 10sec
    function [fnrm,time] =sim3AUTO(x)

%     clear all;
%     format long;
%     clc;
%
    N=7;
    if N==2
        H1=50;
        m2smls2(H1);
    else

        H=zeros(N,1);
        X=zeros(N,1);
        XL=zeros(N-1,1);
        A=zeros(N,N);
        B=zeros(N,N);
        C=zeros(N,N);
        D=zeros(N,N);

        H(1,1)=50;
        X(1,1)=j*0.1;
        R1=2;
        for m=2:N
            H(m,1)=0.1;
            X(m,1)=j*0.1;
        end
        for m=1:N-1
            XL(m,1)=j*0.1;
        end

        % Construct admittance matrix
        % Construct matrices A,B,and C
        for n=1:N
            A(n,n)=1/X(n,1);
            B(n,n)=-1/X(n,1);
            C(n,n)=-1/X(n,1);
        end

        % Construct matrix D

        for n=2:N-1
            %Diagonal exept elements (1,1) & (N,N)

            D(n,n)=1/X(n,1)+1/XL(n,1)+1/XL(n-1,1);
            D(n,n-1)=-1/XL(n-1,1);
            D(n-1,n)=-1/XL(n-1,1);
            D(N,N-1)=-1/XL(n-1,1);
            D(N-1,N)=-1/XL(n-1,1);
        end

        % D elements (1,1) and (N,N)

        D(1,1)=1/R1+1/X(1,1)+1/XL(1,1);
```

```
D(N,N)=1/XL(N-1)+1/X(N);

% Construct reduced admittance matrix
y=A-B*inv(D)*C;
% Calculate power
% EG0=[cos(-1.433*pi/180)+j*sin(-1.433*pi/180); 1];

%Initial E1 and E2
EG0=zeros(N,1);

for n=1:N
    if n==1
       % Initial E1 and E2
        EG0(n,1)=cos(0*pi/180)+j*sin(0*pi/180);
    else
        EG0(n,1)=1;
    end
end
IG0=y*EG0;
Pe0=real(EG0.*conj(IG0));
% Calculate terminal voltage
Vt0=abs(EG0-X.*IG0);


% Define step size
dt=0.0001;
% Define simulation length
t=30;
% Define fault occuring time
```

```
t1=10;
%Calculate size of arrays
s=round(t/dt);

% Declare arrays
time=zeros(s,1);
delta=zeros(s,N);
w=zeros(s,N);
Pe=zeros(s,N);
Pm=zeros(s,N);
Vt=zeros(s,N);
Ef=zeros(s,N);
E=zeros(s,N);


Pref=zeros(N,1);
Vref=zeros(N,1);
Vref(1,1)=1.001;

% Parameters
R=0.00;
K=200;
T3=2.4;
K3=0.3;
K4=1.4;
KD=21;%3.5;
KA=zeros(N,1);
% Define nominal angular velocity
w0=120*pi;
```

```
D_vec=zeros(N,1);
% Get initial values
for n=1:N
    % Parameter D is the load damping constant.
    % D is defined as how many percent the load will
    5drop if the frequency is reduced by 1 percent.
    % D is very small for this case, so 0 is used.
    D_vec(n)=0.0;
    delta(1,n)=angle(EG0(n));
    w(1,n)=0;
    Pe(1,n)=Pe0(n);
    Pm(1,n)=Pe0(n);
    Vt(1,n)=Vt0(n);
    E(1,n)=abs(EG0(n));
    Ef(1,n)=E(1,n)/K3;

    %References
    Pref(n)=Pe0(n);
    Vref(n+1)=Vref(1,1);
    % Calculated parameters
    KA(n)=E(1,n)/(Vref(n)-Vt0(n))/K3;
end

 EG=zeros(N,1);
% Solve equations
% Normal operation for 10 seconds
for k=2:round(t1/dt)
    time(k)=dt*(k-1);
    for m=1:N
    EG(m,1)=E(k-1,m)*exp(1i*delta(k-1,m));
end
IG=y*EG;
Pe_vec=real(EG.*conj(IG));
Vt_vec=abs(EG-X.*IG);

for n=1:N
    delta(k,n)=delta(k-1,n)+w(k-1,n)*w0*dt;
    Pe(k,n)=Pe_vec(n);
    if n==1
        Pm(k,n)=Pm(k-1,n)+K*(R*Pref(n)-R*Pm(k-1,n)-
        w(k-1,n))*dt;%Pm1(k-1)-K*w1(k-1)*dt;
    else
        Pm(k,n)=Pm(k-1,n);
    end
    if Pm(k,n) > 1
        Pm(k,n) = 1;
    elseif Pm(k,n) < 0
        Pm(k,n) = 0;
    end

    if n==2
        w(k,n)=w(k-1,n)+(Pm(k-1,n)-Pe(k-1,n)-D_vec(n)*
        w(k-1,n)+0.1*Pref(n)*sin(2*pi*k*dt))/2/H(n)*dt;
    else
        w(k,n)=w(k-1,n)+(Pm(k-1,n)-Pe(k-1,n)-D_vec(n)*
        w(k-1,n))/2/H(n)*dt;
    end
```

```
            Vt(k,n)=Vt_vec(n);
            Ef(k,n)=KA(n)*(Vref(n)-Vt(k-1,n)+KD*w(k-1,n));
            E(k,n)=E(k-1,n)+1/T3*(K3*(Ef(k-1,n)-K4*delta(k-1,n))-
            E(k-1,n))*dt;

            if n~=N
                if E(k,n) > 1.1
                    E(k,n) = 1.1;
                elseif E(k,n) < 0.9
                    E(k,n)=0.9;
                end
            elseif n==N
                if E(k,n) > 1.1
                    E(k,n) = 1.1;
                elseif E(k,n-1) < 0.9
                    E(k,n)=0.9;
                end
            end

        end
    end

%     f1nrm=(w(:,1)+1)*w0/2/pi;
%     f2nrm=(w(:,2)+1)*w0/2/pi;
%     f3nrm=(w(:,3)+1)*w0/2/pi;
% %     f4nrm=(w(:,4)+1)*w0/2/pi;
%
%     figure;
%     subplot(4,1,1);
%    plot(time, f1nrm);
% %   xlim([0 4]);
% %   ylim([59.997 60.003]);
%    xlabel('Time(Sec)');
%    ylabel('f1nrm(Hz)');
%    subplot(4,1,2);
%    plot(time, f2nrm);
% %   xlim([0 4]);
% %   ylim([59.997 60.003]);
%    xlabel('Time(Sec)');
%    ylabel('f2nrm(Hz)');
%    subplot(4,1,3);
%    plot(time, f3nrm);
% %   xlim([0 4]);
% %   ylim([59.997 60.003]);
%    xlabel('Time(Sec)');
%    ylabel('f3nrm(Hz)');
% %   subplot(4,1,4);
% %   plot(time, f4nrm);
% % %  xlim([0 4]);
% % %  ylim([59.997 60.003]);
% %   xlabel('Time(Sec)');
% %   ylabel('f4nrm(Hz)');


    %Load shedding


    R0=Vt(1,1)^2/Pe(1,1);
    R_new=R0*R1/(R0-R1);
    % Construct new admittance matrix
    N=N-1;
    A=zeros(N,N);
    B=zeros(N,N);
    C=zeros(N,N);
    D=zeros(N,N);
    % Construct new matrices A,B,and C
    for n=1:N
        A(n,n)=1/X(n+1,1);
        B(n,n)=-1/X(n+1,1);
        C(n,n)=-1/X(n+1,1);
    end

    if N==2
        D=[1/(R_new+XL(1,1))+1/XL(2,1)+1/X(2,1) -1/XL(2,1);
        -1/XL(2,1) 1/XL(2,1)+1/X(3,1)];
    else
    % Construct matrix D

        for n=2:N-1
           %Diagonal exept elements (1,1) & (N,N)
D(n,n)=1/X(n+1,1)+1/XL(n+1,1)+1/XL(n,1);
D(n,n-1)=-1/XL(n,1);
            D(n-1,n)=-1/XL(n,1);
            D(N,N-1)=-1/XL(n+1,1);
            D(N-1,N)=-1/XL(n+1,1);
        end
            % D elements (1,1) and (N,N)

            D(1,1)=1/(R_new+XL(1,1))+1/X(N-1,1)+1/XL(N-1,1);
            D(N,N)=1/XL(N)+1/X(N+1);
    end
    %Re-initialize matrix X to meet dimentions
    X=zeros(N,1);
    for n=1:N
        X(n,1)=1i*0.1;
    end
    % Construct reduced admittance matrix
    y=A-B*inv(D)*C;

    EG=zeros(N,1);
    % Loss of machine 1 at 10 sec
    for k=round(t1/dt)+1:s
        time(k)=dt*(k-1);
        for m=1:N
            EG(m,1)=E(k-1,m+1)*exp(1i*delta(k-1,m+1));
        end
        IG=y*EG;
        Pe_vec=real(EG.*conj(IG));
        Vt_vec=abs(EG-X.*IG);

        for n=2:N+1
            delta(k,n)=delta(k-1,n)+w(k-1,n)*w0*dt;
            Pe(k,n)=Pe_vec(n-1);
            Pm(k,n)=Pm(k-1,n);
            if Pm(k,n) > 1
```

```
            Pm(k,n) = 1;
        elseif Pm(k,n) < 0
            Pm(k,n) = 0;
        end

        if n==2
            w(k,n)=w(k-1,n)+(Pm(k-1,n)-Pe(k-1,n)-D_vec(n)*
            w(k-1,n)+0.1*Pref(n)*sin(2*pi*k*dt))/2/H(n)*dt;
        else
            w(k,n)=w(k-1,n)+(Pm(k-1,n)-Pe(k-1,n)-D_vec(n)*
            w(k-1,n))/2/H(n)*dt;
        end

        Vt(k,n)=Vt_vec(n-1);
        Ef(k,n)=KA(n)*(Vref(n)-Vt(k-1,n)+KD*w(k-1,n));
        E(k,n)=E(k-1,n)+1/T3*(K3*(Ef(k-1,n)-K4*delta(k-1,n))-
        E(k-1,n))*dt;

        if E(k,n) > 1.1
                E(k,n) = 1.1;
        elseif E(k,n) < 0.9
                E(k,n)=0.9;
        end

    end
end
fnrm=zeros(s,N+1);
Vtnrm=zeros(s,N+1);
```

```
        for n=1:N+1
           fnrm(:,n)=(w(:,n)+1)*w0/2/pi;
           Vtnrm(:,n)=Vt(:,n)*480/sqrt(3);
        end

%         %PLOT FREQUENCIES
%
%             figure;
%             subplot(4,1,1);
%             plot(time, fnrm(:,1));
% %             xlim([0 30]);
% %             ylim([58.997 60.997]);
%             xlabel('Time(Sec)');
%             ylabel('f1nrm(Hz)');
%             subplot(4,1,2);
%             plot(time, fnrm(:,2));
% %             xlim([0 30]);
% %             ylim([58.997 60.997]);
%             xlabel('Time(Sec)');
%             ylabel('f2nrm(Hz)');
%             subplot(4,1,3);
%             plot(time, fnrm(:,3));
% %             xlim([0 30]);
% %             ylim([58.997 60.997]);
%             xlabel('Time(Sec)');
%             ylabel('f3nrm(Hz)');
%             subplot(4,1,4);
%             plot(time, fnrm(:,4));
% %             xlim([0 30]);
% %         ylim([58.997 60.997]);
%          xlabel(’Time(Sec)’);
%          ylabel(’f4nrm(Hz)’);
%
%          figure;
%          subplot(4,1,1);
%          plot(time, fnrm(:,5));
% %         xlim([0 30]);
% %         ylim([58.997 60.997]);
%          xlabel(’Time(Sec)’);
%          ylabel(’f5nrm(Hz)’);
%          subplot(4,1,2);
%          plot(time, fnrm(:,6));
% %         xlim([0 30]);
% %         ylim([58.997 60.997]);
%          xlabel(’Time(Sec)’);
%          ylabel(’f6nrm(Hz)’);
%          subplot(4,1,3);
%          plot(time, fnrm(:,7));
% %         xlim([0 30]);
% %         ylim([58.997 60.997]);
%          xlabel(’Time(Sec)’);
%          ylabel(’f7nrm(Hz)’);
%
%
%      %  PLOT VOLTAGES
%
%          figure;
%          subplot(4,1,1);
%             plot(time, Vtnrm(:,1));
% %            xlim([0 10]);
% %            ylim([0 400]);
%             xlabel('Time(Sec)');
%             ylabel('Vt1nrm(V)');
%             subplot(4,1,2);
%             plot(time, Vtnrm(:,2));
% %            xlim([0 10]);
% %            ylim([0 400]);
%             xlabel('Time(Sec)');
%             ylabel('Vt2nrm(V)');
%             subplot(4,1,3);
%             plot(time, Vtnrm(:,3));
% %            xlim([0 10]);
% %            ylim([0 400]);
%             xlabel('Time(Sec)');
%             ylabel('Vt3nrm(V)');
%             subplot(4,1,4);
%             plot(time, Vtnrm(:,4));
%         %    xlim([0 10]);
%         %    ylim([0 400]);
%             xlabel('Time(Sec)');
%             ylabel('Vt4nrm(V)');
%         %
%             figure;
%             subplot(4,1,1);
%             plot(time, Vtnrm(:,5));
%         %    xlim([0 10]);
%         %    ylim([0 400]);
%             xlabel('Time(Sec)');
%             ylabel('Vt5nrm(V)');
%             subplot(4,1,2);
%             plot(time, Vtnrm(:,6));
%         %    xlim([0 10]);
%         %    ylim([0 400]);
%             xlabel('Time(Sec)');
%             ylabel('Vt6nrm(V)');
%             subplot(4,1,3);
%             plot(time, Vtnrm(:,7));
%         %    xlim([0 10]);
%         %    ylim([0 400]);
%             xlabel('Time(Sec)');
%             ylabel('Vt7nrm(V)');
    end
```

## Appendix D: MATLAB Code 1Hz Component Calculation

```
    %Calculationg de 1hz component

clear all;
format long;
clc;

[v_t,time]=sim3AUTO(7);
%[v_t,time]=m2smls2(50) ; %Two machine system
dt=0.0001;
t=30;
t1=10;
s=round(t/dt);
index=round(1/dt*1/36);
v_tnew=zeros(s/index,1);
time_new=zeros((s/index)-35,1);
v_t36=zeros(36,1);
n=1;
i=1;

% Y_s=zeros(s,1);
% Y_c=zeros(s,1);
magsq=zeros((s/index)-35,1);

v_c=zeros(36,1);
v_s=zeros(36,1);
v_60=zeros(36,1);
```

```
for m=1:36
    v_c(m,1)=cos(m*pi/18);
    v_s(m,1)=sin(m*pi/18);
    v_60(m,1)=60;
end

while (n<s)
%         v_tnew(i,1)=v_t(n,1); % Column corresponds to Machine # =2
        v_tnew(i,1)=v_t(n,2);% Column corresponds to Machine # >2
        time_new(i,1)=time(n,1);
        n=n+index;
        i=i+1;
end

j=1;
k=1;
i=1;
while k<1045

    for knew=k:k+35
        v_t36(i,1)=v_tnew(knew,1);
        i=i+1;
    end

    i=1;
    Y_s=(2/36)*dot(v_t36-v_60,v_s);
    Y_c=(2/36)*dot(v_t36-v_60,v_c);
    magsq(j+36,1)=Y_c^2+Y_s^2;
    j=j+1;
    k=j;

end

p=324*magsq;
figure;
subplot(2,1,1);
%plot(time, v_t(:,1));%Plot frequency of Machine # = 2
plot(time, v_t(:,2));%Plot frequency of Machine # > 2
%     xlim([0 4]);
%     ylim([59.997 60.003]);
xlabel('Time(Sec)');
ylabel('f2nrm(Hz)');
subplot(2,1,2);
plot(time_new, p(:,1));
%     xlim([0 4]);
%     ylim([59.997 60.003]);
xlabel('Time(Sec)');
ylabel('Index rho');
```