

# Optimizing Control of a Power System during an Emergency

Karen Kyeyune



**LUND**  
UNIVERSITY

Department of Automatic Control

MSc Thesis  
ISRN LUTFD2/TFRT--5960--SE  
ISSN 0280-5316

Department of Automatic Control  
Lund University  
Box 118  
SE-221 00 LUND  
Sweden

© 2014 by Karen Kyeyune. All rights reserved.  
Printed in Sweden by Media-Tryck  
Lund 2014

# Abstract

Population growth, infrastructure and economy puts pressure and demand on the existing power supplies. It puts strains on the current power systems which causes instabilities in the systems. This is an ongoing challenge which needs an immediate solution. The objective of this thesis is voltage stability. This is examined with the help of constructing a small power system using a programming language called *Matlab*. Optimization tools provided by Matlab are used to find the maximum possible pre-contingency load, while still maintaining a stable system. To find feasible solutions in Matlab, system models, such as load models and power line models are simplified.

The results show that a system which has experienced a fault can successfully recover by using a linear load recovery model and an exponential load recovery model. Certain constraints, such as generator ramping and limitations on the field voltages in the generators are implemented. Feasible solutions are found although constraints might have made it more difficult under the course of this study.

These findings are rough approximations of how a small power system can operate. Though, this can give valuable information on how a more complex system might act before and after a contingency as well as suitable recovery paths.

Although the thesis is more suited for those who have some knowledge in control or power systems, a reader without a technical background can enjoy the paper too.



# Acknowledgements

I am sincerely grateful for the guidance and supervision of my two mentors, Prof. Dr. Anders Rantzer and Dr. Magnus Perninge at the Department of Automatic Control have given me during this project. Their advice and wisdom have been valuable and helped me to complete my thesis.

A special thanks to my family and friends who have supported me throughout my university education. Their love and encouragement have helped me to grow into a stronger and determined person and engineer.



# Contents

<b>1. Introduction</b>	<b>12</b>
1.1 Historical Background . . . . .	12
1.2 Motivation and Goal . . . . .	13
1.3 Tools . . . . .	14
<b>2. Network Models of Power Systems</b>	<b>15</b>
2.1 Lines, Loads and Generators . . . . .	15
2.2 Nodal Formulation of the Network Equations . . . . .	20
2.3 Basic Power Flow Problem . . . . .	21
2.4 Power System Stability . . . . .	23
<b>3. The Optimization Problem</b>	<b>31</b>
3.1 Implemented load models . . . . .	33
3.2 Solving the Optimization Problem . . . . .	35
<b>4. Implementation</b>	<b>39</b>
4.1 Overview . . . . .	39
4.2 The Seven-Node System . . . . .	41
4.3 Complementary Building Blocks and Matlab-files . . . . .	42
4.4 Fsolve and Fmincon . . . . .	44
4.5 Nose Curve demonstration on the Seven-Node System . . . . .	48
<b>5. Results</b>	<b>50</b>
5.1 Pre-contingency . . . . .	50
5.2 Post-contingency . . . . .	50
5.3 At end time T=5 . . . . .	50
5.4 Pre, Post and time interval . . . . .	51
<b>6. Discussion</b>	<b>56</b>
<b>7. Conclusion</b>	<b>58</b>
<b>8. Future work</b>	<b>59</b>
<b>9. Appendix</b>	<b>60</b>
9.1 Per Unit System . . . . .	60
9.2 Figures . . . . .	61





# List of abbreviations and keywords

Definition of abbreviations and keywords used in the study.

- Contingency: An event, e.g. a faulty line, that might happen in the future which causes problems in the network
- Pre-contingency load: Load before a fault.
- Post-contingency load: Load after a fault.
- Voltage collapse: When the voltage in a system is so low that it decreases to zero leading to a blackout.
- pre: Pre-contingency (before a fault has occurred).
- post: Post-contingency (after a fault has occurred).
- m-file: A Matlab file.

# 1

## Introduction

The Chapter explains the historical information about a specific Swedish power failure. It also explains the objective and methods used, as well as the outline of the report.

### 1.1 Historical Background

In 1996 the Swedish electricity market was deregulated to introduce competition for trade and production of electricity, according to [ 2014]. Today, the Nordic countries, except Iceland, have a common electricity market. *Swedish Energy*, a trade association which consists of 380 companies in the energy sector, have a goal of having a single European electricity market [ 2014]. However it is not possible to choose the network operator that distributes power, even with the reforms made in the 90's since one single company may have monopoly of a local power grid.

The infrastructure of an electrical grid<sup>1</sup> is of importance for stability. A large part of Swedish electricity is from hydropower plants which are mainly located in northern Sweden, while the highest demand is in the south.

23rd of September, 2003, southern Sweden was hit by a blackout. 1,5 million people were without electricity, some for 5 hours, which resulted in an estimated power loss of 10 million kWh and economical loss of 500 MSEK. The power failure was due to the several reasons. The main triggers were; an internal fault in the nuclear power plant at Oskarshamn and a switchyard failure at Horred close to Varberg. These events put an extreme stress on the backbone of the power system which caused a voltage collapse<sup>2</sup> in Southern Sweden, the south link between Varberg and Norrköping, and Danish Själland and Bornholm [ 2014].

There are various ways of preventing blackouts. One option is to build new lines and spread out the distribution. However, this might be quite expensive. A

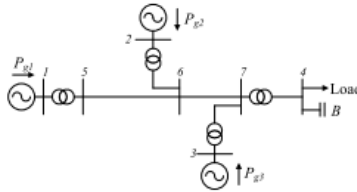
---

<sup>1</sup> An electrical grid is an interconnected network which delivers electricity to consumers from suppliers.

<sup>2</sup> A voltage collapse occurs when the voltage in the system is so low that it can not be saved or increased to a stable level. A voltage collapse leads to a blackout.

more economical approach is to enforce a control system which can be triggered whenever an unusual event occurs in the power system.

A simple example of a power system is shown in Figure 1.1. This power system will be explained greater in a later chapter. It might be useful to give the reader some introduction to components in the system with the help of Figure 1.1. The system in Figure 1.1 contains three circular symbols ( $P_{g1}$ ,  $P_{g2}$  and  $P_{g3}$ ) in the system which symbolize generators. The smaller rings which are connected to each other are transformers. A line can be defined between two busbars and a load is described as an arrow. The two plates against each other ( $B$ ) is a capacitor. How some of these components function will be explained in the Chapter 2.



**Figure 1.1** An example of a power system from [Karystianos et al., 2007]. The power system consists of three generators  $P_{g1}$ ,  $P_{g2}$  and  $P_{g3}$  that are connected to the network via transformer. The transformers are positioned between busbars 1-5, 2-6, 3-7 and 4-7. The load and the capacitor ( $B$ ) are in parallel and connected to busbar 4.

## 1.2 Motivation and Goal

As mentioned in the *Background*, a blackout can be prevented either by reinforcing the power system or by using control actions. This challenge is of great interest since electricity is a vital resource in todays society.

This thesis focuses on control, especially preventing voltage collapses. The interest in the study lies in using control actions as a way of preventing blackouts rather than reinforcing the network, since building new lines is generally regarded as expensive.

Course literature and research papers were used to extract knowledge in this field. Lecture scripts [Andersson and Franck, 2012] and [Andersson, 2012] gave a basic understanding while more advanced theory came from articles such as [Hamon et al., 2013], [Eriksson et al., 2011] and [Perninge, 2013]. Equations taken from lecture notes and articles are presented in Chapter 2.

Studying a small system before and after a fault, might reveal that certain control actions can prevent the system from a total collapse. Stability assessments can also be done to find post-contingency corrective control, i.e. control actions that will

save a system from partly or complete failure after a fault. Paper [Perninge, 2014] examines post-contingency control in a security-constrained optimal power flow. The paper investigates stability after a fault while having certain constraints on the system. The restrictions may be on the generators' production or the field voltage in the generators. This point of view is interesting and relevant in this project. The main focus will be on the pre-contingency load which is the load before the fault occurs. The goal is to maximize this load and still have a stable system after a contingency.

## 1.3 Tools

Most of the programming was done in Matlab. Verifications of the results were also done in Matlab.

### Matlab

*Matlab* is a program used for numerical computation, visualization and programming. Optimization-tools provided by Matlab were essential in the thesis. The two main tools were *fsolve* and *fmincon*. Both these commands can solve nonlinear equations. The main difference between the commands is that *fmincon* solves optimization problems while *fsolve* solves a systems of equations. Another thing which sets the tools apart is possibility to have constraints. *Fsolve* tries to find a static equilibrium with system equations while *fmincon* minimizes an objective function with static equilibrium while satisfying constraints. How these tools were used will be profoundly explained in Chapter 4.

# 2

## Network Models of Power Systems

There are different ways to model power systems using power flow equations for various situations. This chapter will give an overview of common components in a power system and the models used in this master thesis. Most equations are taken from lecture notes [Andersson and Franck, 2012] and [Andersson, 2012], therefore all terminologies and details of power systems will not be explained but can be found in the references.

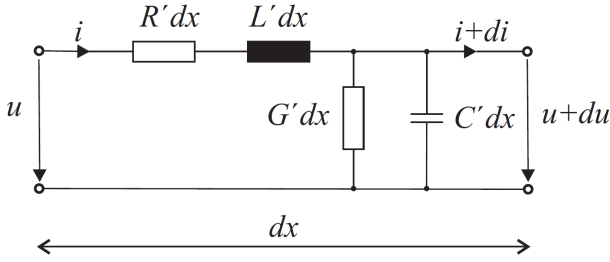
### 2.1 Lines, Loads and Generators

#### Lines

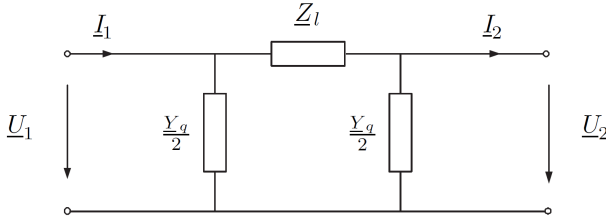
In Figure 1.1, a power line is represented by a straight line (e.g. line 5-6 which is connected to busbars 5 and 6). The lines in the figure are connected to busbars. An equivalent circuit diagram of a line element, which can be seen in Figure 2.1, is derived from a single line element. An infinitesimal line element is considered where voltages and currents can be described by a differential equation of the assumed homogeneous line. By using Maxwell's field theory, the electromagnetic effect of the power line can be analyzed. The per unit length capacitance and inductance are assumed constant along the line and the characteristic of the power line is the propagation of a wave [Andersson and Franck, 2012].

Depending on the length of the line, different assumptions can be introduced. One useful assumption is the lossless power line i.e  $R'=G'=0$ , meaning no active power will be lost across the line.

Another commonly used line model is the two-port model, also known as  $\pi$ -model, see Figure 2.2. The shunt admittance of the line is split in half and connected to each side of the circuit. Even this model can be simplified, further information can be obtained from [Andersson and Franck, 2012]. From [Andersson, 2012], we have:



**Figure 2.1** Equivalent circuit diagram from [Andersson, 2012]. The resistance ( $R'$ ), the inductance ( $L'$ ), the conductance ( $G'$ ) and the capacitance ( $C'$ ) are all in per km.  $i$  is the current and  $u$  is the voltage. The denotations  $dx$ ,  $di$  and  $du$  stands for infinitely small changes in distance, current and voltage.



**Figure 2.2**  $\pi$ -model from [Andersson and Franck, 2012].  $\underline{U}_1$  and  $\underline{I}_1$  are the voltage respective current on the primary side and  $\underline{U}_2$  and  $\underline{I}_2$  are the voltage and current on the secondary side.  $\underline{Z}_l$  is the line impedance and  $\frac{Y_q}{2}$  is the shunt admittance.

$R'$  = series resistance/km per phase ( $\Omega/\text{km}$ )

$X'$  = series reactance/km per phase( $\Omega/\text{km}$ )

$B'$  = shunt susceptance/km per phase (Siemens/km)

$G'$  = shunt conductance/km per phase (Siemens/km)

Assume  $k$  and  $m$  are two nodes, then the parameters above can characterise the series impedance,  $Z_{km}$  ( $\Omega$ ), and shunt admittance,  $Y_{km}^{sh}$  (Siemens) between the nodes.

$$Z_{km} = R_{km} + jX_{km} \quad (2.1)$$

$$Y_{km}^{sh} = G_{km}^{sh} + jB_{km}^{sh} \quad (2.2)$$

Computing the series admittance from node  $k$  to node  $m$  in the nodal admittance

matrix  $y^1$ ,

$$y_{km} = z_{km}^{-1} = g_{km} + jb_{km} \quad (2.3)$$

The complex voltages ( $E$ ) at the terminal nodes  $k$  and  $m$  are known as,

$$E_k = U_k e^{j\theta_k} \quad (2.4)$$

$$E_m = U_m e^{j\theta_m} \quad (2.5)$$

where  $\theta$  is the phase angle at a node, hence  $\theta_{km} = \theta_k - \theta_m$ .

The complex currents ( $I$ ) can be expressed as a function of the difference between two complex voltages ( $E$ ) at their branches and the admittance ( $y$ ) of the power line. The current is defined as positive when injecting a bus and negative when leaving one.

$$I_{km} = y_{km}(E_k - E_m) + Y_{km}^{sh} E_k \quad (2.6)$$

$$I_{mk} = y_{km}(E_m - E_k) + Y_{km}^{sh} E_m \quad (2.7)$$

The complex power ( $S$ ) can be derived using the complex voltage and complex conjugate of the current ( $I^*$ )<sup>2</sup>,

$$S_{km} = E_k * I_{km}^* \quad (2.8)$$

and the active power ( $P$ ) and reactive power ( $Q$ ),

$$P_{km} = U_k^2 g_{km} - U_k U_m g_{km} \cos \theta_{km} - U_k U_m b_{km} \sin \theta_{km} \quad (2.9)$$

$$Q_{km} = -U_k^2 (b_{km} + b_{km}^{sh}) + U_k U_m b_{km} \cos \theta_{km} - U_k U_m g_{km} \sin \theta_{km} \quad (2.10)$$

## Loads

The load is drawn as an arrow in Figure 1.1. Generally loads can be expressed by their current consumption  $I_{km}^{load}$ ,

$$I_{km}^{load} = I_{km}^{load}(U_k) \quad (2.11)$$

<sup>1</sup> Lower case letters are used for components given in the per unit system while upper case letters are used when elements are in SI-units. The per unit system is explained in the appendix.

<sup>2</sup> A star (\*) above a complex variable indicates it's a conjugate.

where the function  $I_{km}^{load}(\cdot)$  describes the steady state load characteristics which is dependent on the voltage  $U_k$ . More often loads can be described by the amount of active and reactive power they use,

$$P_k^{load} = P_k^{load}(U_k) \quad (2.12)$$

$$Q_k^{load} = Q_k^{load}(U_k) \quad (2.13)$$

While loads are current-”consumers”, generators are current injectors. Generator voltage is primarily determined by the reactive power injected into the node.

The exponential load model, equations (2.14) and (2.15), is found in [Van Cutsem and Vournas, 1998], where  $P_0$  and  $Q_0$  are the initial values of active and reactive power.  $V_0$  is the reference voltage.  $\alpha$  and  $\beta$  are exponents which are determined by the type of load one has e.g. lights, motor, heating etc.

$$P = P_0 \left( \frac{V}{V_0} \right)^\alpha \quad (2.14)$$

$$Q = Q_0 \left( \frac{V}{V_0} \right)^\beta \quad (2.15)$$

The exponential load model can be sensitive to low voltages beneath a certain threshold e.g.  $V < 0.6$ . Too low voltage can result in load shedding or different load characteristics.

A simple load model is static, where the load remains constant at all times. For a more accurate thus more complicated description of a load, dynamic behavior has to be included, meaning the load changes with time. From [Van Cutsem and Vournas, 1998], the general load model depends on the instantaneous value of a load state variable, denoted  $x$ :

$$P = P_t(z, V, x) \quad (2.16)$$

$$Q = Q_t(z, V, x) \quad (2.17)$$

where  $P_t$  and  $Q_t$  are smooth functions called the *transient load characteristics*.  $V$  is the voltage and  $z$  is the load demand. A smooth differential equation is included in the dynamic model, see (2.18).

$$\dot{x} = f(z, V, x), \quad (2.18)$$

and at steady state the following algebraic equation holds:



$$0 = f(z, V, x) \quad (2.19)$$

Equation (2.18) will determine the recovery path of the load after a contingency. This equation can take many different form but most important, it can be simplified. With some alterations which can be studied in [Van Cutsem and Vournas, 1998], the two equations below can be obtained:

$$P = P_s(z, V) \quad (2.20)$$

$$Q = Q_s(z, V) \quad (2.21)$$

where  $P_s$  and  $Q_s$  are the steady-state load characteristics which are not dependent on the load state variable  $x$ . Since the dynamic load model can be complicated to implement due to the differential equation, simplifications were done in this thesis. The load equations used in this project are found in Chapter 3.

## Generators

Generators are seen as circles in the power system diagram, see Figure 1.1. The generator model used in the thesis was a synchronous machine called *the Stationary Single Phase Equivalent Circuit* from [Andersson, 2012] and can be seen in Figure 2.3. It is assumed that a generator has an automatic voltage regulator (AVR). An AVR is a device which automatically keeps a constant voltage level. Each generator is described by the equations below, also found in [Eriksson et al., 2011].

$$\dot{\delta} = \omega \quad (2.22)$$

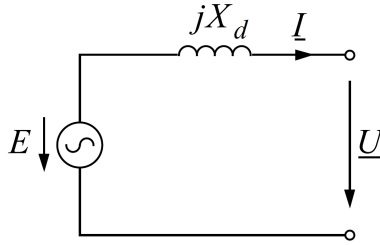
$$\dot{\omega} = \frac{1}{M} \left( P_m - \frac{E'_q U}{x'_d} \sin(\delta - \theta) \right) - D\omega \quad (2.23)$$

$$\dot{E}'_q = \frac{1}{T'_{do}} \left( E_f - \frac{x_q}{x'_d} E'_q + \frac{x_d - x'_d}{x'_d} U \cos(\delta - \theta) \right) \quad (2.24)$$

$$\dot{E}_f = \frac{1}{T_e} \left( -E_f - K_A (U_{ref} - U) \right) \quad (2.25)$$

where  $\delta$  is the rotor speed and  $\omega$  is the rotor speed deviation.  $E'_q \angle \delta$  and  $U \angle \theta$  are voltage phasors in the generator and the terminal respectively.  $T'_{do}$  and  $T_e$  are the time constants for the *d-axis* transient open-circuit<sup>3</sup> and exciter.  $P_m$  is the mechanical

<sup>3</sup> The time constants for the *d-axis* transient open-circuit characterizes the initial decay of transients in the d-axis variables in a synchronous machine with stator windings open-circuited. A synchronous machine has two axis, a d-axis and a q-axis. More information about the axis of a synchronous machine is found in [Andersson, 2012].



**Figure 2.3** The Stationary Single Phase Equivalent Circuit from [Andersson, 2012]

power from the generator shaft while  $D$  is the shaft damping constant. The generator inertia is given as  $M$ . The synchronous reactance and the transient reactance on the  $d$ -axis, are  $x_d$  and  $x'_d$ .  $E_f$  is the generator field voltage and  $U_{ref}$  is the set-value of the connecting terminal bus voltage.  $K_A$  is the gain for the AVR. At steady state  $\dot{E}'_q$  and  $\dot{E}_f$  are zero. Hence, a dot above a capital letter indicates it is a time derivative, e.g.  $\dot{E}_f$  is the derivative of the field voltage.

The expression for active and reactive power generated in a generator,

$$P_g = \frac{UE_q}{x_d} \sin \theta \quad (2.26)$$

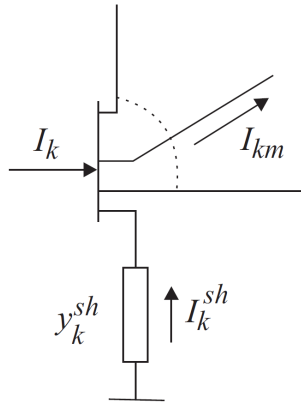
$$Q_g = \frac{-U^2}{x_d} + \frac{UE_q \cos \theta}{x_d} \quad (2.27)$$

## 2.2 Nodal Formulation of the Network Equations

Applying Kirchoff's Current Law e.g. the current injection at each bus can be determined e.g. at bus  $k$ ,

$$I_k + I_k^{sh} = \sum_{m \in \Omega_k} I_{km}, \text{ for } k = 1, \dots, N \quad (2.28)$$

where  $k$  is a generic node,  $I_k$  is the net injection from generators and loads,  $I_k^{sh}$  is the current injection from shunt reactors and  $m$  is a node adjacent to  $k$ .  $\Omega_k$  is the set of nodes connected to  $k$  and  $N$  is the number of nodes in the network. This is also illustrated in Figure 2.4. A relationship between current injection vector  $I$  with elements  $I_k$ ,  $k=1, \dots, N$ , nodal voltage vector  $E$  with elements  $E_k = U_k e^{j\theta_k}$  and nodal



**Figure 2.4** Net injection of a node from [Andersson, 2012]

admittance matrix  $Y = G + jB$ , can be expressed the following way,

$$I = YE \quad (2.29)$$

## 2.3 Basic Power Flow Problem

To formulate a basic power flow problem four variables are needed at each bus  $k$ , [Andersson, 2012]:

- $U_k$  = voltage magnitude
- $\theta_k$  = voltage phase angle
- $P_k$  = net active power (sum of generation and load)
- $Q_k$  = net reactive power (sum of generation and load)

Three types of buses can be determined depending on which variables are known:

- $U\theta$ -bus (slackbus):  $U_k$  and  $\theta_k$  are known;  $P_k$  and  $Q_k$  are calculated
- $PQ$ -bus:  $P_k$  and  $Q_k$  are known;  $U_k$  and  $\theta_k$  are calculated
- $PU$ -bus:  $P_k$  and  $U_k$  are known;  $Q_k$  and  $\theta_k$  are calculated

The slack bus, also known as the reference bus, has two functions in the power flow problem. First it serves as a voltage angle reference and secondly it balances generation, load and losses of active power in the system. PQ-buses usually represent load buses without voltage control and PU-buses are normally buses with voltage control. The most common bus type in systems is the PQ-bus. Let's use Figure 1.1 as an example.  $P_{g1}$  could be a slack bus which balances the production and the consumption in the system.  $P_{g2}$  and  $P_{g3}$  could be PU-buses if they have AVRs<sup>4</sup> that control the voltage at their terminals. The load node would be a PQ-bus since it's missing a voltage regulator.

All voltage magnitudes and voltage angles can be computed in a fully specified system. But as mentioned before, this information is not always given for all buses, only for the slack bus. This problem can be solved by using numerical methods by setting up a power flow problem. How to set up and solve a power flow problem can be read in G. Andersson's lecture script, *Power System Analysis* [Andersson, 2012]. When the problem is formulated, different numerical methods can be used to obtain all voltage angles and magnitude in all nodes. Common methods are *Gauss-Seidel Iteration* and *Newton-Raphsons Method*.

A rough estimation of the Power Flow Problem was used in this thesis. Equations for the rough estimates can be found in [Andersson, 2012]. Approximative methods are mostly used to identify critical cases or to find initial guesses which are then used in the full model for a complete solution of the system. The following equations hold for transmission lines:

$$P_{km} = U_k^2 g_{km} - U_k U_m \cos(\theta_{km}) - U_k U_m b_{km} \sin(\theta_{km}) \quad (2.30)$$

$$P_{mk} = U_m^2 g_{km} - U_k U_m \cos(\theta_{km}) + U_k U_m b_{km} \sin(\theta_{km}) \quad (2.31)$$

$$Q_{km} = -U_k^2 (b_{km} + b_{km}^{sh}) + U_k U_m b_{km} \cos(\theta_{km}) \quad (2.32)$$

$$Q_{mk} = -U_m^2 (b_{km} + b_{km}^{sh}) + U_k U_m b_{km} \cos(\theta_{km}) \quad (2.33)$$

where the nodal admittance matrix is defined as,

$$b_{km} = b_{mk} = -x_{km}^{-1} \quad (2.34)$$

$$b_{kk} = \sum_{m \in \Omega_k} x_{km}^{-1} \quad (2.35)$$

For high-voltage systems there are strong couplings between  $P$  and  $\theta$  and  $Q$  and  $U$  which can be transformed into a linear approximation. In addition to this simplification the shunt admittance and series resistance in the line can be neglected giving

---

<sup>4</sup> Automatic voltage regulator (AVR), controls the voltage at a specific node so it maintains constant.

the following equations,

$$P_{km} = U_k^2 g_{km} - U_k U_m \cos \theta_{km} - U_k U_m b_{km} \sin \theta_{km} \approx -U_k U_m b_{km} \sin \theta_{km} \quad (2.36)$$

$$Q_{km} = -U_k^2 (b_{km} - b_{km}^{sh}) + U_k U_m b_{km} \cos \theta_{km} \approx U_k U_m b_{km} \cos \theta_{km} \quad (2.37)$$

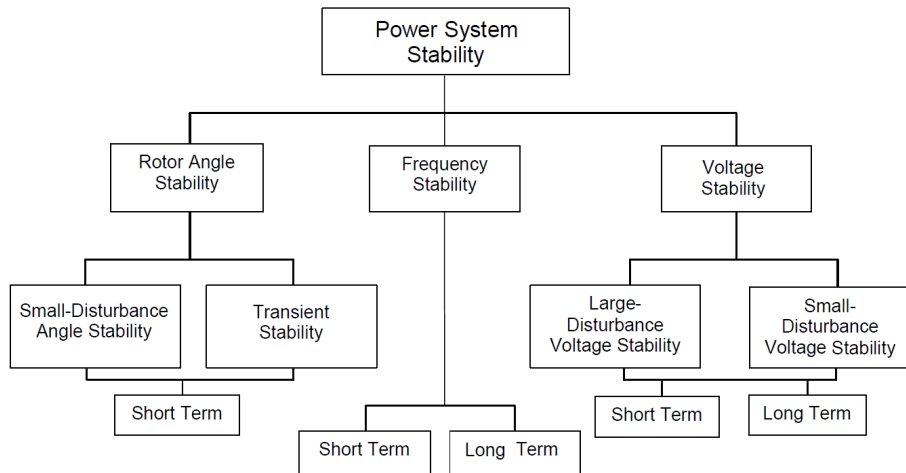
## 2.4 Power System Stability

A correct way of modeling a power system includes dynamic behavior which is modelled by differential equations. A good example of a unit in a power system is an electric load consumer. The load demand might vary spontaneously with time, never bringing the system to a steady state. This problem will mathematically be difficult to solve and can be simplified. Though simplifications may make it easier to evaluate the system, one ought to keep in mind what information is left out.

Since this thesis deals with stability issues in a system when a fault occurs, it is useful to know some general information about faults in networks. There are many types of faults that can occur in a power system which can be classified as *shunt* or *series faults*. A shunt fault occurs when an unwanted connection between phases or a connection between phases and ground occurs. An example of a shunt fault is the very common *Single line to ground fault*, where one phase has a connection to ground. A series fault is a failure along a power line that causes undesirable interruptions or connections in the network. An example of a series fault is when a breaker does not trip on command. This type of event leads to asymmetrical operating conditions which can damage the equipment in the system. The combinations of faults are many but more information about them can be retrieved from chapter 7 in [Andersson and Franck, 2012]. Small disturbances will cause transients in the network which can damp out with time. Larger disturbances on the other hand will cause oscillations that will give rise to stability issues. The definition of stability is quoted from [Andersson, 2012]:

*Power system stability is the ability of an electric power system, for a given initial operating condition, to regain a state of operating equilibrium after being subjected to a physical disturbance, with most system variables bounded so that practically the entire system remains intact.*

This implies that a stable system is able to recover, find an equilibrium and operate after a disturbance. Classification of power system stability can be made depending on active or reactive power imbalance, locally or globally. Figure 2.5 is an overview of the classification of stability in power systems and the main focus will be on voltage stability. The other stability classes will briefly be explained before continuing on to voltage stability.



**Figure 2.5** Classification of power system stability from [Andersson, 2012].

## Synchronous Stability and Frequency Stability

*Synchronous Stability*, also known as *Rotor Stability*, is the ability of synchronous machines to maintain synchronism in a power system after a disturbance. The disturbance may be an active power imbalance between prime mover, e.g. hydro and steam turbines, and loads. Since the net active power from energy producers should be equal to the power demand from loads and losses in the system, machines with rotating parts will try to compensate the mismatch by increasing or decreasing the kinetic energy in their moving parts. The attempt to correct the energy gap will force the machines with rotating parts to fall out of synchronism. Locally a single generator can fall out of step due to a fault occurring close to it. One example is an earth fault close to the generator, which will result in high currents flowing in the generator windings forcing it to disconnect to avoid substantial damages to the machine. This type of disturbance is called *Transient Instability* due to the lack of synchronism of the generator torque. Insufficient damping of oscillations is classified as *Small-Disturbance Rotor Angle Stability*. It determines how well a system is able to maintain in sync during a small disturbance.

The previous power imbalance was local, a corresponding problem on a global scale will affect the frequency of the entire system. As in the past paragraph, active power difference between production and consumption will force rotating parts in synchronous machine to compensate for the imbalance. These machines will supply missing energy by utilizing kinetic energy stored in moving parts or by converting excess power, stored kinetic energy in rotating machine parts. This sort of instability will significantly affect the frequency of a power system. Decreasing or increasing active power generation in the system must be enforced, otherwise it can result in

*Frequency Instability.* The time scale of Frequency Instability is quite varying, from a few seconds to several minutes.

## Voltage Stability

[Andersson, 2012] defines Voltage Stability the following way:

*A system is said to be voltage stable if an increase in load admittance results in an increase in load power.*

The definition says, that a system is stable if an increase in load will lead to an increase in load power, satisfying the energy demand.

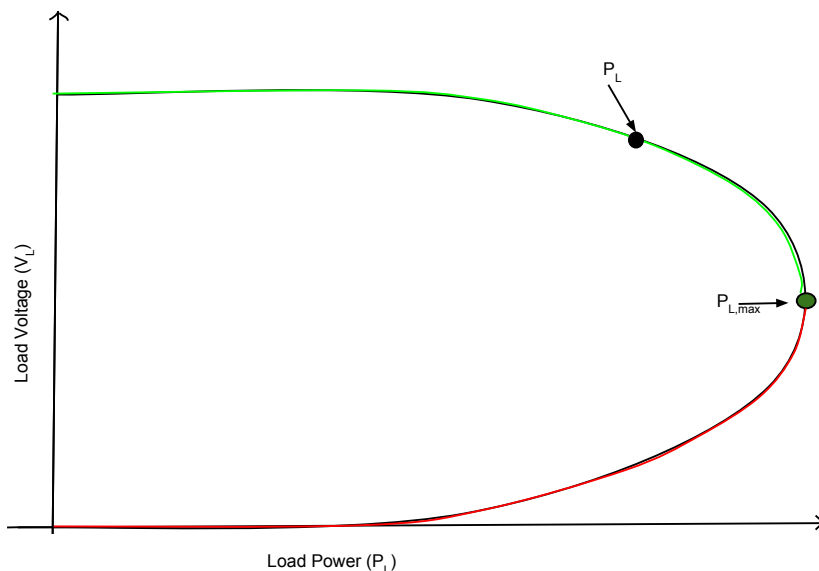
It is easier to balance the active power than the reactive power in a power system. The lack of reactive power injection in a node will cause the voltage at the same node to decrease. (The other way around is also possible. A high reactive power injection can cause a high voltage which might damage the equipment in the power system). It is indeed recognised that reactive power and voltage are coupled to each other. An increase in reactive power will rise the voltage level and vice versa.

While active power is considered as a global quantity, reactive power is regarded local because of the extensive reactive power losses across power lines. Normally the relationship between resistance ( $R$ ) and reactance ( $X$ ) is  $R \ll X$  in power lines, making the reactive power loss more evident than the active power loss.

When a reactive power imbalance occurs in a power system, voltage problems appear, creating *Voltage Instabilities* or worse, a *Voltage Collapse*. Usually, it is the low voltages that create stability issues but in principle, high voltage can also cause problems. The difference is that instabilities caused by low voltages are associated with high load conditions while instabilities induced by high voltages are related to low load conditions.

The time scale of the voltage instabilities divides them into two groups, *short-term* (a few seconds) and *long-term* (seconds to minutes). Short-term voltage instabilities involve fast dynamics such as induction motors, electronically controlled loads, and HVDC converters. Long-term voltage instabilities are typically triggered by tap-changing transformers, thermostatically controlled loads and generator current limiters [Andersson, 2012].

A convenient way of describing the voltage stability of a power system is with the *Nose Curve*. The Nose Curve explains the relationship between the load power ( $P_L$ ) and the voltage ( $V_L$ ) in a node, see Figure 2.6. The shape of the curve is similar to a nose and can be divided into two parts; an upper part and a lower part. The maximum load of the system is on the tip of the curve. As one can see, a single load power can have two voltage solutions (except the maximum load power which only has one load voltage). It is extremely important to know the voltage at the load, since the two voltage values affect the system differently, where one is stable and one is unstable. The upper part of the curve, i.e. above the maximum load, is the area of stability. The lower part of the curve should be avoided since loads with



**Figure 2.6** An example of a nose curve before a contingency. The curve is divided by the maximum load power into two parts, one stable side, coloured green and one unstable side in red. The two areas are divided by a maximum load power.

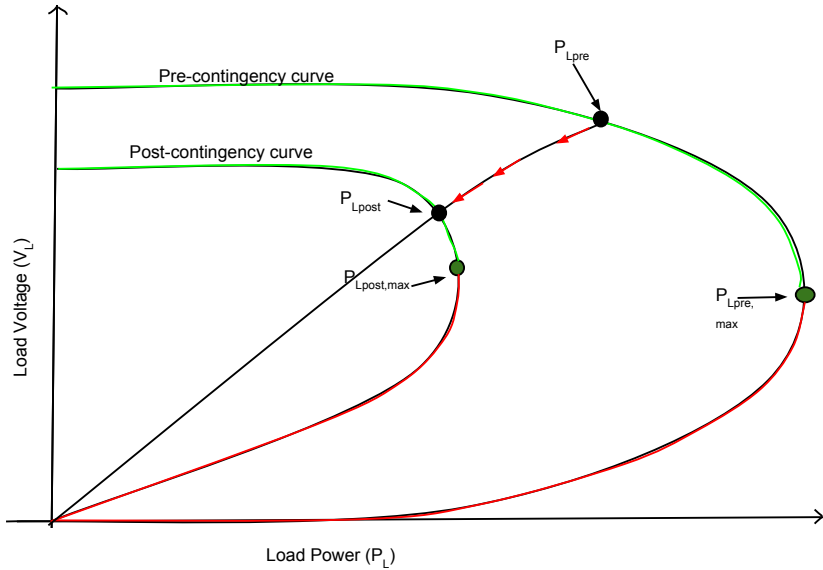
these load voltages result in an unstable system. Normally, a fault is followed by a voltage drop and a load decreases. The new active power and reactive power are derived by using equation (2.16) for the active power and equation (2.17) for the reactive power. The recovery path is determined by (2.18).

A increase of the load power, leads to a voltage decrease. Where the load is located on the Nose Curve is very important, see Figure 2.6. If the load is on the upper side of the Nose Curve, the system is regarded as stable. If the load power is located on the lower side of the Nose Curve, the system is regarded unstable. A good way of describing the Nose Curve is via an example.

Figure 2.7- 2.9 illustrate an example when a fault occurs in a power system by using the *Nose Curve*. The system is stable in Figure 2.6. The maximum possible load power is marked out as  $P_{L,max}$  and the actual load is  $P_L$ . The stable side of the curve is marked green, while the unstable part is red. Why the curve is divided into a stable side and unstable side is clear when the system tries to recover from a fault.

A fault occurs and the new system is represented by a new nose curve, see Figure 2.7. There are two curves, one *Pre-contingency curve* (before the fault) and





**Figure 2.7** An example of a nose curve after a contingency. There are two curves, the *Pre-contingency curve* which is the system before the fault and the *Post-contingency curve* after the fault. The red arrows show a possible way of a location change of the load, from a system without a fault to a faulty system. The change between the two nose curves, is derived by equations (2.16) and (2.17).

one *Post-contingency curve* (after the fault). The fault forces the load power and voltage to decrease, see the black solid line in Figure 2.7. The black line is the path of the load between pre-and post-contingency. No time step is between pre-and post-contingency, meaning that the time dependent equation (2.18) is not needed here. Equations (2.16) and (2.17) are the ones changing the active power and the reactive power of the load due to the change in voltage and load demand. The recovery path of the load power is determined by (2.18) and Figure 2.8 will demonstrated one successful recovery and two which will fail.

The three solid lines (black, blue and purple) in Figure 2.8 are derived by equations (2.16)-(2.18) and describe how the load changes. These equations are highly dependent on (2.18) when there is a time change (not between pre- and post-contingency) since the variable  $x$  needs to be updated at each time step. The two other colours, blue and purple, will give different recovery paths (2.18). The new maximum load  $P_{Lpost,max}$  and the new load power  $P_{Lpost}$  are located on the *Post-*

*contingency curve.*

The power system is stable when (2.19) is fulfilled, i.e. when the system has found a steady state. By that time, the load power should be recovered.

There are three different solid line (blue, black and purple) in Figure 2.8, representing three different scenarios for the same system. The solid lines have three different load powers which will determine their chances of recovering from a fault. The blue line has the lowest pre-contingency load, the black one is in between the blue and the purple, and lastly the purple solid line has the highest pre-contingency load. The only scenario which will result in a fully recovered load power is the blue line. The other two will not find any stable operating points for a fully recovered load. The following examples of scenarios are viewed in Figure 2.8.

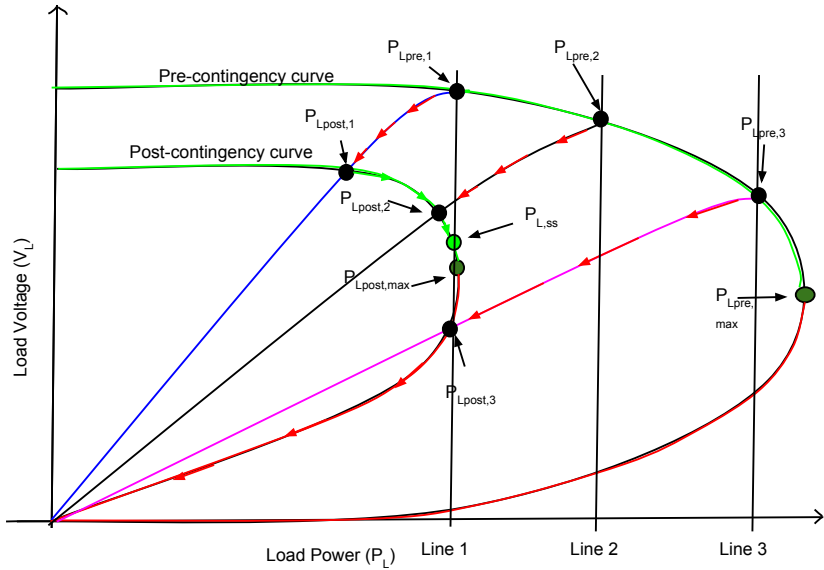
For the "blue line scenario", the load is located at  $P_{Lpre,1}$  on the *Pre-contingency curve*. A fault occurs and the load power changes due to the voltage decrease in load node. The load change is derived by equations (2.16) and (2.17). The load is now located at the  $P_{Lpost,1}$  on the *Post-contingency curve* which is the upper and stable side of the curve. Due to the fault, there is a difference between the load demand and the load which (2.18) will try to eliminate<sup>5</sup>. The state variables are updated using (2.18) so that (2.16) and (2.17) can increase the load power. As expected, the voltage at the load node decreases, but since the load is on the upper part of the nose curve, the voltage decrease leads to a load power increase. The difference between the load demand and the load will keep on decreasing until the load has reached the nominal load power. At each time step (2.18) updates the system state variables and new active and reactive powers are computed using (2.16) and (2.17). The "blue line scenario" in Figure 2.8 will successfully restore the load power while still maintaining stability, meaning that equation (2.19) will be satisfied. The green arrows between  $P_{Lpost,1}$  and  $P_{L,ss}$  presents a load recovery path. Observe that  $P_{L,ss}$  is on the same vertical line (Line 1) as  $P_{Lpre,1}$  and therefore fully recovered.

Lets move on to the "black line scenario". Before the fault, the load is located at  $P_{Lpre,2}$  and after fault, at  $P_{Lpost,2}$  by using (2.16) and (2.17). The load is on the stable side of the *Post-contingency curve* but that will not guarantee a full load recovery for a stable system.

All stable operating points are on the *Post-contingency curve* and loads which are not on any nose curve are regarded as not feasible. Since the load will have to leave the *Post-contingency curve* and be larger than the maximum post-contingency load ( $P_{Lpost,max}$ ) to reach full load recovery (Line 2), the system will never be stable. In other words expression (2.19) will never be fulfilled. Therefore the "black line scenario" is not able give a fully recovered load and a stable system at the same time.

Lastly, the "purple line scenario" is also presented in Figure 2.8. Once again the load is located on the *Pre-contingency curve* before the fault, but at  $P_{Lpre,3}$ . A fault occurs and the load is located at  $P_{Lpost,3}$ , which is on the lower and unstable

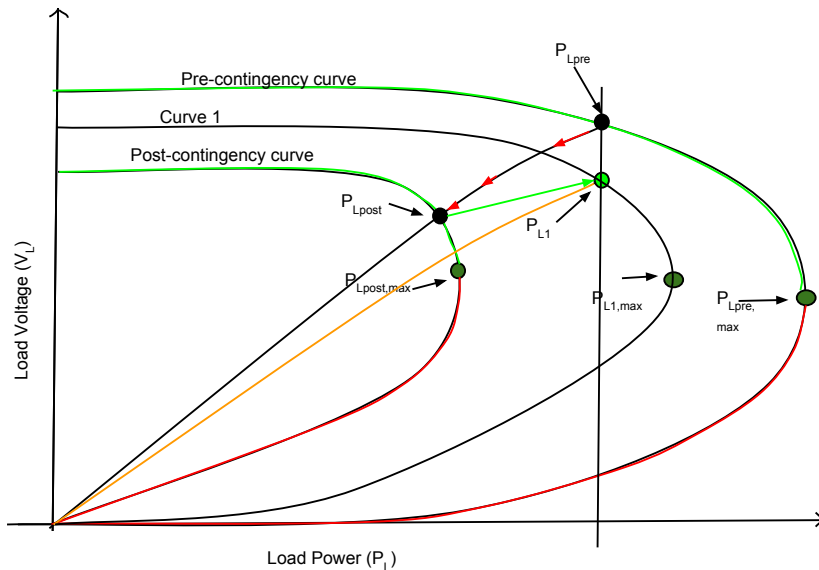
<sup>5</sup> Equation (2.18) can be modelled as a feedback controller that tries to satisfy the load power demand.



**Figure 2.8** An example of a power system after a contingency and during recovery. The generated power is the same as in Figure 2.7. There are three different scenarios, *the blue line scenario, the black line scenario and the purple line scenario*. The blue line is the only stable load recovery path, while the other two lines will not be able to give a fully restored load for a stable system.

side of the *Post-contingency curve*. The difference between the load demand and the load will trigger (2.18) to reduce the power mismatch. (2.18) will compute new state variables which will be used in (2.16) and (2.17) to increase the load power. The voltage will decrease, and unfortunately so will the load power since the load is located on the lower part of the nose curve. The difference between demand and load will increase and force (2.18) to update the state variables and equations (2.16) and (2.17) to compute new load power values. Once again the voltage and the load power will decrease since the load is located on the unstable side of the nose curve. This snowball effect will continue until the system reaches a voltage collapse and probably a blackout. The “purple line scenario” will never reach a full load recovery (Line 3) with a stable power system.

While Figure 2.8 shows one successful recovery path without increasing the generated power in the system, Figure 2.9 has a different solution which involves increasing the power generation. As in the previous examples, the load is located at



**Figure 2.9** An example of a power system after a contingency and during recovery. The generated power is increased and so is the load power. *Curve 1* represents the new system which has a higher generation. There are two load recovery paths (black solid line and orange solid line) which succeed in restoring the load power to its initial value.

$P_{Lpre}$  before the contingency and changes location to  $P_{Lpost}$  after the fault (follow the red arrows on the black solid line in Figure 2.8). The load power and the voltage at the load node are decreased after the fault. To increase both voltage and load, the generated power is increased. The system is now presented by a new curve, *Curve 1*, where the generated power is higher than at the Post-contingency curve. Equation (2.18) will once again try to reduce the power difference between the nominal load power and the current load. (2.18) will update the state variables which are used when (2.16) and (2.17) compute new load power values. There are two examples of feasible recovery paths in Figure 2.9, one black solid line and one orange solid line. The load power is fully restored at  $P_{L1}$ . In Chapter 4 the Nose Curve will be used to explain the recovery path of the *Seven-Node System*.

# 3

## The Optimization Problem

The optimization problem in this thesis, is to find a maximum pre-contingency load in a system which is subjected to a fault and still remaining stable. An example of a fault can be a faulty power line whose impedance will increase after the contingency. The system has to be able to recover from the fault and restore the initial load power.

After a fault the load and the voltages in the system will decrease, but how the system recovers is critical to maintain stability. As stated before, dynamic load behavior are described as differential equations. Since solving differential equations can be difficult, simplifications are usually done. Load model approximations will be presented in this chapter.

The optimization is done on a small system called the *Seven-Node System* or *Seven-Node Model*, see Figure 3.1. The system was presented in Chapter 1 as an example of a power system. The name is inspired by the number of nodes in the system which is seven. There are three generators ( $P_{g1}, P_{g2}$  and  $P_{g3}$ ), one load which is in parallel with the shunt ( $B$ ) at node 4.

The optimization problem can be seen below:

$$\max_{x,y} P_{L,0} \quad (3.1)$$

$$P = P_t(x, y) \quad (3.2)$$

$$Q = Q_t(x, y) \quad (3.3)$$

$$\dot{x} = f(x, y) \quad (3.4)$$

$$0 = \psi_0(x(0^-), y(0^-)) \quad (3.5)$$

$$0 = \psi_1(x, y), \quad \forall t \in [0, \infty) \quad (3.6)$$

$$0 \geq g(x, y) \quad (3.7)$$

$$f^{a,i}(z) \cdot f^{b,i}(z) = 0, \quad i = 1, \dots, n_g, \forall t \in [0, \infty) \quad (3.8)$$

$$f^{a,i}(z) \geq 0, \quad i = 1, \dots, n_g, \forall t \in [0, \infty) \quad (3.9)$$

$$f^{b,i}(z) \geq 0, \quad i = 1, \dots, n_g, \forall t \in [0, \infty) \quad (3.10)$$

$$0 > \frac{dV}{dP}(\infty) \quad (3.11)$$

The variables  $x$  and  $y$  are time dependent, with a time interval  $t \in [0, \infty)$ .  $x$  is the load recovery state variable,  $y$  is a vector of algebraic variables e.g. voltages and phase angles at nodes and internal generator variables.  $\forall t \in [0, \infty)$  tells that the optimization is done from a time interval zero to infinity, i.e. until the system finds an equilibrium.  $n_g$  is the number of generators in the system (which is three in this study).

The first expression (3.1), tells which expression is optimized over defined variables. Our objective function (the variable to be optimized), is maximizing the pre-contingency load ( $P_{L,0}$ ) over  $x$  and  $y$ .

As in Chapter 2, (3.2), (3.3) computes the load power and (3.4) the recovery path of the load power.

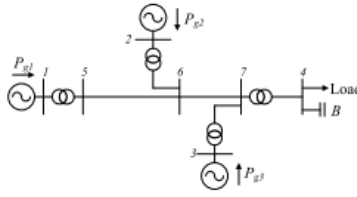
The power system is expressed in (3.5) and (3.6). The system is represented by (3.5) at pre-contingency and (3.6) at time  $t \in [0, \infty)$  (excluding pre-contingency). (3.5) and (3.6) contains power flow equations ((2.34)-(2.37)) and equations for generators ((2.22)-(2.27)) found in Chapter 2.

The search for the maximum possible load is done while subjected to constraints which are expressed in equations (3.7)-(3.10). Some of the constraints, are the ramping of the generators, placed in expression (3.7). An ideal generator can increase and decrease its production infinitely fast, but in reality this is a limiting factor. Generator 2 ( $P_{g2}$  in Figure 3.1) cannot increase or decrease more than 0.04 p.u. per time step while generator 3 ( $P_{g3}$  in Figure 3.1) is limited to 0.1 p.u. ramping per time step. Constraints on the generators internally are also included in the optimization problem and are written in expressions (3.9) and (3.10).  $f^{a,i}$  is the equations of the field voltage of generator  $i$  under voltage control and  $f^{b,i}$  is the equation of the field voltage of generator  $i$  under overexcited voltage control.  $f^{a,i}$  and  $f^{b,i}$  are two very important constraints since they include certain limitations a generator might face.

Equation (3.8) is a constraint that always forces one of (3.9) and (3.10) to be zero. The constraint can be relaxed in order to have a larger search area for the optimization. When and why the complementary-constraint (3.8) can be relaxed is explained in section 3.2.

The optimization is feasible if the system has found a steady state for a fully recovered load power. The stability expression is equation (3.11). It says, that the partial derivative of the voltage at the load node over the load power should be negative. When the load power is decreasing the voltage has to increase and vice versa, when the load power is increasing the voltage should decrease. This translates to being located on the upper part of the *Nose Curve*.

Further more expressions (3.9) and (3.10) describe the first-order automatic voltage regulator (AVR) with overexcitation limiter (OXL) in the generators in equations (3.12)-(3.13). The AVRs can become saturated and therefore two equations are needed to express the field voltages, one before saturation (eq. 3.12) and



**Figure 3.1** An example of a power system from [Karystianos et al., 2007]. The power system consists of three generators  $P_{g1}$ ,  $P_{g2}$  and  $P_{g3}$  that are connected to the network via transformers. The transformers are in between busbars 1-5, 2-6, 3-7 and 4-7. The load and the capacitor ( $B$ ) are in parallel and connected to busbar 4. This power system is also called the *Seven-Node System*. The optimization was performed on this system.

one when the AVR is saturated (eq. 3.13).

$$f^{a,i}(z) = -E_f^i + K_A^i(V_{ref} - V^i), \quad i = 1, \dots, n_g, \forall t \in [0, \infty) \quad (3.12)$$

$$f^{b,i}(z) = -E_f^i + E_f^{lim}, \quad i = 1, \dots, n_g, \forall t \in [0, \infty) \quad (3.13)$$

### 3.1 Implemented load models

When a fault occurs the load decreases. The recovery path is very important for the chances of the system to regain stability and function in a safe matter. Different load models are tested to see if the system will remain stable and recover from a fault. Generally a dynamic load model should include differential equations (2.18), which will make it more difficult to find solutions to the optimization problem. For those reasons, approximations were done on the load model. In Chapter 2, the load is described by three equations, (2.16), (2.17) and (2.18). The two first equations, (2.16) and (2.17) decide the active power and the reactive power of the load, while equation (2.18) is used to approximate the load recovery. Since (2.18) can be expressed in many different ways, equations (3.14) and (3.15) are chosen as load recovery models in this thesis.

The first load recovery model is linear, equation (3.14) which describes how the load ( $P_{LC}$ ) recovers after a fault,

$$P_{LC} = P_{L0} + \frac{t}{T}(P_{L0} - P_{LC0}) \quad (3.14)$$

where  $t$  is the current time,  $T$  is the end time,  $P_{L0}$  is the pre-contingency load and  $P_{LC0}$  is the load right after the contingency. In this study, the recovery time is fixed to five time steps, therefore  $T = 5$ . For the exponential load recovery model, see equation (3.15),

$$P_{LC} = P_{LC0} + \beta(t)(P_{L0} - P_{LC0}), \quad (3.15)$$

where  $\beta(t) = \frac{1-0.5^t}{1-0.5^T}$ .

The voltage dependent load model before a fault (3.16) and (3.17), while immediately after a fault (3.18) and (3.19) can be described with equations below. They correspond to the load power equations (2.16) and (2.17). (2.18) is not included in the change from pre-contingency to post-contingency since there is no time change between these events. (2.18) is only valid at time changes.

$$P_{L0,pre} = P_{L0} = p_0 * V_{pre}^2, \quad (3.16)$$

$$Q_{L0,pre} = \frac{1}{2}P_{L0} \quad (3.17)$$

$$P_{L0,post} = P_{LC0} = p_0 * V_{post}^2, \quad (3.18)$$

$$Q_{L0,post} = \frac{1}{2}P_{LC0} \quad (3.19)$$

where  $p_0$  is a variable computed from knowing the load and the load voltage for either pre-contingency or post-contingency. For example, if  $P_{L,pre}$ ,  $V_{L,pre}$  and  $V_{L,post}$  are known we can set up a very simple equation system and obtain  $P_{L,post}$ . The same thing holds the other way around, when  $P_{L,pre}$  is sought after. This load model is only used right after the contingency and not at the other time steps. The advantage of using a voltage dependent load recovery model, is the incorporation of the voltage but that also makes the implementation more intricate. The load voltage effect the stability factor, as shown earlier by the *Nose Curve*. By using a time dependent load model instead of a voltage dependent load model, the recovery time can be predefined. A voltage dependent load model at load recovery is difficult to find a feasible solution for. Therefore, equations (3.15) and (3.16) are only used at pre- and post-contingency and not at each time step. Adding the voltage characteristic is crucial here since the voltage at post-contingency will determine whether the system will successfully recover or not.



### 3.2 Solving the Optimization Problem

The actual optimization problem solved in this study is presented bellow:

$$\max_{x,y} P_{L,0} \quad (3.20)$$

$$P_{L0,pre} = P_{L0} = p_0 * V_{pre}^2 \quad (3.21)$$

$$Q_{L0,pre} = \frac{1}{2} P_{L0} \quad (3.22)$$

$$P_{L0,post} = P_{LC0} = p_0 * V_{post}^2 \quad (3.23)$$

$$Q_{L0,post} = \frac{1}{2} P_{LC0} \quad (3.24)$$

$$P_{LC} = h(t, P_{L0}, P_{LC0}) \quad (3.25)$$

$$0 = \psi_0(x(0^-), y(0^-)) \quad (3.26)$$

$$0 = \psi_1(x, y), \quad \forall t \in [0, \infty) \quad (3.27)$$

$$-0.04 \leq P_{g,2}(t) - P_{g,2}(t-1) \leq 0.04, \quad t \in [0, 1, 2, 3, 4, 5] \quad (3.28)$$

$$-0.1 \leq P_{g,3}(t) - P_{g,3}(t-1) \leq 0.1, \quad t \in [0, 1, 2, 3, 4, 5] \quad (3.29)$$

$$f^{a,i}(z) \cdot f^{b,i}(z) = 0, \quad i = 1, \dots, n_g, \forall t \in [0, \infty) \quad (3.30)$$

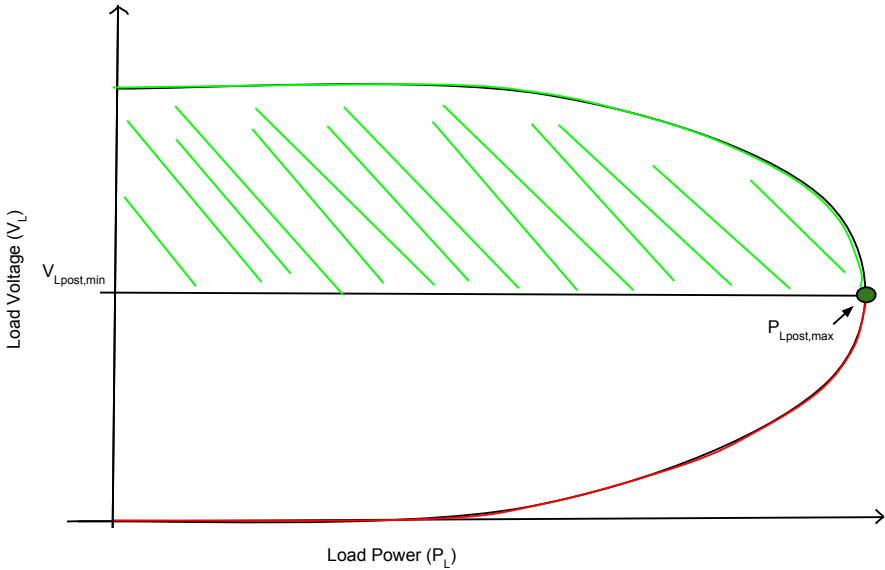
$$f^{a,i}(z) = -E_f^i + K_A^i (V_{ref} - V^i) \geq 0, \quad i = 1, \dots, n_g, \forall t \in [0, \infty) \quad (3.31)$$

$$f^{b,i}(z) = -E_f^i + E_f^{lim} \geq 0, \quad i = 1, \dots, n_g, \forall t \in [0, \infty) \quad (3.32)$$

$$V_{L,post} \geq V_{L,min} \quad (3.33)$$

First expression is the same as before.  $n_g \in [1, 2, 3]$  is the number of generator in the system. The first expression (3.20) is the same as before, the pre-contingency load is to be maximized. (3.21)-(3.24) compute the load powers and correspond to (3.2) and (3.3). (3.4) corresponds to (3.25) which is one of the load recovery models (3.14) or (3.15). (3.26) and (3.27) have the same definitions as (3.5) and (3.6) explaining the power system. (3.28) and (3.29) are the generator constraints in (3.7) and (3.31) and (3.32) are the constraints on the AVRs.

Lastly, (3.33) expresses the stability constraint (3.11) in terms of the voltage level. The voltage at the load node for the maximum post-contingency load is used as a threshold during computations. The post-contingency voltage at the load node will have to be larger than the threshold ( $V_{Lpost,min}$ ). In a nose curve, the voltage has to stay on the upper side of the curve since ( $V_{Lpost,min}$ ) is located on the tip of the curve, see Figure 3.2.

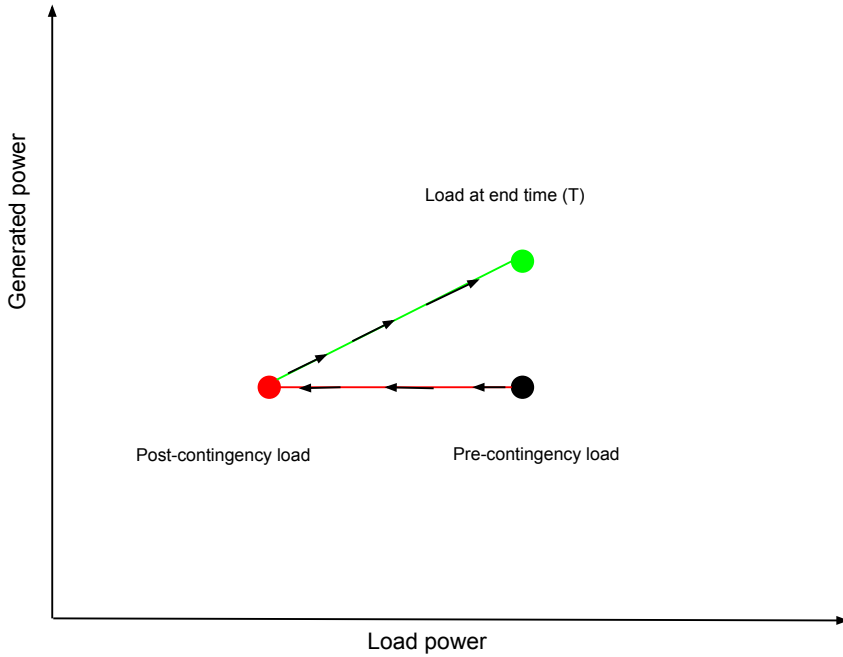


**Figure 3.2** The stability constraint (3.33) expressed in a post-contingency nose curve. The voltage has to be larger than  $V_{L,min}$ .

Figure 3.3 shows the system events for a linear load recovery model in the *Seven-Node System* (Figure 3.1). The horizontal axis is the load power and the vertical axis is the time step. The events in the system are regarded as three, before the fault, right after the fault and when the load power is fully recovered. Before a contingency, we are located at the black dot. A fault occurs in line 5-6 which doubles the line reactance from 0.12 p.u. to 0.24 p.u. The load power decreases to the red dot. The system recovers according to the linear load recovery model (3.14) to the initial load power, the green dot.

The time for the system to recover from a fault is defined as the end time ( $T$ ) and is set to 5 time steps. Generator 1 ( $P_{g,1}$ ) is regarded as a slack bus and initial values on generator 2 ( $P_{g,2}$ ) and generator 3 ( $P_{g,3}$ ) are determined at each computation and may vary between 0-3 p.u.

The optimization problem is divided into three steps. The first step is to find the maximum load at a predefined end time ( $T = 5$ ). The constraints at this step, are on the field voltages in all three generators and the power production. Equations (3.12) and (3.13) are the behaviors of the AVRs in the generators. Constraints on the generator values at the end time are defined as;  $-0.04 * T \leq P_{g,2} \leq 0.04 * T$  and



**Figure 3.3** Events in a power system subjected to a fault and how it recovers. The load power is on the horizontal line and the time is on the vertical line. There are three dots, a black dot representing the load before the fault, a red dot for the load right after the fault and a green dot representing the load at a full load power recovery. The load recovery is linear in this figure but can also be exponential.

$$-0.1 * T \leq P_{g,3} \leq 0.1 * T, \text{ where } T=5.$$

The second step is to find a maximum post-contingency load. The only constraints in this step, are on the AVRs ((3.12) and (3.13)). If a post-contingency load is found, a pre-contingency load is derived using equations (3.16) and (3.18). The pre-contingency load is only feasible if the system has found a static equilibrium. It is very important that the system remains stable at all times, from pre-contingency to the end of the recovery time. Therefore, the newly computed pre-contingency load (from Step 2) and the load at the end time (from Step 1), are compared. The smallest of these two loads, is regarded as the systems pre-contingency load. If the pre-contingency load in Step 2 is replaced by the smaller load at the end time, the post-contingency load is recomputed. Equations (3.16) and (3.18) are used with the updated pre-contingency load to obtain a new post-contingency load.

In Step 3, the linear load recover model (3.14) is included since the optimization is done over the three events, defined in Figure 3.3. A vector of loads are computed

using equation (3.14). These loads are used to find corresponding state variables at a system steady state at each time step (1-4). The results from previous steps are used as initial guesses when optimizing over the three events, pre-contingency, post-contingency and load recovery. Constraints on the AVRs and the rampings of the generators are valid at all time steps. The generator rampings are expressed the following way at each time step;  $-0.04 \leq P_{g,2}(t) - P_{g,2}(t-1) \leq 0.04$  and  $-0.1 \leq P_{g,3}(t) - P_{g,3}(t-1) \leq 0.1$ . At post-contingency the complementary-constraint (3.8) is valid. Step 3 is repeated, but with the exponential load recovery model (3.15).

The complementary-constraint (3.8) is important at post-contingency. The constraint forces the post-contingency load to be located on the nose curve and prevents the load from being very low. Loads larger than  $P_{Lpost,max}$  are not feasible and solutions inside the nose curve are too low. Though, it is possible for the load to recover from a fault when the post-contingency load is low, the issue is not finding feasible solution but rather, how easy it might be and to some extends unrealistic. It is less difficult for the load to recover when the post-contingency load is low than high. By using (3.8) the low load is avoided, since it has to be located on the nose curve. A higher post-contingency load is more realistic but feasible solutions are more difficult to find.

At discrete time steps 1-5, the switching-constraint does not have to be enforced, which expands the search area for the optimization problem. The loads  $P_{L,i}, i \in [1, 2, 3, 4, 5]$  are not forced to be on their nose curve, as long as  $P_{L,i} \leq P_{Lmax,i}$ , where  $i$  stands for the current time step and its nose curve.

How the optimization problem is implemented in Matlab can be read in Chapter 4.

# 4

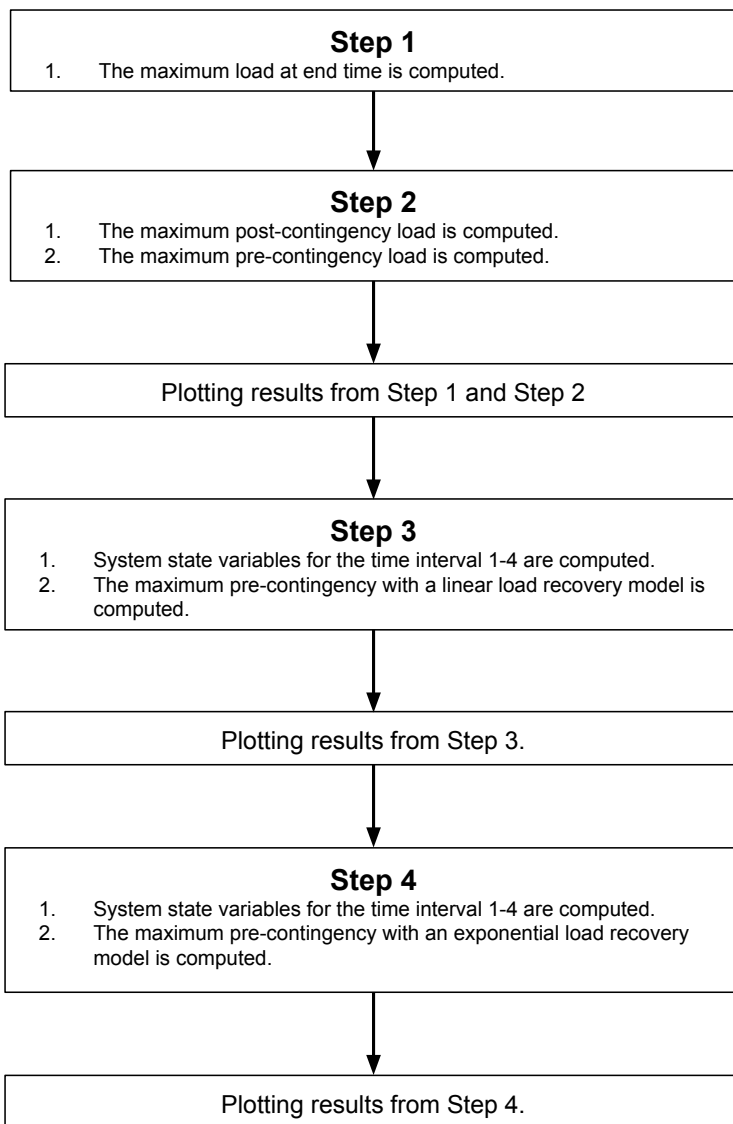
## Implementation

Chapter 2 gives the reader basic information about power systems and in Chapter 3, the optimization problem is presented. This chapter shows how the optimization problem is solved using Matlab. Before any computations, the power system *Seven-Node system* is built in a Matlab-file.

The first section covers the building blocks (Matlab-files) in the Seven Node System, which contain equations from Chapter 2. Some of the equations are simplified to make it possible to find feasible solutions during computations. The overview of the building blocks of the system, is followed by the use of the optimization tools *fsolve* and *fmincon*. This part is also sectioned into parts done in chronological order. For further understanding, the *Nose Curve* is used to explain events in our system. But first an overview of the programming.

### 4.1 Overview

A quick overview of the programming might be helpful before explaining any details. To make the work easier to understand, programming can be divided into four steps. The steps will be named after the order they are executed in. *Step 1* is where the maximum load at end time ( $T = 5$ ) is found. *Step 2* is where the maximum pre-and post-contingency load are evaluated. *Step 3* is where the optimization over all the time step ( pre-and post-contingency and over a time interval 1 – 5) is done to find the maximum pre-contingency load. The linear load recovery is assumed and state variables at time  $t \in [1, 2, 3, 4]$  are computed. The initial guess for this optimization is a vector with state variables computed in Step 1, Step 2 and the state variables for the discrete time interval 1-4 which are computed right before the optimization. *Step 4* is similar to Step 3 but with an exponential load recovery. In between the step, results are plotted. Figure 4.1 is an overview of the programming.



**Figure 4.1** An overview of the programming. *Step 1* computes the maximum load at end time  $T = 5$ . *Step 2* computes the maximum pre-contingency and the maximum post-contingency load. *Step 3* finds the maximum pre-contingency load over a discrete time interval 0-5 with a linear load recovery model. *Step 4* computes the maximum pre-contingency load over a discrete time interval 0-5 with an exponential load recovery model.

**Table 4.1** Power System Details contains values such as line impedances eg. impedance between node 1 and 5 is  $x_{15}$ . The gains of the exciters ( $K_A$ ), the capacitance ( $B$ ), voltage reference ( $V_{ref,i}$ ) and the maximum excitation field voltage ( $E_{f,i}^{lim}$ ) are also included in the table

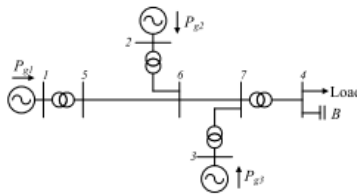
$x_{15} = x_{26}$	$x_{37} = x_{47}$	$x_{56}$	$x_{67}$
0.032	0.016	0.12	0.005625
B	$K_A$	$V_{ref,i}$	$E_{f,i}^{lim}$
0.25	100	1	2.5968

## 4.2 The Seven-Node System

### Overview of the system

The *Seven-Node System* is taken from [Karystianos et al., 2007] and built in Matlab by writing several functions as building blocks (M-files). These M-files will soon be explained but first a quick glance of the system specifications.

The name *Seven-Node System* comes from the number of nodes in the power system, see Figure 4.2 (the same system as the example in chapter 1, Figure 1.1). Study Figure 4.2. There are three buses, two PU buses,  $P_{g2}$  and  $P_{g3}$ , and one slack bus  $P_{g1}$ . Four transformers are located between nodes, 1-5, 2-6, 3-7 and 4-7 and the load is connected at node 4. Table 4.1 contains values of system components such as reactance of lines and transformers. All transformer ratios were set to 1 and the relationship between active and reactive power consumption of the load was  $P_L = 2Q_L$  to keep the power factor at  $2/\sqrt{5}$ . A fault in line 5-6 would occur and this is expressed by doubling the impedance of the line to  $x_{56} = 0.24$ .



**Figure 4.2** The *Seven-Node System* from [Karystianos et al., 2007]. The power system consists of three generators  $P_{g1}$ ,  $P_{g2}$  and  $P_{g3}$  that are connected to the network via transformers. The transformers are in between busbars 1-5, 2-6, 3-7 and 4-7. The load and the capacitor ( $B$ ) are in parallel and connected to busbar 4.

## Building Blocks

The optimization problem is presented in Chapter 3. Since differential equations are hard to solve, simplified equations are used to express the system. The m-file *sjunodsmodell* (Seven-Node model in Swedish), computes the system, excluding the AVRs. *sjunodsmodell* sends back nonlinear equations of power flow equations and generated power values. The inputs of this m-file are state variables ( $x$ ), the initial load ( $P_{L0}$ ), the initial generated power ( $GEN_0 = [P_{g2} \ P_{g3}]$ ) and a fault indicating variable (*fault*)<sup>1</sup>. The main state variables ( $x$ ) are the excitation field voltages ( $E_f$ ), the voltage phasors and amplitudes at all seven nodes ( $\theta_i$  and  $U_i$ , where  $i = 1, \dots, 7$ ). The state variables computes the system nodal admittance matrix, the power flow and the generated active and reactive powers from the generators in m-files *get\_nod\_adm*, *get\_p\_flow* and *get\_p\_gen*. The nonlinear equations which are sent back are the difference between initial generation ( $GEN_0$ ) and the calculated generation from *get\_p\_gen* and the difference between the initial active and reactive powers generated and consumed and the net power flow from *get\_p\_flow*. See Figure 4.3 for a chart of the Matlab-file *sjunodsmodell*.

*sjunodsmodell\_p0* functions as *sjunodsmodell* but instead of an initial load ( $P_{L0}$ ) as input,  $p_0$  is sent in and the load is calculated as a voltage dependent load (3.16).

### 4.3 Complementary Building Blocks and Matlab-files

Some of the files might not build the actual power system but they are vital in the computations. These m-files are gathered under this subsection.

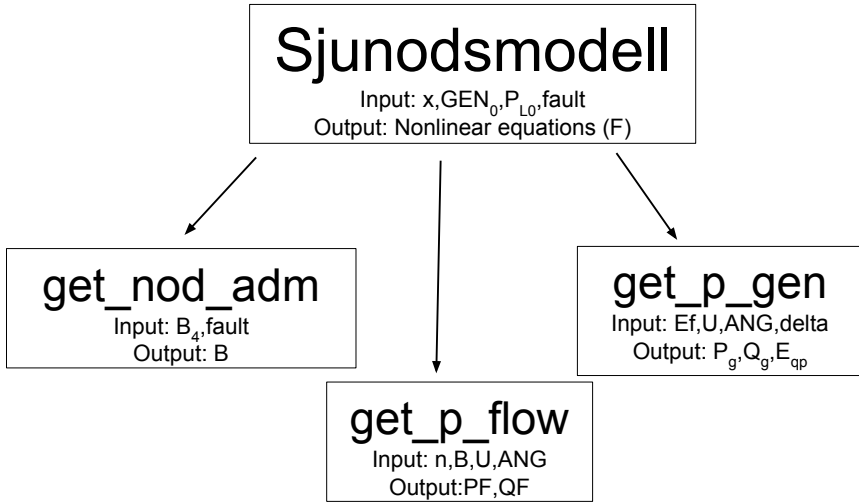
The AVRs are expressed in *get\_g\_fa* and *get\_g\_fb*. When the controllers are not saturated, *get\_g\_fa* (3.12) is used. Saturated generators use (3.13), implemented in *get\_g\_fb*. The generators can switch between these two Matlab-files but to switch between these equations, two vectors, *aset* and *bset* are defined, one contains unsaturated while the other contains saturated generators. *aset* holds the unsaturated generators and *bset* contains the saturated ones. As inputs, both files have the state variables ( $x$ ) and number of generators ( $nbr\_gen = 3$ ). *aset* is an input to *get\_g\_fa* while *bset* is input to *get\_g\_fb*.

Two other important Matlab-files are *get\_PL\_from\_p0* and *getx\_from\_PL* which both use the scripts *sjunodsmodell*, *get\_g\_fa* and *get\_g\_fb*. *get\_PL\_from\_p0* calculates the matching load ( $P_L$ ) for the variable  $p_0$ <sup>2</sup>. The function has the state variables ( $x$ ),  $p_0$ , the initial generation ( $GEN_0$ ) and the variable *fault* as inputs and as outputs, the load ( $P_L$ ), updated state variables and an exitflag which tells if the computations made in *get\_PL\_from\_p0* are feasible. *getx\_from\_PL* is very similar to *get\_PL\_from\_p0* but this m-file delivers the corresponding state variables to the initial load  $P_{L0}$ . The inputs are initial state variables ( $x_0$ ), load ( $P_{L0}$ ) and generation

<sup>1</sup> If *fault* = 0, there is no fault in the system and if *fault* = 1, there is a fault in the system

<sup>2</sup>  $p_0$  is presented in Chapter ?? equations (3.16) and (3.18). It can be obtain by solving a system of equation.





**Figure 4.3** The building blocks of the *Seven-Node System* or *Seven-Node Model* (*sjunodsmoell* in Swedish). The Seven-Node System has four inputs, system state variables ( $x$ ), initial generation for generator  $P_{g2}$  and  $P_{g3}$  ( $GEN_0 = [P_{g2} \ P_{g3}]$ ), initial load ( $P_{L0}$ ) and fault ( $fault = 0 = no \ fault$  and  $fault = 1 = there \ is \ a \ fault$ ). Further, the Matlab-file *sjunodsmoell* contains three m-files, *get\_nod\_adm*, *get\_p\_flow* and *get\_p\_gen*.  $B_4$  is the capacitance at node 4 and  $B$  is the nodal admittance matrix.  $E_f$  is the excitation field voltages,  $U$  and  $ANG$  are the voltage magnitudes and phase angles at all seven nodes while  $delta$  are the voltage phasors at the generator nodes.  $P_g$  and  $Q_g$  are the calculated active and reactive powers from the generators and  $E_{qp}$  are the actual generator voltages (not the terminal voltage  $U$ ).  $n$  is the number of nodes in the system (7 nodes).  $PF$  and  $QF$  are the active and reactive power flows.

( $GEN_0$ ) and the variable *fault*. The outputs are the current state variables and an exitflag that shows feasibility of the computations.

*get\_end* uses the Matlab-tool *fmincon* to find the maximum load at a defined end time  $T$ . Its inputs are initial state variables, end time ( $T$ ) and the initial generation ( $GEN_0$ ) and outputs, the state variables at end time and the exitflag. The maximum load is included in these state variables, placed as the first variable in the vector. In *get\_end*, function *nonlcon\_end* is used to handle all nonlinear equations of the system. *nonlcon\_end* is a part of *fmincon*, where all nonlinear equations are placed while linear equations are put in matrices. The inputs of *nonlcon\_end* are state variables ( $x$ ) and the fault variable (*fault*). The outputs are the nonlinear equations of the *Seven-Node System*. A second function which belongs to *fmincon* is *funmin*. *funmin* is the objective function, it decides what is to be optimized. Our objective is to maximize the load which is done by minimizing the negative load since *fmincon* tries to find the minimum.

*get\_post* is comparable to *get\_end*. The interest here is to find maximum post-contingency load. The inputs are the same as for *get\_end* without the end time and the output related to post-contingency, i.e. state variables at post-contingency with its exitflag and maximum load as the first state variable. As in *get\_end*, *fmincon* is used by *get\_post*.

*get\_interval* computes state variables for the discrete time interval 1-4. The main inputs are previously calculated pre-and post-contingency loads, the end time ( $T$ ), the initial generation ( $GEN_0$ ), the end generation ( $GEN_{end}$ ) and the post-contingency state variables. The outputs are the state variables for the time interval 1 – 4 including power generation at all time steps and lastly, exitflags. Matlab-file *getx\_from\_PL*, is used at each time step. Most important, the loads at each time step are decided in function *get\_P\_LC\_lin* for a linear load recover model and *get\_P\_LC\_exp* for exponential load recovery model.

*pre\_post\_int\_end* finds the maximum pre-contingency load over an entire time interval. The initial guess is a vector with initial state variables, previous computed variables from pre-contingency, post-contingency, at end time and variables for the interval 1 –  $T$ . *Fmincon*, *fsolve* and most of the m-files mentioned are involved. The interested reader can be provided the Matlab-file by contacting the author. The two essential outputs from this file, are the maximum pre-contingency load and the exitflag indicating whether the optimization was successful or not. For an exponential load recovery *nonlcon\_fa\_fb\_lin* is replaced by *nonlcon\_fa\_fb\_exp*. Table 4.2 shows what equations are used in the Matlab-files.

## 4.4 Fsolve and Fmincon

How the two optimization tools, *fsolve* and *fmincon*, are used will be explained in this section.

**Table 4.2** This table shows which equations are included in the Matlab-files. On the left we have m-files, in the center are the m-files that are included in the files to the left and on the right side the equations. For example, if *sjunodsmode*ll is to the left we would find *get\_nod\_adm*, *get\_p\_flow* and *get\_p\_gen* in the center and no equations to the right (since no equations are actually computed in *sjunodsmode*ll, only in its sub-files). *get\_P\_LC\_lin* can be changed to *get\_P\_LC\_exp* if the load recovery is exponential and not linear. The same holds for *nonlcon\_fa\_fb\_lin* which can be replaced by *nonlcon\_fa\_fb\_exp* for exponential load recovery.

Main function	Functions inside the main function	equations in the main function
<i>sjunodsmode</i> ll	<i>get_nod_adm</i> , <i>get_p_flow</i> , <i>get_p_gen</i>	no equations
<i>sjunodsmode</i> ll_p0	<i>get_nod_adm</i> , <i>get_p_flow</i> , <i>get_p_gen</i>	no equations
<i>get_nod_adm</i>	no m-files	eq.2.34-2.35
<i>get_p_flow</i>	no m-files	eq.2.26-2.27
<i>get_p_gen</i>	no m-files	eq.2.24, eq.2.36-2.37
<i>get_g_fa</i>	no m-files	eq.3.12
<i>get_g_fb</i>	no m-files	eq.3.13
<i>get_PL_from_p0</i>	<i>sjunodsmode</i> ll, <i>get_g_fa</i> , <i>get_g_fb</i>	no equations
<i>getx_from_PL</i>	<i>sjunodsmode</i> ll, <i>get_g_fa</i> , <i>get_g_fb</i>	no equations
<i>get_end</i>	<i>nonlcon_end</i> , <i>funmin</i>	eq.3.12-eq.3.13
<i>get_post</i>	<i>nonlcon_post</i> , <i>funmin</i>	eq.3.12-eq.3.13
<i>get_interval</i>	<i>getx_from_PL</i> , <i>get_P_LC_lin</i>	no equations
<i>get_P_LC_lin</i>	no m-files	eq.3.14
<i>get_P_LC_exp</i>	no m-files	eq.3.15
<i>pre_post_int_end</i>	<i>nonlcon_fa_fb_lin</i> , <i>funmin</i>	eq.3.12-eq.3.13
<i>nonlcon_end</i>	<i>sjunodsmode</i> ll	no equations
<i>get_end</i>	<i>sjunodsmode</i> ll	no equations
<i>nonlcon_fa_fb_lin</i>	<i>sjunodsmode</i> ll, <i>sjunodsmode</i> ll_p0, <i>get_P_LC_lin</i> , <i>get_g_fa</i> , <i>get_g_fb</i>	eq.3.14-eq.3.15

*Fsolve* is used to find static equilibrium. By using this tool we can see if the system can find state variables which correspond to a stable system. At the very beginning of the project *fsolve* was used to determine whether the system had stable operating points for different initial generator values. All system equations from *sjunodsmode*ll, *get\_g\_fa* and *get\_g\_fb* are defined as functions and then computed by *fsolve* to find an equilibrium point for the system.

The tool is used to test different production power, meaning the generator power varies from 0 – 3 p.u. Later on, the sum of the power productions was restricted to 5.1 p.u. due to some optimization issues which were discovered when using *fmincon*. After successfully finding feasible solutions for the system *Seven-Node System* with *fsolve* the actual optimization could begin. For now on, *fsolve* is only used to find static equilibriums for the system, e.g. in the Matlab-file *getx\_from\_PL*, where nothing is optimized and the goal is to find a stable operating point. An example of how *fsolve* functions, is presented below.

**An example of how *fsolve* is used. Similar sequences can be found in functions as *getx\_from\_PL* and *get\_PL\_from\_p0*. *x0* is the initial guess, *nbr\_gen* is the number of generators (three in our system), *GEN\_0* initial guesses of generator 2 and 3, and *xsolve* are the optimal state variables. The first row defines the system as a function. The second row, *fsolve* computes state variables.**

```
1)F = @(x)[sjunodsmode
```

*Fmincon* performs the actual optimization. An objective function decides what is to be minimized. Since the load is to be maximized, the object function is to minimize the negative load. In *fsolve* all system equations are defined and computed the same way but in *fmincon* linear and nonlinear equations are separated. The equations are put in matrices and the nonlinear ones are defined as function, the same way as for *fsolve*. Equations (3.12) and (3.13) are placed in the matrices while the rest of the system equations are put in a m-file for nonlinear equations e.g. *nonlcon\_end* or *nonlcon\_fa\_fb\_lin*. There is an example of how to use *fmincon* below.

**An example of how *fmincon* can be used. The linear equations are placed in the matrices A and B where  $Ax \leq B$  must hold. The nonlinear equations are in *nonlcon*. *options* is used to select the type of computation the user wants (e.g. Newton iteration) and if the computation should be displayed or not.**

```
[x,fval,exitflag] = fmincon(@funmin,x0,A,B,[],[],[],@(x)nonlcon(x,GEN0,T),
```

Equations (3.12) and (3.13) are not always placed in the matrices as linear functions. It is critical that equation (3.8) is satisfied at post-contingency. The complementary-

**Table 4.3** Exit conditions of *fsolve*. All exitflags over zero are considered good results.

---

1	<i>fsolve</i> converged to a root.
2	Change in $X$ too small.
3	Change in residual norm too small.
4	Computed search direction too small.
0	Too many function evaluations or iterations.
-1	Stopped by output/plot function.
-2	Converged to a point that is not a root.
-3	Trust region radius too small (Trust-region-dogleg).

---

constraint (3.8) is relaxed at other time steps to expand the search space for the optimization. The advantage with expanding the search area, is the increased chance of finding solutions to the optimization. (3.8) holds at post-contingency due to voltage stability. If the voltage is too low at post-contingency, chances of finding realistic stable solutions are slim. By activating (3.8), the post-contingency load will end up on the nose curve and provide a more stable load recovery and an acceptable voltage.

The complementary-function is implemented using *aset* and *bset* as mentioned earlier in the chapter. The generators which are not saturated, are placed in *aset* while the saturated ones are put in *bset*. Functions *get\_g\_fa* and *get\_g\_fb* are called to compute new *aset* and *bset* for the time step. If a generator is placed wrongly the generator is moved to the right vector (*aset* or *bset*) and *fmincon* is called again to find a new solution. The optimization with *fmincon* is repeated until all generators are in the right set, using the right equations for their AVRs. The same method can be done using *fsolve* instead of *fmincon*, though a steady state is sought after with *fsolve* and not a solution for an optimization problem.

Two other important constraints are the ramping of the generators. Generator 2 ( $P_{g2}$ ) should not decrease or increase more than 0.04 p.u. per time step and generator 2 ( $P_{g3}$ ) has a max ramping of 0.1 p.u. These limitations are put in the matrices for *fmincon* since they are linear while previously discussed the complementary-constraint is defined with the nonlinear equations.

Both optimization tools *fmincon* and *fsolve* return exitflags, which will tell how feasible the solutions are. Tables 4.3 and 4.4 give an overview of the exitflags. Solutions with exitflags over zero were regarded as successful and the pre-contingency load was used in the results.

**Table 4.4** Exit conditions of fmincon. As for fsolve, flags bigger than zero are seen as useful results.

---

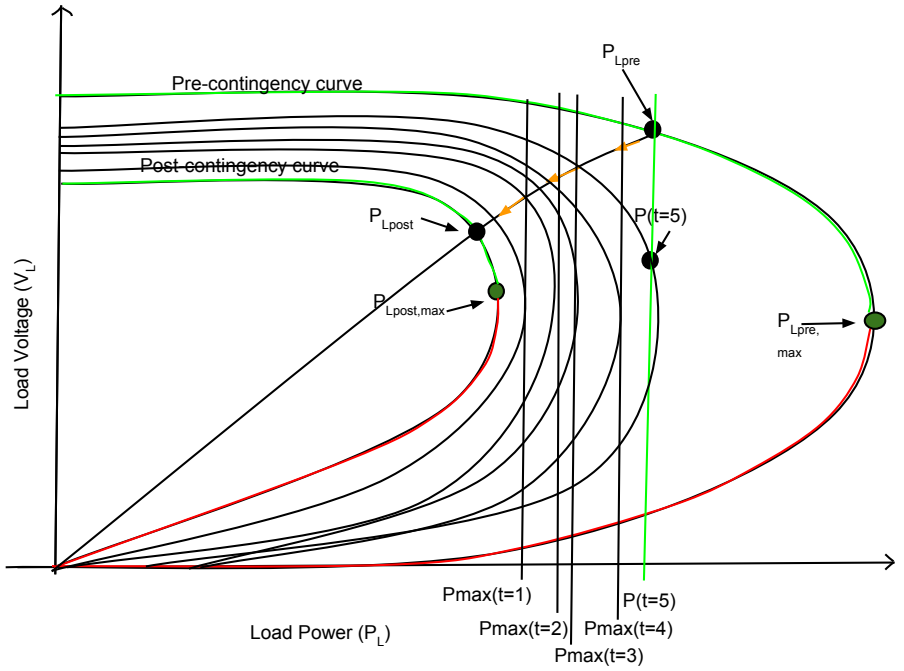
1	First order optimality conditions satisfied.
2	Change in $X$ too small.
3	Change in objective function too small.
4	Computed search direction too small.
5	Predicted change in objective function too small.
0	Too many function evaluations or iterations.
-1	Stopped by output/plot function.
-2	No feasible point found.
-3	Problem seems unbounded.

---

## 4.5 Nose Curve demonstration on the Seven-Node System

The *Seven-Node System* is roughly described using the Nose Curve. Figure 4.4 represents our system. There are three important curves, the *Pre-Contingency Curve*, the *Post-Contingency Curve* and the curve at end time  $T = 5$ . Before the fault, we are located on the *Pre-Contingency Curve* but a fault occurs and the system is represented by the *Post-Contingency Curve*. The load power is to recover in five time steps which are five solid curves between the *Pre-Contingency Curve* and the *Post-Contingency Curve*.

By enforcing the complementary-constraint (3.8) at post-contingency the load voltage is bounded to the *Post-Contingency Curve*, making the optimization more difficult. The voltage constraint (3.33) forces the load to be located on the upper part of the curve. If (3.8) is valid at each time step, it becomes more difficult to find a feasible solution for the optimization problem. But by relaxing the switching-constraint, the search area of the optimization will expand and increasing the chances of finding feasible solutions. It is important that  $(P_{L,i}(t))$  is less than the current maximum load  $(P_{max,i}(t))$ , i.e.  $(P_{L,i}(t) \leq P_{max,i}(t))$ , at each time step  $i$ , for the system stability. At end time  $T=5$ , the load power is fully recovered since it is at the same vertical line as the pre-contingency load.



**Figure 4.4** An example of how the *Nose Curve* for the *Seven-Node System* would look like.

# 5

## Results

The results are presented in four sections, pre-contingency, post-contingency, at a defined end time  $T$  and lastly all sections mentioned in a time interval. The results will be analysed in Chapter 6.

### 5.1 Pre-contingency

The maximum pre-contingency loads can be seen in figures 5.1- 5.2. The two figures contain the same results but represented from two different angles. The initial generation of generator 2 and 3 are on the  $x$ - and the  $y$ -axis and the load is placed on the  $z$ -axis. The optimization problem was solved using `fmincon` and the Matlab-file `sjunodsmoell`. The figures contains only results with feasible solutions meaning no results with exitflags equal or under zero were plotted. Further analysis regarding this outcome will be carried out in the next chapter.

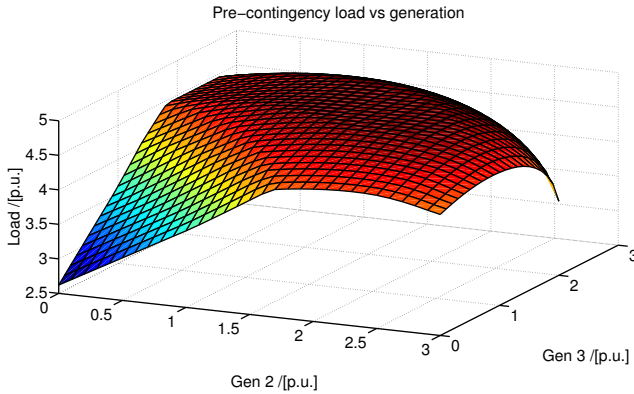
### 5.2 Post-contingency

As in the pre-contingency figures, the  $x$ -and the  $y$ -axis are the initial generations from generator 2 and 3 while the  $z$ -axis is the post-contingency load. Figures 9.3-9.4 show the post-contingency load. Only results from `fmincon` with feasible solutions are plotted, i.e. no results with exitflags under or equal to zero are included. The two post-contingency figures have the same results but plotted from different angles.

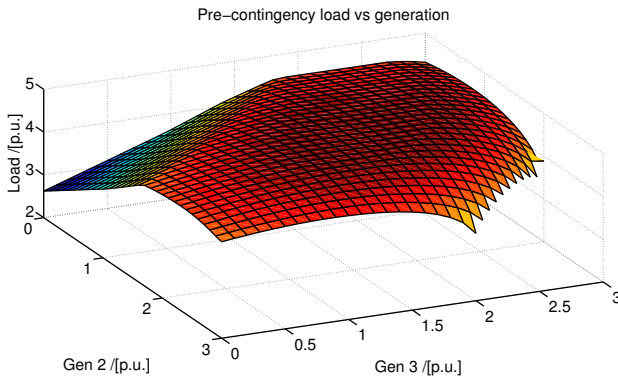
### 5.3 At end time $T=5$

The  $x$ -and  $y$ -axis in figures 9.5 and 9.6 are the same as the ones in pre-and post-contingency figures, i.e. the initial generations from generators 2 and 3. The  $z$ -axis is the maximum load at end time  $T = 5$ .





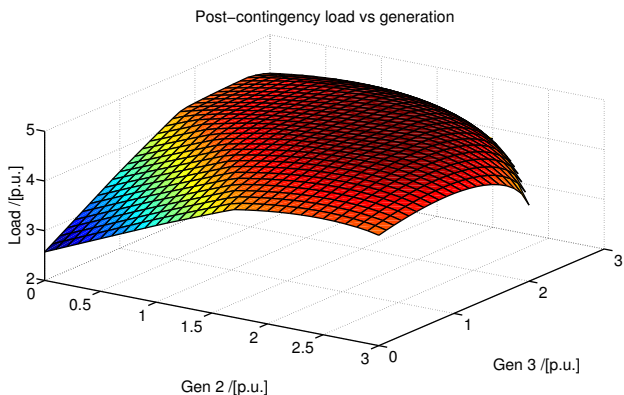
**Figure 5.1** The pre-contingency load is computed by the optimization function `fmincon` from Matlab.  $x$ - and  $y$ -axis are the initial generations and  $z$ -axis is the actual solution of the problem, the pre-contingency load.



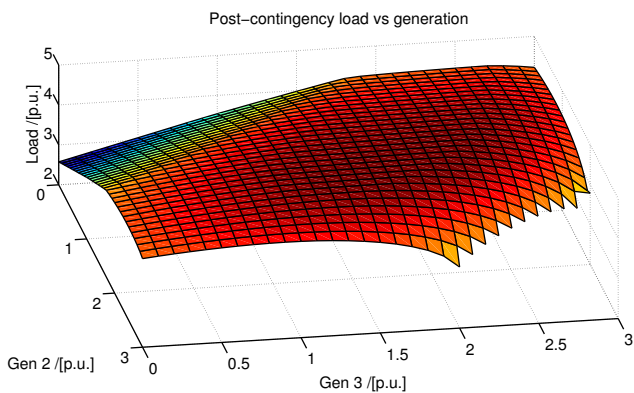
**Figure 5.2** Pre-contingency load, same results as in figure 5.1 but from a different angle.

## 5.4 Pre, Post and time interval

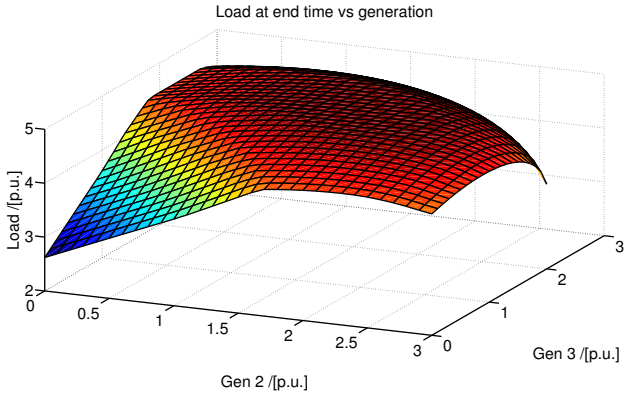
This section contains results for pre-contingency, post-contingency and a time interval  $t \in [1, 2, 3, 4, 5]$  with a linear and exponential load recovery model. The results for the linear load recovery model, using equation (3.14), can be seen in figures 5.7-5.8. Like the previous plots, the  $x$ -axis and  $y$ -axis are initial generated power and the  $z$ -axis is the pre-contingency load. The results for the exponential load model, using



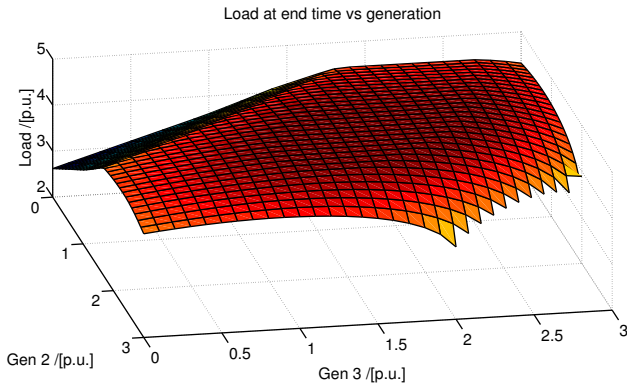
**Figure 5.3** The post-contingency load is obtained from `fmincon`. The  $x$ - and the  $y$ -axis are the initial generations and the  $z$ -axis the post-contingency load. Only feasible results are presented.



**Figure 5.4** Post-contingency load, same results as in figure 9.3 but from another point of view.

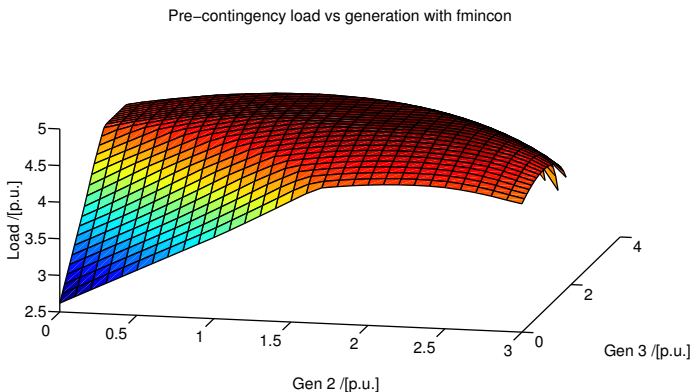


**Figure 5.5** The load at end time ( $T = 5$ ) shows feasible solutions from `fmincon` when searching for the maximum load. The  $x$ -axis and the  $y$ -axis are the initial generated power from generator 2 and 3, while the  $z$ -axis is the sought load.

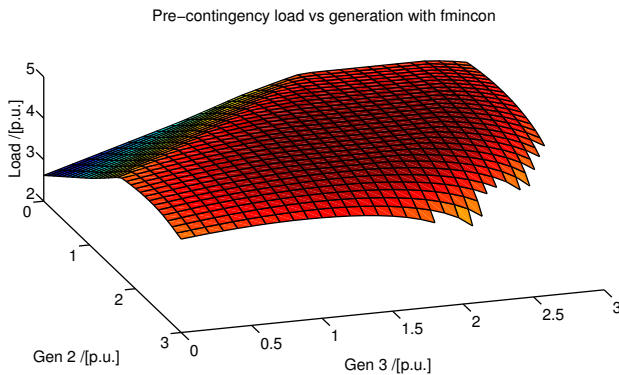


**Figure 5.6** The load at end time. Identical results as 9.5 but from a different angle.

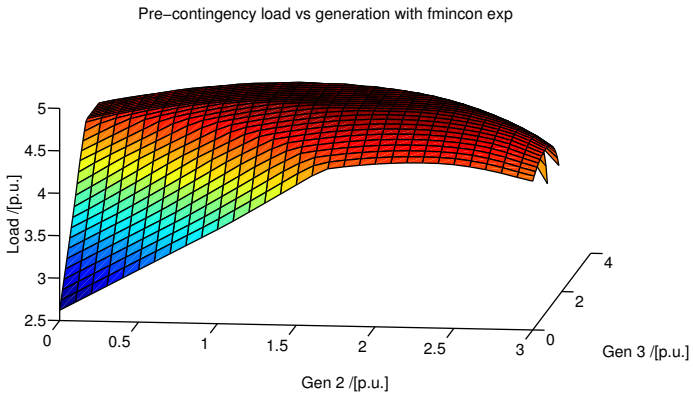
equation (3.15), are seen in figures 5.9- 5.10 and have the same representation of the axis as previous plots. This optimization is more complex due to the number of time steps which are included. *Fmincon* is to optimize and find the maximum pre-contingency load over a time period while still retaining a stable system. Beside a stable system, the optimization had to fulfill certain constraints. The constraints on the generators ramping and the limited field voltage had to be included. The results were different than expected and will be commented on in chapter 6.



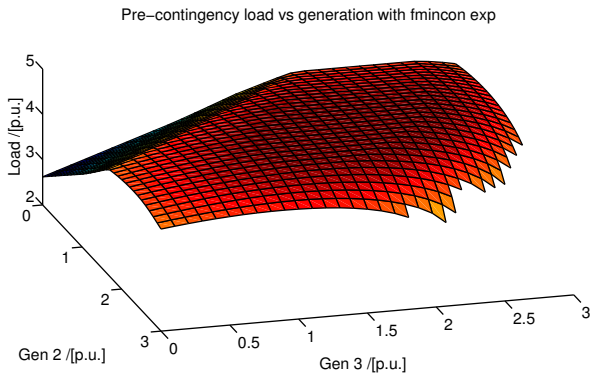
**Figure 5.7** Pre-contingency load with linear load recovery model.



**Figure 5.8** Pre-contingency load with linear load recovery model, same results as in 5.7 but from another angle.



**Figure 5.9** Pre-contingency load with exponential load recovery model.



**Figure 5.10** Pre-contingency load with exponential load recovery model as in 5.9 but another angle.

# 6

## Discussion

The difficulty of finding solutions to an optimization problem varies depending on the system models. The approximations done and constraints subjected onto the system will determine whether feasible solutions will be found. With the approximations and constraints assumed in this study, a pre-contingency load is found in most cases but not when both generators have high values.

At the beginning of the project, the optimization was not broken down into the four steps, as presented in Chapter 4. The entire optimization was done over pre-contingency, post-contingency and the defined discrete time interval 1-5, all at once. This way of solving the problem made it difficult to find feasible solutions and the problem had to be broken down into separate parts. The two first steps would tell if there existed maximum possible loads for a stable system at pre-contingency, post-contingency and at the end time ( $T = 5$ ). This would indicate which areas had solutions and provide good initial guesses for the “overall” optimization done in Step 3 and Step 4. The previous results from Step 1 and Step 2, could therefore determine which areas were worth exploring. As one can see, results where the generator values are both high, are not shown due to discontinuities in the graphs. Since the very high generation combinations (where the sum of the generators is higher than 5.1 p.u) were not interesting for me, they were not computed. The unmodified figures, the ones where high generator values are included, can be seen in Chapter 9.2.

All figures show how the maximum possible load can be found in the middle, where generation of both generators is about 1.5 p.u. It seems like the system finds stable equilibriums which allow high loads in those areas. Another important matter for stability is the voltage level in the system, especially at the load node. The load voltage at post-contingency will determine whether a stable solution to the optimization problem will be found. Too low post-contingency load voltage will make it difficult for the system to recover from the fault. Both linear and exponential load recovery models work excellently and feasible solutions were found. Since the optimization was possible on a small system, it should be possible to implement it on a larger system. The question is how accurate the results will be with all simplifications made.

Another vital discovery, is the importance of the initial values for the optimization. Bad initial guesses could give suboptimal solutions or worse, no solutions at all. This is the reason why the programming was divided into steps, where the two first steps (1 and 2) were mainly for computing good start values for Step 3 and Step 4.

The voltage dependent load model is significantly harder to stabilise than the previous load models. The short time interval 1-5 will not be enough to solve the problem. The amount of time steps required are infinitely many and numerical methods are needed to find feasible solutions. Although the voltage dependent load model is complex, it might give a more accurate behavior of a system.

Verifications were also done in Matlab. It was mentioned that the optimization tool *fmincon*, uses matrices for linear constraints, where  $Ax < B$ . This constraint was controlled at times. Another good way of verifying the results, was to put the computed state variables into the actual system, the *Seven-Node System*. If the sum of these results were smaller than a threshold ( $1e^{-5}$ ), the state variables were considered correct.

# 7

## Conclusion

It is possible to find a pre-contingency load for a small system by using tools in Matlab. The load power is able to recover from a fault using both linear and exponential load recovery models, while maintaining stability. Matlab is capable of solving optimization problems, using approximated models.

Observations shows that, generated power over 5.1 p.u. will not give numerically correct values, since it is difficult to find an equilibrium then. The initial values of the optimization have proven to be important. Therefore, the problem has been broken down in steps, where the first steps provide good initial guesses for the final computation. Although, the defined optimization problem was successfully solved in this study, a more complicated load model, such as a voltage dependent load model during load recovery, will probably have a different outcome.



# 8

## Future work

It would have been interesting to study the optimization done in this thesis, on a large scale system. Another question is, how much the results in this project differ from results of a more complex method. Would more complicated models be worth the advanced computations or is the simple approach good enough? Lastly, verifications in a different program could also be useful to confirm the results.

# 9

## Appendix

### 9.1 Per Unit System

It is common to use per unit values (p.u.) when describing quantities in an electric grid. The values are related to a predefined base unit as a multiple of the base unit. How to transform a value from S.I. units to per unit values is done in equation (9.1).

$$\text{quantity in p.u.} = \frac{\text{actual value}}{\text{base value}} \quad (9.1)$$

The advantage of using p.u.-values is that equipment with different ratings can easily be compared. The p.u. system is very convenient for systems with transformers, where the primary side and secondary side might have values in different orders. By using the p.u. values the primary and secondary side will be comparable.

It is mainly the currents, voltages, powers (active and reactive) and impedances which are transformed to p.u. values. To calculate p.u. values, two nominal values need to be decided as bases. For example is a nominal voltage ( $U_B$ ) and a nominal apparent power ( $S_B$ ) chosen as bases. From Ohm's law a base current ( $I_B$ ) and a base impedance ( $Z_B$ ) are found.

$$I_B = \frac{S_B}{U_B} \quad (9.2)$$

$$Z_B = \frac{U_B}{I_B} \quad (9.3)$$

Now for the per unit values using eq.9.1,

$$s = \frac{S}{S_B} \quad (9.4)$$

$$u = \frac{U}{U_B} \quad (9.5)$$

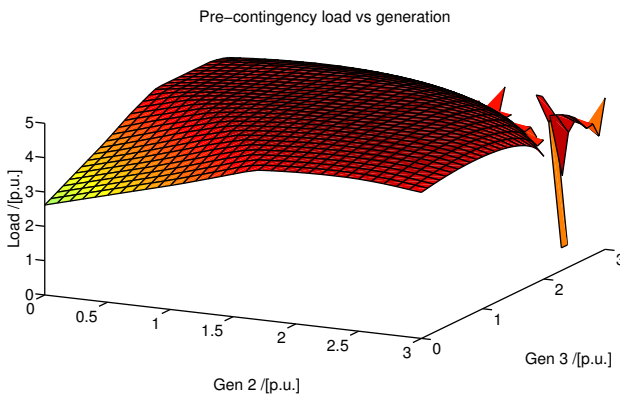
$$i = \frac{I}{I_B} \quad (9.6)$$

$$z = \frac{Z}{Z_B} \quad (9.7)$$

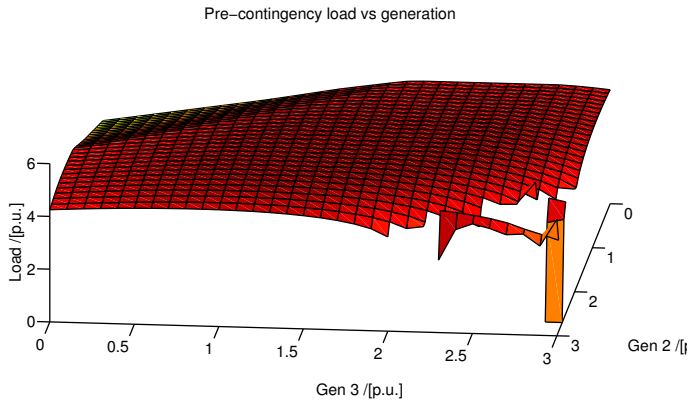
where all quantities in lower case letters are in the p.u. values and the upper case letters are the actual values of the quantities. More examples of the per unit system is found in [Andersson and Franck, 2012].

## 9.2 Figures

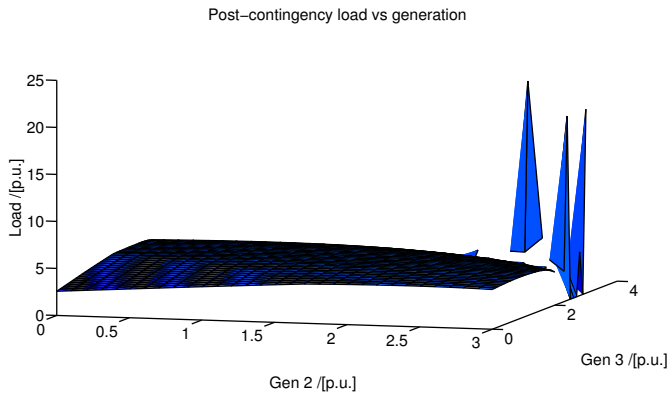
These are the figures which to high values of the generators into account. As discussed in chapter 6, the sum of the generators had to be less than 5.1 p.u. to obtain reasonable results.



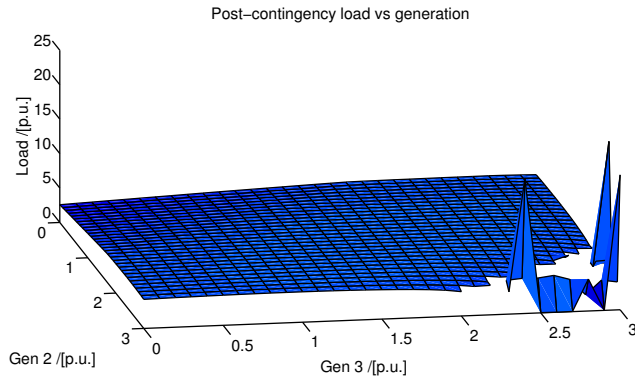
**Figure 9.1** Pre-contingency load without generation restrictions on generators to be less than 5.1 p.u. in total.



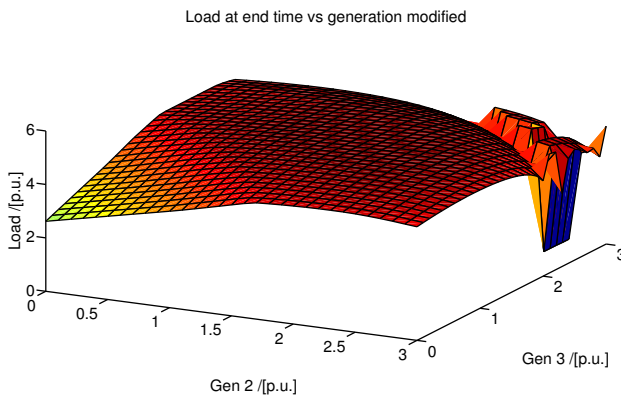
**Figure 9.2** Pre-contingency load without generation restrictions, same results as above but with a different angle.



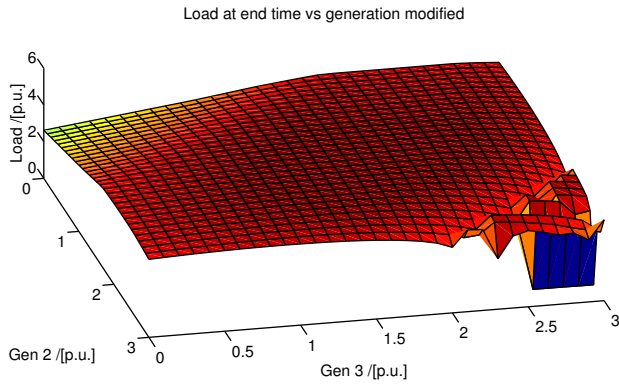
**Figure 9.3** Post-contingency load without generation restrictions on generators.



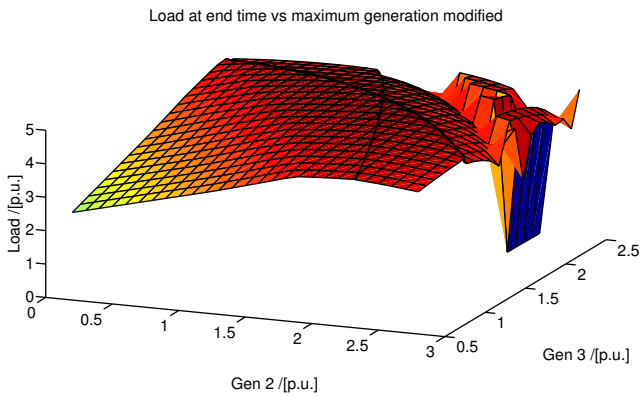
**Figure 9.4** Post-contingency load without generation restrictions.



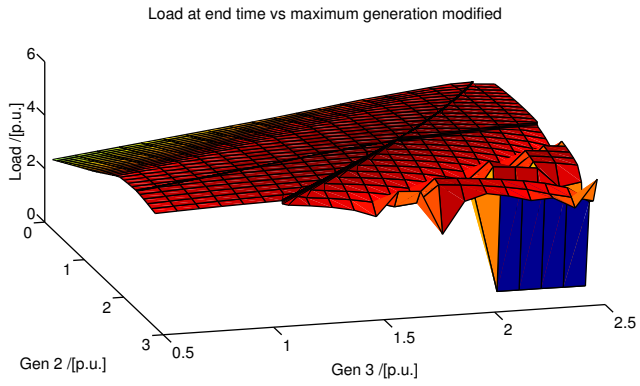
**Figure 9.5** Load at end time without generation restrictions on the generators.



**Figure 9.6** Load at end time without generation restrictions.



**Figure 9.7** Load at end time plotted against maximum generation without generation restrictions



**Figure 9.8** Load at end time plotted against maximum generation without generation restrictions

# Bibliography

- (2014). URL: <http://www.svenskenergi.se/Elfakta/Elmarknaden/>.
- Andersson, G. (2012). *Power System Analysis - 227-0526-00 - 227-0526-00*. Power Systems Laboratory, ETH Zürich.
- Andersson, G. and C. M. Franck (2012). *Electric Power Systems - 227-0122-00*. EEH - Power Systems and High Voltage Laboratories, ETH Zürich.
- Eriksson, R., M. Perninge, and L. Söder (2011). “Transfer capacity enhancement by adaptive coordinated control of hvdc-links based on forecasted load paths”. *European transactions on electrical power* **21**:3, pp. 1455–1466.
- Hamon, C., M. Perninge, and L. Soder (2013). “Applying stochastic optimal power flow to power systems with large amounts of wind power and detailed stability limits”. *Bulk Power System Dynamics and Control-IX Optimization, Security and Control of the Emerging Power Grid (IREP), 2013 IREP Symposium*, pp. 1–13.
- Karystianos, M. E., N. G. Maratos, and C. D. Vournas (2007). “Maximizing power-system loadability in the presence of multiple binding complementarity constraints”. *Circuits and Systems I: Regular Papers, IEEE Transactions on* **54**:8, pp. 1775–1787.
- Perninge, M. (2013). “Approximating the loadability surface in the presence of snb–sll corner points”. *Electric power systems research* **96**, pp. 64–74.
- (2014). “Finding points of maximal loadability considering post-contingency corrective controls”. *Electric Power Systems Research* **116**, pp. 187–200.
- Van Cutsem, T. and C. Vournas (1998). *Voltage Stability of Electric Power Systems*. Ed. by 1st edition. Vol. 441. Springer.



<b>Lund University</b> <b>Department of Automatic Control</b> <b>Box 118</b> <b>SE-221 00 Lund Sweden</b>		<i>Document name</i> <b>MASTER 'S THESIS</b>	
		<i>Date of issue</i> <b>November 2014</b>	
		<i>Document Number</i> <b>ISRN LUTFD2/TFRT--5960--SE</b>	
<i>Author(s)</i> <b>Karen Kyeyune</b>		<i>Supervisor</i> <b>Magnus Perninge, Dept. of Automatic Control, Lund University, Sweden</b> <b>Anders Rantzer, Dept. of Automatic Control, Lund University, Sweden (examiner)</b>	
		<i>Sponsoring organization</i>	
<i>Title and subtitle</i> <b>Optimizing Control of a Power System during an Emergency</b>			
<i>Abstract</i> <p>Population growth, infrastructure and economy puts pressure and demand on the existing power supplies. It puts strains on the current power systems which causes instabilities in the systems. This is an ongoing challenge which needs an immediate solution. The objective of this thesis is voltage stability. This is examined with the help of constructing a small power system using a programming language called Matlab. Optimization tools provided by Matlab are used to find the maximum possible pre-contingency load, while still maintaining a stable system. To find feasible solutions in Matlab, system models, such as load models and power line models are simplified.</p> <p>The results show that a system which has experienced a fault can successfully recover by using a linear load recovery model and an exponential load recovery model. Certain constraints, such as generator ramping and limitations on the field voltages in the generators are implemented. Feasible solutions are found although constraints might have made it more difficult under the course of this study.</p> <p>These findings are rough approximations of how a small power system can operate. Though, this can give valuable information on how a more complex system might act before and after a contingency as well as suitable recovery paths.</p> <p>Although the thesis is more suited for those who have some knowledge in control or power systems, a reader without a technical background can enjoy the paper too.</p>			
<i>Keywords</i>			
<i>Classification system and/or index terms (if any)</i>			
<i>Supplementary bibliographical information</i>			
<i>ISSN and key title</i> <b>0280-5316</b>			<i>ISBN</i>
<i>Language</i> <b>English</b>	<i>Number of pages</i> <b>1-66</b>	<i>Recipient's notes</i>	
<i>Security classification</i>			