

FMH606 Master's Thesis 2023

Electrical Power Engineering

Direct methods for transient stability analysis and contingency screening in power systems

Adrian Knivsflå Toft

Faculty of Technology, Natural Sciences and Maritime Sciences
Campus Porsgrunn

Course: FMH606 Master's Thesis, 2023

Title: Direct methods for transient stability analysis and contingency screening in power systems

Number of pages: 67

Keywords: Transient Stability Analysis, Direct Methods, CUEP, Lyapunov Stability

Student: Adrian Knivsflå Toft

Supervisor: Gunne John Heggliid

Summary:

This master's thesis addresses the challenges associated with power interruption and unstable operation in power systems. It highlights the potential for enhancing transient stability analysis through the utilization of direct methods. Furthermore, it recognizes the increasing importance of online transient stability analysis in the future, driven by factors such as electrification and an influx of renewable energy sources. These factors necessitate operating the grid closer to its operational limits.

The primary aim of this master thesis was to establish the theoretical framework for direct methods and devise a practical approach for implementing the controlling unstable equilibrium point method.

The thesis includes two test cases: a single machine infinite bus test case and a multimachine test case. Results obtained from these test cases align with expectations and existing literature, in computing critical clearing times. For the multimachine test case a homotopy approach was utilized to calculate the CUEP.

Preface

Throughout the course of this master thesis, the scope and objectives have evolved as a result of gaining new knowledge and insight into the subject matter. This is a natural occurrence given the novelty of the subject at USN.

Fortunately, I was able to combine transient stability analysis, one of the pillars of stability analysis in power systems, with a methodology that is evolving and has garnered significant attention both in academic and professional circles. Working in a field where all the solutions are not yet established has been both an inspiring and humbling experience. Additionally, this project has allowed me to expand my knowledge of power systems whilst also improving my programming skills, which prior to this thesis were almost non-existent.

I would like to extend my deepest gratitude to my supervisor, Professor Gunne John Heggli. His initiative in proposing and guiding this thesis, as well as valuable feedback and encouragement has been instrumental in its completion.

The computer programs and data tools used in this thesis are:

- Microsoft Word for thesis writing
- The programming language Python 3.8 and writing in Spyder, together with the modules Numpy, Scipy, Matplotlib and Sympy
- Microsoft Visio for making flowcharts.
- Zotero for referencing

Porsgrunn, 15.05.2023

Adrian Knivsflå Toft

Contents

Preface	3
Contents.....	4
Nomenclature	6
1 Introduction	7
1.1 Background and Motivation	7
1.2 Objectives and scope of work	8
1.2.1 <i>Methods</i>	8
1.2.2 <i>Objectives</i>	8
1.2.3 <i>Scope and limitations of work</i>	8
1.2.4 <i>Thesis overview</i>	9
2 Power system stability	10
2.1 The stability problem.....	10
2.2 Transient Stability	12
2.3 Classical Model with network nodes eliminated	14
2.3.1 <i>The classical model</i>	14
2.3.2 <i>Internal Node Model</i>	15
2.3.3 <i>Center of Inertia formulation</i>	16
2.3.4 <i>Equilibrium points for the center of inertia formulation</i>	17
3 Theoretical foundation of direct methods	18
3.1 Lyapunov Stability	18
3.2 Single Machine energy function.....	20
3.3 Multi Machine energy function	21
3.4 Review of Direct Methods	22
3.4.1 <i>Closest Unstable Equilibrium Point</i>	22
3.4.2 <i>Potential Energy Boundary Surface Method</i>	24
3.4.3 <i>Controlling Unstable Equilibrium Point</i>	25
4 Numerical Methods	28
4.1 Runge Kutta 4 th order method for swing equation	28
4.2 HAM analysis homotopy	29
4.3 Kron reduction	31
5 SMIB Test Case	33
5.1 Model.....	33
5.2 Exit Point	37
5.3 Critical Clearing time and critical clearing angle	39
6 Multimachine Test Case	41
6.1 Determining Initial values	41
6.2 Post-fault SEP	44
6.3 Exit Point	45
6.4 Controlling unstable equilibrium points.....	46
6.5 Critical clearing time.....	48
7 Discussion	49
8 Conclusion and future work.....	51

Contents

8.1 Conclusion	51
8.2 Future work	52
References	53
Appendices	55

Nomenclature

BCU	Boundary of stability region based Controlling Unstable equilibrium point
CCT	Critical Clearing Time
COI	Center Of Inertia
CUEP	Controlling Unstable Equilibrium Point
DAE	Differential Algebraic Equation
FACTS	Flexible AC Transmission System
HVDC	High Voltage Direct Current
PEBS	Potential Energy Boundary Surface
RK4	Runge Kutta 4 th order method
SEP	Stable Equilibrium Poin
SMIB	Single Machine Infinite Bus
UEP	Unstable Equilibrium Point
WSCC	Western System Coordinating Council

1 Introduction

1.1 Background and Motivation

Power interruptions and unstable operation are major challenges for power systems, with significant economic and societal costs. In the United States alone, power outages were estimated to cost around \$80 billion annually in 2004. Addressing these issues would require significant investment, with estimates ranging from \$50 to \$100 billion to modernize the grid[1].

These issues are not unique to the United States and have only grown more complex. In Norway there is an increasing demand for electricity coupled with ambitious climate goals, resulting in a rise in the deployment of renewable energy sources, such as wind and solar. Several wind projects are currently under consideration in Norway. As power generation moves further from consumers, it necessitates new grid investments, and from a socio-economic standpoint, there is a need to maximize the potential of the existing grid. This will likely require utility companies to push the current grid closer to its operational limits.

Operating the power grid to its operational limits necessitates a reevaluation of the criteria for setting the limits. One of the current limitations is related to transient stability considerations, which are typically imposed due to the conservative assumptions applied in off-line time domain simulations for transient stability analysis. To increase the operability of the grid, it is natural to consider conducting on-line transient stability analysis, which can provide a more accurate picture closer to reality, and thus removing some of the conservative assumptions. However, the time-consuming nature of the traditional approach is not suitable for on-line transient stability analysis of practical power systems[2].

The only viable option for an online alternative is direct methods, today direct methods are closely coupled with Lyapunov's stability theorem. The application of Lyapunov stability in various stability cases has been extensively studied since its introduction to the western world in the 1930's. For transient stability, Magnusson made an early contribution in 1947[3], although the methodology did not gain a lot of attention until the 1960s. In the early stages there were two fundamental issues, they were related to the characterization of the systems energy and the determination of its critical energy. In the early 1970s the center of inertia formulation was introduced which was crucial in developing the methodology as it facilitated much simpler ways to characterize the systems energy[4].

In recent years the technology has been tested in real grids with promising results[5]. The reason these approaches are less computationally taxing than the standard time domain approach is because they do not require a full solution of the differential equations, meaning it is not necessary to solve the differential equations for the full period to determine the stability of the system.

1.2 Objectives and scope of work

1.2.1 Methods

During the course of this thesis, a combination of methods was utilized. A literature review was conducted to explore the theoretical foundation for direct methods.

The test cases were simulated and plotted using the python programming language. Several modules in python will be utilized, namely Numpy, Scipy, Matplotlib and Sympy.

A comparative analysis was conducted, comparing the test cases to relevant literature.

1.2.2 Objectives

The primary aim of this thesis is to develop a functional approach to the Controlling Unstable Equilibrium Point (CUEP) method. To achieve this goal, the following objectives have been identified:

- Conduct a literature review on the theoretical foundation of direct methods, with special emphasis on CUEP methodology.
- Describe the mathematical techniques required for the implementation of direct methods.
- Perform a comparative analysis of the results obtained from a SMIB test case using direct methods against results obtained using the equal area criteria.
- Implement the CUEP methodology on the WSCC 3-generator 9-bus system and validate the results by comparing them to credible sources.
- Identify the primary challenges associated with the multimachine case and explore how further research can address these challenges by improving the methodology.

The initial iteration of these objectives can be found in Appendix A. Overall, this thesis aims to contribute to increasing the knowledge of direct methods at the institute for electrical power engineering at USN.

1.2.3 Scope and limitations of work

The multimachine test case is limited to analyzing an ideal fault with zero fault resistance. This might not reflect the actual behavior of faults in real-world scenarios. However, this approach was chosen because the main objective is to test the methodology, including the fault resistance would somewhat complicate the calculations and thus not be appropriate for initial testing.

All test cases presented in this thesis were conducted on power grids operating at a frequency of 60Hz, as this frequency is commonly used in related research papers for comparison and validation. It is important to note that using a 50Hz power grid would not alter the results or conclusions of this thesis in any notable way. In fact, one could argue that a 50Hz power grid is more favorable for transient stability, as machines with the same H constant would

effectively have a higher inertia constant. Nonetheless, the choice of frequency was based on the prevailing convention in literature, and the findings are applicable to both 50 and 60Hz power grids.

For the multimachine transient stability analysis, we will employ the center of inertia formulation as widely adopted in literature. Developing energy functions for a model using relative rotor angles will be beyond the scope of this thesis.

1.2.4 Thesis overview

Chapter 2

Chapter 2 deals with the power system stability problem and the model used later in the thesis.

Chapter 3

Chapter 3 deals with the theoretical foundation of direct methods, with special emphasis on the CUEP methodology.

Chapter 4

Chapter 4 covers the numerical methods employed in the simulation of the test cases.

Chapter 5

Chapter 5 covers the SMIB test case.

Chapter 6

Chapter 6 covers the multimachine test case employing the CUEP methodology.

Chapter 7

Chapter 7 discusses the findings in the report and possible improvements.

Chapter 8

Chapter 8 concludes with the results found in the thesis and gives recommendations for future work.

2 Power system stability

This chapter provides an overview of power system stability with a focus on transient stability and a reduced order model based on the classical model eliminated to the internal nodes of the generator. Each topic is covered in a separate subchapter.

2.1 The stability problem

Power systems are continuously subjected to disturbances, load variations, faults etc. Power system stability may broadly be defined as the property that enables it to be in a state of operating equilibrium under normal operating conditions and after a disturbance[6].

The power system is a dynamic system, consisting of both passive and active components. When a disturbance occurs, the system responds dynamically by shifting its operating point. The classification of a disturbance depends on the type and severity of the disturbance. Generally power system dynamics are analyzed over three different timeframes: short, mid and long term.

Another possible way to classify power system dynamics is to consider the different physical phenomena it can represent and assign a corresponding time scale. This approach may be more appropriate because a disturbance can represent several types of phenomena. The physical phenomena that a power system can be subjected to include wave phenomena, electromagnetic phenomena, electromechanical phenomena, and thermodynamic phenomena. A classification of power system dynamics based on the different phenomena can be seen in Figure 2.1.

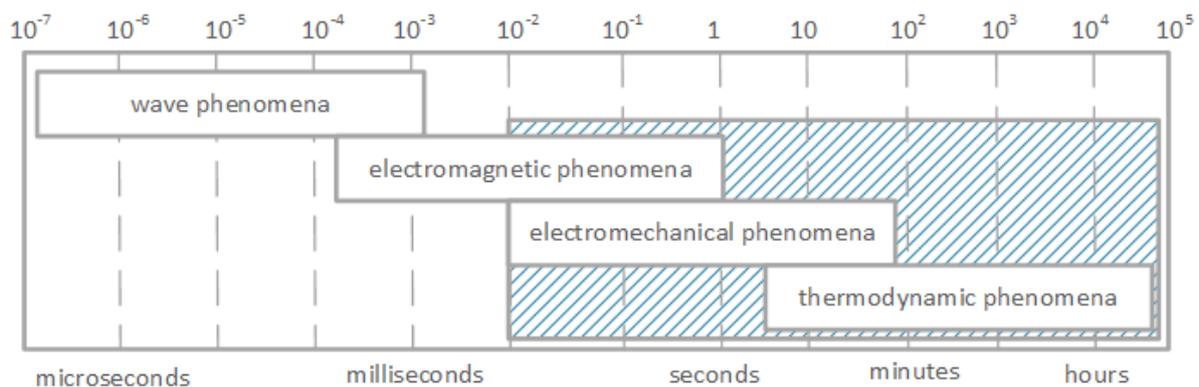


Figure 2.1: Time frames of power system dynamics phenomena [7]

Traditionally power system stability has been characterized as shown in Figure 2.2 which was formalized by IEEE-CIGRE in 2004[8].

Power system stability

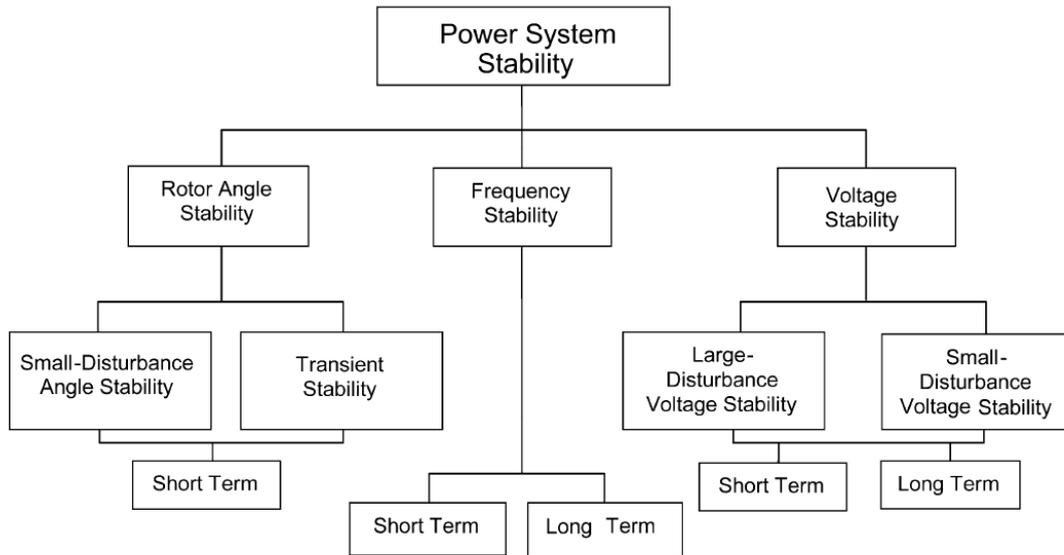


Figure 2.2: Classification of power system stability[8]

The classical definition of power system stability primarily focused on the electromechanical and thermodynamic phenomena. However, with the rapid penetration of power electronic converter interfaced technologies such as wind and photovoltaic generation, as well as HVDC and FACTS, the power system has evolved significantly since the classical definition was formalized. As a result, a new flowchart describing power system stability can be seen in Figure 2.3.

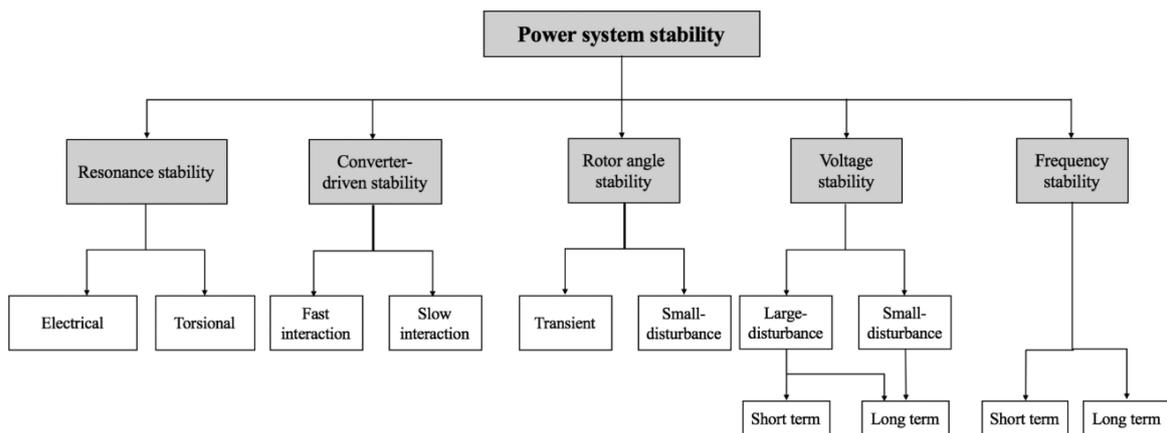


Figure 2.3: Classification of power system stability[9]

It is useful to include the new definition of stability in terms of a timescale, which can be seen in Figure 2.4.

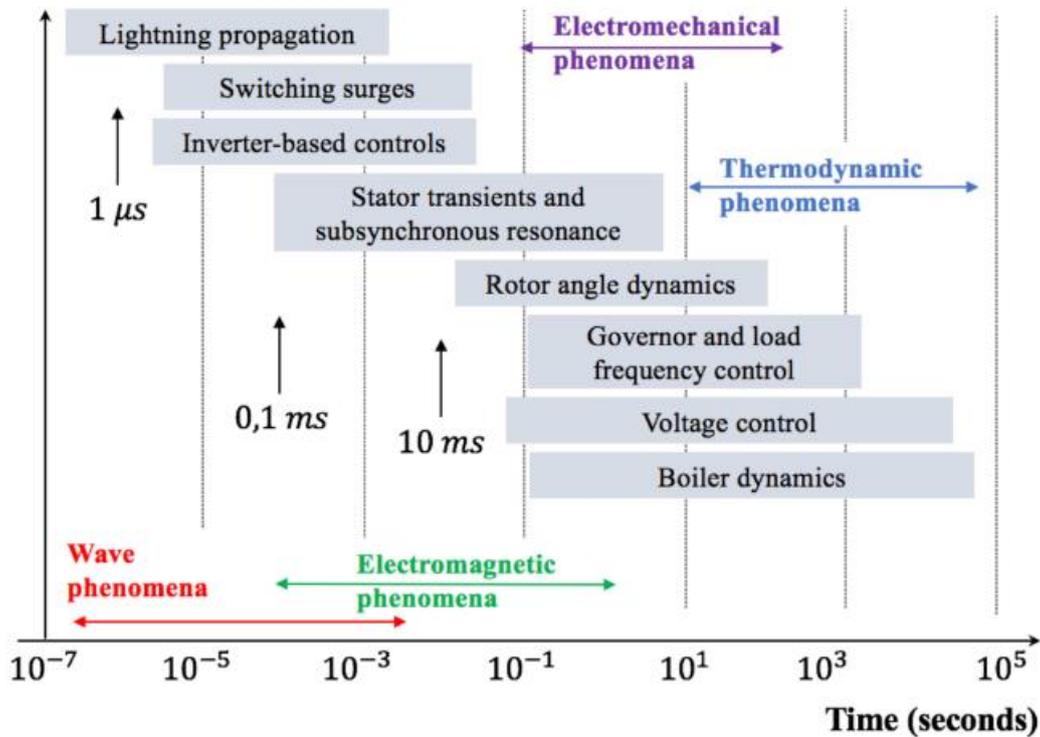


Figure 2.4: Time frame of power system dynamics[9]

This thesis will predominantly concentrate on transient stability, which is a part of rotor angle dynamics. It mainly pertains to the electromechanical phenomena of power systems. However, it is essential to comprehend the prospective influence of emerging technologies on transient stability analysis. Even though the definition of transient stability has remained unchanged, the spread of such technologies has resulted in a decrease in the grid's inertia. Consequently, there is an escalating requirement for precise transient stability analysis, particularly online transient stability analysis, since the traditional off-line transient stability analysis is based on a set of preconceived assumptions. Attaining a set of accurate assumptions will likely become more difficult, with the increasing penetration of photovoltaic and wind generation.

2.2 Transient Stability

Transient stability analysis is a vital aspect of power system analysis, as it helps understanding how power systems behave in response to large disturbances, such as short circuits. These disturbances can cause sudden changes in the operating conditions in the power system and trigger a dynamic response from the system.

Consider a power system operating at a stable equilibrium point prior to a fault. This means that $P_m = P_e$, in this case there is no acceleration or deceleration of the machines. Suddenly a three-phase short circuit occurs, this will change the network topology, hence $P_m \neq P_e$. The machines throughout the system will experience acceleration or deceleration. The result is

that the rotor angle will change with respect to the synchronously moving reference axis. If one or several of the machines accelerates beyond a critical point, they will not be able to return to a stable equilibrium point after the fault. This can lead to loss of synchronism, cascading failures and possibly a blackout. Transient stability analysis deals with this phenomenon, analyzing the implications of different faults and designing the system in such a way that it will be able to safely operate in the event of a fault. This can be determining critical clearing times, available transfer capacity, remedial actions etc. To illustrate the point in Figure 2.5 the rotor angle is plotted for three different cases.

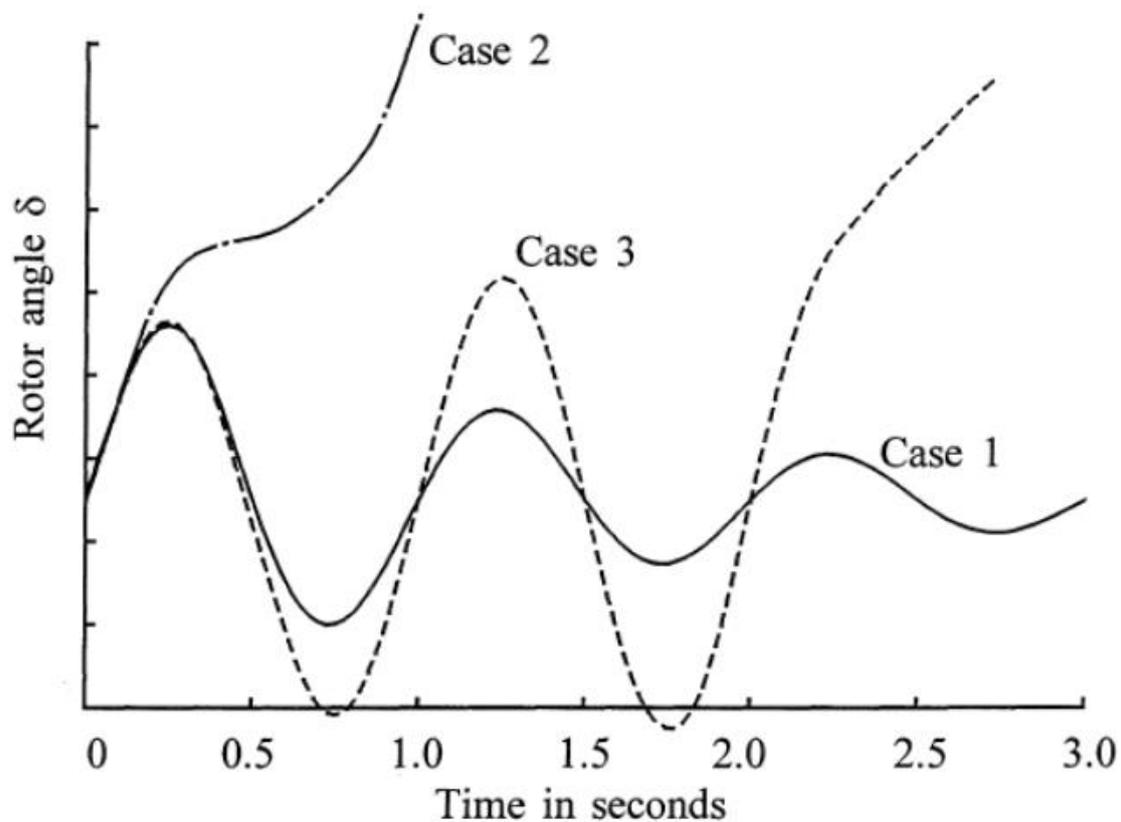


Figure 2.5: 3 cases of transient stability analysis [10]

Where:

- Case 1 is stable.
- Case 2 is first swing unstable.
- Case 3 is also unstable.

Transient stability analysis mostly deals with case 1 or first swing stability. In these situations, it is common to use simpler generator models with a constant voltage behind a transient reactance[11].

The mathematical model for describing a dynamical response in power systems is given by a set of first order differential equations, written in compact form in Equation (2.1).

$$\dot{x} = f(x, y) \quad (2.1)$$

When the power system is subjected to a disturbance, the disturbance can mathematically be described as shown in Equation (2.2).

$$\begin{aligned} \dot{x}(t) &= f^{Pre}(x(t)) \quad -\infty < t \leq 0 \\ \dot{x}(t) &= f^F(x(t)) \quad 0 < t \leq t_{cl} \\ \dot{x}(t) &= f^{post}(x(t)) \quad t_{cl} < t < \infty \end{aligned} \quad (2.2)$$

Prior to time $t=0$, the system is operating in the pre-fault state, denoted “Pre”. At time $t=0$ the system experiences a disturbance, the faulted period denoted “F” lasts until the disturbance is cleared, $0 \leq t \leq t_{cl}$, often referred to as the fault-on period. After the fault is cleared, the system enters the post-fault period denoted “Post”. The objective of transient stability analysis is to determine whether the post fault system will settle down to a stable equilibrium point (SEP), or if the system will spin out of control and possibly experience cascading failures and loss of synchronism.

2.3 Classical Model with network nodes eliminated

This subchapter will introduce a reduced order model commonly used in direct methods of transient stability analysis, specifically the classical model with eliminated network nodes. We will provide an overview of the classical model, followed by a description of the internal node model with both relative rotor angles and a center of inertia formulation. In addition, we will discuss the computation of equilibrium points for the center of inertia representation. Each topic will be presented in separate sections.

2.3.1 The classical model

For this thesis a classical model will be used when solving the differential equations. The classical model can be summarized as follows[11]:

1. Mechanical power remains constant during the swing curve computation.
2. Damping is neglected.
3. The generators can be represented by a constant transient reactance in series with a constant internal voltage.
4. The mechanical rotor angle equals the electrical phase angle of the transient internal voltage.
5. All loads are considered as constant during the computation of the swing equation.

The classical model is said to be adequate for studying first-swing stability, generally considered the first second of the faulted period [12]. Some key limitations of the classical model should be noted:

1. Saying that mechanical power remains constant during the computation of the swing curve is unlikely to be true if we consider longer periods of time.
2. Reduction of the system to internal nodes removes a lot of the system topology and makes it impossible to study how other components in the grid are affected.
3. The worst swing might occur after the first swing.

2.3.2 Internal Node Model

The swing equation for the internal node model with negligible damping can be written as shown in Equation (2.3)[12].

$$\begin{aligned} \frac{d\delta_i}{dt} &= \omega_i \\ M_i \frac{d\omega_i}{dt} &= P_{mi} - P_{ei} \end{aligned} \quad (2.3)$$

The electrical power is given by Equation (2.4).

$$P_{ei} = E_i^2 G_{ii} + \sum_{j=1, j \neq i}^n C_{ij} \sin \delta_{ij} + D_{ij} \cos \delta_{ij} \quad (2.4)$$

Replacing the sum of the constant parameters in Equation (2.3) to a new parameter P_i yields the following set of first order differential equations shown in Equation (2.5).

$$\begin{aligned} \frac{d\delta_i}{dt} &= \omega_i \\ M_i \frac{d\omega_i}{dt} &= P_i - P_{ei} \end{aligned} \quad (2.5)$$

Where:

- $P_i = P_{mi} - E_i^2 G_{ii}$
- $C_{ij} = E_i E_j B_{ij}$
- $D_{ij} = E_i E_j G_{ij}$
- $\delta_{ij} = \delta_i - \delta_j$
- δ_i, ω_i rotor angle and angular velocity with respect to a synchronously moving reference frame.
- M_i the rotor's inertia constant

- G_{ij}, B_{ij} transfer conductance and transfer susceptance in the reduced admittance matrix.
- E_i magnitude of the constant voltage behind the constant transient reactance

According to the classical model only the electrical power of the machines will be changing, this means that from pre-fault, fault-on and post-fault the only thing that changes is the transfer conductance and transfer susceptance in the reduced admittance matrix.

2.3.3 Center of Inertia formulation

The center of inertia formulation is a vital tool for direct methods of transient stability analysis. This is because, unlike the relative rotor angles where each angle is moving with respect to a different synchronously moving reference frame, the center of inertia formulation captures the mean motion of the system. Another benefit of the center of inertia formulation is that it removes the energy associated with the motion of the internal center, which does not contribute to stability deformation[4]. The swing equation according to the COI framework with damping neglected is defined as depicted in Equation (2.6).

$$\begin{aligned} \frac{d\tilde{\delta}_i}{dt} &= \tilde{\omega}_i \\ M_i \frac{d\tilde{\omega}_i}{dt} &= P_i - P_{ei} - \frac{M_i}{M_T} P_{COI} \end{aligned} \quad (2.6)$$

In order to express the angular velocity and rotor angles in the COI-framework it is necessary to initialize them in accordance with Equation (2.7).

$$\begin{aligned} \tilde{\delta}_i &= \delta_i - \delta_0 \\ \tilde{\omega}_i &= \omega - \omega_0 \\ \delta_0 &= \frac{1}{M_T} \sum_{i=1}^n M_i \delta_i \\ \omega_0 &= \frac{1}{M_T} \sum_{i=1}^n M_i \omega_i \end{aligned} \quad (2.7)$$

The constant M_T can be calculated as shown in Equation (2.8).

$$M_T = \sum_{i=1}^n M_i \quad (2.8)$$

The P_{COI} parameter is defined as shown in Equation (2.9).

$$P_{COI} = \sum_{i=1}^n P_i - 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^n D_{ij} \cos \tilde{\delta}_{ij} \quad (2.9)$$

2.3.4 Equilibrium points for the center of inertia formulation

The power system can be mathematically described as a set of DAE's. Like shown in Equation (2.10)[13].

$$\begin{aligned} \dot{x} &= f(x, y) \\ 0 &= g(x, y) \end{aligned} \quad (2.10)$$

The reduced network model in accordance with the classical model is when the y-matrix of the system is reduced to the internal nodes of the generator. This model simplifies the problem further because the algebraic equations disappear as depicted in Equation (2.11).

$$\dot{x} = f(x, y) \quad (2.11)$$

An intuitive understanding of an equilibrium point is to think about the swing equation and say at an equilibrium point $\tilde{\omega} = 0$. This means there is no acceleration. Solving for an equilibrium point is therefore equivalent to saying that the derivative is zero represented by Equation (2.12).

$$0 = f(x, y) \quad (2.12)$$

This gives the system of nonlinear equations depicted in equation (2.13). This equation is the one that needs to be solved for all equilibrium points utilizing this model.

$$f_i = P_i - P_{ei} - \frac{M_i}{M_T} P_{COI} \quad (2.13)$$

It should be noted that due to the center of inertia formulation where $\sum M_i \tilde{\delta}_i = 0$ the number of equations can be reduced, because one of the rotor angles can be expressed as a function of the other rotor angles and inertia constants. It will prove challenging to not reduce the number of equations, because if one wants to find an equilibrium point normally this involves numerical methods, which in turn usually requires the Jacobian matrix. Taking the inverse of a Jacobian if not all the variables are unique is not possible using traditional approaches. Hence, not performing this reduction could make it difficult to solve the system of nonlinear equations correctly.

3 Theoretical foundation of direct methods

This chapter presents the theoretical basis for direct methods used in transient stability analysis. It begins with an introduction to the concept of Lyapunov stability, followed by an overview of the most common energy functions. The chapter then proceeds to discuss the most widely employed direct method with a special emphasis on the CUEP method.

3.1 Lyapunov Stability

The concept of Lyapunov stability, proposed by Aleksandr Lyapunov in his 1892 Ph.D. dissertation, has become a fundamental tool in control theory. It allows the stability of an equilibrium point in a nonlinear dynamic system to be determined without the need of numerical integration, making it a valuable technique in numerous control applications. In this subchapter, we will delve into the basic principles of Lyapunov stability with an emphasis on the parts that pertain to direct methods of transient stability.

To gain an intuitive understanding of Lyapunov stability, consider an example of a football at the bottom of a hill. When the ball is kicked, it can either fail to reach the top of the hill and roll back down, or it can successfully cross over the top and not return to the bottom. One approach to analyzing this behavior is to determine the initial conditions and integrate the differential equations that describe the motion of the ball and see if it crosses the top. This would then be the equivalent of a standard time domain simulation for power systems. However, an alternative approach is to employ a Lyapunov function that satisfies certain criteria that will be addressed later and use the Lyapunov function to determine the stability of the system.

In this context, an intuitive thought for the Lyapunov function would be the system energy. Then one could calculate the energy subjected to the ball and compare it to the energy required to reach the top of the hill. One could then determine the stability of the system without tracking the position as the ball moves up the hill[4].

Consider the nonlinear dynamical system presented in Equation (3.1).

$$\dot{x} = f(x) \tag{3.1}$$

If a scalar function $V(x)$ can be constructed so that it satisfies the mathematical conditions depicted in Equation (3.2) where \hat{x} is an equilibrium point. Then it can be called a Lyapunov function.

$$\begin{aligned}
 V(\hat{x}) &= 0 \\
 V(x) &> 0 \text{ as long as } x \neq \hat{x} \\
 \dot{V}(x) &< 0 \text{ as long as } x \neq \hat{x}
 \end{aligned} \tag{3.2}$$

There are more details to Lyapunov stability but for the purposes of this thesis, it will suffice. From this it can be deduced that if the x -value is put into the Lyapunov function, and the function satisfies the conditions for a Lyapunov function it will converge to an equilibrium point. However, for the purposes of direct methods of transient stability we are interested in what value the scalar function $V(x)$ has at the stability boundary of the system.

For direct methods of transient stability analysis, the Lyapunov functions are constructed as transient energy functions as shown in Equation (3.3).

$$V(\omega, \delta) = V_{KE} + V_{PE} \tag{3.3}$$

To explain this concept in a practical manner, consider Figure 3.1. Where the ball in this case is the power system. The ball starts out operating in at a stable equilibrium point. Meaning that there is no acceleration $P_m = P_e$. When a disturbance occurs, kinetic energy will be injected. It will roll along the surface of the bowl. The trajectory or direction the ball will be moving will depend upon the direction in which the disturbance is applied. If the kinetic energy subjected into the ball is large enough, so that it crosses over the rim, we can no longer guarantee that the ball will return to the SEP and therefore it is considered unstable. As the ball moves, it will acquire potential energy, as the height of the ball with respect to the stable equilibrium point increases. If one pays close attention to Figure 3.1 one will see that the rim of the bowl is not uniform. This means that the kinetic energy the system can absorb will depend greatly on the direction of the ball. Direct methods use different techniques to assess the stability of the system but essentially, they all try and find the potential energy at the top of the rim. Because if the total energy of the system during a fault does not exceed the potential energy at the top of the rim it will return to a stable equilibrium point.

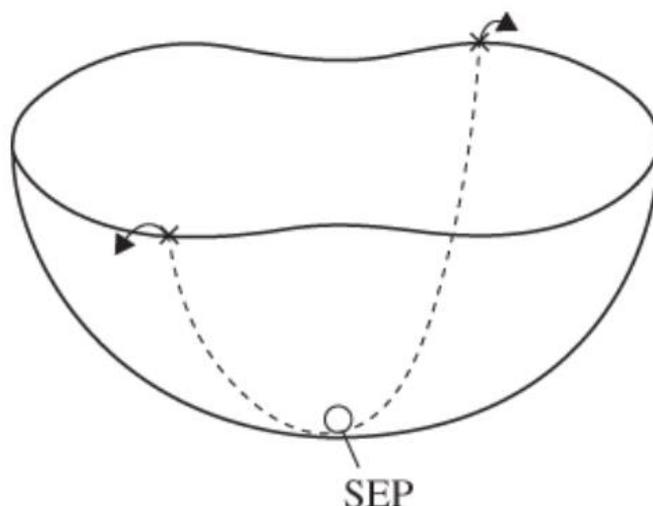


Figure 3.1: Rolling Ball Analogy[14]

While critical energy is important, its value alone is not particularly useful. Instead, critical energy is typically used to calculate the critical clearing time (CCT), which represents the time a fault can persist in the power system before the will be unable to converge to a post-fault SEP. The criterion for determining the CCT involves checking the inequality $V(x) \leq V_{cr}$, the time when the systems energy equals the critical energy is the CCT.

The basic steps for computing the CCT can be expressed as [12]:

1. Construct an energy/Lyapunov function $V(\omega, \delta)$ for the postfault system.
2. Find the critical value of $V(\omega, \delta)$ for a given post-fault system denoted V_{cr} .
3. Integrate the swing equation for the fault-on period until $V(\omega, \delta) = V_{cr}$.

This approach is similar for all direct methods. Where they differ however, is in steps 2 and 3.

3.2 Single Machine energy function

The energy function is always constructed for the post fault system. In Equation (3.4) the energy function for a single machine system is shown[12].

$$V(\delta, \omega) = \frac{1}{2}M\omega^2 + V_{PE}(\delta) \quad (3.4)$$

Where the potential energy is given by Equation (3.5).

$$V_{PE} = -P_m \delta - P_e^{max} \cos \delta \quad (3.5)$$

It is necessary to change the energy function so that the potential energy is zero at the post fault stable equilibrium point. The energy function is then rewritten as shown in Equation (3.6).

$$V(\delta, \omega) = \frac{1}{2} M \omega^2 - P_m (\delta - \delta_s) - P_e^{max} (\cos \delta - \cos \delta^s) \quad (3.6)$$

3.3 Multi Machine energy function

The most used multimachine energy function is depicted in Equation (3.7).

$$V = \frac{1}{2} \sum_{i=1}^n M_i \tilde{\omega}_i^2 - \sum_{i=1}^n P_i (\tilde{\delta}_i - \tilde{\delta}_i^s) - \sum_{i=1}^{n-1} \sum_{j=i+1}^n [C_{ij} (\cos \tilde{\delta}_{ij} - \cos \tilde{\delta}_{ij}^s) - I_{ij}] \quad (3.7)$$

The term I_{ij} is path dependent integral term and we can therefore not say for certain that V is positive-definite. If $D_{ij} \equiv 0$ it can be shown that the energy function is a true Lyapunov function. There are several approximations to the term I_{ij} but one that is often utilized is depicted in Equation (3.8)[12] [5].

$$I_{ij} = D_{ij} \frac{\tilde{\delta}_i + \tilde{\delta}_j - \tilde{\delta}_i^s - \tilde{\delta}_j^s}{\tilde{\delta}_i - \tilde{\delta}_j - \tilde{\delta}_i^s + \tilde{\delta}_j^s} [\sin \tilde{\delta}_{ij} - \sin \tilde{\delta}_{ij}^s] \quad (3.8)$$

Where:

- S denotes the post fault stable equilibrium point.
- $P_i = P_{mi} - E_i^2 G_{ii}$
- $C_{ij} = E_i E_j B_{ij}$
- $\tilde{\delta}_{ij} = \tilde{\delta}_i - \tilde{\delta}_j$
- I_{ij} is the energy dissipated in the network conductances.
- $D_{ij} = E_i E_j G_{ij}$

3.4 Review of Direct Methods

In this subchapter, the working principle of the three most prominent direct methods will be introduced, each in their respective sections. Special emphasis will be placed on the Controlling Unstable Equilibrium Point (CUEP) method since it is the main focus of this thesis.

3.4.1 Closest Unstable Equilibrium Point

The method known as the closest unstable equilibrium point method was first developed in the 1960s also referred to as the lowest energy UEP method. Originally the aim of the method was to find the exact stability boundary, which was deemed a too challenging task for practical power systems. However, by the 1970s researchers shifted their focus to finding a suitable estimate of the stability boundary, which is less computationally taxing[15].

To conceptually illustrate this method, one can consider the rim of the bowl depicted in Figure 3.2. The entire rim represents the stability boundary of the postfault system. The closest UEP method aims to find the lowest point along the rim. The potential energy corresponding this point is V_{cr} . The drawback of this method is that it requires the computation of many UEP's, making it computationally demanding. Moreover, the method often yields to conservative results compared to other direct methods. This is because the trajectory of the fault might be heading in another direction than where the lowest point is located, it can therefore tolerate more energy injected into the system before it is getting close to the stability boundary[12].

To illustrate the concept, consider the SMIB system. The system dynamics are represented by the swing equation represented as two first order differential equations as shown in Equation (3.9). For the purposes of this example damping is negligible.

$$\begin{aligned} \frac{d\delta}{dt} &= \omega \\ M \frac{d\omega}{dt} &= P_m - P_e^{max} \sin\delta \end{aligned} \quad (3.9)$$

The post-fault stable equilibrium point can be found when $\omega = 0$, and is shown in Equation (3.10).

$$\delta_s = \sin^{-1} \left(\frac{P_m}{P_e^{max}} \right) \quad (3.10)$$

The two UEP's of this system is shown in Equation (3.11).

$$\begin{aligned}\delta_1 &= \pi - \sin^{-1}\left(\frac{P_m}{P_e^{max}}\right) \\ \delta_2 &= -\pi - \sin^{-1}\left(\frac{P_m}{P_e^{max}}\right)\end{aligned}\tag{3.11}$$

In Figure 3.2 we can see the stability boundary of the system denoted $\partial A(\delta_s, 0)$, encloses a region $A(\delta_s, 0)$ which is the stability region of the system. The stability region is the union of the stable manifolds of the UEP denoted $W^s(\delta_1, 0)$ and $W^s(\delta_2, 0)$ [2].

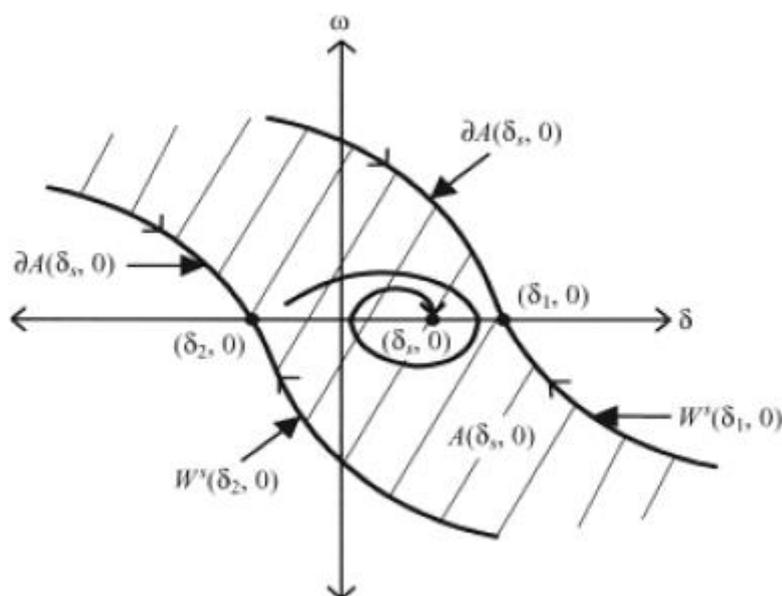


Figure 3.2: SEP enclosed by stability boundary[2]

To conceptually understand what a stable manifold is, it is a set of all the points that converge to the same “location”. For the case of the stable manifold of the UEP’s, it would be all the points in the phase space that converges to the specific UEP.

In the closest UEP method we now proceed to check the energy function given in Equation (3.6), for both UEP’s. The one that yields the lowest value will be the critical energy of the system. In Figure 3.3 the estimated stability region is depicted as the shaded area.

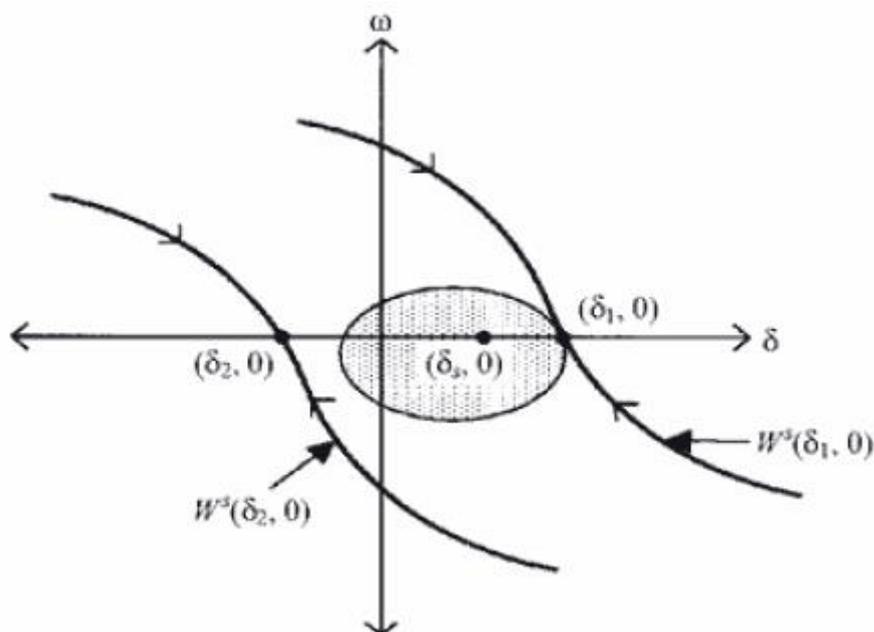


Figure 3.3: Constant energy surface closest UEP method[2]

To summarize the method could be used to find the critical clearing time in the following way:

1. Find all the UEP's of the system.
2. Arrange them in order after their respective energy values.
3. The UEP with the lowest energy value is the closest UEP.
4. The critical energy V_{cr} would then be the potential energy at the closest UEP.
5. Integrate the fault-on trajectory until it reaches V_{cr} the corresponding time is the CCT.

3.4.2 Potential Energy Boundary Surface Method

The Potential Energy Boundary Surface Method (PEBS) was first proposed by Kakimoto et. al and Athay et.al. It has received a lot of attention for its natural relationship with the equal area criteria[12].

To conceptualize the method, the rim of the bowl presented in Figure 3.1 is the potential energy boundary surface. The PEBS method is simple in the sense that it does not require solving the system of nonlinear equations to obtain an equilibrium point. The methodology can be summarized in the following steps:

1. Integrate the fault on trajectory until it crosses the “rim” of the bowl.
2. The exit point or where it crosses the PEBS is used to calculate V_{cr} .
3. The CCT is found by integrating the fault-on trajectory until the sum of the kinetic and potential energy equals V_{cr} .

3.4.3 Controlling Unstable Equilibrium Point

The controlling UEP method, also called the exit point method, was developed in the 1980s. It aims to reduce the conservativeness of the closest UEP method by taking the fault trajectory into account[2].

The method can in one way be said to be a combination of both PEBS and the closest UEP method. The steps to find the CCT using the CUEP method is listed as:

1. Integrate the fault on-trajectory until it crosses the PEBS, this point is called the exit point.
2. Use the exit point as the initial condition and solve for the CUEP.
3. The potential energy at the CUEP is V_{cr} .
4. Integrate the fault on trajectory until it reaches V_{cr} . The corresponding time step is the CCT.

In the context of conservatism, the CUEP method is positioned between the closest UEP method and the PEBS method. The difference between the PEBS and CUEP methods lies in the fact that the exit point is not necessarily an unstable equilibrium point, as discussed previously, the stability boundary of a system is defined as the union of the stable manifolds of the UEP's. This means there is a possibility that the PEBS method may have crossed the stability boundary at the exit point, leading to an unsatisfactory result in terms of critical energy.

For many years, the computation of the CUEP's was a formidable barrier in utilizing the methodology. The system of nonlinear equations is highly nonlinear and can be characterized by their fractal shapes, making many numerical methods unable to converge to a desired solution. The CUEP's also lie on the stability boundary embedded with other UEP's making it difficult to distinguish the CUEP from other UEP's. Finding an initial guess that lies within the convergence region of the CUEP can be challenging [8] [16]. Several methodologies have in recent years made computing the CUEPS easier, the main methods are the BCU methodology and continuation-based approaches. In this thesis, a continuation approach rooted in Homotopy analysis, as proposed by Joydeep Mitra and his colleagues is posed as a viable solution [5].

The algorithm for computing the critical clearing time in this thesis is shown in Figure 3.4.

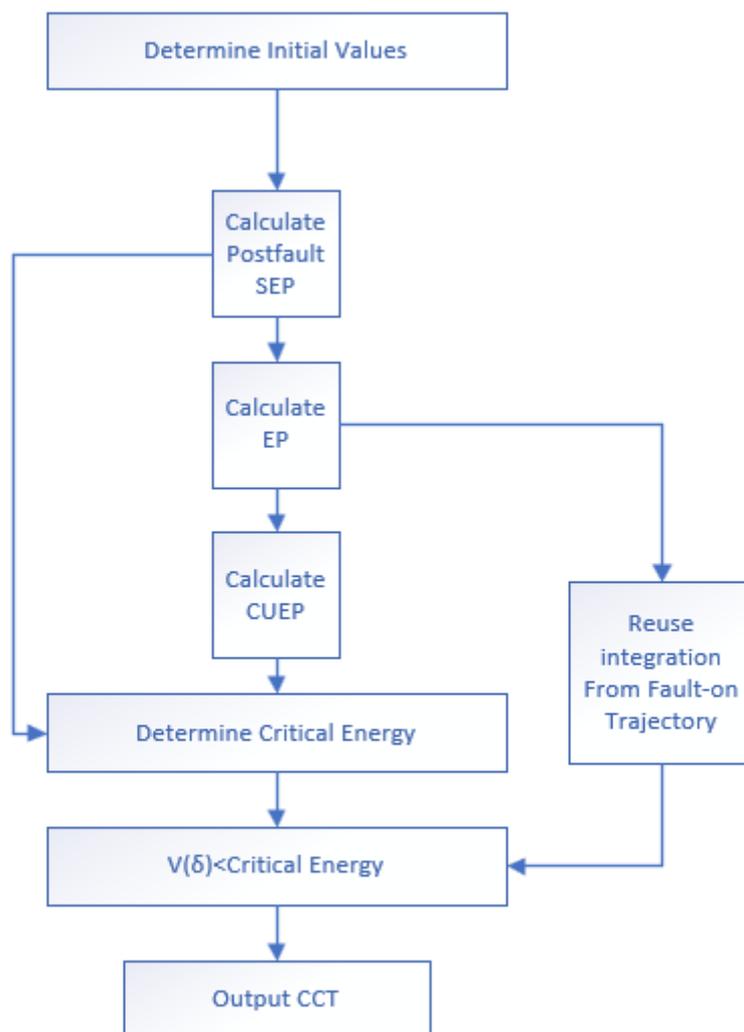


Figure 3.4: Proposed Algorithm for computing CUEP

The first step of the algorithm is to determine the initial conditions of the system. As described in chapter 2 an internal node model using the classical representation will be utilized in computation of the swing curves and the computation of the equilibrium points. The calculations that need to be performed in step 1 can be summarized as follows:

1. From the pre-fault load-flow determine the constant voltage behind the transient reactance.
2. Calculate the initial rotor angle of the machines.
3. Convert the loads to shunt impedances and add them to the corresponding diagonal of the y-matrix.
4. Add the transient reactance to the y-matrix.
5. For the faulted condition assume an ideal fault and remove the corresponding row and column of the y-matrix.
6. For the post-fault system remove the line that is cleared and compute a new y-matrix.
7. Reduce the fault-on and post-fault matrix to an internal node model.

Theoretical foundation of direct methods

The second step is calculating the post-fault SEP. This can be done solving Equation (2.13) for the postfault system. The initial guess should be the pre-fault SEP, using Newtons method will be a good enough method to determine the post-fault SEP.

The third step is calculating the exit point. The exit point can be calculated by integrating the differential equations shown in Equation (2.6). The reduced fault-on matrix acquired in step one is utilized when integrating the equations. The only thing that changes for the different states is the y-matrix, everything else remains constant. The integration continues until the first maximum of the potential energy function is reached.

The fourth step is calculating the CUEP. The post-fault reduced matrix acquired in step one is used to solve the nonlinear system of equations given in Equation (2.13). The exit point is used as the initial condition, the numerical method utilized is a homotopy method that is covered in chapter 4, the mapping direction and practical details of the computation is covered in chapter 6.

The fifth step is calculating the critical energy of the system. The critical energy is found by inputting the CUEP and post-fault SEP into the energy function given in Equation (3.7). Since the CUEP is located on the stability boundary, $\omega = 0$ and the kinetic energy term disappears.

The sixth step is determining the CCT. The integration from step three can be reused and the timestep when $V(\delta, \omega) = V_{cr}$ is the critical clearing time.

4 Numerical Methods

In this chapter, the numerical methods utilized in this thesis are covered. Kron Reduction for reducing matrices, Homotopy analysis for solving nonlinear equations, and the reliable 4th order Runge Kutta method for numerical integration.

4.1 Runge Kutta 4th order method for swing equation

Solving the swing equations in this thesis will be done using, Runge Kutta 4th order method or RK4. It is considered to be the most widely used solver for ordinary differential equations [17].

Solving the swing equation requires the equation to be rewritten as a set of first-order differential equations. The swing equation is solved for both a single machine and several machines, with the same principle applied in both cases. In the case of multiple machines, the equations are iterated per timestep. For the purposes of developing a computational scheme, the solution methodology is presented using a single-machine example. In Equation (4.1) the swing equation is rewritten as a set of two first order differential equations.

$$\begin{aligned}\frac{d\delta}{dt} &= \omega - \omega_s \\ \frac{d\omega}{dt} &= \frac{\pi f}{H} (P_m - P_e)\end{aligned}\tag{4.1}$$

Damping is neglected, P_m is the mechanical power input to the machine and P_e is the electrical power. The RK4 method uses four estimations where h denotes the integration timestep. The first estimation is shown in Equation (4.2)[18].

$$\begin{aligned}K1_\delta &= h(\omega - \omega_s) \\ K1_\omega &= h\left(\frac{\pi f}{H} (P_m - P_e)\right)\end{aligned}\tag{4.2}$$

The second approximation is given by Equation (4.3).

$$\begin{aligned}K2_\delta &= h\left(\omega + \frac{K1_\omega}{2} - \omega_s\right) \\ K2_\omega &= h\left(\frac{\pi f}{H} (P_m - P_e)\right)\end{aligned}\tag{4.3}$$

The third approximation is given by Equation (4.4).

$$\begin{aligned}
 K3_{\delta} &= h \left(\omega + \frac{K2_{\omega}}{2} - \omega_s \right) \\
 K3_{\omega} &= h \left(\frac{\pi f}{H} (P_m - P_e) \right)
 \end{aligned}
 \tag{4.4}$$

The fourth approximation is given by Equation (4.5).

$$\begin{aligned}
 K4_{\delta} &= h(\omega + K3_{\omega} - \omega_s) \\
 K4_{\omega} &= h \left(\frac{\pi f}{H} (P_m - P_e) \right)
 \end{aligned}
 \tag{4.5}$$

These approximations are used to calculate the angular velocity and the rotor angle for the next timestep as shown in Equation (4.6).

$$\begin{aligned}
 \omega_{n+1} &= \omega_n + \frac{1}{6} \cdot (K1_{\omega} + 2K2_{\omega} + 2K3_{\omega} + K4_{\omega}) \\
 \delta_{n+1} &= \delta_n + \frac{1}{6} \cdot (K1_{\delta} + 2K2_{\delta} + 3K3_{\delta} + K4_{\delta})
 \end{aligned}
 \tag{4.6}$$

4.2 HAM analysis homotopy

This thesis employs the homotopy analysis method to identify the controlling unstable equilibrium points. While several homotopy methods exist, in this thesis Newton homotopy will be employed. This section will provide an overview of the fundamental theory and principles associated with this method.

The homotopy analysis method was first proposed by Shijun Liao in 1992. The concept describes a continuous variation. To take an example a circle can be continuously deformed into a square. If a continuous deformation from one object to another is possible the objects are said to be homotopic[19].

In Equation (4.7) the most widely used homotopy function is presented [16].

$$\mathcal{H}(x, t) = tF(x) + (1 - t)G(x) = 0
 \tag{4.7}$$

Where t is the mapping factor and is increased from 0-1. Which in turn will transform the homotopy function to the solution. In Equation (4.8) the function $G(x)$ is chosen. This is called Newton homotopy. This eliminates the process of finding a suitable function.

$$\mathcal{G}(x) = F(x) - F(x^0) \quad (4.8)$$

This can be simplified as shown in Equation (4.9).

$$\mathcal{H}(x, t) = F(x) - (1 - t)F(x^0) = 0 \quad (4.9)$$

The forward mapping approach presented in the example is a useful tool in root finding. However, for certain problems multiple roots may exist, and it may become necessary to search for roots in the opposite direction. This is where Newton homotopy proves advantageous, as it allows for direction of the path to be modified. To change the direction, a slight adjustment in the formula is required, as demonstrated in Equation (4.10). This flexibility of the Newton homotopy method makes it a powerful tool for solving complex problems with multiple roots.

$$\mathcal{H}(x, t) = tF(x) + (1 - t)\mathcal{G}(x) = 0 \quad (4.10)$$

The reason homotopy is used rather than a standard Newton Raphson (NR) approach is because of the region of convergence. If the initial guess to NR does not lie within the region of convergence of the solution it will not be able to converge to a solution. Newton homotopy overcomes this problem because it is globally convergent. This means that it will converge to a solution provided there is no turning point, bifurcation point or singularities in the solution trajectory[16].

An illustration of the operational mechanism of Newton-Homotopy shall be presented using a straightforward example. The numerical method used to solve each homotopy-iteration is Newton Raphson method, the single variable formula is shown in Equation (4.11) and the multi variable formula is shown in Equation (4.12). The system of equations is presented in Equation (4.13).

$$x^{k+1} = x^k - \frac{g(x^k)}{g'(x^k)} \quad (4.11)$$

$$x^{k+1} = x^k - j^{-1}(x^k) \cdot g(x^k) \quad (4.12)$$

$$f(x) = \begin{matrix} x_1 + 2x_2 - 2 \\ x_1^2 - 4x_2^2 - 4 \end{matrix} \quad (4.13)$$

The jacobian matrix is computed and shown in Equation (4.14).

$$j = \begin{matrix} 1 & 2 \\ 2x_1 & 8x_2 \end{matrix} \quad (4.14)$$

The initial guess is presented in Equation (4.15).

$$x^{(0)} = \begin{matrix} 1 \\ 2 \end{matrix} \quad (4.15)$$

Utilizing a standard Newton-Raphson method poses no significant challenge in computing this example. However, to showcase the working principle of Newton-homotopy forward mapping, the jacobian matrix shall remain unchanged while necessitating changes to the system of equations shown in Equation (4.16). The term “y” in the equation is the vector x^0 as shown in Equation (4.16).

$$f(x) = \begin{matrix} (x_1 + 2x_2 - 2) - (1 - t)(y_1 + 2y_2 - 2) \\ (x_1^2 - 4x_2^2 - 4) - (1 - t)(y_1^2 - 4y_2^2 - 4) \end{matrix} \quad (4.16)$$

This means that throughout the iteration of the mapping parameter, the initial condition remains constant for the term that is multiplied with $(1 - t)$. This means a new and updated initial condition will come for each Homotopy-iteration. Each Homotopy-iteration is solved with Newton-raphson. Effectively moving the initial condition for the problem closer to the answer for each iteration.

4.3 Kron reduction

In this thesis, the admittance matrices will be reduced to the internal nodes of the generators in order to implement the formulas presented later in the thesis. This reduction will be accomplished using a commonly known method known as Kron reduction. The present section will describe the underlying principle of this technique.

Kron reduction can be performed utilizing Equation (4.17).

$$Y_{kj(new)} = Y_{kj(orig)} - \frac{Y_{kN(orig)}Y_{Nj(orig)}}{Y_{NN(orig)}} \quad (4.17)$$

Consider a matrix of size 4x4, which is required to be reduced to a 3x3 matrix, as depicted in Equation (4.18).

$$\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & Y_{14} \\ Y_{21} & Y_{22} & Y_{23} & Y_{24} \\ Y_{31} & Y_{32} & Y_{33} & Y_{34} \\ Y_{41} & Y_{42} & Y_{43} & Y_{44} \end{bmatrix} \rightarrow \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{21} & Y_{22} & Y_{23} \\ Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \quad (4.18)$$

The values of k,j and N are determined as k = 3, j = 3 and N = 4, respectively. The elements of the resulting 3x3 matrix are computed using Equation (4.17) and an iterative process over k and j indices. An implementation of this process using Python is included in Appendix B for practical demonstration purposes.

5 SMIB Test Case

The SMIB test case is derived from the power system illustrated in example 13.3 of the book “Power System Analysis” [11]. The system topology is presented in figure Figure 5.1, while Table 5.1 provides a list of the system parameters.

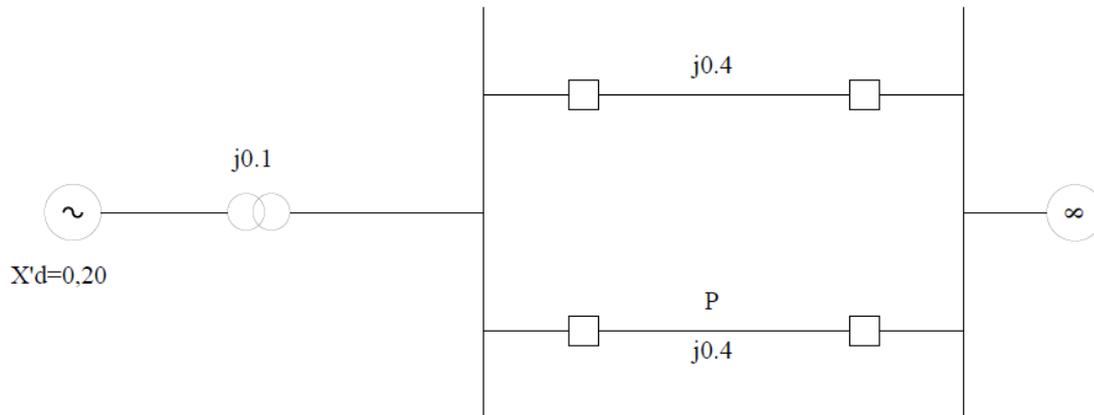


Figure 5.1: One-line diagram for the SMIB test case

Table 5.1: System parameters for the SMIB test case

H	$\frac{5MJ}{MVA}$
P_m	1 p.u
Terminal Voltage	1 p.u
Infinite bus voltage	1 p.u
f	60 Hz

5.1 Model

The classical model described in subchapter 2.3 indicates that during the numerical integration of the two first-order differential equations, both voltage and mechanical power are constant. This suggests that the only parameter that changes during the integration process is electrical power. As a result, it becomes crucial to evaluate the electrical power as a function of the rotor angle in the different system states, namely pre-fault, fault-on and post-fault. In this section, the calculation of the function of the electrical power for each of these states is presented.

SMIB Test Case

The pre-fault state is straight forward, in Equation (5.1) the formula for the electrical power is shown.

$$P_e = \frac{|v_1| \cdot |v_2|}{x} \sin\delta \quad (5.1)$$

Between the terminal voltage and the infinite bus, the electrical power is represented in Equation (5.2).

$$\frac{|V_t| \cdot |V|}{X} \sin\alpha = 1 \rightarrow \frac{1 \cdot 1}{0.3} \sin\alpha = 1 \quad (5.2)$$

The terminal voltage of the machine can be found as shown in Equation (5.3).

$$V_t = 1.0 \angle \sin^{-1} 0.3 = 1.0 \angle 17.458^\circ \quad (5.3)$$

The internal voltage of the machine could then be found as shown in Equation (5.4).

$$E' = \frac{1.0 \angle 17.458^\circ - 1.0}{j0.3} \cdot j0.20 + 1.0 \angle 17.458^\circ = 1.050 \angle 28.44^\circ \quad (5.4)$$

The electrical power in the pre-fault state is then presented in Equation (5.5).

$$P_e = \frac{(1.050)(1.0)}{0.5} \sin\delta = 2.10 \sin\delta \quad (5.5)$$

Now consider a bolted three phase short circuit at the point “P” as illustrated in Figure 5.2.

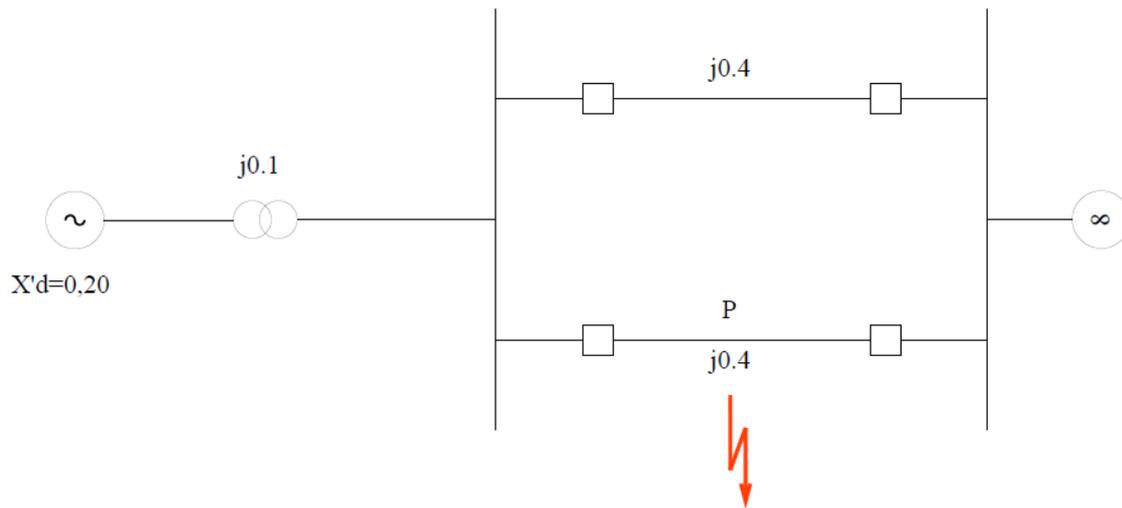


Figure 5.2: One-line diagram for the faulted condition

This makes the connection between the two parallel lines a delta connection as shown in Figure 5.3. This then gets transformed to a Y-connection as shown in Equation (5.6).

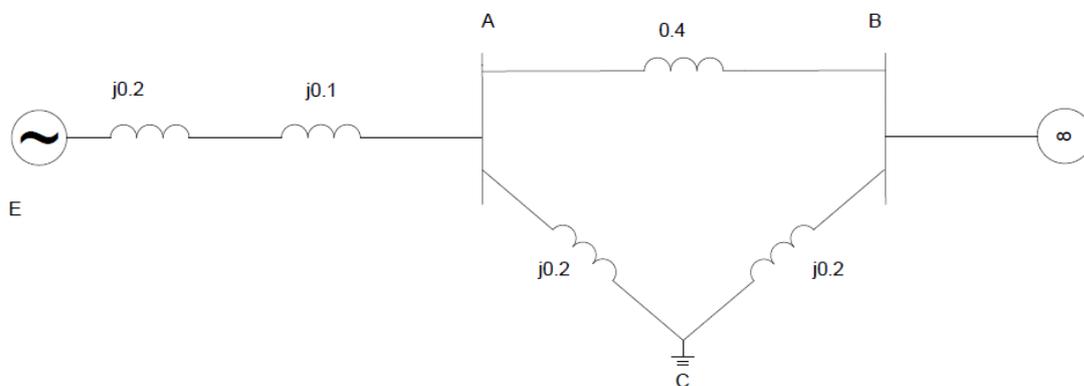


Figure 5.3: Impedance diagram of the faulted delta condition

$$Z_A = \frac{Z_{AB}Z_{CA}}{Z_{AB} + Z_{BC} + Z_{CA}} = j0.1$$

$$Z_B = \frac{Z_{BC}Z_{AB}}{Z_{AB} + Z_{BC} + Z_{CA}} = j0.1$$
(5.6)

$$Z_C = \frac{Z_{CA}Z_{BC}}{Z_{AB} + Z_{BC} + Z_{CA}} = j0.05$$

The system is now represented as a Y-connection, as illustrated in Figure 5.4. To determine the power flow between the generator's internal node and the infinite bus, a straightforward approach involves performing a $Y - \Delta$ transform from node A to node B, as demonstrated in Equation (5.7).

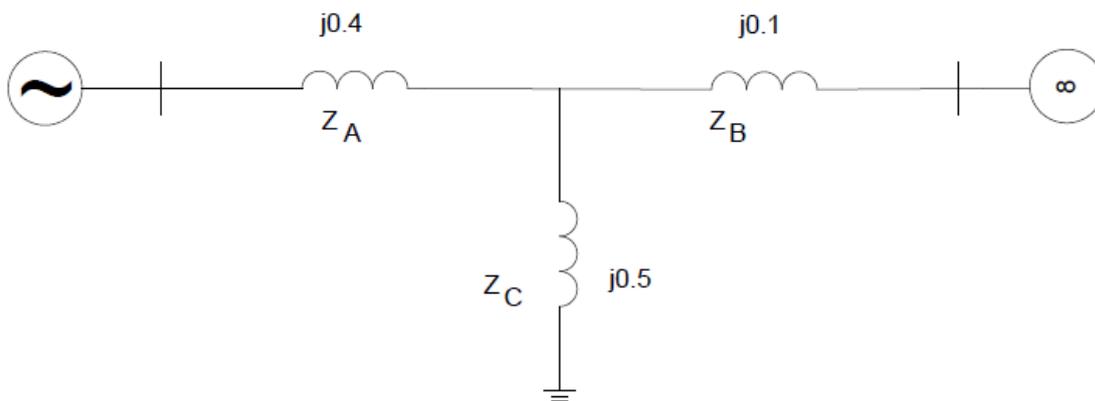


Figure 5.4: Impedance diagram of the faulted y-connection

$$Z_{AB} = \frac{Z_A Z_B + Z_B Z_C + Z_C Z_A}{Z_C} = j1.3 \quad (5.7)$$

The resulting equation representing the electrical power in the fault-on state can be seen in Equation (5.8).

$$P_e = \frac{1.05 \cdot 1}{1.3} \cdot \sin\delta = 0.808 \sin\delta \quad (5.8)$$

The equation for the post-fault state can be obtained in the same manner as the pre-fault state. The post-fault state is represented in Equation (5.9). Table 5.2 summarizes the equations for the three system states.

$$P_e = \frac{1.05 \cdot 1}{0.7} \cdot \sin\delta = 1.5 \sin\delta \quad (5.9)$$

Table 5.2: Overview of the electrical power for each system state

System State	Electrical Power
Pre-Fault	$2.1 \sin\delta$
Faulted	$0.808 \sin\delta$
Post-Fault	$1.500 \sin\delta$

5.2 Exit Point

As discussed in chapter 3 the aim is to find the point where the fault-on trajectory exits the stability boundary of the post-fault system. This is done by integrating the fault-on trajectory until it crosses the stability boundary. In theory this is called the exit point and it is equivalent to the first local maximum of potential energy.

The first thing that needs to be calculated is the post-fault SEP, at a stable equilibrium point P_a is zero. The post-fault SEP can be found as shown in Equation (5.10).

$$\begin{aligned}
 \frac{d\omega}{dt} &= 1 - 1.5 \sin\delta \\
 0 &= 1 - 1.5 \sin\delta \\
 \delta &= \sin^{-1}\left(\frac{1}{1.5}\right) = 41.81^\circ = 0.7297 \text{ rad}
 \end{aligned}
 \tag{5.10}$$

The potential energy as a function of time is plotted in Figure 5.5.

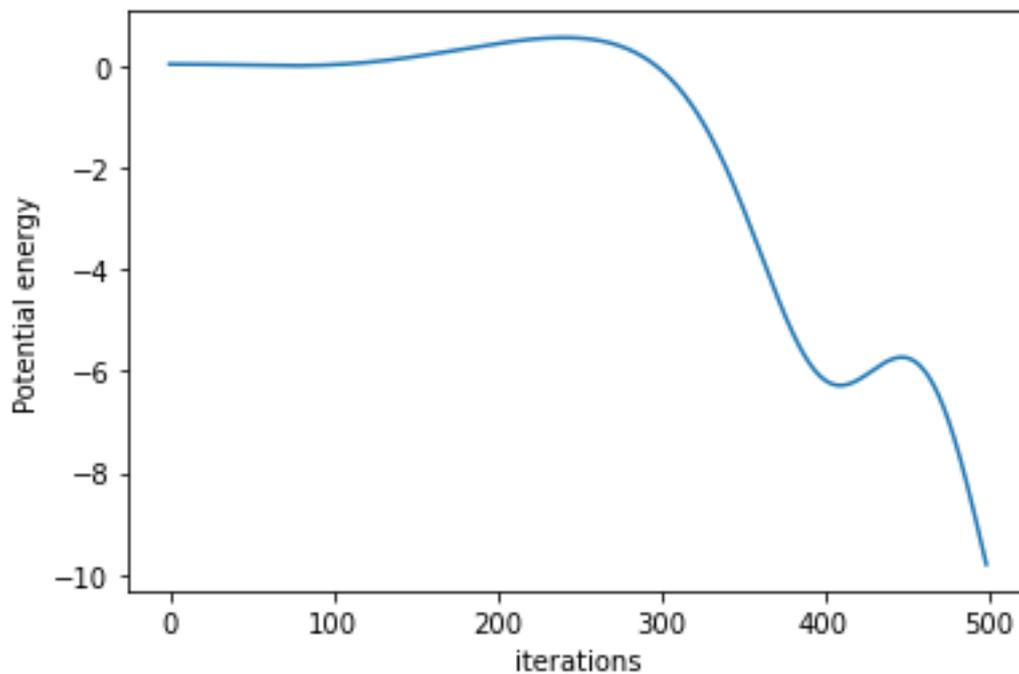


Figure 5.5: Potential energy for the fault-on period

The first maximum is found after 241 iterations, this gives an exit point equal to 2.411 rad. The single machine case is a special case where the exit point is the controlling unstable equilibrium point. This is illustrated in Figure 5.6.

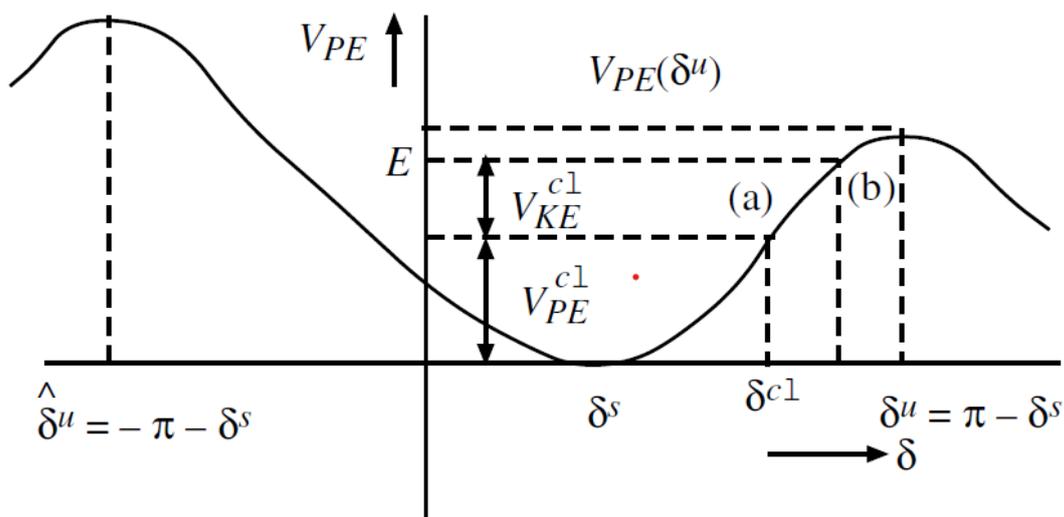


Figure 5.6: Potential Energy “well” for the SMIB case[12]

From the figure it is evident that the post-fault sep is surrounded by, two unstable equilibrium points at coordinates $-\pi - \delta^s$ and $\pi - \delta^s$. From the calculation in Equation (5.10), we know that $\delta^s = 0.7297$. A simple test where one takes $\pi - 0.7297 = 2.411$. Shows that the first

maximum of the potential energy function is in fact the controlling unstable equilibrium point.

5.3 Critical Clearing time and critical clearing angle

It is essential to ensure that the system's total energy does not exceed the critical energy, which is defined as the potential energy at the CUEP or $V(\delta^u)$ to relate it to Figure 5.6. The critical energy level for the present system has been calculated as -1.29. To determine the critical clearing time and angle, the energy of the system is evaluated during each timestep during the fault-on trajectory until the condition until $V(\omega, \delta) < V_{PE}(\delta^u)$ is no longer valid. This process yields a critical clearing time of 0.318 seconds and a critical clearing angle of 83.79 degrees.

To compare it to the equal area criteria, the formula for this specific case can be found in example 13.8 in the book "Power System Analysis"[11], for simplicity the calculation is shown in Equation (5.11).

$$\begin{aligned} \cos\delta_{cr} &= \frac{1.0}{2.10} (2.412 - 0.496) + \frac{0.714\cos(138.19)^\circ - 0.385\cos(28.44)^\circ}{0.714 - 0.385} \\ &= 0.1266 \end{aligned} \tag{5.11}$$

$$\delta_{cr} = \cos^{-1}(0.1266) = 82.726^\circ$$

This is slightly lower than what was achieved using the direct method. This is because of round-off errors in the calculation in the book. To validate that the critical clearing time will not lead to instability a time domain simulation has been performed, as shown in Figure 5.7. The time domain simulation using the classical formulation is simulated for 10 seconds with a damping factor of 0.01. This is done to ensure that the critical clearing time is not set too high and that it goes towards a stable operating point. The numerical integration and calculations for this test case can be found in appendix C.

SMIB Test Case

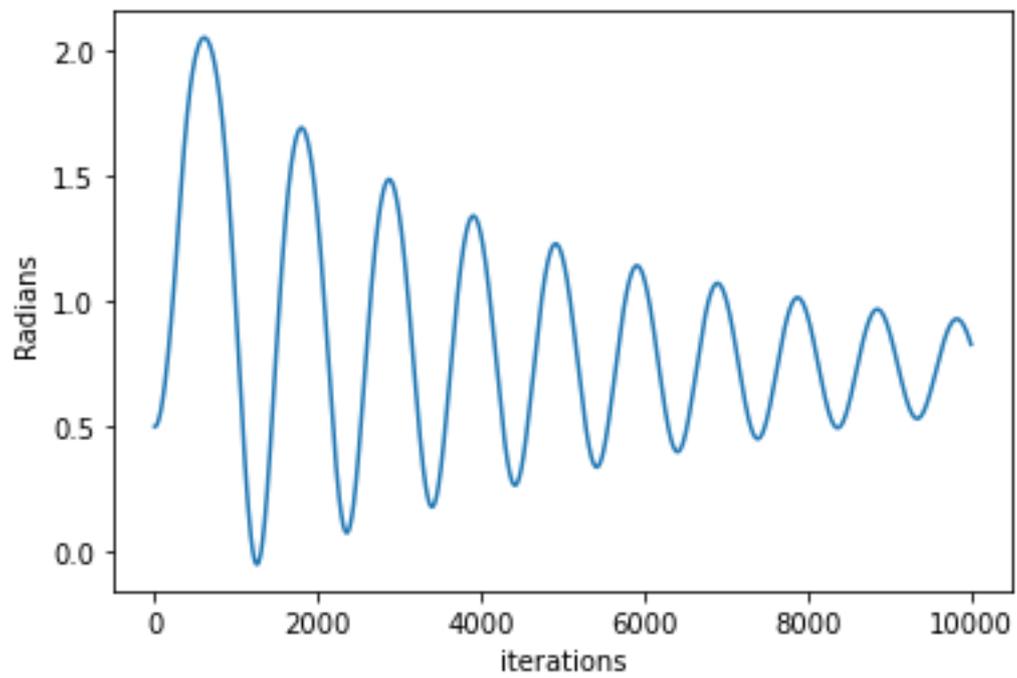


Figure 5.7: Time domain simulation for the SMIB test case

6 Multimachine Test Case

In this chapter, we will introduce the multimachine test case, which is based on the WSCC 3 generator 9 bus system. The algorithm for computing the CCT for a specific contingency was presented in chapter 3. The algorithm will be used to calculate the CCT for a fault occurring close to busbar 4, between lines 4 and 5. The computation will be shown in a stepwise manner. The results of this calculation will be validated against the thesis “Direct Methods for power systems transient stability analysis using BCU method”[15] and the report “A Lyapunov Function Based remedial Action Screening Tool Using real-time data”[5].

6.1 Determining Initial values

The multimachine system that will be used in this report is the WSCC test system. Relevant values for this system is found in the book from Sauer and Pai[12]. The system is shown in Figure 6.1.

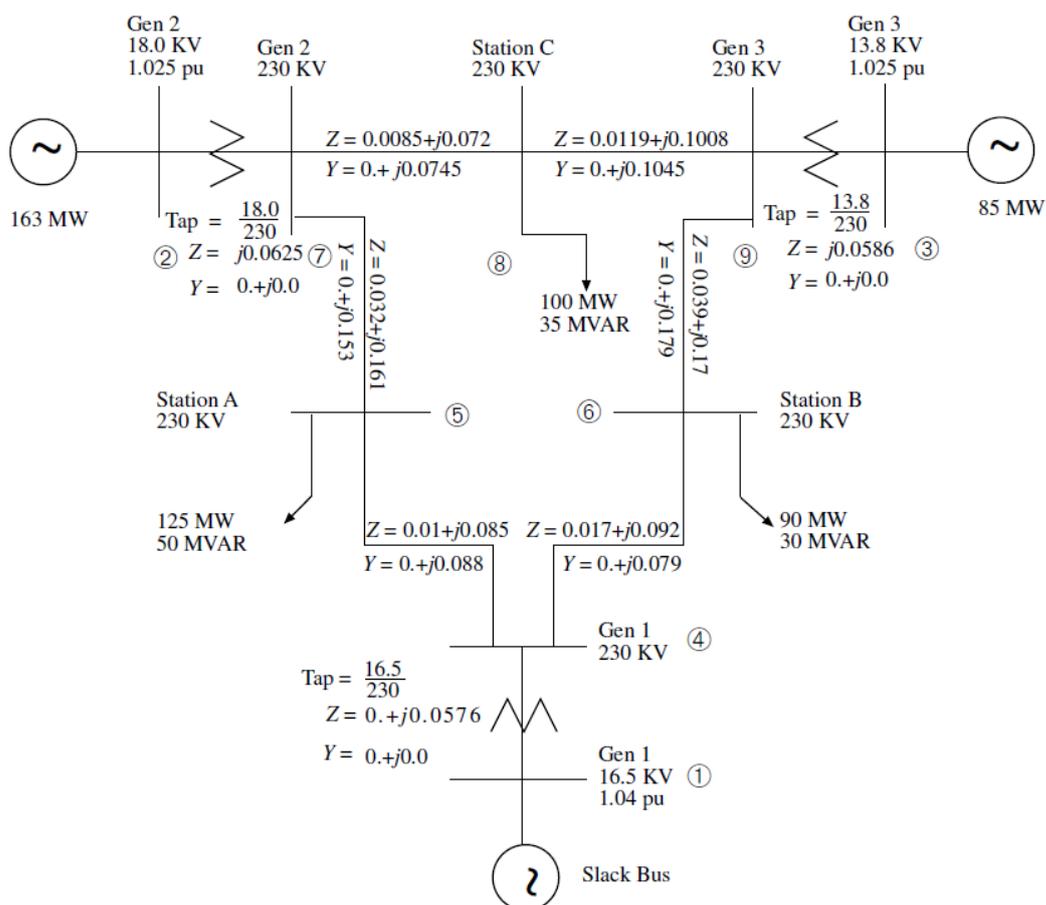


Figure 6.1:WSCC 3 generator 9 bus system [12]

Relevant values from the resulting load-flow analysis can be found in Table 6.1.

Table 6.1: Load-flow results

	Bus	Voltage (Pu)	Pg	Qg	-PL	-QL
1	Swing	1.04∠0°	0.716	0.27	-	-
2	(P-V)	1.025∠9.3°	1.63	0.067	-	-
3	(P-V)	1.025∠4.7°	0.85	-0.109	-	-
4	(P-Q)	1.026∠ - 2.2°	-	-	-	-
5	(‘)	0.996∠ - 4.0°	-	-	1.25	0.5
6	(‘)	1.013∠ - 3.7°	-	-	0.9	0.3
7	(‘)	1.026∠3.7°	-	-	-	-
8	(‘)	1.016∠0.7°	-	-	1.00	0.35
9	(‘)	1.032∠2.0°	-	-	-	-

Determining the initial values in accordance with the classical model and using an internal node model requires five steps:

1. Calculate the constant voltage behind the transient reactance from the pre-fault load-flow analysis.
2. Convert the loads to shunt impedances.
3. Generate a reduced fault-on y-matrix.
4. Generate a reduced post-fault y-matrix.
5. Determine the initial center of inertia parameters.

The data from the synchronous machines required to perform these calculations are given in Table 6.2.

Table 6.2: Relevant Generator values[12]

Generator	X'_d in per unit	M
1	0.0608	0.1254
2	0.1198	0.034
3	0.1813	0.016

The constant voltage behind the transient reactance can be calculated as shown in Equation (6.1).

$$E \angle \delta' = \left(V + \frac{Qx'_d}{V} \right) + j \left(\frac{Px'_d}{V} \right) \quad (6.1)$$

Where V is the magnitude of the terminal voltage. To find the rotor angle of the machine $\delta_0 = \delta' + \alpha$ where α is the angle at the terminal. The resulting internal voltages with corresponding rotor angles are depicted in Equation (6.2).

$$\begin{aligned} E'_1 &= 1.0566 \angle 2.27^\circ \\ E'_2 &= 1.0503 \angle 19.75^\circ \\ E'_3 &= 1.017 \angle 13.2^\circ \end{aligned} \quad (6.2)$$

The initial rotor angles are the pre-fault stable equilibrium point written in the unit of radians this becomes [0.0396,0.344,0.23]. Converting them into the initial angles of the center of inertia formulation as discussed in chapter 3, can be done by utilizing the formulas depicted in Equation (6.3).

$$\begin{aligned} \tilde{\delta}_i &= \delta_i - \delta_0 \\ \delta_0 &= \frac{1}{M_T} \sum_{i=0}^n M_i \delta_i \end{aligned} \quad (6.3)$$

Calculating $\delta_0 = 0.116$, the resulting rotor angles for the center of inertia formulation then becomes [-0.0764,0.229,0.114]. These angles then represent the pre-fault stable equilibrium point in the center of inertia formulation.

The next step is converting the loads to shunt impedances. This is because all loads are to be considered constant during the computation of the swing equation in accordance with the classical requirement. The formula for converting the loads to shunt admittances is shown in Equation (6.4)[12].

$$\bar{y}_{ii} = -\frac{(P_{Li} - jQ_{Li})}{V_i^2} \quad (6.4)$$

The resulting impedances for the loads are shown in Equation (6.5).

$$\begin{aligned} \bar{y}_{55} &= 1.26 - j0.504 \\ \bar{Y}_{66} &= 0.8776 - j0.2925 \\ \bar{Y}_{88} &= 0.9690 - j0.3391 \end{aligned} \quad (6.5)$$

The original y-matrix gets updated with two values, the transient reactance of the synchronous machines is added in series with the corresponding transformer reactance, and the shunt impedances that represent the load are added as a shunt impedance on the busbar where the load is connected.

To create the faulted y-matrix an ideal fault is considered between lines 4 and 5 close to busbar 4. This means that the voltage at busbar 4 = 0. This means that both the column and row of the bus fault location is removed from the original y-matrix and reduced using kron-reduction as explained in chapter 4 to a 3x3 matrix. For the post-fault matrix, the line from 4 to 5 is removed and kron reduction is utilized to generate a 3x3 matrix for the post-fault system.

6.2 Post-fault SEP

The post-fault stable equilibrium point represents the state of the power system when it returns to a steady state after a disturbance. As discussed in chapter 2, this equilibrium point can be obtained by solving the system of nonlinear equations, that for convenience is shown in Equation (6.6).

$$f_i = P_i - P_{ei} - \frac{M_i}{M_T} P_{COI} \quad (6.6)$$

Since the classical model is used the only element that changes is the y-matrix. Using the post-fault matrix and the initial angles as the initial guess this problem can be solved using a standard newton Raphson algorithm, in literature the post-fault SEP normally lies within the region of convergence when the pre-fault SEP is used as the initial guess[13].

For this contingency the number of equations is reduced as discussed in chapter 2. Synchronous machines one and two are chosen as the reference machines and to calculate the angle of machine three Equation (6.7) is used.

$$\tilde{\delta}_3 = -\frac{M_1\tilde{\delta}_1 + M_2\tilde{\delta}_2}{M_3} \quad (6.7)$$

The resulting post fault SEP solved using newtons method is depicted in Equation (6.8). The script for calculating the post fault SEP is shown in appendix (x.x).

$$\begin{aligned} \tilde{\delta}_1 &= -0.6045158 \\ \tilde{\delta}_2 &= 0.18174784 \\ \tilde{\delta}_3 &= 0.08757509825 \end{aligned} \quad (6.8)$$

6.3 Exit Point

The determination of exit points is a crucial step in the CUEP method for transient stability analysis. It is defined as the point where the fault-on trajectory intersects the stability boundary of the post-fault system. There are several methods for computing the exit point, but in this thesis, we employ the first maximum of the potential energy along the fault-on trajectory as the methodology.

Numerical integration is performed using a 4th order Runge-Kutta solver, as outlined in chapter 4. Figure 6.2 illustrates the potential energy, and the delta corresponding to the first maximum of the potential energy function is the exit point. To compare the results obtained from this approach, there has also been included exit points from one master thesis and one technical report that employ the WSCC 3 generator 9 bus system. In Table 6.3 the exit points are presented. Where No.1 corresponds to the work done in this thesis, No.2 is from the master thesis “Direct Methods for Power Systems Transient Stability analysis using BCU method”, and No.3 is from the report “A Lyapunov Function Based Remedial Action Screening Tool Using Real-Time Data”.

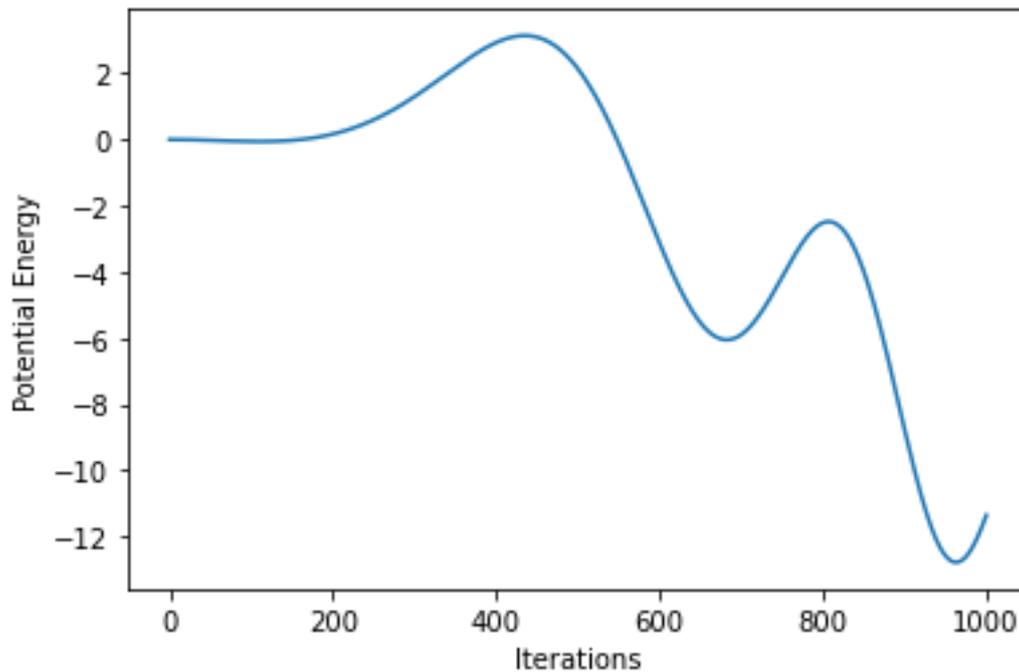


Figure 6.2: Potential Energy for the fault-on period

Table 6.3:Exit Points

No.	Fault Near bus	Line trip		Exit points (rad)		
		From	To	$\tilde{\delta}_1$	$\tilde{\delta}_2$	$\tilde{\delta}_3$
1	4	4	5	-0.77538	1.93921	1.94038
2	4	4	5	-808206	2.02901	2.02450
3	4	4	5	-0.83157	2.05223	2.16748

6.4 Controlling unstable equilibrium points

One of the primary challenges in utilizing the controlling unstable equilibrium point methodology is accurately computing the CUEP. The CUEP is embedded within many unstable equilibrium points on the stability boundary of the post-fault system, making it difficult to distinguish the CUEP from other UEP's. Additionally, the convergence regions of CUEP's are characterized by fractal shapes, which can cause many numerical methods to fail to converge. Finding an initial guess that lies within the region of convergence of the CUEP is also a challenging task. As a result, computing the CUEP is a crucial step in successful implementation of the CUEP methodology[16] [13].

In this thesis a homotopy based approach will be used when computing the CUEP. This is a more computationally taxing method than the BCU method, but in theory it reduces the necessity for accurately computing the exit point.

The principle of homotopy is covered in chapter 4. However, there is some additional information that is required to explain how computation of CUEP's is done utilizing the homotopy approach, the exit point is used as the initial guess. Determining whether to use a backward mapping or forward mapping approach can be assessed by examining Figure 6.3. In the algorithm, a forward mapping approach is assumed by default. After one iteration, if the potential energy decreases with respect to the exit point, this indicates that the algorithm is heading in the direction of the CUEP, and therefore the correct mapping direction has been identified. Conversely, if the potential energy increases with respect to the exit point, the algorithm should switch to a backward mapping approach to ensure an accurate identification of the CUEP. Thus, the mapping direction of the mapping approach can be adjusted based on the behavior of the potential energy with respect to the exit point. It should be noted that in this thesis fixed point homotopy was used, meaning that the mapping factor t had a predetermined value which was used throughout the process. However, to increase the computational speed of the method an adjustable mapping factor should be utilized[16].

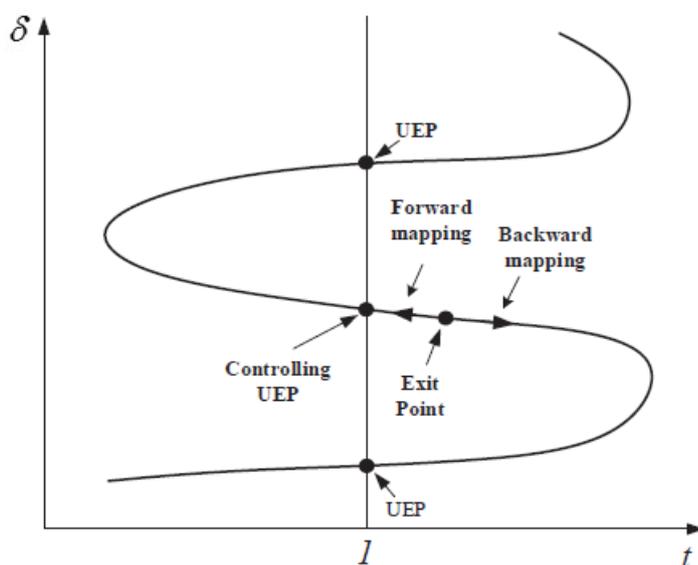


Figure 6.3:Forward and Backward mapping CUEP

The computation of the CUEP is done solving the same system of nonlinear equations as depicted in Equation (2.13). In Table 6.4 three sets of CUEP's are shown. Where No.1 corresponds to the work done in this thesis, No.2 is from the master thesis "Direct Methods for Power Systems Transient Stability analysis using BCU method", and No.3 is from the report "A Lyapunov Function Based Remedial Action Screening Tool Using Real-Time Data".

Table 6.4:CUEP's

No.	Fault Near bus	Line trip		Controlling UEPs, (rad)		
		From	To	$\tilde{\delta}_1$	$\tilde{\delta}_2$	$\tilde{\delta}_3$
1	4	4	5	-0.83065	2.16207	1.9158
2	4	4	5	-0.79264	2.16069	1.94184
3	4	4	5	-0.83157	2.05223	2.16748

6.5 Critical clearing time

Finding the critical clearing time is obtained by reusing the integration from the exit point calculation. Then the inequality $V(x) \leq V_{cr}$ is checked until the systems energy equals the critical energy. The CUEP computed in this thesis along with the CUEPs from the thesis and report for comparison is presented in Table 6.5.

Table 6.5:Critical Energy and Critical Clearing Time

No.	Critical Energy	Critical Clearing time
No.1	3.14345	0.332
No.2	3.094	0.33
No.3	3.05	0.328

7 Discussion

An intriguing discovery in the SMIB-test case, is that all three direct methods discussed in Chapter 3 would produce identical results. For the closest UEP method the UEP found would in fact be the closest, for the PEBS method the UEP found would be the first maximum of the potential energy, and if one were to use a larger timestep so that the exit point did not equal the UEP and solved it using Newtons method it would also converge to the same UEP. Therefore, the SMIB-test case is a special case where all methods would yield the same result.

When analyzing the multimachine case, it is advisable to exercise caution while comparing the exit point, CUEP, critical energy and critical clearing times, as the assumptions made in the documents used for validation are not entirely clear. This lack of information could lead to misconceptions. However, it should be noted that the selected contingency is a prevalent one for the WSCC 3 generator 9 bus system, and thus the findings in this thesis are consistent with exiting literature. Concluding on which method gave the most conservative results is not possible to know seeing as all the assumptions is not known.

Two assumptions made in the multimachine case of this thesis may increase the conservatism of the results. Firstly, assuming an ideal fault condition with zero fault resistance may exaggerate the mismatch between electrical and mechanical power, leading to an overestimation of the initial acceleration. Secondly, neglecting damping may result in an overestimation of the angular velocity, further increasing conservatism in the analysis. However, it should be noted that these assumptions do not reflect the conditions of practical power systems. One of the challenges is to find a balance between a conservative analysis and a realistic one to maximize the potential of utilizing direct methods.

Another simplification made in this thesis is that loads are converted to shunt impedances and remain constant during the simulation. However, in practical power systems, dynamic loads, like induction motors and power electronic devices are present. They affect the stability of the system, by utilizing a reduced network model it is not possible to study the behavior of dynamic loads. The penetration of power electronic converter interfaced technologies could potentially make this more of a problem in the future and is definitively something that should be investigated further.

This thesis showcases how direct methods can be utilized to assess the transient stability of a power system. In the test cases off-line analysis has been performed. However, as mentioned in the introduction it is desirable to utilize this method for online transient stability analysis. Implementing this as a online approach requires the methodology to be incorporated in existing control systems, where the analysis can be performed continuously and be utilized for remedial actions in the power system.

Discussion

Remedial action can be classified as either corrective or preventive. Corrective actions, such as line tripping, load shedding, or generation tripping are triggered immediately if a significant disturbance occurs, whereas preventive actions, such as load rescheduling or load shedding, are triggered periodically or upon request of system operators.

An important concept in this approach is the energy sensitivity margin, which is the difference between the energy at the CUEP and the energy at clearing. If the margin is larger than zero, the contingency is deemed stable. The energy sensitivity margin is depicted in Equation (7.1).

$$\Delta V = V^u - V^{cl} \quad (7.1)$$

The energy margin can also be utilized for determining how load rescheduling is performed, but further details on this aspect are beyond the scope of this thesis. By integrating online transient stability analysis, the power system can be operated more realistically, closer to its practical limits, instead of relying on limits determined using offline analysis, which may not accurately reflect the actual operating conditions of the grid[5].

8 Conclusion and future work

8.1 Conclusion

In test case 1 the single machine infinite bus test case, obtained results that aligned with the expectations and were consistent with the outcomes derived from the equal area criteria. This confirms the effectiveness of the implementation of direct methods, demonstrating its validity and reliability.

Furthermore, the CUEP homotopy approach was employed and tested for a contingency within the multimachine test case. The results obtained were found to be in accordance with existing literature, and validation was conducted using two reputable sources. This successful application of the homotopy approach establishes its viability in determining critical clearing times for contingencies in the power system.

During the literature review, it was revealed that different methods for assessing transient stability have varying levels of conservatism. The closest UEP method was identified as the most conservative, followed by CUEP, while PEBS method was determined to be the least conservative.

Another significant aspect worth mentioning is the report titled “A Lyapunov Function Based Remedial Action Screening Tool Using Real-time Data”. In this report, the CUEP homotopy approach was applied to a large-scale real-time digital simulator. Utilizing data from Southern California Edison one of the largest utility companies in California. The simulator facilitated real-time testing of a system with up to 286 busbars. Although specific details of the performed tests are not publicly available, the report indicates a successful outcome, suggesting broader testing and commercialization. The project was completed in 2016 and has led to more research within the field.

The successful implementation of direct methods conducted in the test cases demonstrates the feasibility of utilizing these approaches for assessing transient stability of power systems. The extensive research conducted in the field, along with ongoing investigations, demonstrates the promising prospects for enhancing power system operability, stability and overall grid resilience.

8.2 Future work

In terms of future work, several recommendations emerge from this thesis. Firstly, it would be valuable to implement the proposed methodology in a larger-scale model, such as the Nordic Power system. This expanded application would provide insights into the effectiveness and scalability of the approach.

Furthermore, incorporating structure preserving models into the analysis would significantly enhance the level of information obtained from transient stability assessments. By considering the characteristics and dynamics of the system components, a more comprehensive understanding of the transient stability issue can be achieved.

Additionally, exploring the impact of power electronic converter interfaced technologies would be useful. Assessing how the penetration of these technologies influence stability can help in understanding how they affect stability.

References

- [1] K. H. LaCommare and J. H. Eto, ‘Understanding the cost of power interruptions to U.S. electricity consumers’, LBNL--55718, 834270, Sep. 2004. doi: 10.2172/834270.
- [2] H. Chiang, *Direct methods for stability analysis of electric power systems: theoretical foundation, BCU methodologies, and applications*. Hoboken, N.J: John Wiley & Sons, 2011.
- [3] P. C. Magnusson, ‘The Transient-Energy Method of Calculating Stability’, *Trans. Am. Inst. Electr. Eng.*, vol. 66, no. 1, pp. 747–755, Jan. 1947, doi: 10.1109/T-AIEE.1947.5059502.
- [4] L. L. Grigsby, Ed., *Power system stability and control*. Boca Raton: CRC Press, 2007.
- [5] J. Mitra, M. Ben-Idris, O. Faruque, S. Backhaus, S. Deb, and Southern California Edison, ‘A Lyapunov Function Based Remedial Action Screening Tool Using Real-Time Data’, DOE-MSU--00625, 1421846, Mar. 2016. doi: 10.2172/1421846.
- [6] P. Kundur, N. J. Balu, and M. G. Lauby, *Power system stability and control*. in The EPRI power system engineering series. New York: McGraw-Hill, 1994.
- [7] K. Mazur, M. Wydra, and B. Ksiezopolski, ‘Secure and Time-Aware Communication of Wireless Sensors Monitoring Overhead Transmission Lines’, *Sensors*, vol. 17, no. 7, p. 1610, Jul. 2017, doi: 10.3390/s17071610.
- [8] ‘Definition and Classification of Power System Stability IEEE/CIGRE Joint Task Force on Stability Terms and Definitions’, *IEEE Trans. Power Syst.*, vol. 19, no. 3, pp. 1387–1401, Aug. 2004, doi: 10.1109/TPWRS.2004.825981.
- [9] N. Hatziargyriou *et al.*, ‘Definition and Classification of Power System Stability – Revisited & Extended’, *IEEE Trans. Power Syst.*, vol. 36, no. 4, pp. 3271–3281, Jul. 2021, doi: 10.1109/TPWRS.2020.3041774.
- [10] ‘Introduction_Power_System_Stability.pdf’. Accessed: May 05, 2023. [Online]. Available: https://www.iitp.ac.in/~siva/2022/ee549/Introduction_Power_System_Stability.pdf
- [11] J. J. Grainger, W. D. Stevenson, and G. W. Chang, *Power system analysis*, International edition. New York, NY: McGraw Hill Education, 2016.
- [12] P. W. Sauer and M. A. Pai, *Power system dynamics and stability*, Updated ed. Champaign, IL.: Stipes Publishing L.L.C., 2006.
- [13] Owusu-Mireku, Robert, ‘IMPROVEMENTS OF BCU-BASED DIRECT METHODS AND APPLICATIONS OF BCU TO LINE SWITCHING DESIGNS FOR ENHANCING ONLINE TRANSIENT STABILITY OF LOOK-AHEAD POWER SYSTEMS’, doi: 10.7298/1QE4-W408.
- [14] K. Sun, ‘ECE 522 - Power Systems Analysis II Spring 2021 Transient Stability’, 2021.
- [15] C. Dai, ‘Direct methods for power system transient stability analysis using BCU method.’, The University of Texas, Austin, 2013. [Online]. Available: <https://repositories.lib.utexas.edu/handle/2152/21765>

References

- [16] J. Mitra and M. Benidris, 'A Homotopy-Based Method for Robust Computation of Controlling Unstable Equilibrium Points', *IEEE Trans. Power Syst.*, vol. 35, no. 2, pp. 1422–1431, Mar. 2020, doi: 10.1109/TPWRS.2019.2942948.
- [17] S. Linge and H. P. Langtangen, *Programming for computations - Python: a gentle introduction to numerical simulations with Python*, Second edition. in Texts in Computational Science and Engineering, no. 15. Cham: Springer Open, 2020.
- [18] A. B. Rehiara and S. Setiawidayat, 'An Implemented Software for Transient Stability Analysis of SMIB Based on Runge- Kutta Method', 2014.
- [19] S. Liao, *Homotopy analysis method in nonlinear differential equations*. Berlin, Heidelberg: Springer, 2012.

Appendices

Appendix A Task Description for the Master Thesis

Appendix B Kron Reduction

Appendix C Test case 1: Single machine infinite bus

Appendix D Test case 2: WSCC 3 generator 9 bus system

Appendix A: Task description for the master thesis

USN University of
South-Eastern Norway
Faculty of Technology, Natural Sciences and Maritime Sciences, Campus Porsgrunn

FMH606 Master's Thesis

Title: Direct methods for transient stability analysis and contingency screening in power systems

USN supervisor: Gunne John Heggliid

External partner: Skagerak Kraft AS

Task background:

The USN department has research within electric power system theory including transient stability evaluation in connection with contingency analysis. The institute wants to build up knowledge within this field. The most used technique for such analysis is time-domain simulation of a model containing a set of ordinary first order nonlinear differential equations which are solved together with a set of algebraic equations describing the power transmission/distribution system. Earlier these type of calculations have been used in system planning including establishing of basis for setting of protection equipment and following limits for transmission capacities in part of or whole of a grid system. When the penetration of new renewables (wind and sun) has increased the need of such capacity evaluation has also increased. On this background the system operators (TSO & DSO) will look for effective online/realtime methods to make evaluation on stability properties of the generation and grid systems. (Realtime is often described within a time fram of minutes and online within a timeframe of quarter to whole hour as part of operational planning activities.)

Task description:

Stability assessment methods based on energy function have been used since the 60-ties. Such methods was first developed by Lyapunov and was mostly used in complex nonlinear control system and does not require full package time-domain simulations. Stability performance can be clarified by far less calculation effort than detailed time-domain simulations. In the later years such energy-based methods have also been developed to evaluate the degree of stability and also exact stability limits for a complex grid system with numerous synchronous generators.

The following tasks should be carried out:

1. A literature study on the theoretical foundation for Direct methods for Stability analysis with emphasis on CUEP.
2. Make a script implementing a homotopy based method to find the CUEP for an n-bus system with a classical generator model (constant voltage behind a reactance)
3. Perform a simulation on a simple system, 3 generators and 9 bus system.
4. Utilize the data to determine the available transfer capability.

Student category: EPE

Is the task suitable for online students (not present at the campus)? No

Practical arrangements:

The student needs to be present at campus at least for the supervision meetings.

Supervision:

As a general rule, the student is entitled to 15-20 hours of supervision. This includes necessary time for the supervisor to prepare for supervision meetings (reading material to be discussed, etc).

Signatures:

Supervisor (date and signature): 31/1-2023 

Student (write clearly in all capitalized letters):

Student (date and signature): 31/1-2023 

Appendix B: Kron Reduction

Methods created in python to perform kron reduction to create the fault on and post-fault reduced matrices.

```
def kronreduction(Y):
    reduced_Y = np.zeros((len(Y)-1,len(Y)-1),dtype =complex)

    for i in range(0,len(reduced_Y),1):
        for j in range(0,len(reduced_Y),1):
            reduced_Y[i][j] = Y[i][j]-(Y[i][len(Y)-1]*Y[len(Y)-1][j])/Y[len(Y)-1][len(Y)-1]

    return reduced_Y

def kronreduction_comp(Ybus, num):
    for x in range(num):
        Ybus = kronreduction(Ybus)
    return Ybus

def complete(lines,From,to,faultbus):

    Postfault = [line for line in lines if not (line['from'] == From and line['to'] == to)]
    Postfault = y_matrix(Postfault,9)

    Postfault[4][4] = Postfault[4][4]+(1.2610-0.5044j)
    Postfault[5][5] = Postfault[5][5]+(0.8776-0.2925j)
    Postfault[7][7] = Postfault[7][7]+(0.9690-0.3391j)

    Fault_on = y_matrix(lines, 9)
    Fault_on[4][4] = Fault_on[4][4]+(1.2610-0.5044j)
    Fault_on[5][5] = Fault_on[5][5]+(0.8776-0.2925j)
    Fault_on[7][7] = Fault_on[7][7]+(0.9690-0.3391j)
    Fault_on = np.delete(Fault_on,faultbus-1,axis=1)
    Fault_on = np.delete(Fault_on,faultbus-1,axis=0)

    Reduced_Fault_on = kronreduction_comp(Fault_on, 5)

    Reduced_Postfault = kronreduction_comp(Postfault, 6)

    return Reduced_Fault_on,Reduced_Postfault
```

Appendix C: Test Case 1 SMIB

Runge Kutta 4th order solver for numerical integration of the fault-on trajectory.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import argrelextrema

def runge_kutta(delta, w, w_s, f, H, Pm, v1,v2, prefault, h, t_0, t_final,fault,postfault):
    t = t_0
    delta_list = [delta]
    w_list = [w]
    time = [t_0]
    constant = fault
    while t < t_final:

        if t >= 0:
            constant = fault

        Pe = constant*np.sin(delta)

        k1_delta = h * (w - w_s)
        k1_w = h * (f * np.pi / H) * (Pm - Pe)

        k2_delta = h * ((w + k1_w / 2) - w_s)
        k2_w = h * (f * np.pi / H) * (Pm - Pe)

        k3_delta = h * (w + k2_w / 2 - w_s)
        k3_w = h * (f * np.pi / H) * (Pm - Pe)

        k4_delta = h * (w + k3_w - w_s)
        k4_w = h * (f * np.pi / H) * (Pm - Pe)

        delta = delta + (k1_delta + 2 * k2_delta + 2 * k3_delta + k4_delta) / 6
        w = w + (k1_w + 2 * k2_w + 2 * k3_w + k4_w) / 6
        t = t + h

        delta_list.append(delta)
        w_list.append(w)
        time.append(t)

    return time, delta_list, w_list
```

Finding the first maximum of the potential energy, the critical energy and corresponding critical clearing time and angle for comparison with the equal area criterion.

```

def potentialEnergy(systemstate,x,postfaultsep):
    return -1*(x-postfaultsep)-systemstate*(np.cos(x)-np.cos(postfaultsep))

def maxPotential(systemstate,cuep):
    return -1*cuep-systemstate*np.cos(cuep)

def entireEnergyfunction(M,omega,delta,postfaultsep,P,systemstate):
    energy = 1/2*M*omega**2-P*(delta)-systemstate*(np.cos(delta))
    return energy

Pot = []
for i in range(0,500,1):
    X = potentialEnergy(postfault, De[i], postfaultsep)
    Pot.append(X)

First = np.array(Pot)

FirstMaximum = argrelextrema(First,np.greater)

Criticalenergy = maxPotential(postfault, De[241])
print(Criticalenergy)

print(De[241])
print(FirstMaximum)

b = 0

energy = entireEnergyfunction(M, W[0], De[0], postfaultsep, Pm, postfault)
print(energy)

while energy < Criticalenergy:
    b=b+1
    energy = entireEnergyfunction(M, W[b], De[b], postfaultsep, Pm, postfault)

print(b)

print(De[b]*180/np.pi)
print(t[b])

```

For the time domain simulation, the RK4 solver was updated to include damping and clear the fault at the CCT, which can be seen in the code.

```
def runge_kutta(delta, w, w_s, f, H, Pm, v1,v2, pefault, h, t_0, t_final,fault,postfault,damping):
    t = t_0
    delta_list = [delta]
    w_list = [w]
    time = [t_0]
    constant = fault
    while t < t_final:

        if t <= 0.318:
            constant = fault
        else:
            constant = postfault

        Pe = constant*np.sin(delta)

        k1_delta = h * (w - w_s)
        k1_w = h * (f * np.pi / H) * (Pm - Pe-damping*(w-w_s))

        k2_delta = h * ((w + k1_w / 2) - w_s)
        k2_w = h * (f * np.pi / H) * (Pm - Pe-damping*(w-w_s))

        k3_delta = h * (w + k2_w / 2 - w_s)
        k3_w = h * (f * np.pi / H) * (Pm - Pe-damping*(w-w_s))

        k4_delta = h * (w + k3_w - w_s)
        k4_w = h * (f * np.pi / H) * (Pm - Pe-damping*(w-w_s))

        delta = delta + (k1_delta + 2 * k2_delta + 2 * k3_delta + k4_delta) / 6
        w = w + (k1_w + 2 * k2_w + 2 * k3_w + k4_w) / 6
        t = t + h

        delta_list.append(delta)
        w_list.append(w)
        time.append(t)

    return time, delta_list, w_list
```

Appendix D: Test Case 2 Multi Machine

To solve for equilibrium points the Sympy library in Python has been utilized. This enables symbolic math for computing the jacobian of Equation (2.13) and return numerical values for both the jacobian and the function.

```
def f(E,D,Ym,M,mt,P):
    d1 = Symbol('d1')
    d2 = Symbol('d2')
    d3 = Symbol('d3')
    E1 = Symbol('E1')
    E2 = Symbol('E2')
    E3 = Symbol('E3')
    G11 = Symbol('G11')
    G12 = Symbol('G12')
    G13 = Symbol('G13')
    G21 = Symbol('G21')
    G22 = Symbol('G22')
    G23 = Symbol('G23')
    G33 = Symbol('G33')
    G31 = Symbol('G31')
    G32 = Symbol('G32')
    B12 = Symbol('B12')
    B13 = Symbol('B13')
    B21 = Symbol('B21')
    B23 = Symbol('B23')
    B31 = Symbol('B31')
    B32 = Symbol('B32')
    MT = Symbol('MT')
    Pm1 = Symbol('Pm1')
    Pm2 = Symbol('Pm2')
    M1 = Symbol('M1')
    M2 = Symbol('M2')
    M3 = Symbol('M3')
    Pm3 = Symbol('Pm3')

    Y = Matrix([d1,d2])

    X=Matrix([[Pm1-E1**2*G11-((E1*E2*(B12*sin(d1-d2)+G12*cos(d1-d2)))+E1*E3*
    (B13*sin(d1+(M1*d1+M2*d2)/M3)+G13*cos(d1+(M1*d1+M2*d2)/M3)))-
    M1/MT*((Pm1-E1**2*G11)+(Pm2-E2**2*G22)+(Pm3-E3**2*G33))-2*
    (E1*E2*G12*cos(d1-d2)+E1*E3*G13*cos(d1+(M1*d1+M2*d2)/M3)+E2*E3*G23*cos(d2+(M1*d1+M2*d2)/M3))],
    [Pm2-E2**2*G22-((E2*E1*(B21*sin(d2-d1)+G21*cos(d2-d1)))+E2*E3*
    (B23*sin(d2+(M1*d1+M2*d2)/M3)+G23*cos(d2+(M1*d1+M2*d2)/M3)))-
    M2/MT*((Pm1-E1**2*G11)+(Pm2-E2**2*G22)+(Pm3-E3**2*G33))-2*(E1*E2*G12*
    cos(d1-d2)+E1*E3*G13*cos(d1+(M1*d1+M2*d2)/M3)+E2*E3*G23*cos(d2+(M1*d1+M2*d2)/M3))],
    ])
```

```

Bφ = X.jacobian(Y)
function = X.subs({d1:D[0],d2:D[1],M1:M[0],M2:M[1],M3:M[2],
                  E1:E[0],E2:E[1],E3:E[2],MT:mt,
                  B12:np.imag(Ym[0][1]),B13:np.imag(Ym[0][2]),
                  B21:np.imag(Ym[1][0]),B23:np.imag(Ym[1][2]),
                  G12:np.real(Ym[0][1]),G13:np.real(Ym[0][2]),
                  G21:np.real(Ym[1][0]),G23:np.real(Ym[1][2]),
                  Pm1:P[0],Pm2:P[1],Pm3:P[2],
                  G11:np.real(Ym[0][0]),G22:np.real(Ym[1][1]),
                  G33:np.real(Ym[2][2]),G31:np.real(Ym[2][0]),
                  G32:np.real(Ym[2][1]),B32:np.imag(Ym[2][1]),
                  B31:np.imag(Ym[2][0])
                  })

numericalJacobian = Bφ.subs({d1:D[0],d2:D[1],M1:M[0],M2:M[1],M3:M[2],MT:mt,
                             E1:E[0],E2:E[1],E3:E[2],
                             B12:np.imag(Ym[0][1]),B13:np.imag(Ym[0][2]),
                             B21:np.imag(Ym[1][0]),B23:np.imag(Ym[1][2]),
                             B31:np.imag(Ym[2][0]),B32:np.imag(Ym[2][1]),
                             G11:np.real(Ym[0][0]),G12:np.real(Ym[0][1]),
                             G13:np.real(Ym[0][2]),G21:np.real(Ym[1][0]),
                             G22:np.real(Ym[1][1]),G23:np.real(Ym[1][1]),
                             G31:np.real(Ym[2][0]),G32:np.real(Ym[2][1]),
                             G33:np.real(Ym[2][2])})

return function,numericalJacobian

```

The data coming out from this method cannot be used directly in a numerical solver, therefore the data needs to be reshaped.

```

def Making_it_readable(E,D,Y,M,MT,Pm):
    By,Bφ = f(E,D,Y,M,MT,Pm)

    By1 = np.array(By)
    By2 = By1.reshape(2,1)
    F = By2.astype(np.float64())

    Bφ1 = np.array(Bφ)
    Bφ2 = Bφ1.reshape(2,2)
    J = Bφ2.astype(np.float64())
    return F,J

```

Newtons method is used both for calculating the postfault SEP and the CUEP. However, when calculating the CUEP a full solution of Newtons method is used per homotopy iteration.

```

def Newton(E,D,Ybus,M,MT,Pm,eps):
    D3 = D.reshape(2,1)
    F,J = Making_it_readable(E, D, Ybus, M, MT, Pm)
    F_norm = np.linalg.norm(F,ord=2)
    maxiteration = 100
    counter =0
    while abs(F_norm)>eps and counter < maxiteration:
        F,J = Making_it_readable(E, D, Ybus, M, MT, Pm)
        X = D3 - np.dot(np.linalg.inv(J),F)
        D3 = X
        D = X.flatten()
        counter +=1
        F_norm = np.linalg.norm(F,ord=2)
    return D

```

Defining P_{COI} and P_{ei}

```

def PEI(Ybus,num,E,D):
    Pei = 0
    for i in range(0,len(E),1):
        if num != i:
            Pei += E[num]*E[i]*(np.imag(Ybus[num][i])*np.sin(D[num]-D[i])+np.real(Ybus[num][i])*np.cos(D[num]-D[i]))
    return Pei

def Pcoi(Ybus,E,D,Pm):
    Pcoi = 0
    Pi = 0
    coiterm = 0

    for i in range(0,len(E),1):
        Pi += Pm[i]-(E[i]**2)*np.real(Ybus[i][i])

    for j in range(0,len(E)-1,1):
        for b in range(0,len(E),j+1):
            coiterm += E[j]*E[b]*np.real(Ybus[j][b])*np.cos(D[j]-D[b])

    Pcoi = Pi-2*coiterm
    return Pcoi

```

The fault-on matrix gets integrated using an RK4 solver.

```

def runge_kutta(voltages,Ybus,iDelta,iomega,M,Pm,MT,h,simulationtime):

    number_of_rows = simulationtime/h
    delta_tilde = np.zeros((int(number_of_rows)+1,len(voltages)))
    omega_tilde = np.zeros((int(number_of_rows)+1,len(voltages)))

    for i in range(0,len(voltages),1): #inserting initial values
        delta_tilde[0][i] = iDelta[i]
        omega_tilde[0][i] = iomega[i]

    delta_Pcoi_PEI = np.zeros(len(voltages))

    for i in range(0,int(number_of_rows),1):

        for j in range(0,3,1):
            delta_Pcoi_PEI[j] = delta_tilde[i][j]
            num = j
            Pei = PEI(Ybus,num,voltages,delta_Pcoi_PEI)
            PCOI= Pcoi(Ybus,voltages,delta_Pcoi_PEI,Pm)
            k1_d = h*omega_tilde[i][j]
            k1_w = h*(1/M[j]*((Pm[j]-voltages[j]**2*np.real(Ybus[j][j]))-Pei)-1/MT*PCOI)
            k2_d = h*(omega_tilde[i][j]+k1_w/2)
            k2_w = h*(1/M[j]*((Pm[j]-voltages[j]**2*np.real(Ybus[j][j]))-Pei)-1/MT*PCOI)
            k3_d = h*(omega_tilde[i][j]+k2_w/2)
            k3_w = h*(1/M[j]*((Pm[j]-voltages[j]**2*np.real(Ybus[j][j]))-Pei)-1/MT*PCOI)
            k4_d = h*(omega_tilde[i][j]+k3_w/2)
            k4_w = h*(1/M[j]*((Pm[j]-voltages[j]**2*np.real(Ybus[j][j]))-Pei)-1/MT*PCOI)

            delta = (k1_d+2*k2_d+2*k3_d+k4_d)/6
            omega = (k1_w+2*k2_w+2*k3_w+k4_w)/6

            delta_tilde[i+1][j]=delta_tilde[i][j]+delta
            omega_tilde[i+1][j]=omega_tilde[i][j]+omega

    return delta_tilde,omega_tilde

```

The energy function is defined.

```

def PotentialEnergy(Pmech,Voltages,Ybus,Idelta,Delta):
    by = []
    n = len(Pmech)
    Pi = np.zeros(n)

    for i in range(0,n,1):
        Pi[i]=Pmech[i]*Voltages[i]*np.real(Ybus[i][i])*(Delta[i]-Idelta[i])
    x = np.sum(Pi)*(-1)

    for i in range(0,n-1,1):
        for j in range(0,n,i+1):
            Z = (Voltages[i]*Voltages[j]*np.imag(Ybus[i][j]))*
                (np.cos(Delta[i]-Delta[j])-np.cos(Idelta[i]-Idelta[j]))
                - (Voltages[i]*Voltages[j]*np.real(Ybus[i][j])*
                  (Delta[i]+Delta[j]-Idelta[i]-Idelta[j])/Delta[i]-Delta[j]-Idelta[i]+Idelta[j]))
                *(np.sin(Delta[i]-Delta[j])-np.sin(Idelta[i]-Idelta[j])))
            by.append(Z)
    y = np.array(by)
    B = y
    Z = np.sum(B)

    fullpot = x-Z

    return fullpot

def Kinetic_Energy(Inertia,omega):
    n = len(Inertia)
    by = []

    for i in range(0,n,1):
        Z =Inertia[i]*omega[i]**2
        by.append(Z)
    Y = np.array(by)
    X = np.sum(Y)

    X = X*1/2
    return X

```

Finding the exit point is the first maximum of the fault-on trajectory.

```

## Calculating ExitPoint

for i in range(0,len(delta),1):
    pot = PotentialEnergy(MechPower, Voltages, Post_fault, PostfaultSEP, delta[i])
    x.append(pot)

plt.xlabel('Iterations')
plt.ylabel('Potential Energy')
plt.plot(x)

y = np.array(x)

EP = argrelextrema(y, np.greater)

print(EP)

print(delta[434])

```

Finding the critical clearing time is done by reusing the integration from the fault-on trajectory, the full energy function is used and figuring out where $V(x) = V_{cr}$.

```

j = 0

Vcrit = PotentialEnergy(MechPower, Voltages, Post_fault, PostfaultSEP, CUEP)
print(Vcrit)

Summen = Kinetic_Energy(Mconstant, omega[j])+PotentialEnergy(MechPower, Voltages, Post_fault, D, delta[0])

while Summen <= Vcrit:
    j = j+1
    Summen = Kinetic_Energy(Mconstant, omega[j])+PotentialEnergy(MechPower, Voltages, Post_fault, D, delta[j])

print(j*h)

```