Protection and DC Fault Location

For MT-HVDC Transmission Systems

Pieter Michiel Rogier Gommers

Delft University of Technology

TU Delft

Protection and DC Fault Location

For MT-HVDC Transmission Systems

by

Pieter Michiel Rogier Gommers

Pieter Gommers 4923189

Instructor:	Prof. Dr. Ir. Marjan Popov
Thesis Commitee:	Prof. Dr. Ir. Marjan Popov & Dr. Ir. Jianning Dong
Teaching Assistant:	Dr. Ir. Le Liu
Project Duration:	December, 2023 - Juli, 2024
Faculty:	Faculty of Electrical Engineering, Mathematics & Computer Science, Delft University
Cover:	Render of Offshore 2GW HVDC Converter (TenneT: https://www.tennet.eu/about-tennet/innovations/2gw-program).

TU Delft

Preface

This thesis rounds up my MSc programme in Electrical Power Engineering. Since the beginning of the BSc degree in Electrical Engineering, I have been interested by how electrical energy is generated, distributed, transmitted and consumpted. My interest in the energy transition and how electrical power systems play a great role in this transition motivated me to delve deeper into this research area, with the hope of contributing to a better understanding of modern electrical transmission systems.

I would like to extend my sincere gratitude to my supervisor Prof. Marjan Popov, for his unwavering support and guidance throughout this research project. His expertise has been instrumental during the project, especially in the moments I found myself being stuck. I am also grateful for the help and support provided by Dr. Le Liu, who allocated time for me whenever was needed.

Conducting research has both been challenging and rewarding. Because of the impossibilities of the chosen simulation platform, this project has taught me how to enhance my problem-solving skills. This makes me satisfied with the obtained results.

Concluding this thesis, I look forward to the first steps in my professional journey. I hope that the conducted research not only remains academic but also inspires further applicable studies as the topic of the protection of HVDC is of utmost importance.

Pieter Michiel Rogier Gommers
Delft, July 2024

Abstract

This thesis explores a Travelling Wave (TW) parameter fitting-based approach for both fault location and protection in a multi-terminal High Voltage Direct Current (HVDC) network. Utilizing a three-terminal HVDC network model within a Real-Time Digital Simulator (RTDS) environment, a custom parameter fitting control component is developed using the Adaptive Multi-step Levenberg-Marquardt (AMLM) algorithm.

For fault location, the methodology analyzes the line-mode backwards travelling voltage wave (Vb1) following an internal DC cable fault. Despite the successful integration of the AMLM parameter fitting algorithm into the real time environment, the achieved average absolute error of 9.80% (29.4 km) for faults spaced 50 km along a 300 km cable was above the acceptable threshold, rendering the proposed method impractical for precise fault location. Improvements through specific signal truncation and optimal cable parameter selection reduced the error to 5.15% (15.45 km), which remains insufficient for practical applications.

In contrast, the TW parameter fitting method proves highly effective for fast and fully selective protection. The protection scheme accurately discriminates between internal and external faults using the Vb1 signal and determines fault types through Vb0 signal analysis. Extensive testing revealed a fault detection rate of 100%, with an overall accuracy of 99.91% for fault resistances up to 200Ω. Severe internal faults are isolated in 1.68 milliseconds, while non-severe internal faults are typically isolated in 3.84 milliseconds. The method completely eliminates relay deadzone, providing robust performance even under noisy conditions with an accuracy of 99.85% for faults with an impedance up to 50Ω.

These findings highlight the potential of the TW parameter fitting approach to significantly enhance the reliability and promptness of fault isolation in HVDC systems, while offering insights into the challenges and limitations of fault location accuracy on a real time platform.

Contents

Nomenclature

Abbreviation	Definition
AC	Alternating Current
AMLM	Adaptive Multi-step Levenberg-Marguardt (algorithm)
CSV	Comma Seperated Values
DC	Direct Current
DCCB	Direct Current Circuit Breaker
HIF	High Impedance Fault
HVAC	High Voltage Alternating Current
HVDC	High Voltage Direct Current
LIF	Low Impedance Fault
MMC	Modular Multilevel Converter
MT-HVDC	Multi-Terminal High Voltage Direct Current
NTG	Negative pole To Ground
PTG	Positive pole To Ground
PTP	Pole To Pole
RTDS	Real-Time Digital Simulator
SNR	Signal to Noise Ratio
TAE	Total Average Error
THR	Threshold
TSO	Transmission System Operator
TW	Traveling Wave
VSC	Voltage Source Converter

1

Introduction

Although most of the population takes it for granted, electricity is the second most used form of final energy consumption in Europe [1], after gas and petroleum products. Driven by European Union policies such as *The Green Deal*, the share of electricity in both the primary energy mix and final energy consumption will only increase in the future [2]. Among other renewable energy sources, wind energy is said to be the main source of Europe's power demand by 2050, mainly from offshore wind parks [3]. These trends drive offshore wind projects all over Europe. In particular, large-scale developments in the north-sea, where capacity targets increase from 65GW by 2030 to 150GW by 2050 [4], [5].

This bulk power is not needed offshore and must be transported to shore over long distances. Traditional high-voltage alternating current (HVAC) systems are not the preferred option for this power transfer. Apart from technological limitations, such as increased reactive power in subsea cables, HVAC systems are not economically the most viable option [6]. Instead, modern voltage source converter-based high-voltage direct current (VSC-HVDC) systems are gaining popularity for offshore bulk power transfer. As seen in Figure 1.1, the initial investment cost for HVAC is lower but catches up with the HVDC investment cost as the cable length increases. This is because an HVDC converter station is more expensive than an HVAC station due to the converters, although transmission lines are less expensive because there are only two conductors, instead of three. Furthermore, HVDC transmission has lower conductor losses, which makes the break-even distance between HVAC and HVDC transmission even lower than 100km. Considering that offshore wind parks or transnational power connections can reach hundreds of kilometers, one can see why HVDC systems are economically more attractive.

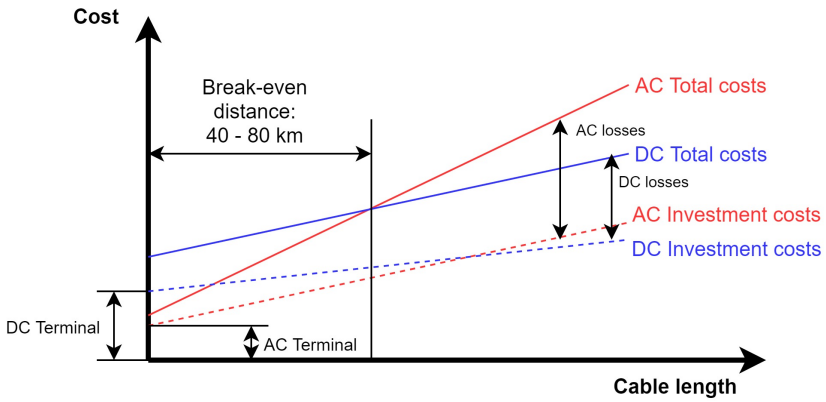


Figure 1.1: HVDC vs. HVAC cost comparison [7]

To increase the reliability of the connections to the wind parks, multi-terminal HVDC (MT-HVDC) grids are planned instead of point-to-point connections. However, this makes the control and protection

of the system complex. When a DC fault occurs on a point-to-point system, there are two strategies to clear the fault [8]. The first is to apply AC circuit breakers that can be used to isolate the fault. This action brings the whole system to a halt. However, this action is easy to perform because there is a lot of experience in operating AC circuit breakers, and is a well-known technology in power systems. The second one, is applying DC circuit breakers (DCCB) to isolate the fault by disconnecting the faulted DC line from the converters and keeping the converters energized from the AC side. In the case of a point-to-point system, both methods can successfully isolate the fault, and the power through the system will drop to zero. Apart from the added cost and complexity of the DCCB, there is no grave difference between the two methods. In the case of HVDC grids with multiple interconnected converter stations, the opening of all circuit breakers on the AC sides of the system to clear a DC fault would imply that not only the faulted DC line but also the whole DC grid is taken out of operation, including all the healthy parts. This is highly undesired, as there will be no power from offshore wind parks connected to the HVDC grid, which could be several GW's of power. It is not an unusual scenario that the onshore AC grid cannot remain stable from this power loss, especially for higher power losses and lower inertia AC systems [9]. To increase the level of system reliability and protection selectivity, MT-HVDC protection must be implemented using DCCBs.

Keeping the MT-HVDC system secure and operational is of great importance. For the transmission system operator (TSO) to restore the system after the event of a DC line fault, the fault location is twofold. From a system perspective, it is necessary to detect on which cable the fault is located. Using this information, the fault can be isolated and the rest of the system can resume its operation without the faulty part almost immediately. However, the faulted cable section must be repaired as soon as possible. Considering the cables are subsea and hundreds of kilometers long, it is unfeasible for the TSO's maintenance crew to visually find the fault location on the cable. Instead, an integrated fault locator is needed, which can estimate the fault based on measured signals within several kilometers of the fault location. This drastically reduces the search area and positively impacts the repair time and system restoration.

The possible reason for DC short circuit faults is insulation failure [10]. Factors such as the aging of cables or mechanical damage can compromise the insulation, which is used to isolate the conductor from its surroundings. This compromised insulation creates a path for the current to flow to the ground, leading to DC faults. If these faults are not detected promptly and accurately, they can result in substantial disruptions to system operation, thereby increasing the risk of widespread blackouts. Selective protection is essential for the reliable operation of the MT-HVDC system. Selectivity is achieved when the protection system can identify the faulted transmission line section correctly, and operate the respective DCCBs. Because the VSC-HVDC system has low damping and inertia, the fault current increases rapidly during a DC line fault [11]. The fault current withstand capability of the VSC-MMC converters is low, typically twice the nominal current rating of the converter [12]. To avoid damage to the converter and keep the fault current below the maximum interruptible current of the DCCB, the fault current interruption must be quick, in the order of milliseconds. Typically, the fault classification should be done within 3ms, after which it takes another 3ms for the DCCB to break the current [13]. It is known for DC line protection that high impedance faults (HIF) are particularly challenging to detect and classify. During HIF, the fault impedance keeps the voltage at a low value though significantly higher than zero, and reduces the fault current. As most protection algorithms are based either on a sudden increase in current or a decrease in voltage, this poses difficulties that are yet to be solved. However, especially for cable-based systems, HIF does not occur often as cable faults are mostly direct connections to the grounded cable sheath, resulting in a low fault impedance.

1.1. Literature Review

1.1.1. State of the Art DC Fault Location

The current state-of-the-art research on fault location mainly considers four types of methods including the natural frequency method, fault analysis method, active injection method, and traveling wave-based method. The natural frequency method is based on the relationship between the natural frequency of the travelling wave, fault distance, and reflection coefficient at the transmission line terminal [14]. A 10 ms current data window is used to perform multiple signal classification for extracting the dominant natural frequency. With this, the traveling wave velocity and reflection coefficient are calculated for

fault location estimation. A similar approach is done in [15] and [16], however, the natural frequency is obtained by the PRONY and MUSIC algorithm, respectively. These methods require a high sampling frequency (50 - 100 kHz) and fail to detect faults close to the measurement point. All of them are however single-ended, which means no communication is required with other measurement points. In fault analysis-based methods voltage and current traveling waves are used to estimate transmission line parameters up to the fault point. In [17], the current TW signal is reconstructed using the least squares algorithm for parameter fitting. From this, the line inductance is estimated resulting in the fault location. In [18], a simplified R-L representation is used to estimate the R-L values of the fault, resulting in the location. Both methods require a lower sampling frequency than other fault location methods. However, because the computations are based on the aggregated model of transmission lines, the traveling wave propagation is not taken into account. Hence, the fault location estimation accuracy is not guaranteed for long transmission lines. The active injection-based method relies on the measured response after injecting specific signals in the faulted transmission line at the line terminal [19]. In [20], the healthy pole MMC injects DC voltage perturbations, which result in induced characteristics in the faulted line. These contain information about the fault properties when analysed by the wavelet transform. In [21], the single-frequency signal is injected in the faulted pole. Parameter identification is performed on the measured response which provides the location estimation. Generally, the active injection-based method accuracy is prone to be affected by the fault resistance and DC inductor size. Traveling wave-based methods are typically defined as reliable and high precision. By applying the telegrapher's equations mathematical relations can be established among the propagation time, propagation velocity, and fault distance. Double-ended methods as presented in [22] and [23] show good performance, however, they need very accurate signal measurements and synchronized communication channels. This needs more investments due to additional communication networks and also poses a risk of data corruption during synchronization. Single-end methods only use local measurement data to avoid communication shortcomings. The method presented in [24] measures time delays in TWs, requiring a very high sampling rate of 1MHz. In [25] fault information is extracted from the measured traveling wave through parameter fitting, however, it requires a high sampling rate. The work presented in [26] seems promising, as it provides a TW parameter fitting-based method utilizing a relatively low sampling frequency. This work is taken as inspiration for this thesis work.

1.1.2. State of the Art MT-HVDC Line Protection

Even though MT-HVDC systems are relatively new, extensive research has been conducted in the protection of transmission lines regarding DC faults. In practice, DC line protection can be classified into two categories; Unit protection and non-unit protection.

Unit protection, also known as pilot protection, depends on communication channels to transmit measurement data between relays that are located at each end of a transmission line. Various methods for unit protection have been proposed. In [27], the Hausdorff algorithm is applied to compare the similarity of current waveforms sampled at both ends of the line following a DC fault. In [28], the difference in currents measured at two different points is taken to identify the DC fault. The main advantage of unit protection is that selectivity can be guaranteed. However, these methods need precise and synchronized measurement data and a strong communication network. For long-distance transmission, this is difficult and costly to achieve. Only the communication latency can cause enough delay to fail the protection speed necessary for DC line faults.

In contrast, non-unit protection refers to protection without communication and relies on faulted line identification with only local measurement values. In existing research, three main categories of non-unit protection are defined time domain TW-based, DC inductor voltage-based, and time-frequency-based. Time domain TW-based protection relies on the analysis of the measured traveling wave voltages at currents at the relay locations. In [29], the rate of change of voltage (ROCOV) is used to design a fault detection scheme. The fault type identification is improved in [30] by using the ratio of transient voltage (ROTV). In [31], a combination is used based on under-voltage, DC voltage derivative, and directional over-current criteria to design the protection scheme. According to [32], practical HVDC systems in China utilize the derivative of decoupled waves, or voltage and current TW protection. Generally, time domain TW-based protection eliminates dead zones (fault zones undetectable by the protection) and offers fast detection and classification (<3ms). However, as they rely on TW amplitude, these methods often face insensitivity to HIF and are susceptible to noise interference.

DC inductor voltage-based protection methods are based on large inductors installed at specific loca-

tions in the MT-HVDC grid. The main use of these inductors is to limit the rate of change in fault current, giving the DCCB more time to operate. When placed at both ends of a DC transmission line, they also form the boundary elements of that line. There are multiple ways to utilize this. In [33], the change rate of the reactor voltage is used to design a protection scheme. It is robust to fault resistance variations, however, it is not able to detect single-pole-to-ground faults. The voltage across the reactors from both poles is used as the main detection in [34]. This method is fast (<2ms), but fails to detect faults under HIF. In [35], modal analysis is performed to use the line-mode inductor voltage for fault detection and the zero-mode voltage for fault type discrimination. Compared to time-domain TW-based methods, reactor-based methods are robust to noise interference and require a lower sampling frequency. However, threshold value selection is difficult, as dead zones occur at the ends of faulted transmission lines. Sensitivity during HIF can also still be improved.
Next to limiting the rise in fault current and acting as boundary components, the DC inductors also attenuate specific frequency bands more than others, acting like a filter. The presence of high-frequency components in measured signals can identify faults selectively. The signal analysis in the frequency domain forms the basis of time-frequency analyses-based protection. Extensive research has been done on this topic. Fourier transform methods proposed in [36] (SFFT) and [37] (FFT) have a fast processing speed, but are limited in performance due to fixed window lengths and noise interference. The Hilbert Huang Transform (HHT) proposed in [38] is used to decompose measurement data in various datasets of different frequencies. Although a good intrinsic performance, the noise interference renders this method less suitable for industrial applications. Of the different time-frequency methods, the Wavelet Transform (WT) shows the best overall performance, as it has the best noise filtering capability. In [39], the Continuous Wavelet Transform (CWT) is used for the current TW analysis. However, it requires a very high sampling rate of 2MHz. Other WT methods, such as [40], face difficulty with fault type discrimination and HIF detection: maximum of 200Ω.

1.2. Problem Definition and Scope

1.2.1. Fault Location

As becomes apparent in subsection 1.1.1, there are several methods to estimate the location of a DC line fault. The methodology applied in this thesis is strongly inspired by the work presented in [26]. In this work, DC line faults are simulated in the RTDS/RSCAD environment, after which the sampled line-mode backward travelling wave V_{b1} data is manually exported to MATLAB for analysis. The data is sampled with a time-step of 50 µs, resulting in a sampling frequency of 20 kHz. In MATLAB the Adaptive Multi-step Levenberg-Marquardt (AMLM) algorithm as presented in [41] is implemented which performs parameter fitting on the sampled V_{b1} data. Even with the addition of 40 dB of noise estimation errors are below 1.00 % for all simulated fault cases, where attention is paid to different fault types, locations, and resistances.

To extend the work presented in [26], the analysis part that is executed in MATLAB will be implemented and fully integrated into the RTDS / RSCAD environment. The goal is to remove human intervention from the location estimation and to conduct the analysis in real time. This should provide the system operator with an almost instantaneous fault location estimation following a DC line fault. The fault locator will comprise the parameter fitting algorithm, an automatic fault detector, and the final location computation. The parameter fitting algorithm will be translated from MATLAB code to C code and created as an RSCAD component using the CBuilder environment.

Research Question 1: *Can the proposed fault location method using the AMLM parameter fitting algorithm be successfully integrated into the RTDS / RSCAD environment?*

> The AMLM algorithm is an iterative process. Although it is highly optimised, it is still relatively computationally expensive, consisting of multiple matrix multiplications and inversions per iteration. This means that the execution of the parameter fitting component takes quite a bit of time. A drawback of real-time simulation is that all computations belonging to a time-step have to be finished in time before the RTDS computer can move on to the next simulation step. Therefore, the parameter fitting algorithm cannot be done in RSCAD's *small-timestep* environment. If this means that sampling can not be executed in the *small-timestep* either, this might severely influence the location estimation accuracy in a bad way.

Research Question 2: *Is the interventionless and instantaneous location estimation worth the loss in accuracy?*

> As stated in RQ1, the real time integration of the fault locator will likely lead to a loss of accuracy. Even if the possibility of successful operation of the fault locator in RSCAD is proven, it depends if this implementation is worth it. After all, for a post-mortem fault location analysis computation speed in the range of milliseconds is not a *must* have, as it will take days, weeks, or even months to do repair work on the damaged submarine cables.

The research will be executed on a single 300km long subsea cable in a three-terminal HVDC network model in RSCAD. Faults up to 1.0Ω will be analysed, which means no care is taken for HIF. A successful integration goal is set to be below 1% error for the location estimation, leading to a maximum deviation of 3km.

1.2.2. Protection

Non-unit protection is necessary due to the required fault-clearing speed. This means that fault detection and classification must happen exclusively by local measurements. As a fault can occur anywhere on the system, the protection relays must be able to classify faults as either internal or external. Each cable will have two DCCBs, one on each end of the line. The DCCB location is also the measurement location for the input signals for the protection relay.

Research into fault location indicates that the V_{b1} signal contains a substantial amount of fault data. Therefore, it is hypothesized that this signal can be used to identify external faults from internal faults. When large boundary inductors are installed at the ends of each transmission line, the response is attenuated by these inductors, and the HVDC converters. Based on the difference caused by this attenuation, backward travelling fault responses can be differentiated. Forward travelling fault responses from the neighbouring transmission lines are not present in the V_{b1} signal as it only contains backward TWs. This facilitates an even more straightforward differentiation. The protection method determines the internal or external faults by utilizing parameter fitting results for the V_{b1} signal. The faulty pole identification will be done by utilizing parameter fitting results for the V_{b0} signal.

Research Question 3: *Can the V_{b1} signal be used for fault discrimination using AMLM-based parameter fitting when integrated into the RTDS environment?*

> Due to the boundary components present in the HVDC model there is a difference in internal and external faults, which will be present in the V_{b1} signal. However, the question remains if the parameter fitting results are accurate and diverse enough to correctly and timely differentiate between faults of different types, resistances, and locations. Sizing L_{dc} must be taken into account as its value greatly influences both the rise in fault current and the amount of signal attenuation for external faults. Given that the fault current interruption process must be prompt, sampling and computation time must be limited, presumably leading to less accurate fitting results. This is even degraded additionally by real-time simulation limitations. High impedance faults (HIF) have a great impact on fault response, which will have to be analysed to find the protection boundaries in terms of fault impedance.

Research Question 4: *Is the algorithm and proposed protection method robust enough to account for measurement noise?*

> In practical HVDC systems, a significant level of noise will always be present for measured quantities. For the protection to be reliable, this noise should not adversely impact the relay's operation. To find the impact of noise, gaussian white noise with a level of +40dB SNR will be added to the voltage and current measurements. The parameter fitting results are expected to deteriorate in accuracy due to noise and therefore HIF internal faults are expected to be indistinguishable from low-impedance external faults.

The proposed protection part of the research work comprises only the detection and classification of DC cable faults. For this, a steady state is assumed, meaning that no transients are present during the analysis. Furthermore, interference from other protection or control systems is not investigated. System restoration after a fault will only be part of the project for successive fault experiments and is not deemed realistic. The benchmark model used is considered to be an accurate representation

of a realistic system. Control systems are not changed, although changes to the model will be made consisting of the created protection relays and system verification testing. DCCBs are average models, and therefore considered ideal DCCBs, not ideal switches. Successive faults (one fault leading to the next one) are out of the scope of this project. The detection and classification of a fault are successful if they are completed within 3ms. The fault impedance is taken into account: 10Ω or higher is identified as HIF.

1.3. Thesis Outline

The thesis is structured as follows: Chapter 2 provides a comprehensive background into basic theory regarding travelling waves. These basics comprise of the decoupling of poles (symmetrical component decomposition), mathematical properties of DC faults, and the definition of time domain functions for signals of interest at the measurement locations. In chapter 3 the RTDS technology is briefly discussed. Furthermore, the model used during the research is detailed. The simulation setups are also discussed for better result interpretation in later chapters. In chapter 4 the research into the DC fault location estimation is presented. The methodology is explained and the custom parameter fitting component is explained in detail. The location estimation results are shown, analysed and discussed. In chapter 5 the research into MT-HVDC DC fault protection is presented. The design process and protection scheme are explained. The protection system is tested for performance in correctness and operation time. Also, a deadzone test is performed. Finally, the results are analysed and explained. Chapter 6 concludes the thesis by answering the research questions posed in chapter 1.

2

Travelling Wave Theory

A travelling wave is a transient electrical disturbance that propagates along a transmission line or cable and can be expressed by voltages or currents. It can be caused by several contingencies, such as switching surges, lightning surges, or short-circuit fault-induced surges, and it moves away from the faulty point in both directions. A transient is present in the power system for a short time - in the range of microseconds - after which the system reaches a new steady state [42]. To properly model transmission lines considering high-frequency transient signals, distributed elements are used instead of lumped elements. This gives rise to travelling waves, in which it takes time for a signal to propagate over the length of the transmission line due to the electrical components present. During propagation, the signal gets attenuated due to losses along the transmission line, which means that the amplitude of the voltage and current travelling waves is exponentially decreased.

2.1. Directional Travelling Waves

To describe a travelling wave, it is important to analyse the medium over which it propagates. The distributed parameter model is a basic model of a transmission line and is shown in Figure 2.1. It consists of many segments with small lengths Δx, each described by resistance R, inductance L, conductance G, and capacitance C per unit length. The voltage $u(x,t)$ on and the current $i(x,t)$ through the transmission line are functions of both time t and position x [43]. These functions are known as *telegraphers' equations*.

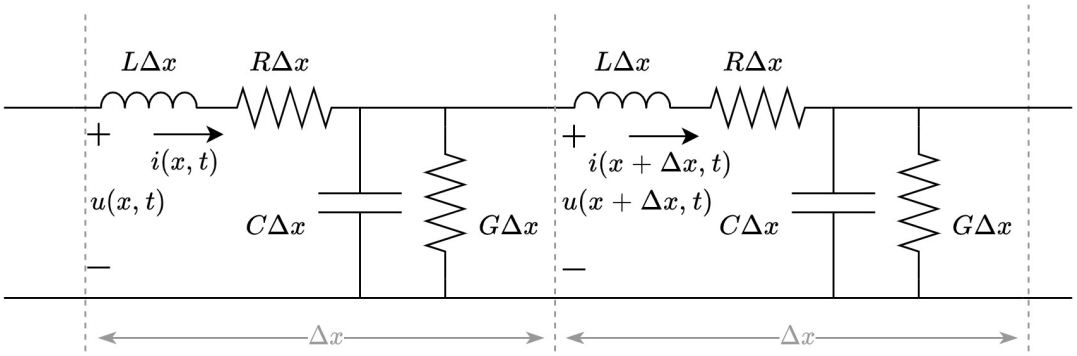


Figure 2.1: Distributed parameter model of a transmission line

From Figure 2.1 the following equation can be obtained:

$$\begin{cases} u(x,t) - u(x+\Delta x,t) = \left(L\dfrac{\partial i(x,t)}{\partial t} + Ri(x,t)\right)dx \\ i(x,t) - i(x+\Delta x,t) = \left(C\dfrac{\partial u(x+\Delta x,t)}{\partial t} + Gu(x+\Delta x,t)\right)\Delta x \end{cases} \tag{2.1}$$

Considering that the resistance and conductance of a transmission line are small (this holds for a simplified lossless line) and can be neglected, Equation 2.1 can be simplified as:

$$\begin{cases} \dfrac{\partial^2 u(x,t)}{\partial x^2} = LC\dfrac{\partial^2 u(x,t)}{\partial t^2} \\ \dfrac{\partial^2 i(x,t)}{\partial x^2} = LC\dfrac{\partial^2 i(x,t)}{\partial t^2} \end{cases} \tag{2.2}$$

Equation 2.2 is in the form of a standard homogeneous one-dimensional wave equation of which the general solution is given as:

$$\begin{cases} u(x,t) = u_f\big(t - \dfrac{x}{v}\big) + u_b\big(t + \dfrac{x}{v}\big) \\ i(x,t) = i_f\big(t - \dfrac{x}{v}\big) + i_b\big(t + \dfrac{x}{v}\big) \end{cases} \tag{2.3}$$

In Equation 2.3, the subscripts f and b indicate forward and backward waves, respectively. A forward-travelling wave travels along the positive x-axis, while the backward-travelling wave travels in the opposite direction along the x-axis. The wave velocity is a function of the transmission line's inductance and capacitance:

$$v = \frac{1}{\sqrt{LC}} \tag{2.4}$$

The voltage and current are related by the characteristic impedance of the transmission line which is given by:

$$Z_c = \frac{u}{i} = \sqrt{\frac{L}{C}} \tag{2.5}$$

Given this, the following constraint relation holds in Equation 2.3:

$$\begin{cases} i_f(t) = \dfrac{u_f(t)}{Z_c} \\ i_b(t) = \dfrac{u_b(t)}{Z_c} \end{cases} \tag{2.6}$$

From Equation 2.3 it can be concluded that the voltage and current on the transmission line can be regarded as the superposition of the forward and backward travelling wave. Conversely, if the voltage and current at a certain point are known, the forward and backward voltage wave at this point can be calculated by:

$$\begin{cases} u_f(t) = \dfrac{u(t) + Z_c i(t)}{2} \\ u_b(t) = \dfrac{u(t) - Z_c i(t)}{2} \end{cases} \tag{2.7}$$

With this, when simulating, the backwards voltage travelling wave can be computed by measuring voltage and current for the use of travelling wave-based applications.

2.2. Symmetrical Components

Present power systems are mainly three-phase AC systems. Contingencies occurring in the three-phase power system may lead to asymmetrical events, e.g. when a single-phase fault occurs in the system. In 1918, C.L. Fortescue proposed a method for transforming asymmetrical multiphase sequence networks into symmetrical sequences ones, which drastically simplified the computations for power systems by enabling analysis using positive, negative, and zero sequence quantities [44]. The Fortescue transformation is defined by

$$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 \\ 1 & \alpha & \alpha^2 \\ 1 & \alpha^2 & \alpha \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix}, \tag{2.8}$$

in which $\alpha = e^{j\frac{2\pi}{3}}$, $[x_a \ x_b \ x_c]$ are the phase variables and $[x_0 \ x_1 \ x_2]$ are the symmetrical components known as sequence variables. The square root of three is in place to keep the transformation power invariant.

The bipolar DC system used in this thesis can have asymmetrical events as well, e.g. when a single-pole-to-ground fault occurs. However, the pole voltages and currents are coupled. Therefore, it would simplify the analysis when a similar transformation is used. The bipolar DC system can be seen as a two-phase system with zero frequency [45]. By using this, a transformation can be derived from Equation 2.8 by changing α to $e^{j\pi}$ and reducing the system to two dimensions. The resulting transformation and its inverse are

$$\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_p \\ x_n \end{bmatrix}, \quad \begin{bmatrix} x_p \\ x_n \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}, \tag{2.9}$$

in which $[x_p \ x_n]$ are the positive and negative pole variables and $[x_0 \ x_1]$ are the symmetrical components, also known as the zero- and line-mode components, respectively [46]. Note that there is a power conversion relationship in the transformation

$$v_p \cdot i_p + v_n \cdot i_n = v_1 \cdot i_1 + v_0 \cdot i_0. \tag{2.10}$$

If the voltages are balanced, as in the case of normal steady-state operation, then $v_1 = v_p = -v_n$ and $v_0 = 0$. In this case,

$$v_p \cdot i_p + v_n \cdot i_n = v_1 \cdot i_1. \tag{2.11}$$

This implies that the line-mode component represents the combined power transfer of both poles, while the zero-mode describes the interaction between poles caused by an imbalance. Due to the invariant nature of the transformation, the relation between symmetrical and asymmetrical components is 1.

Decoupled equivalent circuits can be derived for both modes, called sequence networks using the transformation. They can be applied for a simplified evaluation of the modes independently and describe the static and dynamic behaviours of the systems under study, especially for fault conditions.

2.3. Initial Voltage at Fault Point

By superposition, a faulted network can be replaced by two symmetrical networks, each analysed by a corresponding sequence network shown in Figure 2.2 [47]. Due to the fast nature of the fault phenomenon, it is enough only to model the local initial voltage, current, and impedance at the fault point to find the initial wave. The positive sequence voltage sources represent the steady-state system voltages.

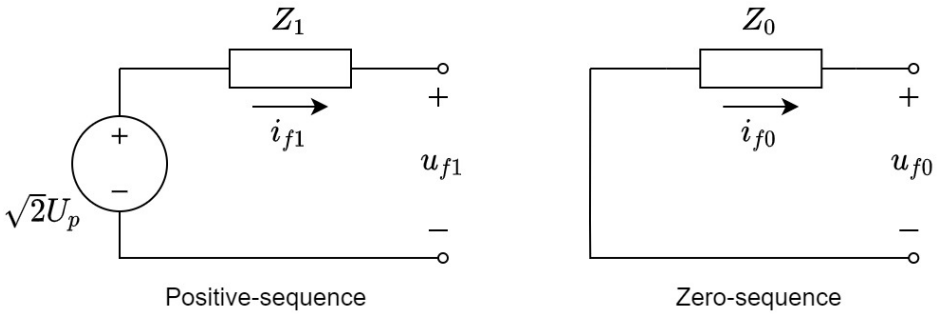


Figure 2.2: Sequence networks

The relationships between voltage and current in Figure 2.2 are given by

$$\begin{aligned} u_1 - i_{f1}Z_1 &= u_{f1} \\ 0 - i_{f0}Z_0 &= u_{f0} \end{aligned} \tag{2.12}$$

In (2.12), Z_1 and Z_0 are the Thévenin-equivalent positive- and zero-sequence network impedances as seen from the fault location.

2.3.1. Single Pole to Ground Fault

The simplified fault circuit for a positive pole-to-ground (PTG) fault is shown in Figure 2.3a. The p and n lines represent an arbitrary bipolar HVDC transmission line. Considering the fault resistance R_f, which is known as a grounding resistance, the boundary conditions are

$$\begin{cases} i_{fn} = 0 \\ ufp = i_{fp}R_f \end{cases} \tag{2.13}$$

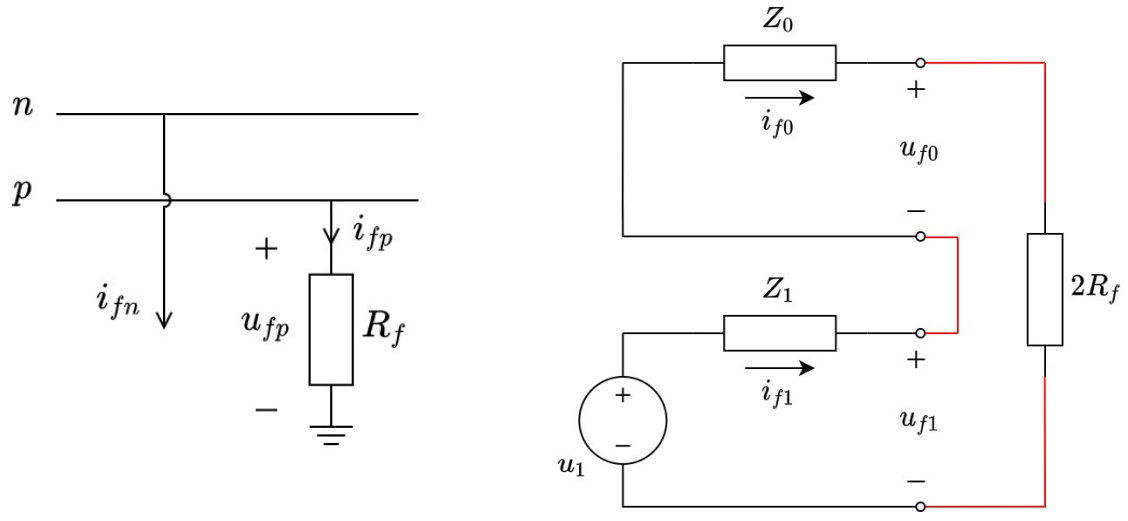
Next, the sequence components are

$$\begin{cases} i_{f0} = i_{f1} \\ u_{f0} + u_{f1} = (i_{f0} + i_{f1})R_f \end{cases} \tag{2.14}$$

Together with (2.12), the fault-induced voltage sequence components are computed as

$$\begin{cases} u_{f0} = \dfrac{u_1 Z_0}{Z_1 + Z_0 + 2R_f} \\ u_{f1} = \dfrac{u_1(Z_0 + 2R_f)}{Z_1 + Z_0 + 2R_f} \end{cases} \tag{2.15}$$

The result in (2.15) also follows from the analysis of the sequence network connection shown in Figure 2.3b. Here, the positive- and zero-sequence networks are connected in this way because the fault is single-phase [48]. The analysis for a negative pole-to-ground (NTG) fault is similar because of the symmetrical nature of the bipolar HVDC system.



(a) Fault circuit

(b) Sequence network connection

Figure 2.3: Positive pole to ground fault

Finally, the transient sequence component is superimposed on the pre-fault steady-state condition, and in this way, the condition after the fault can be analysed. $u_1 = \sqrt{2}V_f$ where V_f equals the steady-state nominal voltage of the faulted pole (in this case: $V_f = V_p$). Because the fault brings the voltage close

to zero, the change in voltage is considered negative. This results in the following PTG fault-induced voltage change:

$$\Delta u_{f0} = u_{f0} - 0 = \frac{-\sqrt{2}V_f Z_0}{Z_1 + Z_0 + 2R_f} \tag{2.16}$$

$$\Delta u_{f1} = u_{f1} - u_1 = \frac{-\sqrt{2}V_f Z_1}{Z_1 + Z_0 + 2R_f} \tag{2.17}$$

2.3.2. Pole to Pole Fault

The simplified fault circuit for a pole-to-pole (PTP) fault is shown in Figure 2.4a. Considering the fault resistance R_f, the boundary conditions are

$$\begin{cases} i_{fp} = -i_{fn} \\ ufp - u_{fn} = i_{fp} R_f \end{cases} \tag{2.18}$$

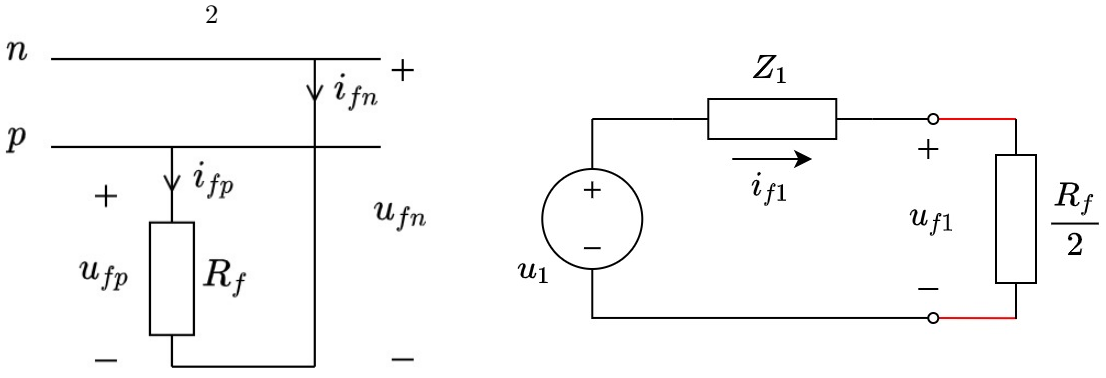
Using (2.9), the instantaneous sequence components are

$$\begin{cases} i_{f0} = 0 \\ 2u_{f1} = i_{f1} R_f \end{cases} \tag{2.19}$$

Together with (2.12), the fault-induced voltage sequence components are computed as

$$\begin{cases} u_{f0} = 0 \\ u_{f1} = \dfrac{u_1 R_f}{2Z_1 + R_f} \end{cases} \tag{2.20}$$

The result in (2.20) also follows from the analysis of the sequence network connection shown in Figure 2.4b. Here, only the positive-sequence network is connected in this way because the fault is a phase-to-phase [48] (positive to negative pole). Due to the absence of ground in the fault, the zero-sequence is not used.



(a) Fault circuit **(b)** Sequence network connection

Figure 2.4: Pole to pole fault

Finally, the transient sequence component is superimposed on the pre-fault steady-state condition so that the condition after the fault can be analysed. $u_1 = \sqrt{2}V_f$ where V_f equals the steady-state nominal voltage of the faulted pole (in this case, it is taken: $V_f = V_p$). Because the fault brings the voltage close to zero, the change in voltage is considered negative. This results in the following PTP fault-induced voltage change:

$$\Delta u_{f0} = u_{f0} = 0 \tag{2.21}$$

$$\Delta u_{f1} = u_{f1} - u_1 = \frac{-2\sqrt{2}V_f Z_1}{2Z_1 + R_f} \tag{2.22}$$

2.4. Propagation Characteristics

In contrast to the simplification made for the lossless line in Equation 2.2, the lossy component of a transmission line is important to be taken into account as it attenuates the signal concerning the traveled distance. At the initial stage of propagation of the backward voltage TW, the measured TW at the beginning of the transmission line is described in spectral S-domain as:

$$u_b(0,s) = e^{\gamma(s)x} u_b(x,s) \tag{2.23}$$

where $u_b(x,s)$ describes the TW at the fault location x and $e^{\gamma(s)x}$ the attenuation effect [49]. This attenuation effect can be expressed as [50], [26]:

$$e^{\gamma(s)x} \approx \frac{1 - k_a \frac{x}{l}}{(1 + sT_a)} e^{-s\frac{x}{v}} \tag{2.24}$$

where $e^{-s\frac{x}{v}}$ denotes the phase-change characteristics of TWs. T_a is the dispersion time constant, and it is highly dependent on, and proportional to distance. It is further defined as $T_a = \tau x$, where τ is a known cable parameter and x is the fault distance. Constant k_a denotes the amplitude attenuation per unit length:

$$k_a = \mathrm{Re}\left[\sqrt{(r_0 + j\omega L_0)(g_0 + j\omega C_0)}\right] \tag{2.25}$$

where r_0, g_0, L_0 & C_0 are the per-unit parameters of the distributed parameter model of the transmission line, as shown in Figure 2.1.

2.5. Voltage at Relay Unit for Internal Faults

The objective of this section is to find analytical expressions for the backward line-mode voltage travelling wave (V_{b1}) at the relay units R_{12} & R_{21} for faults on Cable12. These types of faults are considered to be internal, as they are on the cable that is protected by the relays. To identify proper signal labels, it is assumed that an internal fault takes place on Cable12 as shown in Figure 2.5. Considering the steps taken in Sections 2.1 and 2.2, the general line-mode backward voltage TW can be calculated in spectral S-domain as:

$$u_{b1}(s) = \frac{u_1(s) - Z_{c1}(s) i_1(s)}{2} \tag{2.26}$$

In Figure 2.5, the fault is denoted by f_{12} because it is on the cable that connects MMC1 and MMC2. With the relay R_{12} as reference, taking the voltage at the fault point into account from section 2.3 and using Equation 2.24 where the fault distance l is substituted for x, the fault induced TW is labelled $V_{b1,12}$ and expressed as:

$$\begin{aligned} V_{b1,12}(s) &= \frac{\Delta v_{F1}}{s} e^{\gamma(s)x} \\ &= \frac{\Delta v_{F1}}{s} \left(\frac{1 - k_a l}{1 + s \cdot \tau l} e^{s \cdot \frac{l}{v_1}} \right) \\ &= A_0 \left(\frac{1}{s} - \frac{1}{s + 1/\tau_0} \right) e^{-s \cdot T_{d0}}. \end{aligned} \tag{2.27}$$

By applying the inverse Laplace transform, the time-domain expression for $V_{b1,12}$ is obtained:

$$V_{b1,12}(t) = A_0 \big(1 - e^{-\frac{t - T_{d0}}{\tau_0}}\big) u(t - T_{d0}) \tag{2.28}$$

The coefficients in Equation 2.27 and 2.28 can be calculated by:

$$A_0 = \Delta v_{F1}(1 - k_a l), \quad \tau_0 = \tau l, \quad T_{d0} = l/v_1; \tag{2.29}$$

Assuming the system is in a steady state before the fault occurs, the incoming wavefront of the backward TW monitored at the relay unit is not refracted or reflected by the line boundaries. Therefore, it is

the preferred component to study the initial fault characteristics. Additionally, because this is the first available information on the fault, analysing it should result in the fastest reaction possible - depending on processing speed - which is important for DC fault detection. As can be seen by the coefficients in Equation 2.29, the incoming wave contains information about the fault type, impedance, and location. To avoid losing fault information due to distortions caused by reflections, only the wavefront of V_{b1} is of interest, which results in the sampling window shown in Figure 2.5 as t_s.

The line-mode propagation velocity v_1 is a known transmission line parameter, which determines the propagation time T_{d0}. However, it is not possible for a relay to accurately determine the time instant of the fault; therefore making this coefficient incapable of estimating the fault location. As τ_0 and A_0 are coefficients that are possible to fit the parameters for, it is theoretically possible to use a parameter fitting algorithm to estimate these coefficients and calculate the fault distance, type, and impedance.

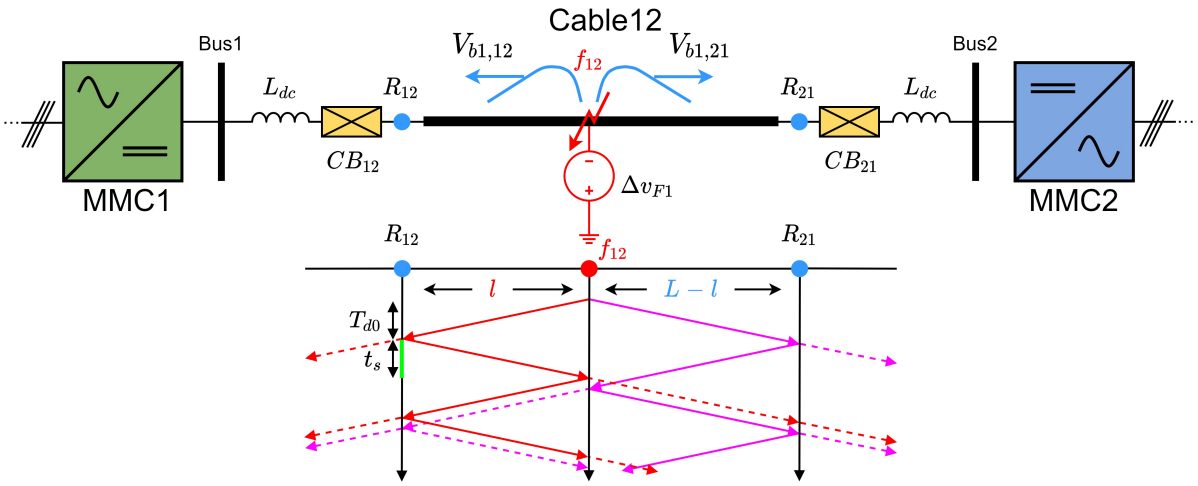


Figure 2.5: Faulty Cable12 and TW propagation (lattice diagram).

2.6. Voltage at Relay Unit for External Faults

The objective of this section is to discuss the backward line-mode voltage travelling wave (V_{b1}) at the relay units R_{12} & R_{21} for faults not on Cable12, i.e. on neighbouring cables. These faults are considered external, as they are not on the cable protected by the relays. Two main types of external faults are considered. In Figure 2.6, relay R_{12} is taken as a reference. Note the small black arrow below the relay that indicates the looking direction.

The first external fault (in green) occurs on Cable13. In this case, the fault response TW travels through Bus1 towards R_{12}. Based on the direction of propagation with respect to R_{12}, this fault TW is identified as a forward travelling wave, as the solid green arrow is in the looking direction of R_{12}. As such, the backward TW V_{b1} measured at R_{12} does not exist yet for this fault, and will theoretically remain unchanged at the steady-state level. For R_{12}, fault response TWs caused by faults on Cable13 are called *forward external faults*, as the response TW is a forward TW. The fault itself does not travel or propagate, but this is named as such for documentation purposes.

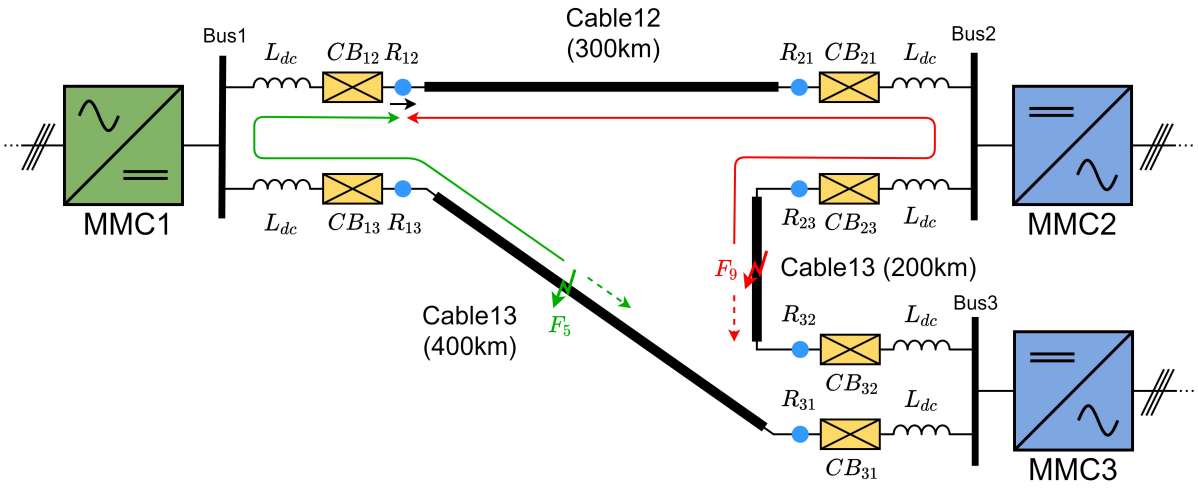


Figure 2.6: TW propagation for faults on neighbouring cables.

The second type of external fault as seen for R_{12} is when the fault (in red) occurs on Cable23. The fault-induced TW will first travel to bus 2, where it is partly split into a wave proceeding to the MMC2 and another one traveling along Cable12. From R_{21}, the response follows the same path as an internal fault. This fault response is called a *backward external fault*, as the TWs propagation direction is opposite of the looking direction of R_{12}. The possibility of discriminating between internal and external follows from the additional attenuation provided by the line-end inductors L_{dc} and the converter. These will smoothen the fault response, making the amplitude difference caused by the fault less severe and the rate of change is reduced. Finding the analytical expression for this function is superfluous for this research project, as the protection relay can only perform parameter fitting for a single function. Practically, this is thus done for the approximated V_{b1} signal following an internal fault, as given in Equation 2.28.

3

DC Grid System Model

3.1. Real Time Simulation Environment

The work presented in this paper is based solely on simulations performed on a Real-Time Digital Simulator (RTDS). This simulator is a powerful computer that is designed particularly to run electrical power system simulations. These simulations are performed in real time, which means that the simulation time is equal to the designated time of the case to be simulated. This is different than the off-line simulations, in which the duration may last much longer than the duration of waveforms that the user wishes to simulate. For instance, 1 second in simulation can take several minutes to compute, depending on the model complexity. Computing simulations in real time provides opportunities to interface physical components, such as hardware control systems or protection relays. Such components can therefore be tested realistically, given that RTDS simulates an electrical grid as it is in real life.

Before a simulation can start, a model must be created. This is done in RSCAD, the accompanying RTDS software. RSCAD provides a draft environment in which it is possible to build an electrical network by interconnecting components found in a large library, ranging from simple RCL branches to detailed models of MMC converters. Custom components can be made using CBuilder. Control systems for these components can also be designed. The runtime environment is used for interaction with the simulation as it is running. The RSCAD software is installed on a personal computer. This computer is connected to RTDS using a VPN connection over the internet. When a simulation is launched from RSCAD, a simulation package is sent to RTDS where it will start. The runtime environment provides set switches, push buttons, and data collection from the simulation over the VPN connection.

Digital simulation has a discrete timestep. From one time step to the next, the network model is solved so that new voltage and current values are computed for all nodes and branches. For real-time simulation, this means that all computations for the next timestep must be completed in the current timestep, which for RTDS is in the range of 50 µs. This results in a high computational burden, which is why the RTDS used in this project is the NovaCor rack equipped with a 7 core processor. If there is no sufficient computing power, multiple racks can be used in unison. Even with multiple racks, real-time simulation imposes limits on what a single rack can accomplish, as it is impossible for some tasks to be divided over multiple racks. These limits will be found and tested in this thesis project. Some components (e.g. VSC-HVDC converters) of an electrical power network model need a much smaller timestep to accurately simulate transient behavior. This can be done with RSCAD's small-timestep functionality, which can bring the update rate down to the range of 1 µs, or even lower if necessary. The part that is simulated using small-timestep must be relatively simple, as its computations need to finish within the respective current small-timestep.

During simulation, a user can interface with RTDS using the runtime environment. Faults can be applied by setting its location and clicking a button. In this project, exhaustive simulation cases are performed, which means that many different fault scenarios are tested. Variables like fault type, location and impedance are changed, as well as test cases for different component settings, and protection scheme threshold values. Many of these will be repeated several times, as RTDS will always give a slightly

different result based on initial conditions. This means that over the course of the project, tens of thousands of fault cases are simulated. This is a cumbersome task, considering that it takes about 10 seconds for the system to restore to a steady state after an experiment. To avoid manual experiment execution, the runtime scripting functionality of RSCAD is used. It provides an environment in which scripting in a C-like language can be performed to automate user tasks such as starting/stopping a simulation, changing component parameters, applying a fault, and exporting simulation data. As the research is data-driven, this functionality is needed and will be used extensively.

3.2. HVDC grid modeling

Developing a model for an HVDC system is a task of considerable complexity, and therefore, this is outside the scope of this project. However, a comprehensive model is necessary to simulate DC line faults accurately. Hence, one of the benchmark HVDC models presented by the authors of [51], [52] is used for the research presented in this thesis. The model represents a three-terminal 2GW HVDC system, as shown in Figure 3.1. Each VSC-MMC converter converts 2GW of power from AC to DC and vice versa. The technology used is a bipole arrangement with a dedicated metallic return. The nominal pole voltages are ±525kV. The system can be divided into three subsystems: the onshore 400kV AC system, the DC system, and the offshore 66kV AC system. On the positive pole of Cable 1, two average model DCCBs are installed.

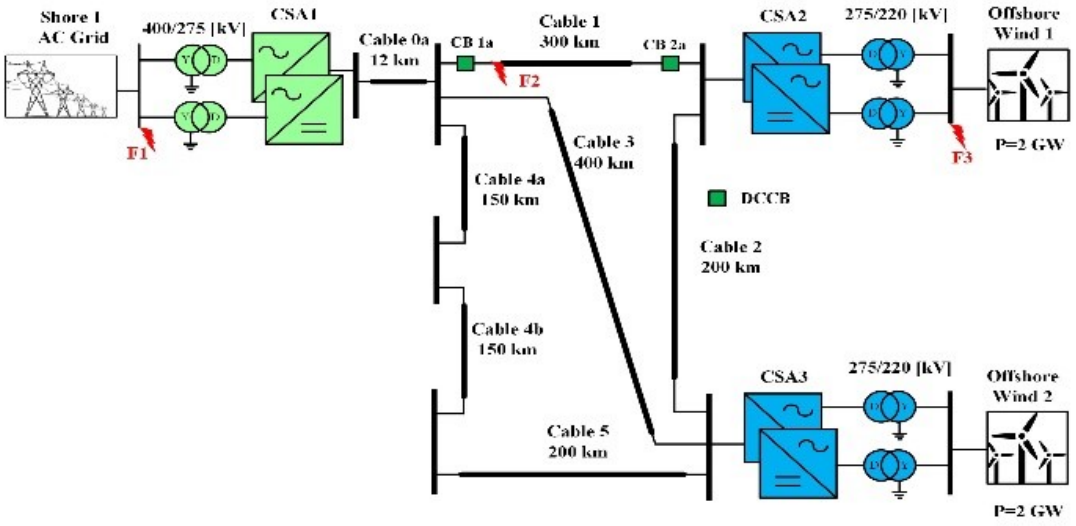


Figure 3.1: HVDC RTDS model presented in [52].

To simplify the model, cables 4a, 4b, and 5 are removed since the empty buses are irrelevant to the performed studies. For this research, only the DC system is of interest. The model has a main RTDS timestep of 60 µs, which results in an update frequency of 16.67 kHz. For further details about the models used in this thesis, the reader is referred to [52].

3.2.1. Protection System

In the model, one protection system is installed, the converter arm current protection. This protection detects if the current through the MMC arm exceeds a certain threshold which is set to 6 kA. Although this protection is necessary in real life, it destabilises the model, making it inappropriate to do successive simulations. After consultation with the model designer, it was chosen to disable it. This thesis work assumes a steady state of the system before application of a fault. The arm current protection is not active in steady state. The research in this thesis is done on the primary travelling wave response which is not influenced by the arm current protection, as this is too slow. Hence, turning off the arm current protection to keep the system stable is considered not to be of significant influence for the scope of this project.

Fault Application and DCCB control

Fault application control is shown in Figure 3.2 for a positive-pole-to-ground (PTG) fault. When the button is pressed, a signal is created at the output of the leftmost flip-flop. The edge detector triggers a pulse with a width of 100ms (i.e. semi-permanent in terms of DC faults) which triggers the fault. By multiplying this pulse with the fault location, the fault type can be selected. This location bitword is set by a dial knob in the runtime environment. Simultaneously, the main breaker of the DCCB for the positive pole is opened after a 3ms delay, and 7ms later, the auxiliary breaker of the DCCB is also opened. The trip signal is sent to the right DCCBs based on location to only close the breakers on the faulted line and pole. This method is referred to as forced protection, because correct DCCB trip signals are created with fault location in mind. Hence, no DCCB actions are taken incorrectly, therefore, the system is kept perfectly stable. A negative-pole-to-ground (NTG) fault is created similarly as a PTG fault, as for its DCCB trip signals. The same holds for a pole-to-pole (PTP) fault creation, although it triggers both positive and negative pole DCCBs. Reclosing the DCCBs is done by resetting the flip-flops. This is performed by pressing the corresponding pole reclose button in the runtime environment.

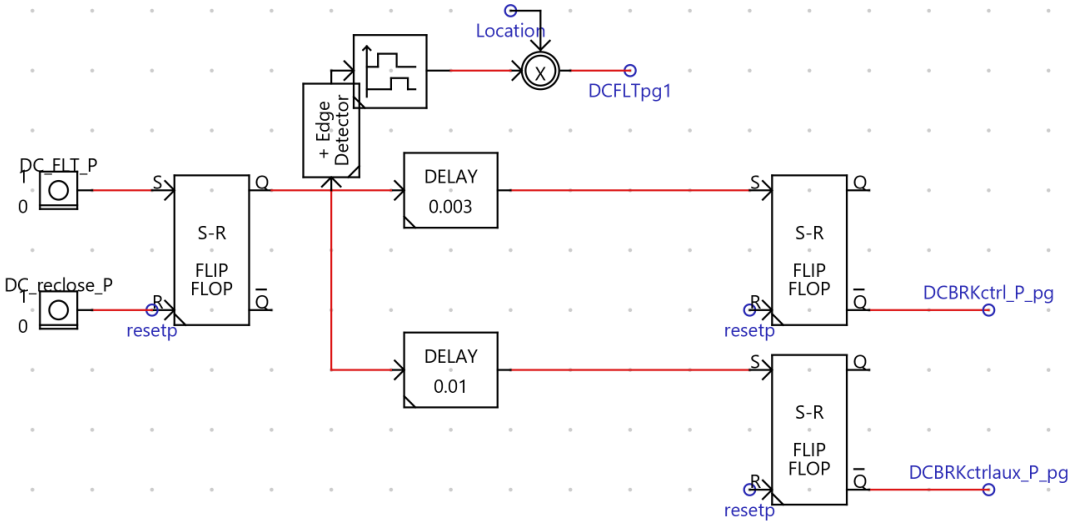


Figure 3.2: Fault application and forced DCCB trip signal generation.

Figure 3.3 shows the fault component in RSCAD. The upper red line is a node along the positive pole, the middle is the negative pole, and the bottom is the dedicated metallic return (connected to the ground). To apply a PTG fault, the positive pole is connected to the ground for the given duration of the pulse. Similarly, an NTG fault is made by connecting the negative pole to the ground. A PTP fault is created by shorting the positive and negative poles together. The fault resistance is a parameter of these components, for which an impedance value is inserted in series with the drawn arc.

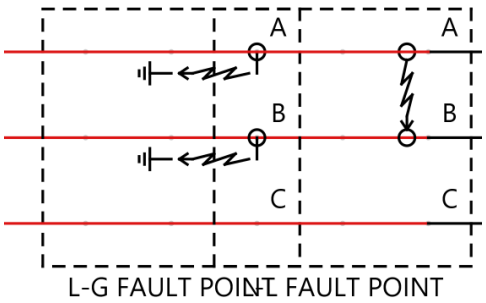


Figure 3.3: Fault application for different types.

3.3. Test Setup

3.3.1. Fault Location

For testing the fault location detection algorithm faults must be simulated with different locations along the cable. To achieve this, Cable12 shown in Figure 3.4 is cut into six pieces of 50km. With relay R_{12} as a reference, faults are simulated at locations spaced evenly at 50km intervals. Fault 1 is at the end of Cable12, 300km away from the measurement point at R_{12}. There is no fault at R_{12} itself, as this could not be measured because of zero voltage and current. Based on the experiment conducted, the fault type is varied as PTG, NTG, and PTP. The fault impedance is varied between 0.001Ω and 1.0Ω.

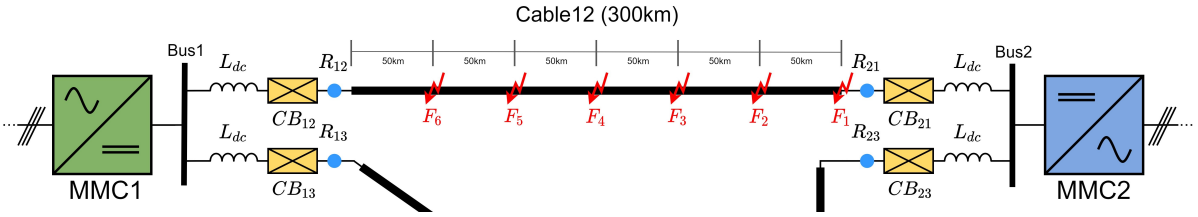


Figure 3.4: Experimental setup for fault location testing.

3.3.2. Protection

To test selective protection, many different fault scenarios must be simulated. The most important fault parameter for protection experiments is the fault location, especially for different cable connections present in the HVDC grid. The ten locations where faults are applied in the network are shown in Figure 3.5. These fault locations are spread evenly over the cables.

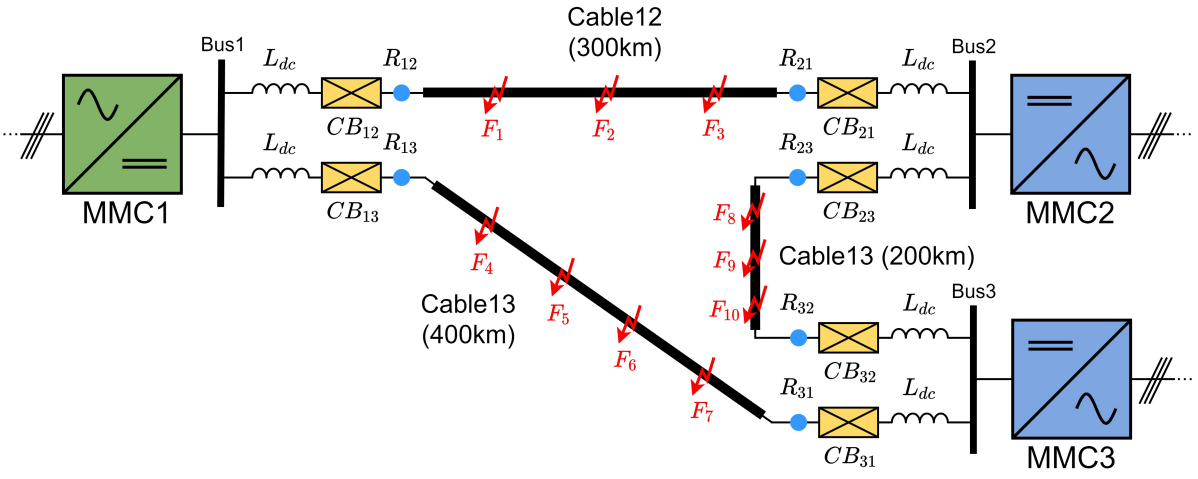


Figure 3.5: Experimental setup for protection testing.

All fault types (PTG, NTG, and PTP) are tested in each case. The fault resistance is varied from 0.0001Ω up to 500.0Ω, depending on the requirements of the experiment. When applying successive faults, a wait time of 10 seconds is required because the system needs to be able to restore stable operation after the transient period. When applying successive faults with different fault resistances, the model should be stopped, recompiled, and restarted. Starting the model requires the start-up procedure, detailed in [52]. An exhaustive experiment with 10 different fault resistances, 10 locations, and 3 fault types will simulate $10 \times 10 \times 3 = 300$ fault scenarios, which takes over an hour to simulate. Although this seems short, it must be considered that model stability issues impose many forced repetitions of simulations due to file corruption or other instabilities. Often, this is only found during data analysis, rendering a complete simulation run unusable.

The relay performance tests are done with forced protection enabled. The trip signals generated by the designed relays are therefore only collected to verify their correctness. Automatic protection done by the relays can however be activated by a switch.

4

DC Fault Location

In offshore HVDC systems, it is challenging to locate the fault occurring on a subsea cable connection. DC faults may significantly damage the cable, necessitating repair work. Underwater visual inspection by humans or machines is possible, however, considering that cable lengths may reach hundreds of kilometers, this is a particular challenge. Providing an accurate fault location greatly reduces the search area, thereby decreasing the overall repair time and enabling the system to return to nominal operation as soon as possible. The fault location is not used for discriminating the faulty transmission line from healthy ones used for protection purposes, but rather for providing support to asset management/maintenance.

4.1. Methodology

As explained in chapter 2, V_{b1} contains information on the fault location due to the distortion effect of the cable on the fault response. By using parameter fitting, the signal's shape can be estimated according to the negative exponential found in Equation 2.28. This leads to the estimation of τ_0, and considering τ is a known cable parameter, the fault location l can be determined. The goal is to create a fully automatic location estimation algorithm that does not rely on communication. This system must be fully integrated in the RTDS / RSCAD environment, operating in real-time without human intervention, and accurately provide the fault location directly following a fault.

4.1.1. Modeling Approach

The AMLM algorithm implemented in MATLAB as used in [26] was provided earlier in this thesis work. First, the algorithm is translated into C and tested. When the C code operates indistinguishably from the MATLAB code, an RSCAD component is created so that RTDS can run the algorithm. This parameter fitting component will also execute the sampling of the V_{b1} signal. In RSCAD, a new draft window is created which computes the V_{b1} signal, detects if a fault occurs, activates and executes the parameter fitting, and computes the fault location. Some assumptions are made based on [26].

Assumptions

- τ is a cable parameter and therefore constant for all location estimations. Its value can be chosen so that it fits the data best.
- The type of fault does not significantly affect the accuracy of the fault location because it only affects the initial value of $\Delta v_f 1$.
- The resistance of the fault does not significantly affect the accuracy of the location estimation because it only affects the initial value of $\Delta v_f 1$.
- A satisfactory result must have an average absolute error below 1.0%, which is assumed to be possible.

4.1.2. Tools

C code is developed in Visual Studio Code, a convenient code editor capable of doing numerical tests of the code. As it is not possible to easily plot the data on a graph, the numerical results are exported to CSV format and imported into MATLAB for visual inspection and code comparison. RSCAD FX 2.1.1 is used for all things related to simulation. It is mainly used to create control blocks for the analysis of measured quantities within the electrical system model. RSCAD's CBuilder (Component Builder, which comes pre-installed with RSCAD) is applied to create the parameter fitting algorithm component. In addition, Excel is used for quick manual data analysis throughout the development of the fault locator.

4.2. Design Process

4.2.1. V_{b1} in Simulation

Both the fault detection and the fault location will be based on the analysis of the V_{b1} signal. The very first step in the project is to compute this signal in RSCAD. This can be executed using the simple control circuit shown in Figure 4.1. This circuit is directly implemented from Equations 2.9 and 2.7, respectively. The `R12` label shows that this TW measurement is located at the relay R_{12}, located at converter 1 on Cable12. The `meas` label shows that these are measured quantities from the HVDC model, namely pole voltages and currents at the relay location. The positive current direction is taken from the DC bus into the cable.

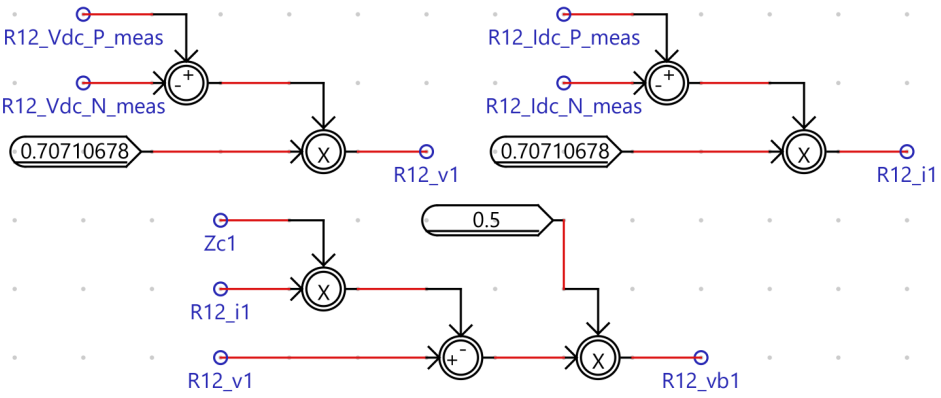


Figure 4.1: V_{b1} computation in RSCAD.

From Equation 2.28, it is known that the wavefront of V_{b1} following an internal fault is a decaying exponential function until it is reflected in the fault point. However, faults can vary in a multitude of characteristics, of which three are in the scope of this thesis, namely fault type, location, and resistance.

Effect of Fault Type

Considering that the line-mode voltage or current as calculated in Equation 2.9 decouples the poles, a ground fault on a positive or negative pole (PTG or NTG) should result in the same line-mode response. Because the signal of interest is V_{b1}, a similar response is expected for both types of faults. This can be seen in Figure 4.2. Similarly, if both poles are faulted in a PTP fault, the fault-induced response is effectively doubled compared to a PTG or NTG fault because of the decoupling effect.

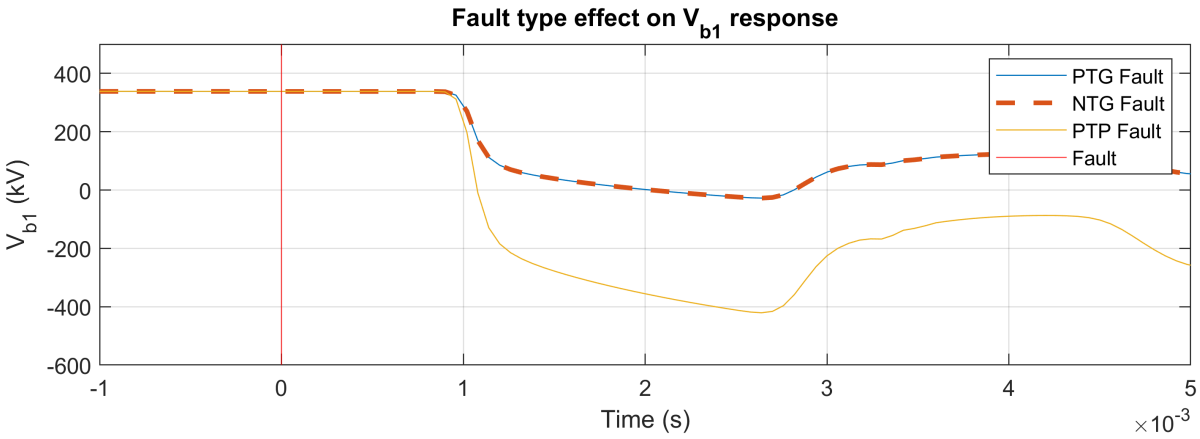


Figure 4.2: Effect of the fault type on V_{b1} response ($R_f = 1.0\Omega$, $l = 150$km).

Effect of Fault Resistance

When a fault has an internal resistance, the voltage at the fault point is not entirely zero, and the fault current will be lower than that of a metallic fault. This effect is transferred to the V_{b1} signal and should therefore be taken into account. Figure 4.3 shows this effect for three fault resistances: 1.0Ω, 0.1Ω, and 0.001Ω. The latter is considered to be a metallic fault. It can be seen that the 0.1Ω fault is not visually distinguishable from the metallic fault. However, the 1.0Ω fault is different and so the response is less severe. The severity of a fault is mainly given by the drop in amplitude in the V_{b1} response. Higher impedance faults will be even harder to detect and classify correctly, as they will be even less severe.

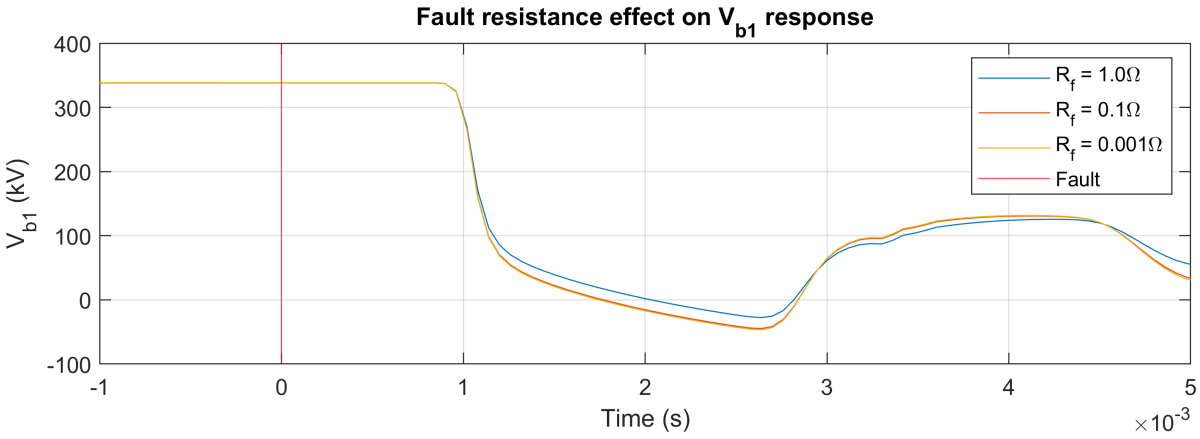


Figure 4.3: Effect of the fault resistance on V_{b1} response (PTG fault, $l = 150$km).

Effect of Fault Location

The fault location has a great effect on the measured response, as can be seen in Figure 4.4. All faults are applied at $t = 0s$. However, their responses are measured later as a result of the propagation speed of the TW. Given that the faults are evenly spread across the cable with a 50 km interval, the incident wavefronts are $50\text{km}/1.67 \cdot 10^8\text{m/s} \approx 0.3s$ spaced apart. This would be the most valuable information to determine fault location; however, as the time of application of the fault is not known apart from the simulation, there is no reference to use for the location estimation.

Secondly, the steepness of the incident wave is related to the amount of signal attenuation that is governed by the travel distance. The response of the fault with $l = 50$km is very steep, whereas this gradually gets flatter for faults with increasing fault distance. This signal property can be estimated by the parameter fitting and, therefore, will be exploited for location estimation.

Lastly, the closer a fault is to the relay, the sooner the signal is reflected into the fault point. The TW will oscillate with a frequency related to the fault distance. As the fault information is contained in the

wavefront until the first reflection, faults that are relatively close to the measuring point have a short window and therefore only a few input data points for the parameter fitting algorithm. This may lead to lower fitting accuracy.

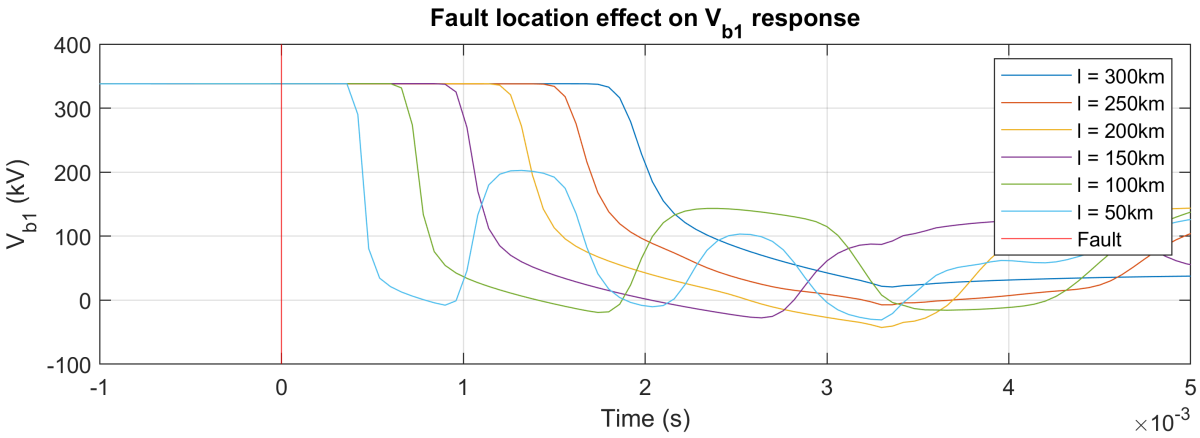


Figure 4.4: Effect of the fault location on V_{b1} response (PTG fault, $R_f = 1.0\Omega$).

4.2.2. Fault Detection

The fault detection control block must activate the sampling of the fault signal. It should therefore produce the trigger signal as soon as the fault occurs. The control block is shown in Figure 4.5. The `delay` signal is a last timestep's 1% scaled down version of the V_{b1} signal: $delay[n] = 0.99 \times V_{b1}[n-1]$. Next, it is compared with the current V_{b1} and `detected` is raised if the current input signal is lower than or equal to the delayed signal: $V_{b1}[n] <= 0.99 \times V_{b1}[n-1]$. `detected` stays high as long as the comparison holds, which depends on the fault response. A gated set-reset flip-flop is used to control the output signal `sampTrig`. As soon as the set signal (S) of the flip-flop is raised, its output signal (Q) is set to 1, regardless of subsequent changes in S. This ensures that the trigger is created only once after the fault. The timer block following the flip-flop creates a pulse with a width of 6ms. This is the `sampTrig` output signal. The flip-flop can be reset by raising the reset input signal (R). It can be manually raised by pressing a reset button, but it is also automatically reset at the moment when a fault is applied by `FaultEvent`.

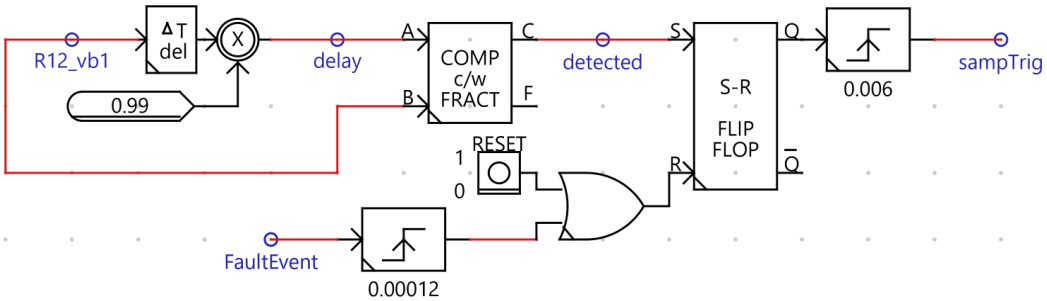


Figure 4.5: Fault detection control block.

In Figure 4.6 the timing diagram of the detection is shown. The comparison made in Figure 4.5 checks whether the V_{b1} signal is below the `delay` signal. As soon as this occurs, `detected` is raised which in turn immediately raises `sampTrig`. It is interesting to note that `sampTrig` is raised one time-step too late since the first sample of interest is the one in which the V_{b1} signal is still above the `delay` signal. This needs to be corrected by the sampling strategy.

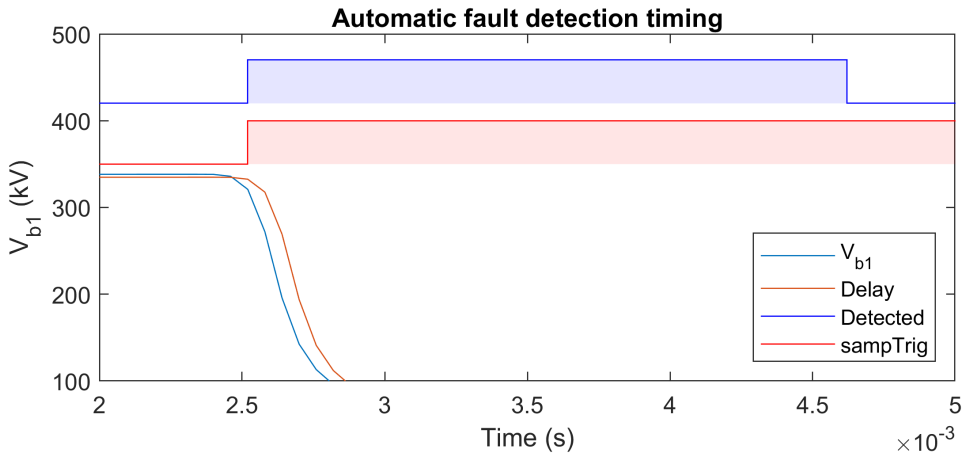


Figure 4.6: Fault detection timing.

4.2.3. Parameter Fitting Component using AMLM

In the RTDS / RSCAD environment, arrays of data cannot be passed from one component to another. Data are passed one sample at a time, once per time-step. This means that the block that performs signal sampling and the parameter fitting component must be on the same simulation time-step, as otherwise the sampled data cannot be read by the algorithm. As passing arrays of data is not possible efficiently, both functions need to be performed by the same component. In this way, the sampled data can be efficiently stored in an array in C, after which it is available directly to the AMLM algorithm. The component design in Cbuilder starts by setting the component's parameters. These are variables organized in tabs that can be changed inside the model drafts after the component is taken from the library. Next, the graphics are drawn for the component, and the input and output connections are defined, as shown in Figure 4.7. In the figure, dots with a white vertical stripe show the input variables, and dots with a white circle show the output variables. The blue and green dots are integer and real-valued variables (floating point), respectively. The design of the graphic is relatively large compared to simpler native RSCAD components, which is necessary due to the large number of IO variables.

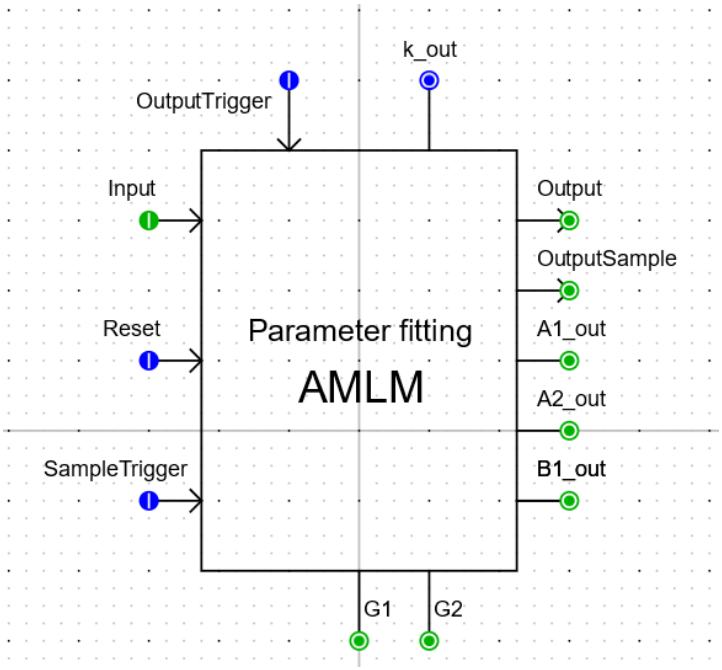


Figure 4.7: Graphic of custom RSCAD parameter fitting component

After the parameters, graphics, and IO variables are defined the component is given its functionality in the form of C code. CBuilder automatically compiles a header file from the parameters and IO variables, which are then accessible to the custom C file. This C file is where the logic of the component is defined. It consists of three sections. The *STATIC* section contains variable declarations. All data types in C (e.g. integer and float) and structures (e.g. array and matrix) must be declared to the compiler before they are used because the compiler needs to create storage spaces in the memory for all these variables. The *RAM* section contains the code that is run once prior to the start of the simulation case. It is meant to compute constants based on the component parameters that are set by the user. It is also meant to initialize arrays and matrices in this section. The final section is the *CODE* section which is executed once every time-step during a simulation. It houses all the functions for the algorithm, as well as a control sequence for the main component logic. To organize the component's functionality, a state machine is created in which each state is assigned as a function, as shown in Figure 4.8.

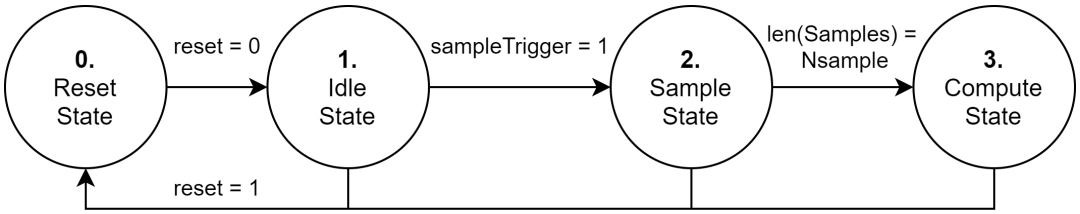


Figure 4.8: State machine schematic of custom RSCAD parameter fitting component

The state machine consists of three action states and a reset state. The component can only be in one state at a time.

State 0: Reset State

In the reset state, all internal variables are reset, which means set equal to zero. The variables are reinitialized so that a new fitting event can start as soon as this is expected. Some variables, such as the input array, are reinitialised to a very high number instead of 0. Because the actual input data are around 0, having this as a default value may lead to problems. The component can go into the reset state from any other state, as shown in Figure 4.8. The *reset* signal is the integer input signal shown in Figure 4.7. As long as the signal is high, the component is kept in the reset state. As soon as the reset signal is low, the component moves to state 1.

State 1: Idle State

The idle state makes sure that the component is ready to start sampling by moving to state 2 as soon as the *sampleTrigger* signal is set to high. One of the user-defined component parameters is the number of history terms *histTrm*. Because the sample state is activated only after a fault is detected, it is noticed that sometimes the first one or two samples are lost. Because the signal of interest has the steepest downward curve at the beginning of the sample, losing the first sample greatly influences the algorithms' performance. With the history term parameter, the user can define how many samples to add to the sample array before the sampling is triggered. The idle state temporarily stores these history terms as the first few samples in the sampling array. It is possible to set *hstTrm* to 0, 1, or 2 terms, depending on the needs of the user's trigger system, which is fault detection in regards to this research.

State 2: Sample State

The sample state is responsible for sampling the input signal and storing it in an array. When the *sampleTrigger* signal is raised, the component moves into this state. At this point, the first few samples are already stored in the array in state 1, based on the *histTrm* parameter setting. Each timestep, the internal *samples* array is extended by the value at the *Input* port as shown in Figure 4.7. As the timestep is 60 µs, the sampling frequency is $f_s = 1/60\mu s = 16.67$kHz. A maximum *samples* array length is set to 100. Defining the length of the sample (sample window) is done automatically: A sample is only added to the array if its value is lower than the last timestep's sample, i.e. $V_{b1}[n] < V_{b1}[n-1]$. If $V_{b1}[n] >= V_{b1}[n-1]$, the window is considered to be completed, because this happens when V_{b1} TW is reflected at the fault point as explained in section 2.5. The sampled signal should have the same length

as t_s as shown in Figure 2.5. Lastly, if the difference between the first two samples is greater than 10 (kV in the case of V_{b1}), the first one is omitted for better performance.

The data window is automatically selected based on the mentioned criteria. This is required because the right input data is crucial for an accurate result. Because the component runs in real-time on the RTDS computer, manual intervention is cannot be made for window selection. The automatic window selection can be seen in Figure 4.9. The PTP fault is applied at $t = 0s$ with $R_f = 0.001\Omega$. The distance from the fault to the relay is 150 km, hence the delay in response of 0.9 s due to the propagation time. It can be seen that the window is selected from the point where the V_{b1} signal is no longer constant at $t = 0.89s$ to the point where the response bottoms out at $t = 2.64s$. It is useful to note that the fault was detected only at the second sample, which shows the importance of the historical terms. In this case, the first sample is a history term such as the one necessary in subsection 4.2.2.

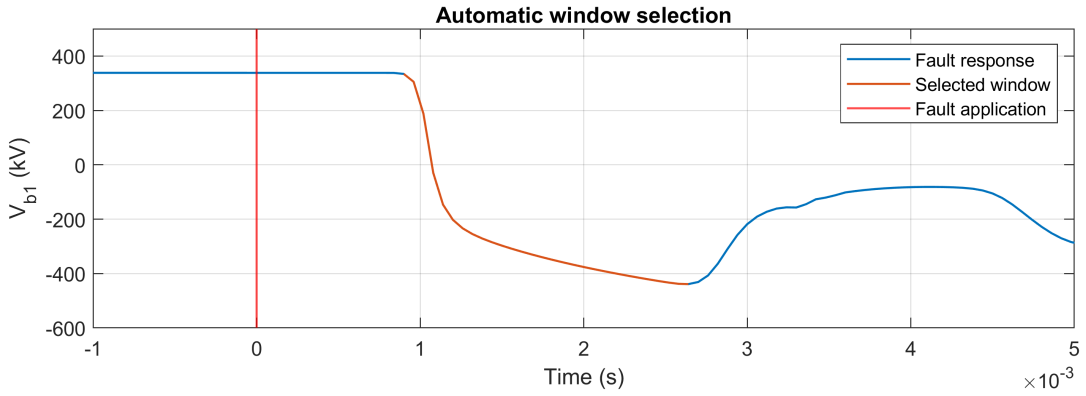


Figure 4.9: Automatic window selection (Fault: PTP, $R_f = 0.001\Omega$).

When the input window is selected, the last step in the sampling process is to remove the steady state. This is done by scaling the sample array down so that the first sample is equal to zero. This results in the signal only containing data on the fault-induced response, which leads to better parameter-fitting results. The samples ready for use by the algorithm are then stored in the *outputArray*, which is the input to the AMLM algorithm in state 3.

The sampled responses or the signals in Figure 4.4 are shown in Figure 4.10. It is seen that the steady state value is removed, as the first sample is at 0 kV. In the figure, it is clearly visible that the propagation along the cable causes the sampled response to change the steepness, becoming more flat for longer propagation distances.

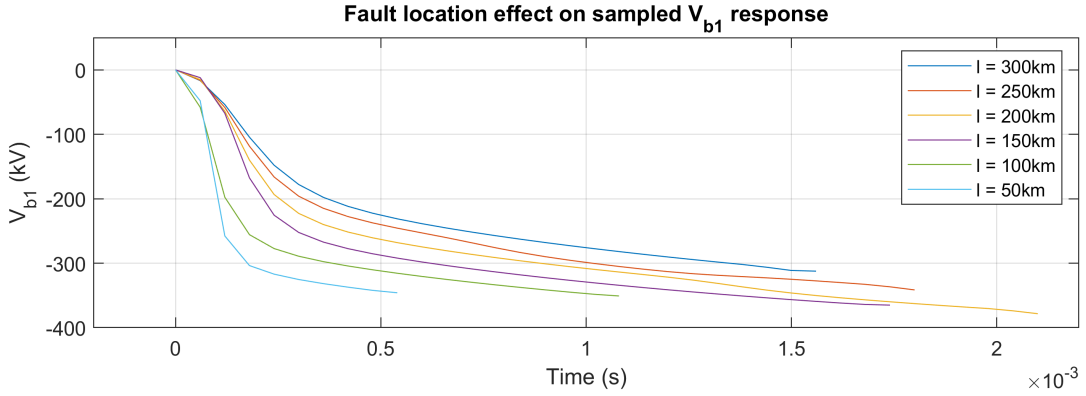


Figure 4.10: Propagation effect on sampled response (Fault: PTG, $R_f = 1.0\Omega$).

State 3: Compute State

The compute state's goal is to do the parameter fitting and provide the parameters as output signals from the component. The parameter fitting is a minimization problem that is executed by the Adaptive

multi-Step Levenberg-Marquardt algorithm (AMLM). At the beginning of the project, the author of [26] provided this algorithm implemented in Matlab [53]. More details on the AMLM algorithm itself and the math behind it can be found in Appendix A.
To create the AMLM component's compute state in CBuilder, the algorithm was translated from MATLAB to C. Where in MATLAB, all linear algebra array and matrix functions are readily available, these had to be manually implemented in C. To get the algorithm to work, C code was developed several numerical operations: functions were implemented for the dot product, matrix multiplication, matrix transposition, and matrix inversion. Following this, dummy data for testing were collected using the sample state of the component. This data was then used to translate the algorithm in MATLAB to C line by line while testing to ensure that both algorithms would perform identically. This step is critical for the comparison of the results with [26] because this means that only the input data of the algorithm can influence it. Figure 4.11 shows the fitting results for a sampled PTG fault for both algorithm implementations. As can be seen, the resulting graphs are on top of each other, showing very similar performance.

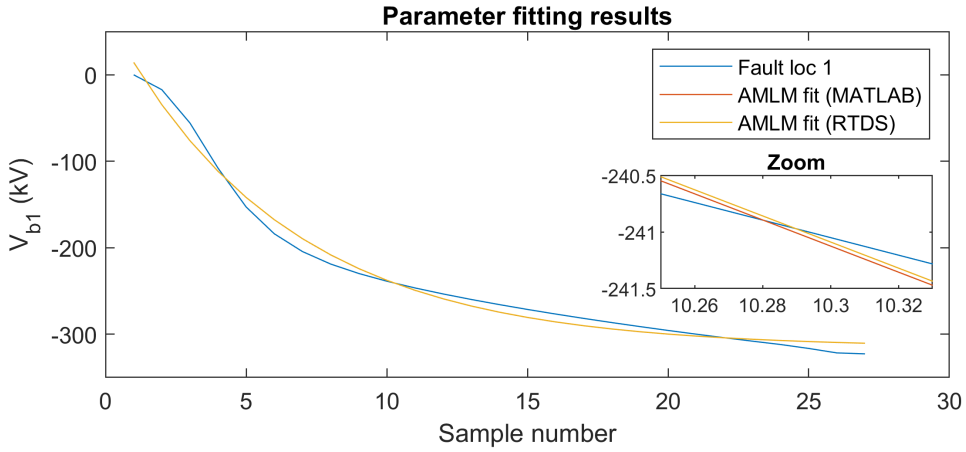


Figure 4.11: Fitting results MATLAB vs. RTDS (PTG fault, $l = 300$km).

The parameters that are used for the fitted graphs are given in Table 4.1. The error between the two fitting algorithms is very small. This could be caused by numerical roundings imposed by the different languages. In general, the translation of the algorithm is successful.

Parameter	MATLAB	RTDS	Error
A1	-315.54	-315.51	0.010%
A2	329.91	329.87	0.012%
B1	2673.43	2673.09	0.013%

Table 4.1: Fitted parameters (PTG fault, $l = 300$km).

As the AMLM algorithm is a minimization algorithm, it keeps iterating and minimizing the cost function until a certain threshold. Given that a single iteration of this process must execute multiple loops over arrays and matrices, matrix multiplications, and even a matrix inversion, which is computationally expensive, the full minimization operation will take a significant amount of time. To minimize the computation time, the algorithm was significantly optimized, even more so than the MATLAB implementation. The CPU time needed for the algorithm to converge is timed on a laptop. The test data sample array had a length of 30 samples, which is almost the maximum length, implying a maximum computation time. The CPU time for this case was 189 µs, which is more than three times the time of a full timestep of 60 µs in the RSCAD model. On RTDS, this measured CPU time of 189 µs will likely decrease, as the RTDS CPU's are more powerful than the CPU's of a generic laptop. However, it still indicates that if not accounted for, RTDS will give the following message: `ERROR - TIME STEP OVERFLOW`. This error means that the simulator could not finish all computations for a certain timestep in the given time frame of 60 µs. To avoid this problem, algorithm iterations must be divided over the

course of multiple simulation time-steps. Because minimizing processing speed is not the main goal of fault location, the maximum AMLM iterations per timestep was set to one. Having a margin in computation time is important, because the parameter fitting component is not the only component or process requiring computation time on the RTDS CPU, even when it is assigned a single core for itself. Forcing only one iteration per timestep completely avoids the timestep overflow error, which makes the parameter fitting component useful in simulation.

4.3. Results & Analysis

To test the algorithm performance, faults are applied along Cable12 with a 50km interval. The B1 parameter is fitted by the AMLM parameter fitting component in the RSCAD model and is stored. The window length is monitored during the experiment. All samples are also stored during the experiment, and they can be processed in MATLAB too. The fault location estimation is numerically calculated from the B1 parameter:

$$l^* = \frac{1}{\tau \cdot B1}. \tag{4.1}$$

The absolute error is calculated by:

$$\text{Error} = \left| \frac{l - l^*}{L_{Cable12}} \right| \%. \tag{4.2}$$

The results of the experiments are given in Table 4.2. The parameter τ is chosen to be $1.55e-06$ to obtain the lowest possible average error. The Total Average Error (TAE) is 9.80%, which is generally very bad. Although some estimates are within 3 km, others are more than 50 km from the actual fault location. Because the location varies with 50 km intervals, the estimated location is assumed to be linear. However, based on the given data, this does not seem to be true. Secondly, the assumption that the type of fault does not matter for parameter B1 also does not hold for this experiment. Especially for $l = 100$ km, a difference of 40km is visible in the estimation.

Fault type	**Location** l	**B1**	**Nsample**	**Estimation** l^*	**Error**
PTG	300	2674.79	27	241.20	19.60%
NTG	300	2674.83	27	241.20	19.60%
PTP	300	2630.71	27	245.24	18.25%
PTG	250	2635.50	31	244.80	1.73%
NTG	250	2635.64	31	244.78	1.74%
PTP	250	2659.60	31	242.58	2.47%
PTG	200	2580.11	36	250.05	16.68%
NTG	200	2579.79	36	250.08	16.69%
PTP	200	2657.95	36	242.73	14.24%
PTG	150	3671.69	30	175.71	8.57%
NTG	150	3671.86	30	175.70	8.57%
PTP	150	3639.01	30	177.29	9.10%
PTG	100	6651.25	19	97.00	1.00%
NTG	100	6651.52	19	96.99	1.00%
PTP	100	4725.49	20	136.53	12.18%
PTG	50	8598.97	10	75.03	8.34%
NTG	50	8598.99	10	75.03	8.34%
PTP	50	8635.43	10	74.71	8.24%

Table 4.2: Location estimation results ($R_f = 1.0\Omega$, $\tau = 1.55e-06$, TAE = 9.80%)

The fault location estimation results for experiments with lower fault resistance R_f showed similar results and are therefore not shown.

It is observed that the sample length *Nsample* is quite long in Table 4.2. Given that the window for input data starts when V_{b1} arrives at the relay point and ends when it is reflected at the fault point,

the data length of each signal (both in number of samples and in time) can be calculated based on the fault location and wave velocity.

Location l	Travel time	Nsample
300	1.80ms	30
250	1.50ms	25
200	1.20ms	20
150	0.90ms	15
100	0.60ms	10
50	0.30ms	5

Table 4.3: Optimal window length (theoretically).

Comparing Table 4.3 with Table 4.2, it becomes evident that automatic window selection selects windows that are too long comparing to the theoretical window length. To remedy this effect, three different truncation strategies are examined. Truncation is the process of shortening the input window. Optimal truncation occurs when signals are truncated according to the calculated value in Table 4.3. However, this is only possible if the fault location is known in advance. Since the relay is not supposed to have this information, the second truncation strategy is to limit the input sample to a maximum of 27 samples. This means that the normal sample length for faults with $l = 300$km is reached, but the other windows that are too long are cut short by at least a few samples. Lastly, as it is known that most location information is stored in the steepest part of the wavefront of the V_{b1} signal, all input windows are limited to a maximum of 10 samples. This will capture only the first part of the signal. If a signal is already shorter than the length at which it is truncated, its length will not change.

To test truncation strategies, the PTG data collected in the experiment of Table 4.2 are exported to MATLAB. The sample length is then set to the correct value in the code. The parameter fitting is then performed using the MATLAB algorithm, which practically yields the same results as the RSCAD component, as shown in Figure 4.11. The results are again collected with a $\tau = 1.55e{-}06$ and are shown in Tables 4.4, 4.5 and 4.6, respectively. It can be seen that the average absolute error is reduced by all truncation strategies, whereas a maximum *Nsample* of 10 samples shows the best results, although still all results are less than satisfactory and do not meet the expected standards of the average error being below 1.0%. It is noted that the fit for the maximum *Nsample* of 10 is poor for the different estimations, which is expected due to the small number of data samples in the input window. For many cases, the fitted signal by the parameter fitting algorithm does not accurately represent the V_{b1} response.

B1	l*	l	**Error**
2674.3	241.2	300	19.59
2790.8	231.2	250	6.28
3246.9	198.7	200	0.43
3835.1	168.2	150	6.08
6635.6	97.2	100	0.92
8628.4	74.8	50	8.26
			6.93

Table 4.4: Truncated to maximum *Nsample* of 27.

B1	l*	l	**Error**
2674.3	241.2	300	19.59
2856.9	225.8	250	8.06
3246.9	172.0	200	9.32
4315.8	149.5	150	0.17
7445.4	86.7	100	4.45
8628.4	74.8	50	8.26
			8.31

Table 4.5: Truncated to theoretical optimum (Table 4.3).

B1	l*	l	**Error**
2425.1	266.0	300	11.32
2791.3	231.1	250	6.29
3207.9	201.1	200	0.37
3642.9	177.1	150	9.03
7445.4	86.7	100	4.45
8628.4	74.8	50	8.23
			6.62

Table 4.6: Truncated to maximum *Nsample* of 10.

The values of the fitted parameter B1 for the original and truncated experiment cases and their corresponding location estimations are plotted in Figure 4.12. The target line on the left graph shows what the B1 values should be for a specific τ ($\tau = 1.55e - 06$). It is calculated as follows:

$$B1 = \frac{1}{\tau \cdot l}, \tag{4.3}$$

following directly from Equation 4.1, but with the actual fault location l, instead of its estimate l*. This target line can be shifted up or down by decreasing or increasing the chosen τ, respectively. It

will remain a decaying exponential as derived from Equation 4.3. Based on the plotted B1 values in Figure 4.12 one cannot find a value for τ for which the target graph meets all fitted B1 points. Because of this, the fault distance estimation values plotted in the right graph of Figure 4.12 cannot be placed on the target line either. As long as the fitted B1 values are not in the exact shape of decaying exponential, significant estimation errors will be present.

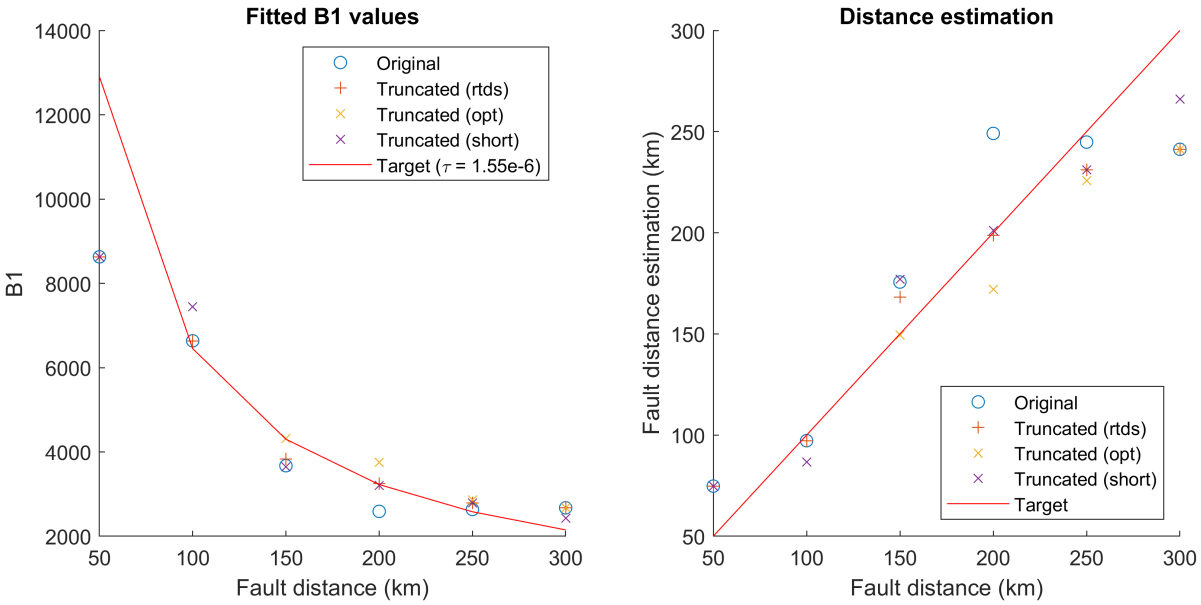


Figure 4.12: Truncation strategies.

However, for each truncation strategy, the chosen τ can still be optimized to reduce the average absolute error. The results of this are shown in Tables 4.7, 4.8, and 4.9.

B1	l*	l	Error
2674.3	249.3	300	16.91
2790.8	238.9	250	3.71
3246.9	205.3	200	1.78
3835.1	173.8	150	7.94
6635.6	100.5	100	0.16
8628.4	77.3	50	9.09
			6.60

Table 4.7: Truncated to maximum *Nsample* of 27 & $\tau = 1.50e-6$.

B1	l*	l	Error
2674.3	277.0	300	7.67
2856.9	259.3	250	3.09
3246.9	197.5	200	0.82
4315.8	171.6	150	7.21
7445.4	99.5	100	0.17
8628.4	85.8	50	11.95
			5.15

Table 4.8: Truncated to theoretical optimum (Table 4.3) & $\tau = 1.35e-6$.

B1	l*	l	Error
2425.1	286.4	300	4.55
2791.3	248.8	250	0.40
3207.9	216.5	200	5.49
3642.9	190.6	150	13.54
7445.4	93.3	100	2.24
8628.4	80.5	50	10.16
			6.07

Table 4.9: Truncated to maximum *Nsample* of 10 & $\tau = 1.44e-6$.

The best TAE of 5.15% is given for the case where the input data concerning a PTG fault is truncated to the theoretical optimum sample length. This error value is optimistic, as it is only based on PTG fault data. Considering that different B1 fittings are found for PTP faults, this error value will be higher.

In conclusion, based on the results, only some of the assumptions made in subsection 4.1.1 are correct. In fact, fault resistance is a parameter that does not significantly affect fault location estimation, at least for low impedance below 1.0Ω. In contrast to the assumption, the type of fault appears to have an impact on location estimation, because the PTP fault is more severe in the V_{b1} fault response than the PTG or NTG faults. The parameter τ has a strong influence on the final location estimation, but choosing it correctly requires accurate fitted B1 values. From the results, it is not conclusive whether there is a single τ that is perfect for all types and locations of faults. Finally, it is evident from the results obtained that an average absolute error below 1.0% is impossible for this approach of fault location, especially for an automatically chosen fault window.

4.4. Discussion

The results obtained are unsatisfactory. Compared to the results obtained in [26], the performance is lacking. Given that both methods are virtually the same (i.e., the same parameter fitting algorithm and similar input data), it is not entirely clear what causes the discrepancy. The higher sampling frequency used in [26] of 20 kHz could have a minor positive effect on the estimation, over the 16.7 kHz used in this thesis. However, given the short window length, this will only add 1-5 samples, which is not sufficient to cause a significant difference in accuracy. Furthermore, the research is not performed using the same network model, which potentially causes different results. The different cable models can lead to different propagation characteristics in the traveling wave. In addition, the HVDC converter model can influence the results as well. Lastly, control and protection systems present in the model may influence the measured signals.

The iterative AMLM algorithm component on the sequential RTDS platform can be used for fault location estimation just milliseconds after a given fault. However, it is less useful when accuracy is lacking. It is argued that integrating the location estimation in a real-time environment is unnecessary. Because a DC fault is generally considered permanent, it is assumed that the protection system of the DC line or cable (either AC or DC protection) disconnects the faulted transmission line permanently, until the fault is repaired, which easily exceeds hours. This means that the analysis of the measured fault data does not have a speed requirement in the range of milliseconds. However, since the analysis portion in MATLAB or RSCAD provides the same output given the same input, it does not matter whether it is in real-time or offline. Hence, there are two possible improvements in the estimation process. The first is the length of the sampling window. It is difficult to select it automatically, however, this process can be optimized by comparing the slope of the signal to the threshold value. Threshold selection is complex because the slope of the signal depends on many factors related to the characteristics of the fault and propagation. The second and most promising optimization is to use a higher sampling frequency. This leads to more data points in the sampling window, which should increase the parameter fitting accuracy. A better representation of the measured signal provides more information about the fault distance. Hence, the location estimation improves. However, it must be noted that the measured signal is not a purely mathematical decaying exponential, and the parameter fitting cannot provide a perfect result.

5

DC Fault protection

Electrical power system protection has existed since power systems were introduced. In conventional AC systems, practically all equipment is protected by different protection strategies. The main protection system for a transformer is the differential protection, whereas for long lines, distance protection is used. What is similar for all types of protection is that fault analysis computations are performed by protection relays. Today, digital relays collect measured voltage and current signals, analyse these data, and create trip signals to their respective circuit breakers. The goal of this chapter is to design a protection concept against DC line faults.

5.1. Methodology

5.1.1. Design Approach

In chapter 4 the AMLM parameter fitting component for RSCAD has been developed and tested. The relay will be built around this simulation component. Firstly, the effects of fault type, resistance, and location on the V_{b1} TW are studied for both internal and external faults, as explained in section 2.5 and 2.6, respectively. The V_{b0} TW is also investigated, especially for different fault types. Additionally, the effect of line-end inductors L_{dc} is analyzed and a fixed value is chosen. Next, a protection scheme is designed to create trip signals based on the parameter fitting results. The relay will comprise the input signal computation (including noise addition), the fault detection algorithm, parameter fitting components, and trip signal generation (protection scheme). During the development process, all components are created by keeping flexibility in mind. This means that addition of noise can be turned on or off, threshold values can be easily changed at sensible locations in the model, and other component specific parameters can be easily set manually and/or automatically. The fault detection method from subsection 4.2.2 is improved for better performance under different circumstances. Robustness upgrades are also done for the AMLM algorithm component. The creation of the protection scheme is data-driven, by which it is meant that internal thresholds are found by means of simulation. Some correlations in the data give rise to more complex information flows in the protection scheme to differentiate internal from external faults for R_f as large as possible. To collect data from many different fault conditions, runtime scripting is used to apply faults and record the data to CSV files. During data collection, DCCBs are operated by forced operation based on the applied fault. This ensures that no wrong trips are made and keeps the system stable, making it possible to perform many faults in succession. The relays will still create their respective trip signals, but these are only used for concept verification.

Finally, the relay concept will be extensively tested. The first test is for all fault scenarios presented in the development phase. The second test is similar, however, it includes the addition of noise. The final tests consist of finding the boundaries of the system, especially considering High Impedance Faults (HIF) that are challenging to detect. The protection deadzone is also identified, which is the zone in which faults are gone unnoticed by a relay.

Assumptions

- It is assumed that the HVDC system can handle several milliseconds of an active fault; An acceptable relay operation is when the correct trip signals are provided to the DCCB within 3 ms after the fault detection.
- A DCCB is assumed to operate swiftly, and it must operate immediately when the trip signal is given.
- High Impedance Faults (HIF) have a fault resistance of at least 10 Ω.

5.1.2. Tools

Similarly to chapter 4, the RSCAD is used for modeling the test circuit and protection concept. For parameter fitting component upgrades, Cbuilder is used in conjunction with Visual Studio Code for programming in C. For automated testing and data collection, the runtime scripting functionality in RSCAD is used. Data are collected in CSV format and analyzed in Excel. Signal data is also collected in CSV format, although analysis is performed in MATLAB.

5.2. Protection Relay Concept Development

This section explains the iterative process of the development of the protection relay concept. For reporting purposes, the explanation is referred to relay R_{12}, which is the relay on Cable12 at the MMC1 converter station. In terms of control, the other five relays operate in the same way, although thresholds might differ as the network is not symmetrical.

5.2.1. Signal analysis

Before designing the protection scheme, the measurement data must be analyzed to find the relations between the different fault scenarios to be exploited by the relay. Recall that the V_{b1} TW is the first available local measurement data for the relay; hence, the optimal source of fault information for the swift operation of the protection algorithms. Based on the result found in section 2.6, the V_{b1} signal can be used for the identification of the fault locale (i.e. whether or not the fault is internal or external). The term locale is used to avoid confusion with the term location, as the meaning is more generic and differs from that explained in chapter 4. Figure 5.1 shows the difference between faults with different locale for a PTG fault, measured at R_{12}. Three different kinds of locale are present: the locations of the faults are F2 (internal), F5 (forward external), and F9 (backward external), as given in subsection 3.3.2. Note that a forward external fault signifies the fault response measured at R_{12} for a fault occurrence at F5, for which the measured TW is forward. This process is explained in section 2.6. Similarly, a backward external fault signifies a fault for which the response measured at R_{12} is a backward TW because the fault occurs at F9. The fault itself is not traveling, however, it called this way for documentation purposes. The solid lines have $R_f = 1.0\Omega$ and the dashed lines have $R_f = 100.0\Omega$. It can be seen that the internal fault has a more severe response compared to external faults (more severe because the amplitude drops more). In contrast to the theoretical analysis in section 2.6, the V_{b1} signal is, in reality, not zero following a forward external fault. Instead, it rises from the point of arrival at R_{12} (at $t = 1.8ms$). The V_{b1} response following a backward external fault decreases after arrival at R_{12} (at $t = 2.9ms$). Note that the fault resistance has a large impact on the response. The signals behave similar to the lower resistance responses, albeit with a severely reduced amplitude. Visually, it is difficult to differentiate between external HIF; however, the internal fault is still easily differentiated from an external fault, which is an ultimate goal.

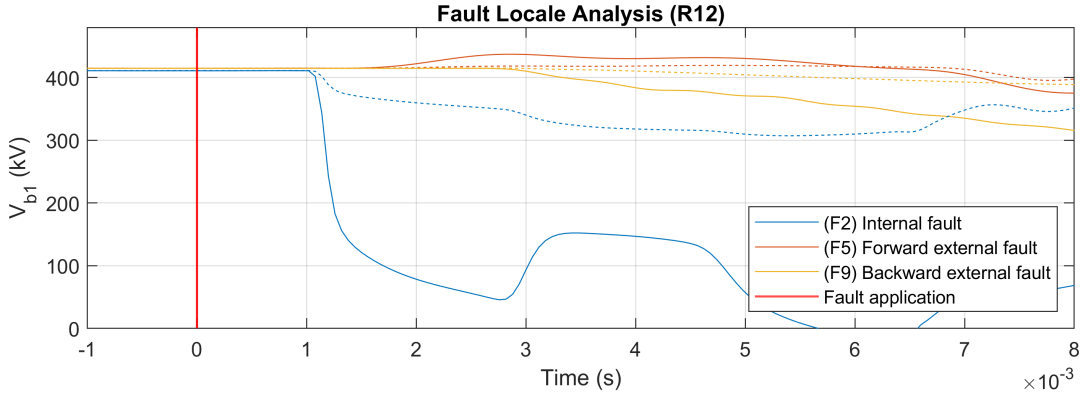


Figure 5.1: Fault V_{b1} response for different locale. Solid: $R_f = 1.0\Omega$, Dashed: $R_f = 100.0\Omega$ (PTG fault).

If a fault is identified as an internal fault, the fault type must be identified. This is necessary to send the trip signal to the correct DCCBs, as the fault type shows which pole is faulty. As explained in section 2.2, the zero-mode component in voltage and current refers to an imbalance between the pole quantities. Because of this, the V_{b0} TW can be used to determine whether a fault is balanced or not, since it will be zero for a balanced PTP fault. Equation 2.9 shows that the zero-mode components are negative for PTG faults and positive for NTG faults. This effect is shown in Figure 5.2 where the V_{b0} response is measured for internal faults. The solid lines have $R_f = 1.0\Omega$, the dashed lines have $R_f = 100.0\Omega$. It can be seen that fault resistance has a significant effect on the response, but the three types of faults are easily distinguishable from each other.

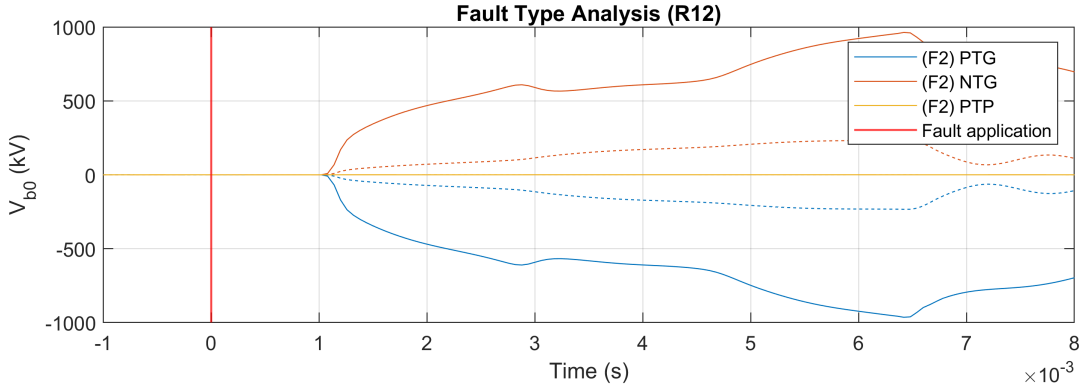


Figure 5.2: Internal fault V_{b0} response for different fault type. Solid: $R_f = 1.0\Omega$, Dashed: $R_f = 100.0\Omega$.

During propagation from one cable to another, the V_{b1} TW is influenced by the line-end inductors and the converter station impedance. The converter impedance is considered an unchangeable parameter of the converter. However, L_{dc} is a design choice, and it is chosen for the best fault locale differentiation performance. The effect of L_{dc} on V_{b1} is shown in Figure 5.3. The tested values for L_{dc} are those in the range of commonly found values in the literature [46], [50]. In the upper graph, it can be seen that the line-end inductors only marginally affect the V_{b1} response following an internal fault. The wavefront is the same for all four responses, however, because higher values for L_{dc} may hold the change in the current component of V_{b1} more, the sag becomes lower. In the second graph, higher values of L_{dc} tend to flatten the response for the forward external fault. However, the effect is minimal as V_{b1} is supposed to measure only backward fault TWs. Still, for all four graphs the characteristic rise in the signal is observed. The bottom graph shows the effect of the backward external fault, which is the largest of the three fault cases. A higher value lowers the rate of change of the signal.

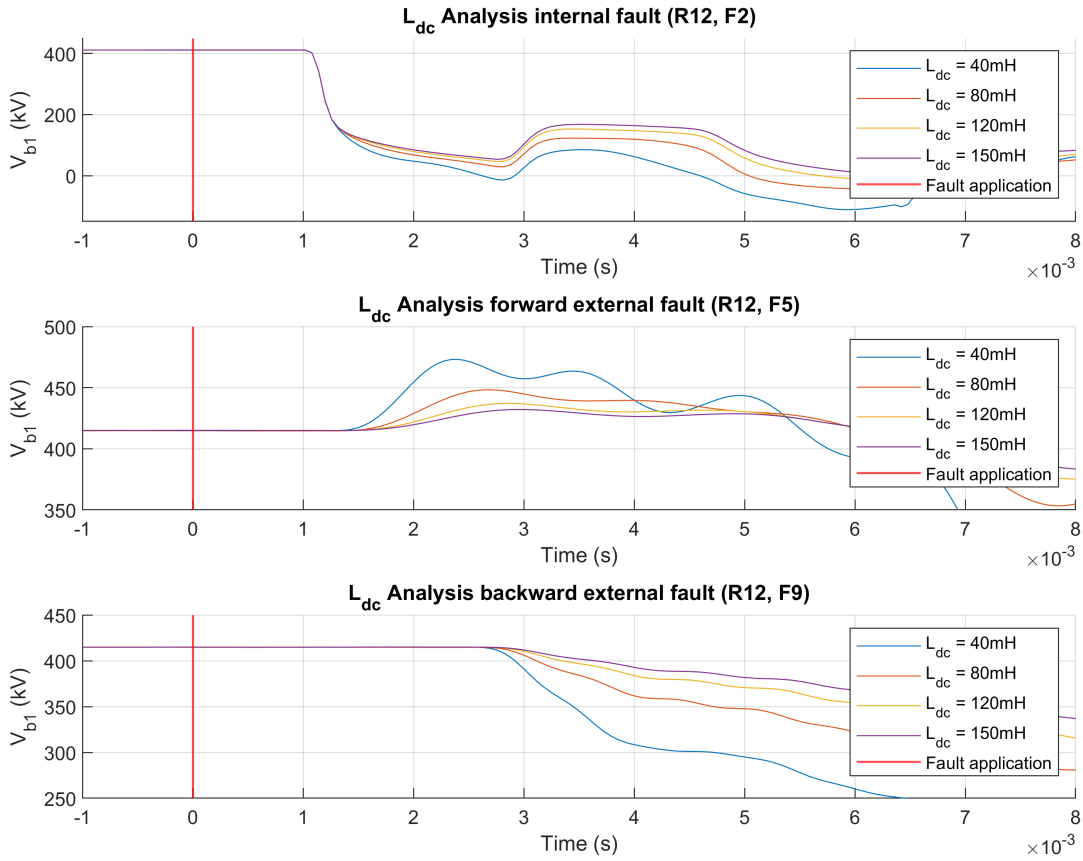


Figure 5.3: Effect of L_{dc} on fault response for different fault locale (PTG, $R_f = 1.0\Omega$).

Taking a higher value for L_{dc}, it is easier to differentiate the internal fault from the backward external fault, but it is also more difficult to detect the fault, especially considering HIF and noise. Based on this, a value of $L_{dc} = 120mH$ is chosen. This choice has been empirically verified by manually testing the fault detection algorithm for various circumstances.

5.2.2. Input Measurements

Measured voltage and current quantities in a power system simulation give very good results; there is no uncertainty observed of whether the value is incorrect. However, measuring high voltage and corresponding currents is challenging, often resulting in signals with a significant amount of noise. Additionally, actual HVDC systems are known to have this noise, too. To design a robust protection algorithm, this noise must be taken into account, as it will adversely influence its performance. To simulate the effects of noise, white Gaussian noise is superimposed on the pole voltage. The goal is achieve a 40dB of Signal-to-Noise Ratio (SNR). This is achieved by adding a noise amplitude of 1%:

$$\mathrm{SNR}_{dB} = 10\log_{10}\left(\frac{P_{signal}}{P_{noise}}\right) = 10\log_{10}\left(\left(\frac{A_{signal}}{A_{noise}}\right)^2\right) = 10\log_{10}\left(\left(\frac{100}{1}\right)^2\right) = 40\mathrm{dB}. \tag{5.1}$$

The control block shown in Figure 5.4 performs the measurement noise addition. The *Random Generator* blocks generate a random number following a normal distribution with a zero mean and a standard deviation of one. This value is then scaled by 1 percent of the nominal system voltage `Vdc_nom`. For development and performance testing purposes, the addition of noise must have the ability to be toggled on or off. This is performed by multiplying the generated noise by the integer value `NOISE`, which is 1 if noise is enabled or 0 if noise is disabled. Hence, when noise is disabled, the random number signal is nullified. Finally, the noise value is added to the positive and negative pole voltages (`R12_Vdc_P` and `R12_Vdc_N`) to obtain the measured voltages `R12_Vdc_P_meas` and `R12_Vdc_N_meas`. Using the same method, noise is also added to the pole currents. The nominal voltage and current are taken as 525 kV

and 2 kA, respectively. As random number generation is not fully random but based on a given seed, care is taken that all generator blocks present in the model have a different seed.

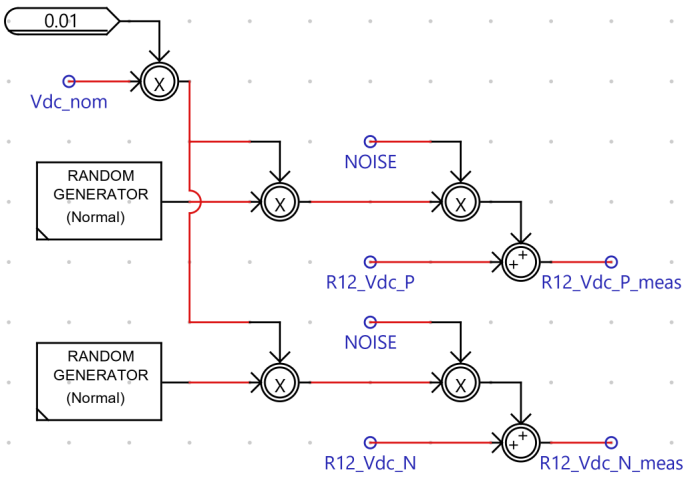


Figure 5.4: Addition of measurement noise.

The calculations of the line-mode voltage and current components are done in the same manner as presented in subsection 4.2.1, as well as the computation of V_{b1}. Similarly, the zero-mode components and V_{b0} are computed. These calculations are based on the measured signals, which could include noise. To verify that the superimposed noise has an SNR of 40dB, 20ms-long samples of voltage, current, and V_{b1} are exported to MATLAB. The *snr()* function is used to measure the SNR of the signal. The results are given in Table 5.1. Although the current SNR is slightly less than 40dB, the signal used for the fault analysis V_{b1} has an SNR of above 40dB, resulting in an average superimposed noise amplitude of 1.5%. This addition of noise is therefore considered sufficient.

Signal	Measured SNR
Voltage (P)	39.99 dB
Current (P)	38.92 dB
V_{b1}	43.72 dB

Table 5.1: Noise measurement.

5.2.3. Fault Detection

The fault detection control shown in Figure 4.5 is not adequate for protection because of several reasons. The sensitivity threshold for the detection algorithm is not easily changeable. Secondly, the detection method does not consider the noise, which has a large impact on the sensitivity of fault detection. Lastly, it only detects a fault when the V_{b1} signal decreases by more than 1% of the value of the previous timestep. However, in Figure 5.1 it is seen that a forward external fault first increases the V_{b1} signal, after which it starts to decrease 1ms later. It can be discussed if forward fault responses are not supposed to be analysed given the signal of interest is a backward TW, however, the forward fault starts decreasing after it has reached its peak, which will be detected nonetheless. Therefore, it is better to immediately detect the forward fault as well, so that the analysis of the fitted parameters can classify the fault as external.

It is found in subsection 5.2.1 that PTG or NTG faults have a less severe response in V_{b1} compared to PTP faults, making them more difficult to detect. However, as shown in Figure 5.2, these unbalanced faults have a detectable fault response in V_{b0} as well. During testing with these signals, it was empirically found that for unbalanced HIF, the V_{b0} response was better for fault detection than the V_{b1} response, mainly due to the detection speed. Hence, it is integrated in the upgraded fault detection control block.

The fault detection algorithm is shown in Figure 5.5. The current V_{b1} value `R12_vb1` is subtracted from

the average of the last six values R12_vb1_avg6, and the ch1 signal is created by taking the absolute value of the difference. Next, ch1 is compared to a threshold value Vb1_T, and if higher, a pulse of 1ms is created on the Vb1_trig signal. The threshold value is different when noise is enabled, because the detection algorithm cannot be as sensitive. If noise is enabled (NOISE_I = 1), the noise threshold Vb1_Tn is used instead. The same logic is used to generate the Vb0_trig signal, but it is based on the V_{b0} signal R12_Vb0. These signals are given to the OR-gate, which ensures either can detect the fault, set the flip-flop, and start the sample trigger sampTrig_R12. The pulse width of the trigger signal is set to be $triggerTime, which means that its value can be set elsewhere in the model. In this case, the threshold values and the trigger time are global variables, meaning that they are equal for all relays. Setting the sampTrig_R12 starts the sampling process done by the AMLM algorithm components.

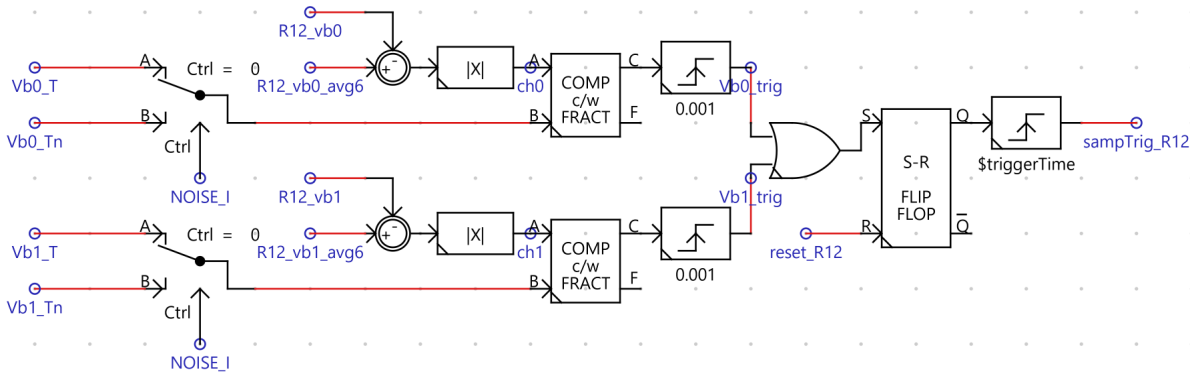


Figure 5.5: Fault detection control.

Noise Mitigation

The fault detection is designed to detect faults as soon as possible to give the parameter fitting algorithm the best input data as feasible. To achieve this, the algorithm must be as sensitive as possible, without detecting power system events other than faults. This means that the noise should not be picked up by the detection algorithm. However, the noise measured on the R12_vb1 signal exceeds 1.5%, which means that the threshold of 1.0, as is the case for the noiseless scenario, is not feasible. The noise present can jump from +1.5% to -1.5% from one timestep to another, which is a huge difference. To mitigate the sporadic behaviour of the noise, a rolling average filter of 6 samples is used for comparison instead of a single delay value. The filtered value is computed by

$$\mathrm{MA}_6(V_{b1}) = \frac{1}{6} \cdot \sum_{i=1}^{6} V_{b1}[n-i]. \tag{5.2}$$

The implementation of the filter in RSCAD can be seen in Figure 5.6. The $\Delta Tdel$ block denotes a delay of exactly one timestep. This delay can be extended by sequentially placing the block for the desired length. Consequently, four sequential delay blocks provide the signal of four timesteps ago. The summation block adds all values together, after which the total is divided by six to obtain the average.

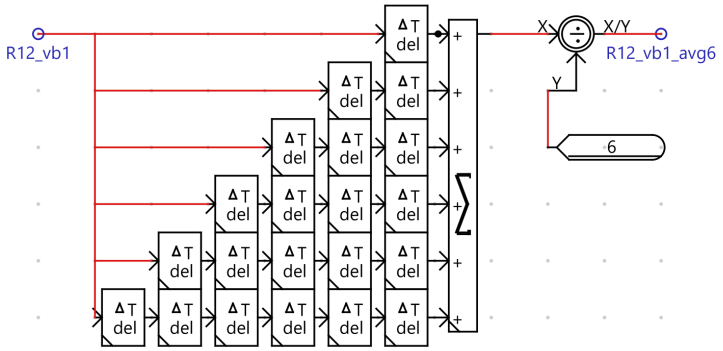


Figure 5.6: Rolling average filter (N=6).

In the top graph of Figure 5.7, the result of the moving average is shown in orange. The V_{b1} signal with the addition of noise is plotted in blue. It can be seen that the moving average has a smoothing effect on the signal, making the signal less jagged and with lower peaks than the input signal. In the bottom graph, the absolute difference `ch1` signal is shown with and without the moving average filter. It can be seen that the addition of the moving filter results in lower peaks in the difference signal, making it less susceptible to noise triggering. If noise is disabled, the rolling average filter is still active. However, as a steady state is assumed, the signal does not change. This means the average is the same as all six signals used for its computation. Hence, it is practically identical to using just a single delayed V_{b1} sample.

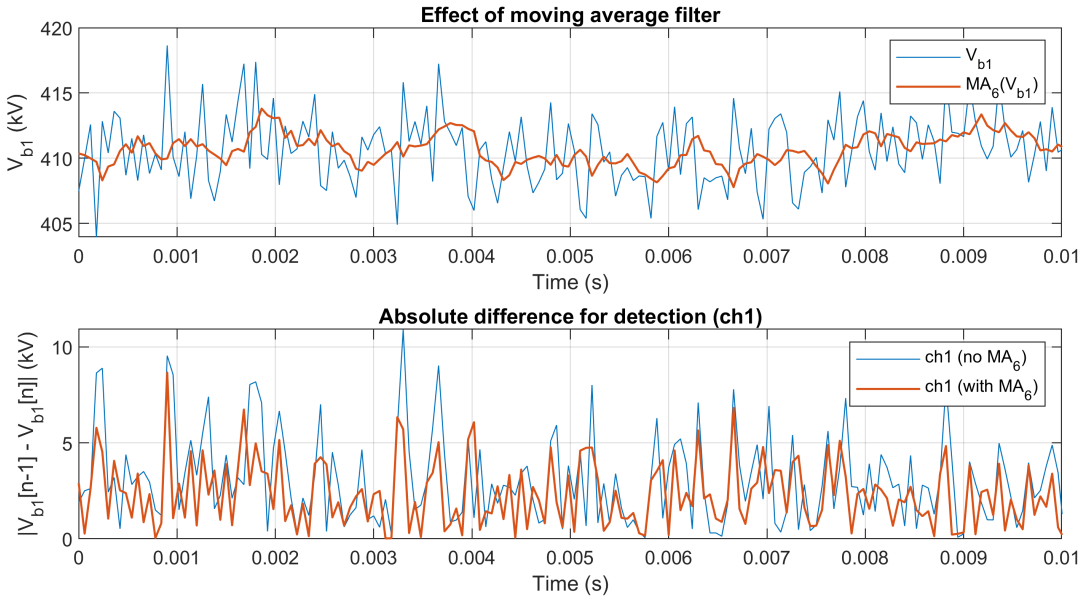


Figure 5.7: Rolling average filter (N=6).

The thresholds are empirically found. Because of the random and probabilistic nature of the noise, very high peaks may occur in the signal, higher still than shown in Figure 5.7. From the bottom graph, it seems a threshold for `ch1` can be set at 10. However, over time, a random peak in the noise level could still trigger the detection algorithm. Hence, values were manually tested by increasing by 10.0 and increasing steps of 1.0. If the active detection algorithm was not being triggered for more than five minutes given a certain threshold, this threshold was deemed sufficient. This method resulted in the thresholds in Table 5.2. For reference, implementing the moving average filter reduced the noise threshold from 22.0 to 16.0. Although this seems just a small improvement, many more faults, in particular external and HIF, are being detected and therefore classified this way, improving the robustness of the relay greatly.

$V_{b1,thr}$	$V_{b0,thr}$	$V_{b1,thr,N}$	$V_{b0,thr,N}$
1.0	1.0	16.0	16.0

Table 5.2: Fault detection threshold values.

5.2.4. Parameter Fitting

The next step in the protection concept design is analyzing the measured TW signals. Information about the fault is extracted from these signals using parameter fitting performed by the AMLM algorithm. The AMLM component responsible for this analysis is taken from chapter 4 where it was developed. Because the signal of interest is the same as for the location estimation algorithm, only minor changes are made to the component.

The analysis process commences by sampling the relevant signals V_{b1} and V_{b0} as soon as the sample trigger is given by the fault detection control. The history term of the parameter fitting component is set to 1 to ensure a correct start of the sampling window, given the lessons learned in chapter 4. However, not the same automatic window selection is used for protection. Recall that it worked by the V_{b1} signal bottoming out and therefore changing the sign in the slope. This worked because the fault of the study was always internal. For protection, this is not true. Forward faults that have a positive slope at the beginning of the sample, would in that case not be sampled at all. Instead, a fixed number of samples is chosen for the signal analysis. In contrast, to location estimation, where a very accurate fit is required to extract a good estimation, determining the fault locale is hypothesized to require less accurate results. Therefore, a less-than-optimal parameter fit result is supposed to be sufficient. Sample lengths of 20, 25, and 30 were tested by collecting parameters given by the algorithm, for numerous fault scenarios varying locale, impedance, and type. Generally, performing parameter fitting with 20 samples had worse results than for 25 or 30 samples. 25 samples were chosen as the fixed number of samples, as 30 is regarded to take too much sampling time. Given the RTDS timestep is 60 µs, only sampling 25 samples already takes up 1.5ms, which is half of the maximum relay operation time of 3ms.

Next, the maximum number of iterations (parameter `k`) done by the AMLM algorithm was defined. The algorithm works by minimizing the cost function of the mathematical function with the parameters compared to the given input data. For external faults, the mathematical approximation of the negative exponential does not match the input data. Therefore, for many cases, the cost function cannot be satisfied. This could mean that the algorithm keeps iterating without improvement. Secondly, it is noted that the initial few iterations have considerably the most impact on the accuracy of the fit, regardless of the input data being an external or internal fault. As a result, the maximum number of AMLM iterations is fixed at 20 iterations, as lower values led to significant losses in accuracy for the same testing strategy as for sample length.

The iterative process of the AMLM was found to cause RTDS timestep overflow errors in simulation. Hence, component was limited to 1 algorithm iteration per timestep. 20 iterations of computation thus results in 1.2ms of analysis, right after the 1.5ms of sampling. However, the limits of the component's performance are not pushed in this way. To optimize for performance, all components in the relay unit are set to a specific control group. This means, that it is assigned a dedicated processing core on the RTDS NOVACOR rack's processor. In this case, the algorithm can do up to five algorithm iterations per timestep, reducing the parameter fitting time to only 0.24ms. However, to avoid the error altogether a standard of four iterations per timestep is taken, as five still caused the error to appear on some occasions. This results in a parameter fitting time of 0.3ms. For large-scale testing in which many simulations are done in succession, the value is even lowered to 3, to avoid the risk of corrupting a simulation batch. Most of such tests are done to find protection threshold values, or to simply test control systems, and thus relay speed performance is not critical.

The final modification in the parameter fitting component concerns the method used to eliminate the steady state value from the sampled signal. As is seen in Figure 5.1 the steady state value of the V_{b1} signal before the fault response arrives is about 410 kV. Removing this constant value from the sampled response makes sure only the fault induced response is analyzed. Due to the presence of noise, removing the steady-state by using the first sample can cause the entire sample to be shifted upwards or downwards relative to the noise present in the initial sample. Consequently, a new input to the component has been devised, wherein the rolling average filter value is employed as a more accurate

representation of the steady-state value. It is important to note that this adjustment does not affect scenarios without the addition of noise.

5.2.5. Protection Scheme

The design of the protection scheme is an iterative and data-driven process. The scheme works by comparing the fitted parameters to a set of carefully predefined threshold values. Noise is initially disabled during the design, but will be incorporated later on in the process. The protection scheme is first designed only for R_{12}, because copying a finished relay to fit on the other five relay locations saves a lot of model development time. The goal is therefore to finish the relay as far as possible for a single location, before integrating the full network protection system.

Design Iteration 1

To commence the protection development process, a dataset of fault parameters is collected for R_{12}: the fault location is varied according to the test setup given in subsection 3.3.2. PTG, NTG, and PTP faults are applied at each location. The fault impedance is varied for 0.0001Ω, 0.1Ω, 10.0Ω, and 200.0Ω. This leads to 120 total different fault scenarios. The forced protection system is used to keep the system stable during the simulations, as having no protection during a fault results in a total breakdown of the network. From the data it becomes clear that a perfect separation using a single threshold is not possible when considering HIF. To show this effect, a small portion of the collected data is shown in Table 5.3, where data for fault impedances 0.0001Ω and 500Ω is left out for demonstration purposes. The table is sorted in ascending order for `V_b1_A1`. The fault location column is colored to easily show the fault locale: Green for Cable12, red for Cable23, and orange for Cable13. In the fault type column the PTP fault is given a light yellow shade. All forward external faults have an A1 value higher than -4.33, which means that they are easily separable from the internal faults. However, it can be seen that the internal faults and backward external faults are mixed. This means that a single threshold for A1 is enough to separate these. Upon closer inspection, only the high impedance internal faults are mixed with backward external faults. This is expected, as shown in Figure 5.1, in which internal HIF approaches backward external faults. For all internal faults the 200Ω NTG fault at location 2 has the lowest A1 of -29.55, and the highest backward external fault A1 is -89.36. However, the internal A1 parameter drops to -12.29, which makes it very difficult to separate from external faults. Based on the collected data a perfect locale separation with only a single threshold value is possible for faults with an impedance up to 10Ω. Based on the full data table, faults with $R_f = 500.0\Omega$ are deemed too difficult to separate, and are therefore not further considered.

As expected in Figure 5.2, `V_b0_A1` in Table 5.3 shows a perfect fault type separation for internal faults for any R_f, resulting in the threshold values of:

$$\text{PTG} < -5.00 < \text{PTP} < 5.00 < \text{NTG}. \tag{5.3}$$

Detecting the type of the fault is only useful for internal faults, as for external faults no DCCB trip signals will be generated. Still, the fault type detection can be done for backward external faults with the same threshold values. For forward faults, the threshold settings are reversed (i.e., PTG $\Leftrightarrow$ NTG in Equation 5.3). However, this is obsolete and not used in the protection concept.

Design Iteration 2

The goal of the second design iteration is to determine the possibility to filter internal from external faults in Table 5.3. To provide more information on fault data, relay R_{13} is also used collect parameter fitting results. The same fault locations and types are applied as last iteration, the fault resistance is varied for 0.0001Ω, 0.1Ω, 1.0Ω, 10.0Ω, 100.0Ω, 200.0Ω, and 300.0Ω. This leads to a total of 210 fault scenarios. Again, when looking at the A1 parameters for R_{12}, a perfect separation of internal and external faults with a single threshold value can be achieved for faults up to 10Ω. For HIF up to 300.0Ω quite some overlap between backward external and internal faults is happening. Even a single forward fault (PTP, 300.0Ω) ends up with a negative A1 parameter. This could be due to a bad fitting result, but it needs to be filtered out nonetheless. Two observations are made in the data which could be the basis of the fault locale filtering.

1. PTP Filter

Generally, PTP faults are the most severe. LIF PTP will therefore have a very negative A1 parameter,

F_loc	F_type	F_Res (Ω)	V_b1_A1 ↑	V_b0_A1
⋮	⋮	⋮	⋮	⋮
1	NTG	0.1	-144.24	427.16
2	PTP	200	-109.58	-0.02
1	PTP	200	-108.86	0.12
9	PTP	0.1	-89.36	0.25
9	PTP	10	-70.91	3.35
3	PTG	200	-64.59	-69.59
3	NTG	200	-63.57	69.01
8	NTG	10	-48.40	34.69
9	PTG	0.1	-44.56	-60.83
9	NTG	0.1	-43.95	60.56
10	NTG	0.1	-43.44	56.12
10	NTG	10	-33.00	43.44
8	NTG	0.1	-32.06	47.94
1	NTG	200	-30.74	61.43
1	PTG	200	-30.57	-61.20
9	NTG	10	-29.96	40.52
2	PTG	200	-29.71	-56.42
2	NTG	200	-29.55	56.12
9	PTG	10	-29.54	-38.18
10	PTP	0.1	-28.73	2.41
⋮	⋮	⋮	⋮	⋮

Table 5.3: First design iteration fault parameters (computed by R_{12}).

more so than unbalanced faults. A similar but opposite effect is observed in backward faults, where PTP faults generally have the highest A1 parameter. This means that there is a range in all Vb0_A1 parameters in which all PTP faults are forward external and all unbalanced faults are internal. Hence, based on the Vb0_A1 parameter, the fault locale can be determined for these faults. However, this can be only applied to faults up to 200.0Ω, as 300.0Ω internal PTP faults are similar in Vb1_A1 to backward external LIF. This method gives rise to a range for which external PTP faults can be distinguished from internal faults. The forward fault (PTP) would also be filtered using this method.

2. External Fault Filter

It is observed that internal faults, as indicated for R_{12}, have a positive A1 parameter for R_{13}, whereas backward external faults (for R_{12}) are characterized by a negative A1 parameter for R_{13}. This effect is demonstrated in Table 5.4. This observation is logical, as internal faults for R_{12} correspond to forward external faults for R_{13}, thereby resulting in the positive A1 parameter. Backward external faults for R_{12} are also backward external faults for R_{13} because the fault takes place on Cable23. Both relays R_{12} and R_{13} will regard faults on Cable23 initially as non-severe internal faults, but can use each others data to find that the fault is actually external for both. Consequently, this leads to the development of a second method for external fault filtering.

One of the goals of the protection design is to be non-unit protection. This means that the protection systems do not rely on communication between relays as this requires precise and synchronised measurement data which is hard to achieve over long distances. However, relays R_{12} and R_{13} are installed at the same geographical location: A substation or an offshore platform. This means that a communication channel in the form of a optical fibre of negligible length can be established between these relays. Given that both relays detect and analyse the fault, the locale can be established by examining both A1 parameters. This method is considered to be non-unit, as the communication channel itself does not cause significant delay in the fault analysis method. In some fault scenarios, this method will however cause delays in relay operation. If R_{12} detects an internal fault and the analysis result is not conclusive (e.g. for HIF), it needs to wait for R_{13} to provide more information. Because the wave propagation takes time and distortion by the inductors, it is possible that R_{13} detects the fault later than R_{12}.

F_loc	F_type	F_Res (Ω)	Vb1_A1_R12 ↑	Vb1_A1_R13
⋮	⋮	⋮	⋮	⋮
10	NTG	10	-33.59	-20.8
10	PTP	100	-31.62	-20.59
1	NTG	200	-30.75	2.14
1	PTG	200	-30.56	1.95
9	PTG	10	-30.22	-25.69
9	NTG	10	-29.94	-25.40
2	NTG	200	-29.20	2.08
2	PTG	200	-29.01	1.97
9	PTP	100	-26.98	-23.98
8	PTP	100	-23.87	-28.47
1	NTG	300	-21.39	1.30
1	PTG	300	-21.28	1.15
2	PTG	300	-20.16	1.15
2	NTG	300	-20.06	1.00
10	PTP	200	-19.53	-12.62
⋮	⋮	⋮	⋮	⋮

Table 5.4: Second design iteration fault parameters (computed by R_{12} & R_{13}).

Hence, the A1 value provided by R_{13} can take a significant amount of time before it is available to R_{12}. It is proposed to split internal faults in severe and non-severe faults. The non-severe faults are regarded as inconclusive and will therefore wait for a signal provided by the accompanying relay. Severe internal faults however, typically PTP or HIF, make the relay create trip signals immediately. This split is assumed to be possible as non-severe internal faults have a much lower rise in fault current due to high fault resistance. This makes for easier fault current breaking by the DCCB, essentially extending the 3 ms time criterion for fault detection.

Although promising, this method is not suitable when noise is enabled. Because the fault detection algorithm is less sensitive to avoid faulty triggers, relays are often not triggered at all for external HIF. Waiting for data that never becomes available because of the accompanying relay not detecting a fault is a waste of time that may lead to averse protection performance.

Design Iteration 3: Implementation

For the implementation of the relay protection algorithm it is proposed to divide the A1 parameter range in several sections, which is done by threshold values. The first threshold value `THR1` divides the range such that all internal faults have a lower A1 value. The second value `THR2` determines the severe internal faults. The PTP filter section is defined between `THR1` and `THR3`. For rare cases, the PTP filter also depends on a certain range of the A1 parameter defined by the accompanying relay, which is realized by `THR4`. `THR5` is the threshold for comparing the accompanying relay A1 parameter to implement the forward fault filter. A schematic overview of the given threshold values and their relative positions is given in Figure 5.8. Given threshold values are subject to change, especially for the addition of noise. If noise is enabled, only `THR1`, `THR2`, and `THR3` are used, as the fault analysis result of the accompanying relay is considered unstable. `THR6` and `THR7` are for fault type identification.

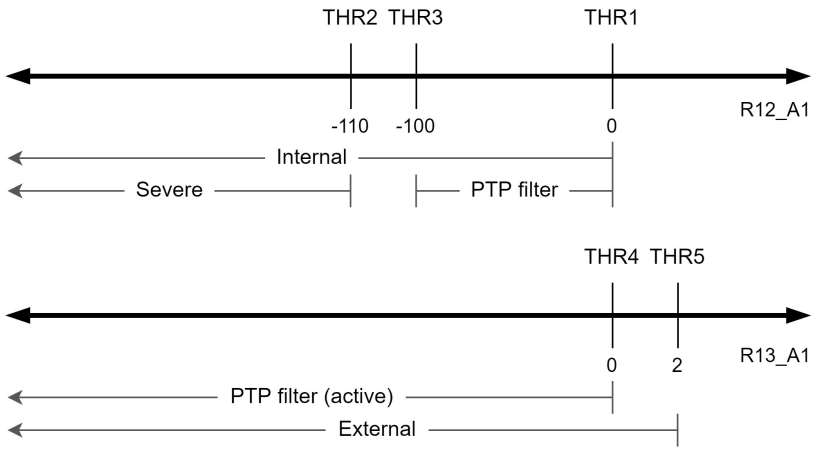


Figure 5.8: Schematic overview of threshold values for fault locate identification based on A1 parameter.

The protection scheme overview for the relay is given in Figure 5.9. It comprises all functional control blocks discussed in this chapter. When noise is enabled, all functional blocks regarding the accompanying relay are not present: The PTP filter is based solely for parameters generated by the relay of interest and forward fault detection on the accompanying relay is not performed.

The RSCAD implementation of the `internal` signal generation is given in Figure 5.10. Firstly, the right `THR1` value is chosen given the noise state `NOISE_I`. The If/Else block compares the fitted A1 signal to `THR1`, and outputs a high signal if it is lower. To check if the fitted A1 signal is reliable, the algorithm iteration number `k` must be a positive integer, indicating that the parameter fitting component was activated and executed correctly. If it is still zero, no parameter fitting has taken place. If both these conditions are met, a signal is created indicating a possible internal fault has been identified. Next, a delay is imposed on this signal, because the relay needs to wait on the accompanying relays result before deciding the actual fault action. If noise is enabled, this delay is only two timesteps, which is necessary due to RTDS imposed control delays. Similar comparison blocks are created for the PTP filter and the Forward fault filter.

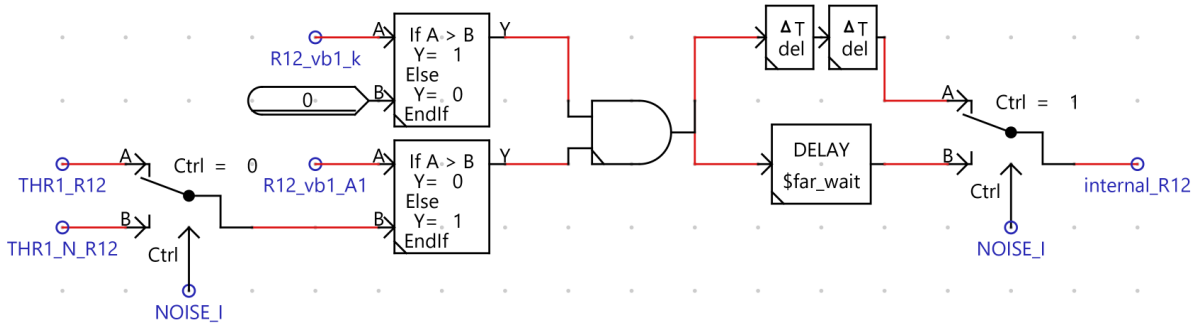


Figure 5.10: RSCAD implementation for internal signal generation.

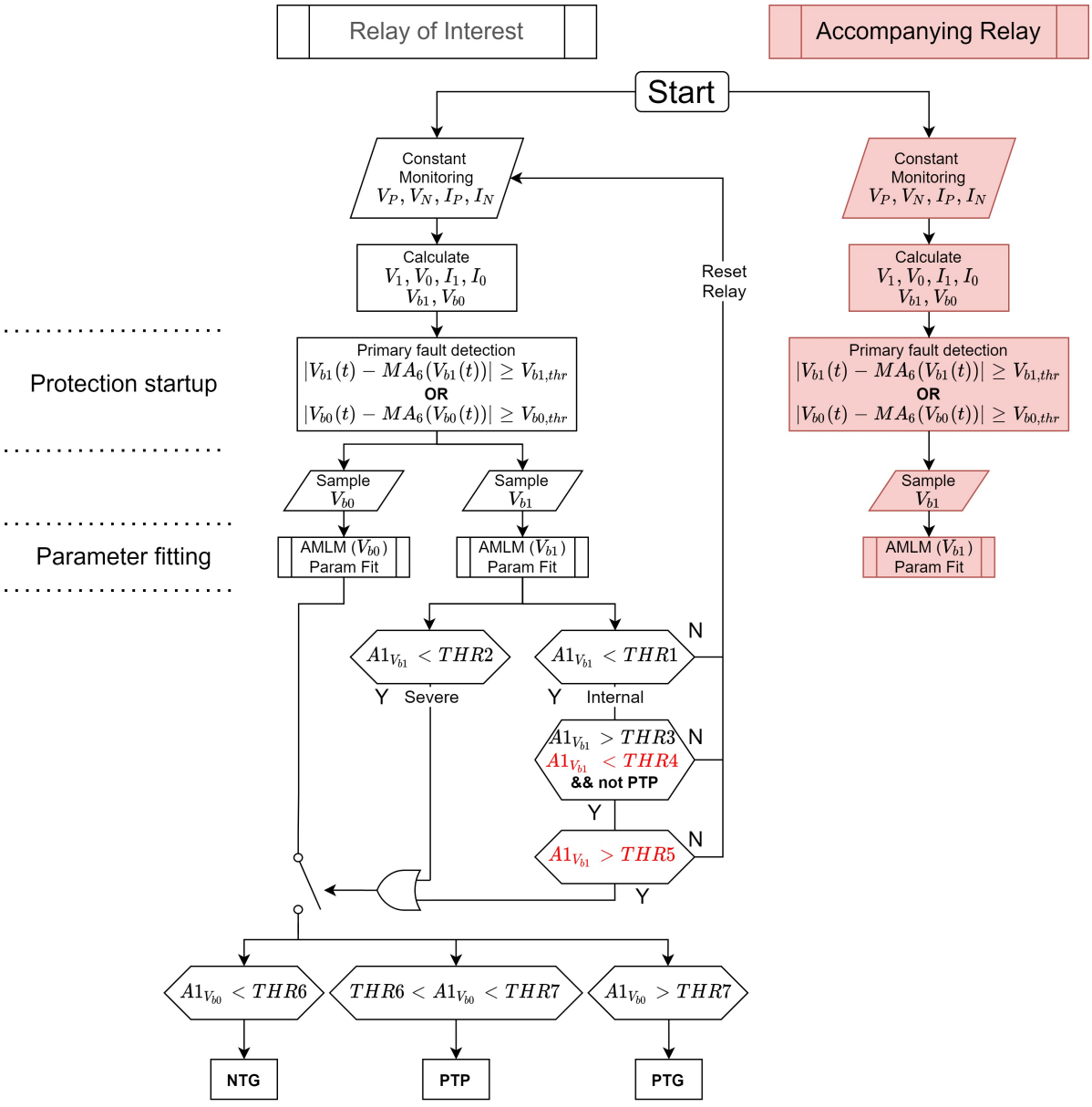


Figure 5.9: Overview of DC line fault protection scheme.

The implementation of the fault classification logic is shown in Figure 5.11. If either the external filter (`Ext_filter`) or the PTP filter (`PTP_filter`) are triggered, the fault is classified as an external one, as indicated by the `act_ext` signal. Next, an internal fault (`int_FAULT`) is given as a final decision. If this happens, the relay will definitely create a trip signal, based on the fault type. The (`int_FAULT`) is created when the following few statements are true: the delay detects a possible internal fault (`internal` = 1, only happens after its delay), the relay is not in reset mode (`ResetRelay` != 1), the relay does not detect the actual external fault (`act_ext` != 1), and the sample trigger (`sampTrig` = 1) is still enabled. If a severe internal fault (`sev_int`) is detected, this check is overruled and the trip signal is passed immediately.

The fault type is classified by comparing (`vb0_A1`) to the threshold values. The AND gates ensure the V_{b0} parameter fitting went well. An internal type fault such as (`NTG_int`) is only generated if `int_FAULT` is active, the corresponding type is active, and the protection is enabled. During the startup sequence for the converters, this is not the case.

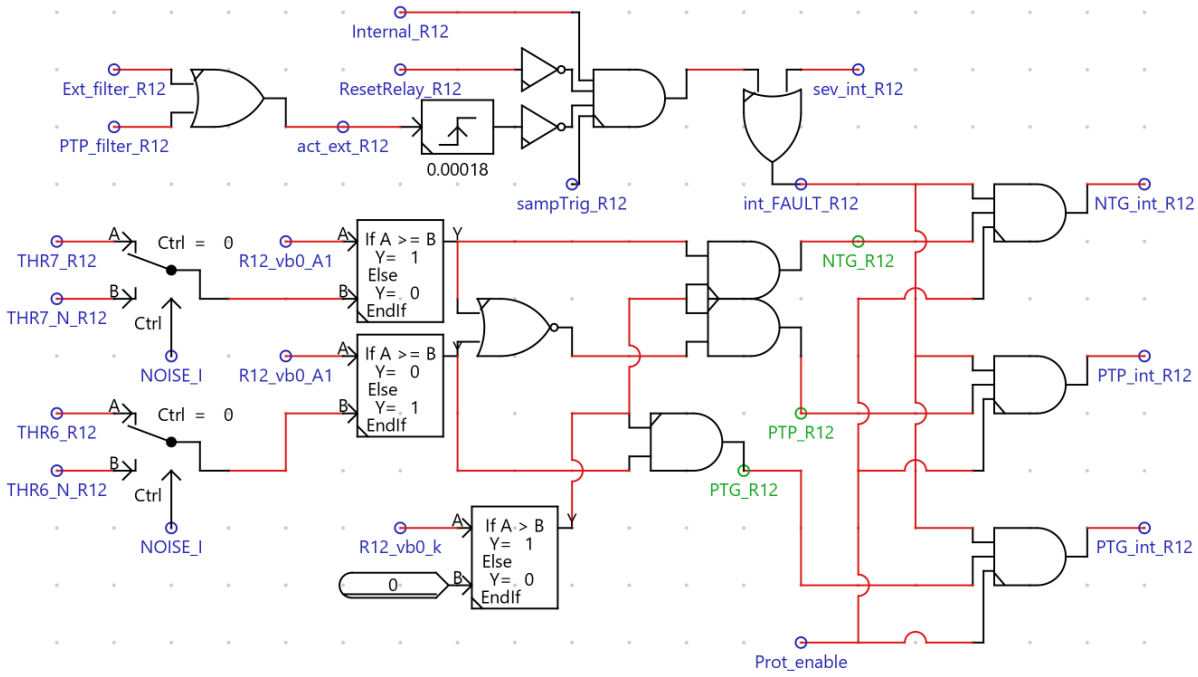


Figure 5.11: RSCAD implementation of the protection scheme.

The classified fault created in Figure 5.11 is used to create the DCCB trip signals in Figure 5.12. It is similar to the fault application logic explained in subsection 3.2.1. The trip signals are a direct consequence of the classified fault signal. A ground fault trips its corresponding positive or negative DCCB, whereas a PTP fault gives the trip signals for both. The flipflops are reset when the reclose signal is given, similar to the forced protection. The relay trip signals are passed to the DCCBs if automatic protection is enabled by a switch in the runtime environment in the model. If this is disabled, forced protection is used to trip the correct DCCBs.

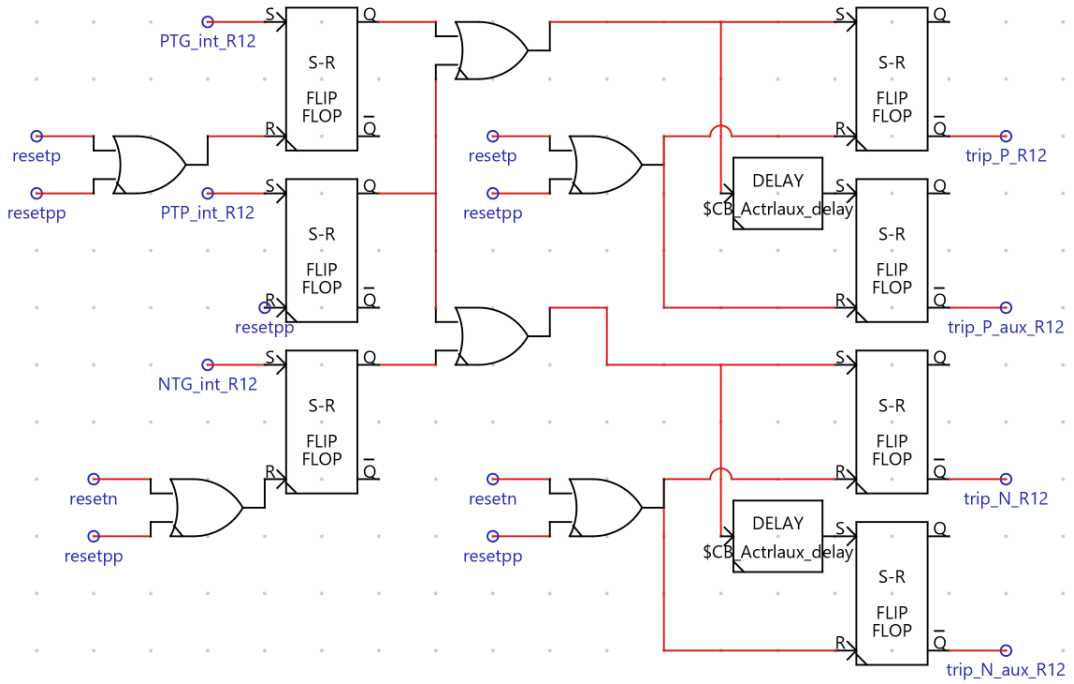


Figure 5.12: RSCAD implementation for the creation of DCCB trip signals.

The final step of the implementation design iteration is installing the relay at all six locations and setting the protection thresholds. This is done by collecting A1 parameters for a large number of fault scenarios, by running the main performance test. During this test he fault location is varied according to the test setup given in subsection 3.3.2. PTG, NTG, and PTP faults are applied at each location. The fault impedance is varied for 0.0001Ω, 0.1Ω, 1.0Ω, 10.0Ω, 50.0Ω, 100.0Ω, 200.0Ω, 250.0Ω, and 300.0Ω. This leads to 240 total different fault scenarios. Based on the data the design decision was made to optimize for a maximum fault impedance of 200Ω, as higher faults were not possible to fully separate, even with the proposed filtering functionalities. For the noise case, the system is optimized for a maximum fault impedance of 50Ω, as parameter fitting results for higher HIF are unreliable due to the noise. The collected threshold settings are shown in Table 5.5.

Threshold	R12	R13	R21	R23	R31	R32
THR1	-25.0	-25.0	-25.0	-25.0	-25.0	-25.0
THR2	-110.0	-110.0	-125.0	-120.0	-110.0	-110.0
THR3	-100.0	-100.0	-100.0	-100.0	-100.0	-100.0
THR4	0.0	0.0	0.0	0.0	0.0	0.0
THR5	0.0	0.0	0.0	2.0	0.0	1.9
THR6	-5.5	-5.5	-5.5	-5.5	-5.5	-5.5
THR7	5.5	5.5	5.5	5.5	5.5	5.5
THR1_N	-75.0	-70.0	-70.0	-70.0	-70.0	-70.0
THR2_N	-200.0	-140.0	-190.0	-150.0	-190.0	-190.0
THR3_N	-200.0	-140.0	-190.0	-150.0	-190.0	-190.0
THR4_N	X	X	X	X	X	X
THR5_N	X	X	X	X	X	X
THR6_N	-8.5	-8.5	-10.5	-10.5	-8.5	-8.5
THR7_N	8.5	8.5	10.5	10.5	8.5	8.5

Table 5.5: Protection threshold settings.

5.3. Results and Analysis

5.3.1. Protection Performance Testing

To asses the protection system performance, the test case outlined in subsection 3.3.2 is performed. PTG, NTG, and PTP faults are applied at all 10 fault locations. The fault resistance is varied from 0.0001Ω to 250Ω. Because the relays are optimised up to 200Ω, the 250Ω HIF case is tested to show the relay boundary. It is not expected to have a high accuracy. Each test is executed three times to account for differences caused by the real time platform. This results in 90 simulation cases for each fault resistance. During the simulations, both fault detection and the corresponding action are recorded. The performance accuracy is based on the correct actions taken. The performance testing results are shown in Table 5.6. It is seen that the detection rate of the faults is 100%. Including the 250Ω cases, the total accuracy is 98.19%. Without the 250Ω cases, this improves to 99.91%. Considering only the simulation cases up to and including 200Ω, four incorrect actions are taken (out of the in total 4320 actions):

- A 0.0001Ω mistake is made by R_{23} (PTG), which is due to an external fault classified as internal. The fitted V_{b1} A1 parameter of accompanying relay R_{21} is just above the threshold THR5.
- A 0.0001Ω mistake is made by R_{32} (PTG), which is due to an external fault classified as internal. The fitted V_{b1} A1 parameter of accompanying relay R_{31} is just above the threshold THR5.
- Two 200Ω mistakes are made by R_{32} (1 PTG, 1 NTG fault), which are due to internal faults classified as external. The fitted V_{b1} A1 parameter of accompanying relay R_{31} is just below the threshold THR5.

Relay		**Fault Resistance (Ω)**									**Total**
		0.0001	**0.1**	**1**	**10**	**50**	**100**	**150**	**200**	**250**	
R12	**Detected Faults**	90	90	90	90	90	90	90	90	90	810 / 810
	Correct Actions	90	90	90	90	90	90	90	90	78	798 / 810
	Accuracy	100%	100%	100%	100%	100%	100%	100%	100%	86.67%	98.52%
R13	**Detected Faults**	90	90	90	90	90	90	90	90	90	810 / 810
	Correct Actions	90	90	90	90	90	90	90	90	72	792 / 810
	Accuracy	100%	100%	100%	100%	100%	100%	100%	100%	80.00%	97.78%
R21	**Detected Faults**	90	90	90	90	90	90	90	90	90	810 / 810
	Correct Actions	90	90	90	90	90	90	90	90	78	798 / 810
	Accuracy	100%	100%	100%	100%	100%	100%	100%	100%	86.67%	98.52%
R23	**Detected Faults**	90	90	90	90	90	90	90	90	90	810 / 810
	Correct Actions	89	90	90	90	90	90	90	90	79	798 / 810
	Accuracy	98.89%	100%	100%	100%	100%	100%	100%	100%	87.78%	98.52%
R31	**Detected Faults**	90	90	90	90	90	90	90	90	90	810 / 810
	Correct Actions	90	90	90	90	90	90	90	90	72	792 / 810
	Accuracy	100%	100%	100%	100%	100%	100%	100%	100%	80.00%	97.78%
R32	**Detected Faults**	90	90	90	90	90	90	90	90	90	810 / 810
	Correct Actions	89	90	90	90	90	90	90	88	77	794 / 810
	Accuracy	98.89%	100%	100%	100%	100%	100%	100%	97.78%	85.56%	98.02%

Table 5.6: Performance Metrics for Different Fault Resistances

5.3.2. Protection Performance Testing by considering Noise

Similar simulation tests are executed for the noise case. The fault resistance is varied form 0.0001 Ω to 100 Ω. Because the relays are optimized up to 50 Ω, the 100 Ω HIF case is performed to show the relay boundary. Each test is executed three times. During the simulations, both fault detection and the corresponding action are recorded. Also, the detection rate is found for internal and external faults specifically. The performance testing results are shown in Table 5.7. It can be seen that the fault detection is deteriorated due to the addition of noise. However, this is only the case for external faults. Out of all 1080 internal fault cases, only one internal fault (R_{32}, 50Ω, NTG, location 8) has not been detected, leading to an internal fault detection rate of 99.91%. An undetected internal fault will always lead to a wrong relay decision, as the analysis is not performed and thus no trip signals are created. External faults are going undetected more often than internal faults, especially with increasing fault resistance. The total external fault detection rate is 87.54%. Undetected external faults do not cause relay inaccuracies, as trip signals should not be created. Almost all incorrect actions happen for 100Ω faults. This is expected, as the relays are optimised for faults up to 50Ω. The total protection system accuracy is 96.76%. Without the 100Ω cases, the total accuracy increases to 99.85%. Considering only the simulation cases up to and including 50Ω, four incorrect actions are taken (out of the in total 2455 detected faults):

- A 50Ω mistake is made by R_{12} (NTG), which is due to an backward external fault classified as internal. The fitted V_{b1} A1 parameter is just below the threshold THR1_N, possibly due to noise distortion.
- A 0.1Ω mistake is made by R_{23} (NTG), which is due to an backward external fault classified as internal. The fitted V_{b1} A1 parameter is just below the threshold THR1_N. possibly due to noise distortion.
- A 0.1Ω mistake is made by R_{13} (PTP), which is due to the fault classified as NTG instead of PTP. Therefore, only the negative pole trip signal was created. The fitted V_{b0} A1 parameter is just above THR7_N. Changing the threshold value from -8.5 to -10.5 solves this mistake, as there is room for this threshold to be increased without causing NTG faults being classified as PTP.
- A 50Ω fault is not detected by R_{32} (NTG). Analysis of this internal fault is not performed. Hence, no trip signal is created.

Relay		Fault Resistance (Ω)						Total
		0.0001	**0.1**	**1**	**10**	**50**	**100**	
R12	**Detected Internal**	100%	100%	100%	100%	100%	100%	100%
	Detected External	98.41%	100%	98.41%	98.41%	95.24%	80.95%	95.24%
	Correct Actions	90	90	90	90	89	72	521 / 540
	Accuracy	100%	100%	100%	100%	98.89%	80.00%	96.48%
R13	**Detected Internal**	100%	100%	100%	100%	100%	100%	100%
	Detected External	90.74%	92.59%	90.74%	81.48%	72.22%	64.81%	82.10%
	Correct Actions	90	90	90	90	90	69	519 / 540
	Accuracy	100%	100%	100%	100%	100%	76.67%	96.11%
R21	**Detected Internal**	100%	100%	100%	100%	100%	100%	100%
	Detected External	84.13%	77.78%	82.54%	71.43%	77.78%	61.90%	75.93%
	Correct Actions	90	90	90	90	90	72	522 / 540
	Accuracy	100%	100%	100%	100%	100%	80.00%	96.67%
R23	**Detected Internal**	100%	100%	100%	100%	100%	100%	100%
	Detected External	100%	100%	100%	100%	100%	69.84%	94.97%
	Correct Actions	90	89	90	90	90	82	531 / 540
	Accuracy	100%	98.89%	100%	100%	100%	91.11%	98.33%
R31	**Detected Internal**	100%	100%	100%	100%	100%	100%	100%
	Detected External	85.19%	79.63%	83.33%	83.33%	92.59%	66.67%	81.79%
	Correct Actions	90	89	90	90	90	63	512 / 540
	Accuracy	100%	98.89%	100%	100%	100%	70.00%	94.81%
R32	**Detected Internal**	100%	100%	100%	100%	96.30%	100%	99.38%
	Detected External	100%	100%	98.41%	100%	93.65%	79.37%	95.24%
	Correct Actions	90	90	90	90	89	81	530 / 540
	Accuracy	100%	100%	100%	100%	98.89%	90.00%	98.15%

Table 5.7: Performance Metrics for Different Fault Resistances (with noise).

5.3.3. Time tests

Because relay operation speed is an important parameter for HVDC protection, several fault scenarios are simulated in detail to find relay timing characteristics. Given that there are five possible relay decision paths, each one is tested. Everything tested from the perspective of relay R_{12}.

Severe Internal Fault

To find the relay decision timings for a severe internal fault, a 1.0Ω PTG fault is applied at location 2. The timing graph is shown in Figure 5.13. The fault is applied at $t = 0$ ms. The propagation time is short because the fault is internal, so the detection is realized at $t = 1.08$ ms, raising the `SampTrig` signal. As soon as the parameter fitting for V_{b1} is completed at $t = 2.76$ ms, both the `Severe` signal and the temporary `Internal` signal are raised. The `Severe` signal skips waiting time and causes the trip signal `INTERNAL` to be raised immediately. Because the `PTG` signal was already high, the trip signal is sent to the positive pole DCCB. The relay operation time from detection until trip signal creation is 1.68 ms in total.

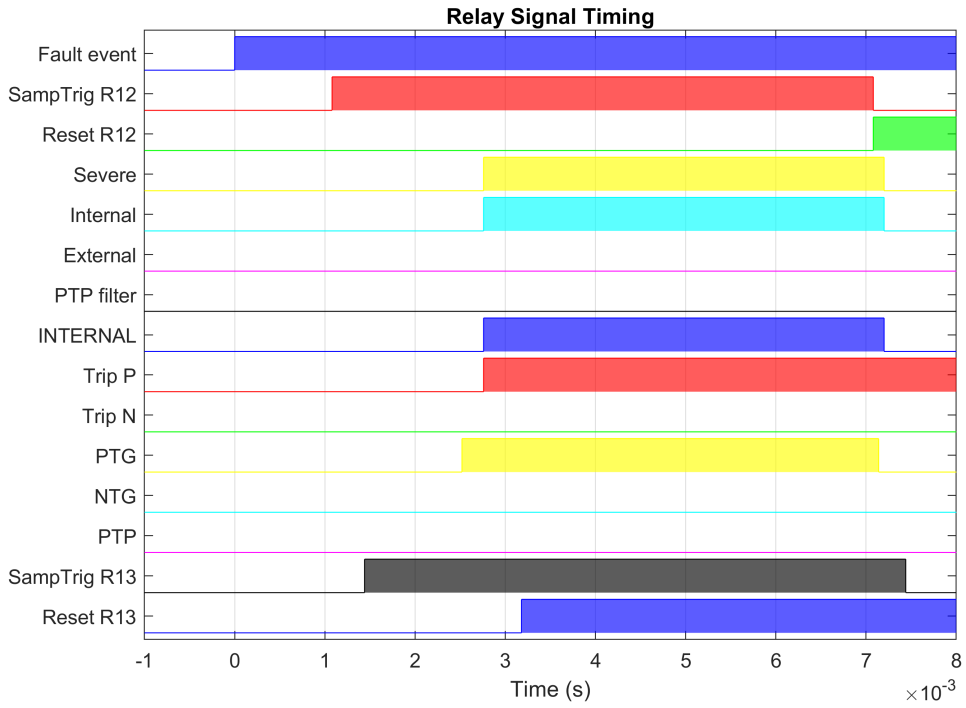


Figure 5.13: Relay timing following a severe internal PTG fault.

Internal Fault

To find the relay decision timings for an internal non-severe fault, a 50Ω PTG fault is applied at location 2. The timing graph is shown in Figure 5.14. Fault detection for R_{12} occurs at $t = 1.08$ ms. The V_{b0} analysis finishes 1.44 ms later, detecting a PTG fault. At $t = 2.70$ ms, 1.62 ms after detection, the `Internal` signal is raised. However, the `Severe` signal stays low. Hence, the relay starts the forced waiting process for signals coming from the accompanying relay R_{13}. R_{13} detects the fault at $t = 1.62$ ms and decides for an forward external fault at $t = 3.30$ ms, therefore forcing the `Reset R13` signal. This means that at this time, the signal is ready to be used by R_{12}, and the decision could have been made at $t = 3.30$ ms, just 2.22 ms after fault detection. However, as the forced `$far_wait` delay signal in Figure 5.10 is set to 1 ms, the decision is taken only at $t = 3.84$ ms, 2.80 ms after fault detection. Although there are cases where this long wait time is necessary, it is too long in this case. Functionality can be added to overwrite the wait time as soon as the signal from the accompanying relay is ready, reducing the relay operation speed for non-severe internal faults to 2.22 ms.

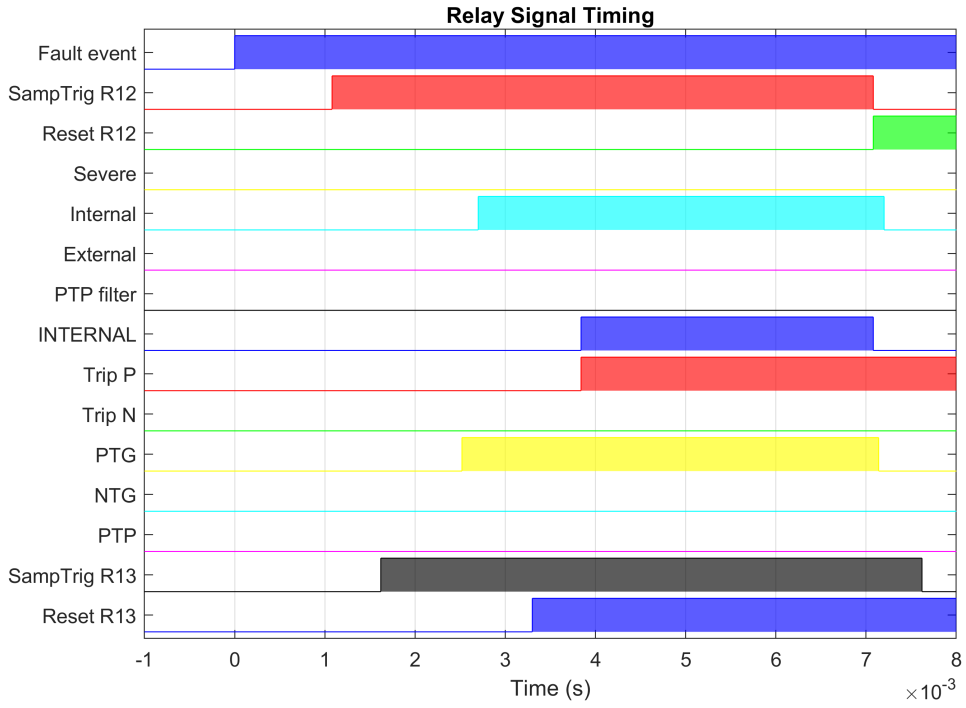


Figure 5.14: Relay timing following an internal PTG fault.

For some non-severe fault cases with resistances higher than 100Ω the `$far_wait` time must be set to 3 ms instead of 1 ms shown in Figure 5.14. In these cases, the V_{b1} fault response is difficult to detect, especially for the accompanying relay. Cases have been found for which the accompanying relay detects the fault up to 2 ms later. Exceeding the wait time and thus not acting on data from an accompanying relay is not a problem for internal faults, as a trip signal will be created successfully. However, for external faults, the trip signal will also be created, which is not correct. This leads to the relay being over-sensitive. Increasing the `$far_wait` time to 3 ms solves this issue, but imposes longer relay operation times with a maximum of 4.5 ms in total.

Forward External Fault

To find the relay decision timings for a forward external fault, a 1.0Ω PTG fault is applied at location 5. The timing graph is shown in Figure 5.15. Fault detection for R_{12} occurs at $t = 1.44$ ms. It is seen that the `Internal` signal is never set to high, so a trip signal will never be created. In addition, the `External` signal is set to high, as R_{13} correctly detects an internal fault. This signal coming from R_{13} makes R_{12} decide to reset immediately, signaling the analysis is over. Notice that setting the `External` signal to high is superfluous in this case, but happens nonetheless. The total relay operation time from fault detection to relay resetting is 1.74 ms.

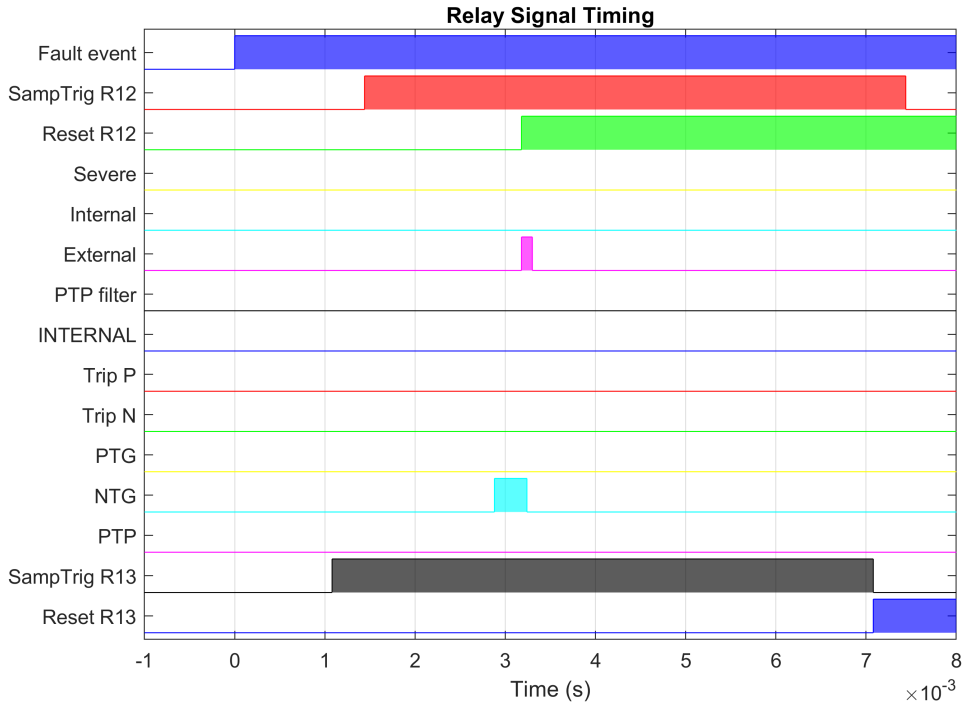


Figure 5.15: Relay timing following a forward external PTG fault.

Backward External Fault

To find the relay decision timings for a backward external fault, a 1.0Ω PTG fault is applied at location 8. The timing graph is shown in Figure 5.16. Fault detection for R_{12} occurs at $t = 2.52$ ms. At $t = 4.26$ ms, the `Internal` signal is raised, however, because `Severe` remains low, the forced waiting period is started. At $t = 5.28$ ms (0.96ms after `Internal` was set to high), relay R_{13} is done with its V_{b1} analysis and sets the `External` signal to high, signaling the fault is actually external from the point of R_{12}. Hence, no trip signal is created and the relay is reset. The total relay operation time from fault detection to relay resetting is 2.76 ms.

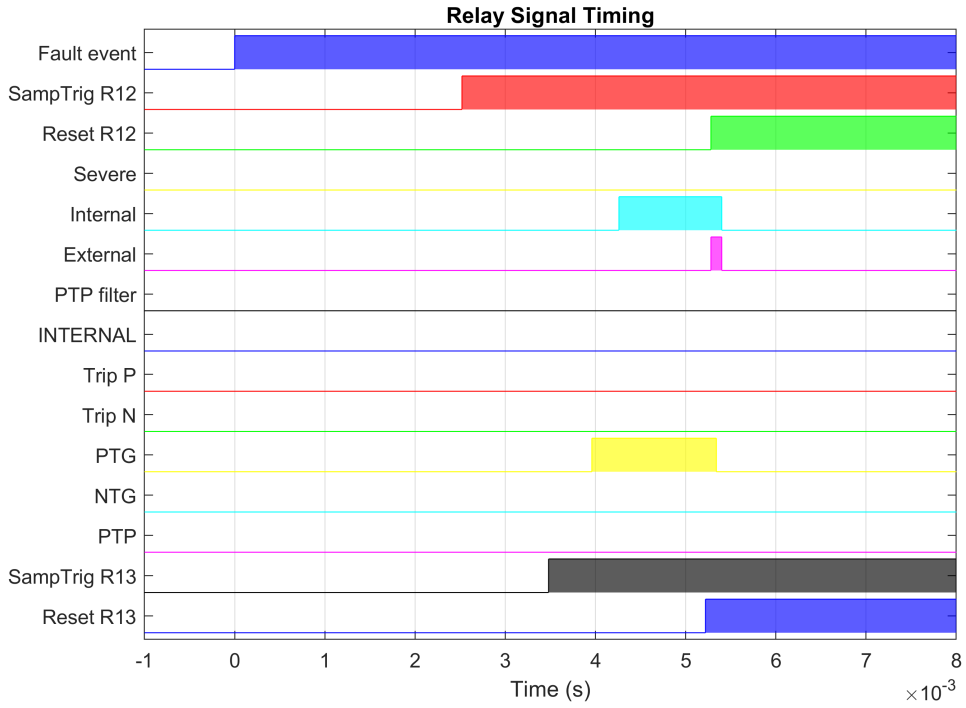


Figure 5.16: Relay timing following a backward external PTG fault.

Backward PTP Fault (with Noise)

To find the relay decision timings for a case using the PTP filter, a 1.0Ω PTP fault is applied at location 10. Noise is enabled. The timing graph is shown in Figure 5.17. Fault detection occurs at $t = 3.48$ ms. The parameter fitting analysis of V_{b0} is finished at $t = 4.92$ ms, showing the fault is of type PTP. At $t = 5.16$ ms, the V_{b1} analysis is also finished. The fault is classified as internal, but not as severe. The value of the V_{b1} A1 parameter is inside the PTP filter window, and because the fault type is classified as PTP, the relay correctly decides the fault is external by raising the `PTPfilter` signal. Hence, no trip signals are created. Notice that the signals for R_{13} remain zero, as communication between relays is not performed when noise is enabled. Hence, no waiting times are present. The total relay operation time from fault detection to relay resetting is 1.68 ms.

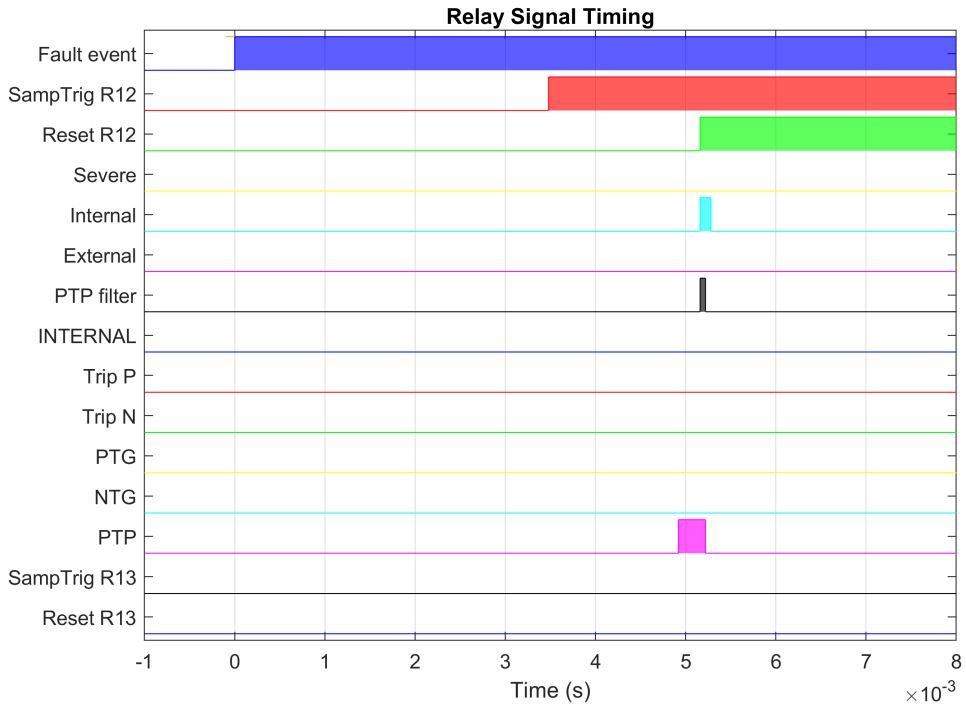


Figure 5.17: Relay timing following a backward external PTP fault (with noise).

5.3.4. Deadzone

It is important to find a relay's deadzone. A deadzone is defined as the region for which a fault is at such short distance to the relay and is not detectable. The shortest section of a cable model in RSCAD is 15 km because of a minimum travel time requirement inside the software. Hence, this is the cable length for the deadzone test. In addition, a fault is applied right at the DCCBs terminal (relay R_{12} location), with a cable length of 0 km. For both fault locations, the fault type is varied as PTG, NTG, and PTP. The fault resistance is varied as 0.0001Ω, 1.0Ω, and 200Ω. Each fault is applied three times. As seen from R_{12}, all simulated faults are internal. Hence, a correct protection operation by R_{12} is required to keep the system stable. The results of the simulations are found in Table 5.8. The table the percentage of the simulations in which all relays (including testing relay R_{12}) detected the fault and acted correctly.

	$l = 0$ km			$l = 15$ km		
	0.0001 Ω	1.0 Ω	200 Ω	0.0001 Ω	1.0 Ω	200 Ω
PTG	100%	100%	100%	100%	100%	100%
NTG	100%	100%	100%	100%	100%	100%
PTP	100%	100%	100%	100%	100%	100%

Table 5.8: Deadzone test results.

It is seen that all simulated faults were successfully identified by the relays, with corresponding relaying actions taken successfully. Therefore, it is concluded that for this method of DC line protection no deadzone exists.

5.4. Discussion

The protection performance shown in section 5.3 is good. Without noise, the system is able to identify faults with an impedance up to 200Ω with an accuracy of 99.91%. With noise, the system is able to identify faults with an impedance up to 50Ω with an accuracy of 99.85%. Still, improvements can be made to the relays. A higher sampling frequency results in more samples in the window, possibly improving the parameter fitting such that the fitted parameters are of better quality. This should result in a better distinction between faults. In addition, more samples in the data window will improve robustness to noise, as the parameter fitting will be more accurate. The amplitude shift caused by the random nature of the noise is reduced because of the weak law of large numbers, which states that for a sufficiently large number of samples of noise, the signal will be close toe the expected value with high probability. Effectively, this implies that increasing the number of samples mitigates the impact of noise.

To optimise the parameter fitting, a different target function can be used for fitting external faults. Although this might lead to better results, it adds a lot of complexity in the system, potentially leading to the analysis lasting for a longer time. In addition, the parameters of the AMLM are not optimized. Although this is a complex process and likely does not lead to a significant increase in performance, it can lead to faster processing.

Setting threshold values for a large variety of fault impedances presents a challenge. The V_{b1} A1 parameter is similar for low impedance backward external faults and high impedance internal faults. Increasing the fault impedance over 50Ω presents overlap in these values, implying that this value alone is not enough for selective protection. The data from the accompanying relay can be used for further fault locale identification, but imposes a waiting time. Currently, this waiting time is forced, but a control sequence should be implemented to bypass the waiting time as soon as data is available. This will make sure all internal faults are correctly identified within 2.5 ms. Optimising the waiting time could also enable communication with the accompanying relay when noise is enabled. Data will be available when it detects a fault. If the accompanying relay does not detect the fault, the main relay will decide for itself when the waiting time is over. This mechanism could increase the relays performance for faults with an impedance over 50Ω.

From a realistic perspective, a threshold-based system is challenging to implement. Because the threshold settings are data-driven, numerous simulations must be executed to determine the appropriate values. This approach is impractical for a real system, as a single short circuit can cause extensive damage. Therefore, the real HVDC system must be modeled to identify the protection settings. This is feasible only if the models' absolute quality and accuracy are guaranteed. The model used in this thesis is very accurate, but for real-world application each HVDC project needs a project specific model to run the simulations, specifically with project details such as layout and accurate component specifications.

For both the noisy and noise-free scenarios, the accuracy is not 100%. In the noise-free case, this is due to the fault impedance boundary pushed to the limit, or slightly beyond. The threshold values are set in such a way that small changes in fault response imposed by the real-time platform or parameter fitting inaccuracy can adversely affect the relay decision. Classifying an external fault as an internal one, generates trip signals where they are not needed. Essentially, the protection is acting over-sensitive. This is not necessarily a problem, as expensive system components are well protected. However, when internal faults are classified as external, no trip signals are created, and system damage is probable. In this case, the protection is acting with a lack of sensitivity. Having a protection accuracy of 100% is an ongoing topic of debate, as HIF have a low probability of occurrence in subsea cable systems. Hence, in the rare case such a fault occurs, backup AC protection is able to shut down the HVDC system completely, after which it can be restored one healthy DC connection at the time. Based on this conviction, the accuracies demonstrated using the proposed method are sufficient.

The proposed method works well for a three-terminal system as shown. Future research must conclude if this method functions effectively for DC grids with more terminals and connections. It is hypothesized that it will, because of two main reasons. The first is that faults that are even further away from an arbitrary relay than those shown in this work are simple to correctly identify as external. The most severe external faults to identify are those that are backward external. The faults investigated in this work occur on a neighbouring cable. In this case, the fault is similar to an internal fault, but it is attenuated by two line-end inductors and a converter station. If a fault would occur on a cable after this one, as seen from a relay, the fault response would be attenuated by four line-end conductors, two converter stations, and at least two full-length cables. Therefore, the attenuation level is high, and fault locale identification is feasibly achievable. Secondly, for a converter station with more than two connections to other converter stations, more relays are present at the same geographical location. Hence, there is more data from accompanying relays present to utilize in the protection scheme, on which the trip signal creations can be based. While potentially inducing longer wait times because the data from accompanying relays is not yet available (especially for far away HIF), a higher relay accuracy is feasible.

5.4.1. Future research

The DC line protection method presented in this thesis is promising, however, it is compromised due to RTDS limitations. The relatively slow sampling speed affects parameter fitting accuracy and the fixed time-step slows down the algorithm tremendously. Integrating the protection method on field-programmable gate array (FPGA) units connected to RTDS could solve both problems. The small timestep environment in RSCAD can be used to sample the measurement signals with a higher frequency and pass the signals to the external component. The AMLM algorithm can run as fast as possible on the FPGA unit, as it is not limited by RTDS time steps. There will not be dead time between AMLM iterations. This can lead to faster processing or higher accuracy. The relay logic can be executed on the FPGA unit as well. It must be researched if the additional communication time between the FPGA units and the RTDS computers has a low enough latency such that this method can be executed successfully. However, given the extensive work done by RTDS engineers testing relays, this is not believed to be an issue.

The proposed method could also be performed on an external fast computing device. All relay logic can be programmed in a language with quick operation times, such as C. The AMLM algorithm fits three parameters to the measured TW. However, only one of these parameters (A1) is used for simplicity. It is hypothesized that machine learning techniques such as neural networks could be used to process more available data, possibly pushing the relay performance to correctly identify HIF over 200Ω. This does mean that the trip signal creation takes longer due to an additional algorithm, though when

programmed in a quick language such as C/C++ and trained well, the minimal additional computation time might be worth the increase in relay performance.

6

Conclusion

This thesis deals with developing a TW parameter fitting-based approach toward fault location and protection following a DC cable fault in a multi-terminal HVDC network. The research is conducted on a three-terminal HVDC network model in the RTDS environment. A custom parameter fitting RSCAD control component is developed based on the AMLM algorithm. The line-mode backward traveling voltage wave following an internal DC cable fault is analyzed for the fault location estimation. The same signal and its zero-mode counterpart are analyzed to create a fully selective protection system.

6.1. Fault Location

Research Question 1: *Can the proposed fault location method using the AMLM parameter fitting algorithm be successfully integrated into the RTDS / RSCAD environment?*

Despite the successful implementation of the AMLM parameter fitting in the RSCAD environment, it has not been possible to successfully implement the proposed fault location, as the average absolute error is 9.80% (corresponding to 29.4 km) for faults located 50 km along a 300 km cable. The integration of the AMLM-based parameter fitting algorithm into an RSCAD component imposes significant drawbacks, such as a low sampling frequency and compromised iterative parameter fitting process due to the real-time nature of RTDS. The objective of an average absolute error in location estimation of 1.0% has not been achieved. Although intermediate results were promising and the cable attenuation effect was shown, estimation of the actual location remains challenging. Efforts to improve the result by specific truncation of the measured signal and optimal choice of cable parameter τ reduced the average error to 5.15% (corresponding to 15.45 km), which is still insufficient.

Research Question 2: *Is the interventionless and instantaneous location estimation worth the loss in accuracy?*

No, the benefits of interventionless and instantaneous location estimation do not justify the loss in accuracy. Because the location is practically only useful for maintenance crews to repair the damaged cable which takes days or months, fast fault location is not required. Instead, care must be taken to provide an accurate estimation.

6.2. Protection

Research Question 3: *Can the V_{b1} signal be used for fault discrimination using AMLM-based parameter fitting when integrated into the RTDS environment?*

The proposed TW parameter fitting DC line fault protection method can provide fast-acting protection with full selectivity. The information provided in V_{b1} is extracted by parameter fitting and the associated parameter is used to discriminate between internal or external faults. The boundary line-end inductors used in the HVDC system model provide adequate signal attenuation to discriminate the faulty from healthy state. The V_{b0} signal is analysed to determine the fault

type. A distinction is made between severe and non-severe faults. Severe faults are typically low-impedance faults. Non-severe faults can be either internal or external, for which the relay operation is supported by data from the accompanying relay on the neighboring cable. This is not considered aunit protection, as both relays are present at the same geographical location. The proposed protection scheme can identify faults with a fault resistance of up to 200Ω with an accuracy of 99.91% and provide corresponding DCCB trip signals. Severe internal faults are isolated in 1.68 ms after the fault detection. Non-severe internal faults are typically isolated in 3.84 ms. The proposed protection method eliminates relay dead zones.

Research Question 4: *Is the algorithm and proposed protection method robust enough to account for measurement noise?*

To find the noise robustness of the protection method white Gaussian noise with an SNR level of 40 dB is added to the measured voltages and currents. Relay fault detection capability is reduced accordingly, as it may not trigger the protection based on noise. This results in data from the accompanying relay being rendered unusable, as it might fail to detect fault cases and provide no data accoridngly. This reduces the overall relay performance of the protection in case of high-impedance faults. However, the proposed protection method can detect faults with an impedance up to 50Ω with 99.85% accuracy, and create corresponding DCCB trip signals. Relay operation time is 1.68 ms.

References

[1] Eurostat (European Commission), *Energy, transport and environment statistics*, 2020 ed. Luxembourg: European Union, 2020.

[2] C. Fetting, "The european green deal," ESDN Report, Dec. 2020, ESDN Office, Vienna.

[3] BVGA Associates, Recognis, WindEurope, "Our energy, our future: How offshore wind will help europe go carbon-neutral," WindEurope, Nov. 2019.

[4] North Sea Wind Power Hub consortium, "Hubs and spokes - viable beyond theory," Oct. 2022.

[5] H. Mjahed, "The north sea: Europe's energy powerhouse," Policy Center for the New South, Feb. 2023.

[6] J. Song-Manguelle, M. Todorovic, S. Chi, S. Gunturi, and R. Datta, "Power transfer capability of hvac cables for subsea transmission and distribution systems," *IEEE Transactions on Industry Applications*, vol. 50, pp. 2382–2391, Jul. 2014.

[7] J. Liang, O. Bellmunt, J. Ekanayake, and N. Jenkins, "Control of multi-terminal vsc-hvdc transmission for offshore wind power," Oct. 2009, pp. 1–10.

[8] G. Chaffey, D. Van Hertem, N. Svensson, S. Norrga, and L. Ängquist, "The impact of point-to-point vsc-hvdc protection on short-term ac system voltage," in *2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, 2018, pp. 1–6. DOI: 10.1109/ISGTEurope.2018.8571812.

[9] M. Nedd, J. Browell, K. Bell, and C. Booth, "Containing a credible loss to within frequency stability limits in a low-inertia gb power system," *IEEE Transactions on Industry Applications*, vol. 56, no. 2, pp. 1031–1039, 2020. DOI: 10.1109/TIA.2019.2959996.

[10] W. Wang, X. Yan, S. Li, L. Zhang, J. Ouyang, and X. Ni, "Failure of submarine cables used in high-voltage power transmission: Characteristics, mechanisms, key issues and prospects," *IET Generation, Transmission & Distribution*, vol. 15, no. 9, pp. 1387–1402, 2021. DOI: https://doi.org/10.1049/gtd2.12117. eprint: https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/gtd2.12117. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/gtd2.12117.

[11] G. Liu, F. Xu, Z. Xu, Z. Zhang, and G. Tang, "Assembly hvdc breaker for hvdc grids with modular multilevel converters," *IEEE Transactions on Power Electronics*, vol. 32, no. 2, pp. 931–941, 2017. DOI: 10.1109/TPEL.2016.2540808.

[12] Y. Xue and Z. Xu, "On the bipolar mmc-hvdc topology suitable for bulk power overhead line transmission: Configuration, control, and dc fault analysis," *IEEE Transactions on Power Delivery*, vol. 29, no. 6, pp. 2420–2429, 2014. DOI: 10.1109/TPWRD.2014.2352313.

[13] X. Han, W. Sima, M. Yang, L. Li, T. Yuan, and Y. Si, "Transient characteristics under ground and short-circuit faults in a ±500 kV mmc-based hvdc system with hybrid dc circuit breakers," *IEEE Transactions on Power Delivery*, vol. 33, no. 3, pp. 1378–1387, 2018. DOI: 10.1109/TPWRD.2018.2795800.

[14] Z.-Y. He, K. Liao, X.-P. Li, S. Lin, J.-W. Yang, and R.-K. Mai, "Natural frequency-based line fault location in hvdc lines," *IEEE Transactions on Power Delivery*, vol. 29, no. 2, pp. 851–859, 2014. DOI: 10.1109/TPWRD.2013.2269769.

[15] G. Song, X. Chu, X. Cai, S. Gao, and M. Ran, "A fault-location method for vsc-hvdc transmission lines based on natural frequency of current," *International Journal of Electrical Power & Energy Systems*, vol. 63, pp. 347–352, 2014, ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2014.05.069. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061514003408.

[16] J. He, B. Li, Q. Sun, *et al.*, "The improved fault location method based on natural frequency in mmc-hvdc grid by combining fft and music algorithms," *International Journal of Electrical Power & Energy Systems*, vol. 137, p. 107 816, 2022, ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2021.107816. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061521010322.

[17] D. Li and A. Ukil, "Fault location estimation in voltage-source-converter-based dc system: The L location," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 11, pp. 11 198–11 209, 2022. DOI: 10.1109/TIE.2021.3125662.

[18] J. Xu, Y. Lü, C. Zhao, and J. Liang, "A model-based dc fault location scheme for multi-terminal mmc-hvdc systems using a simplified transmission line representation," *IEEE Transactions on Power Delivery*, vol. 35, no. 1, pp. 386–395, 2020. DOI: 10.1109/TPWRD.2019.2932989.

[19] T. Bi, S. Wang, and K. Jia, "Single pole-to-ground fault location method for mmc-hvdc system using active pulse," *IET Generation, Transmission & Distribution*, vol. 12, no. 2, pp. 272–278, 2018. DOI: https://doi-org.tudelft.idm.oclc.org/10.1049/iet-gtd.2017.0116. eprint: https://ietresearch-onlinelibrary-wiley-com.tudelft.idm.oclc.org/doi/pdf/10.1049/iet-gtd.2017.0116. [Online]. Available: https://ietresearch-onlinelibrary-wiley-com.tudelft.idm.oclc.org/doi/abs/10.1049/iet-gtd.2017.0116.

[20] T. Wang, G. Song, and K. S. T. Hussain, "Adaptive single-pole auto-reclosing scheme for hybrid mmc-hvdc systems," *IEEE Transactions on Power Delivery*, vol. 34, no. 6, pp. 2194–2203, 2019. DOI: 10.1109/TPWRD.2019.2921674.

[21] G. Song, J. Hou, B. Guo, T. Wang, B. Masood, and S. T. H. Kazmi, "Single-ended active injection for fault location in hybrid mmc-hvdc systems," *International Journal of Electrical Power & Energy Systems*, vol. 124, p. 106 344, 2021, ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2020.106344. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061519343133.

[22] L. Yuansheng, W. Gang, and L. Haifeng, "Time-domain fault-location method on hvdc transmission lines under unsynchronized two-end measurement and uncertain line parameters," *IEEE Transactions on Power Delivery*, vol. 30, no. 3, pp. 1031–1038, 2015. DOI: 10.1109/TPWRD.2014.2335748.

[23] S. Lan, M.-J. Chen, and D.-Y. Chen, "A novel hvdc double-terminal non-synchronous fault location method based on convolutional neural network," *IEEE Transactions on Power Delivery*, vol. 34, no. 3, pp. 848–857, 2019. DOI: 10.1109/TPWRD.2019.2901594.

[24] P. Chen, B. Xu, J. Li, and Y. Ge, "Modern travelling wave based fault location techniques for hvdc transmission lines," *Transactions of Tianjin University*, vol. 14, no. 2, pp. 139–143, Apr. 1, 2008. DOI: 10.1007/s12209-008-0025-7. [Online]. Available: https://doi.org/10.1007/s12209-008-0025-7.

[25] C. Zhang, G. Song, T. Wang, and L. Yang, "Single-ended traveling wave fault location method in dc transmission line based on wave front information," *IEEE Transactions on Power Delivery*, vol. 34, no. 5, pp. 2028–2038, 2019. DOI: 10.1109/TPWRD.2019.2922654.

[26] L. Liu, F. Xie, M. Popov, Z. Hao, and A. Lekić, "Single-ended dc fault location method for mmc-based hvdc power system using adaptive multi-step levenberg-marquardt algorithm," in *2023 IEEE Belgrade PowerTech*, 2023, pp. 1–6. DOI: 10.1109/PowerTech55446.2023.10202991.

[27] L. Chen, X. Lin, Z. Li, *et al.*, "Similarity comparison based high-speed pilot protection for transmission line," *IEEE Transactions on Power Delivery*, vol. 33, no. 2, pp. 938–948, 2018. DOI: 10.1109/TPWRD.2017.2731994.

[28] S.-p. Gao, Q. Liu, and G.-b. Song, "Current differential protection principle of hvdc transmission system," *IET Generation, Transmission & Distribution*, vol. 11, no. 5, pp. 1286–1292, 2017. DOI: https://doi-org.tudelft.idm.oclc.org/10.1049/iet-gtd.2016.1380. eprint: https://ietresearch-onlinelibrary-wiley-com.tudelft.idm.oclc.org/doi/pdf/10.1049/iet-gtd.2016.1380. [Online]. Available: https://ietresearch-onlinelibrary-wiley-com.tudelft.idm.oclc.org/doi/abs/10.1049/iet-gtd.2016.1380.

[29] J. Sneath and A. D. Rajapakse, "Fault detection and interruption in an earthed hvdc grid using rocov and hybrid dc breakers," *IEEE Transactions on Power Delivery*, vol. 31, no. 3, pp. 973–981, 2016. DOI: 10.1109/TPWRD.2014.2364547.

[30] J. Liu, N. Tai, and C. Fan, "Transient-voltage-based protection scheme for dc line faults in the multiterminal vsc-hvdc system," *IEEE Transactions on Power Delivery*, vol. 32, no. 3, pp. 1483–1494, 2017. DOI: 10.1109/TPWRD.2016.2608986.

[31] W. Leterme, J. Beerten, and D. Van Hertem, "Nonunit protection of hvdc grids with inductive dc cable termination," *IEEE Transactions on Power Delivery*, vol. 31, no. 2, pp. 820–828, 2016. DOI: 10.1109/TPWRD.2015.2422145.

[32] C. Zhang, J. Huang, G. Song, and X. Dong, "Non-unit ultra-high-speed line protection for multi-terminal hybrid lcc/mmc hvdc system and its application research," *IEEE Transactions on Power Delivery*, vol. 36, no. 5, pp. 2825–2838, 2021. DOI: 10.1109/TPWRD.2020.3027723.

[33] R. Li, L. Xu, and L. Yao, "Dc fault detection and location in meshed multiterminal hvdc systems based on dc reactor voltage change rate," *IEEE Transactions on Power Delivery*, vol. 32, no. 3, pp. 1516–1526, 2017. DOI: 10.1109/TPWRD.2016.2590501.

[34] C. Li, A. M. Gole, and C. Zhao, "A fast dc fault detection method using dc reactor voltages in hvdc grids," *IEEE Transactions on Power Delivery*, vol. 33, no. 5, pp. 2254–2264, 2018. DOI: 10.1109/TPWRD.2018.2825779.

[35] S. Yang, W. Xiang, R. Li, X. Lu, W. Zuo, and J. Wen, "An improved dc fault protection algorithm for mmc hvdc grids based on modal-domain analysis," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 8, no. 4, pp. 4086–4099, 2020. DOI: 10.1109/JESTPE.2019.2945200.

[36] K. Satpathi, Y. M. Yeap, A. Ukil, and N. Geddada, "Short-time fourier transform based transient analysis of vsc interfaced point-to-point dc system," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4080–4091, 2018. DOI: 10.1109/TIE.2017.2758745.

[37] Z. Zheng, T. Tai, J. S. Thorp, and Y. Yang, "A transient harmonic current protection scheme for hvdc transmission line," *IEEE Transactions on Power Delivery*, vol. 27, no. 4, pp. 2278–2285, 2012. DOI: 10.1109/TPWRD.2012.2201509.

[38] V. Albernaz Lacerda, R. M. Monaro, D. Campos-Gaona, D. V. Coury, and O. Anaya-Lara, "Distance protection algorithm for multiterminal hvdc systems using the hilbert–huang transform," *IET Generation, Transmission & Distribution*, vol. 14, no. 15, pp. 3022–3032, 2020. DOI: https://doi-org.tudelft.idm.oclc.org/10.1049/iet-gtd.2019.1551. eprint: https://ietresearch-onlinelibrary-wiley-com.tudelft.idm.oclc.org/doi/pdf/10.1049/iet-gtd.2019.1551. [Online]. Available: https://ietresearch-onlinelibrary-wiley-com.tudelft.idm.oclc.org/doi/abs/10.1049/iet-gtd.2019.1551.

[39] O. M. K. K. Nanayakkara, A. D. Rajapakse, and R. Wachal, "Traveling-wave-based line fault location in star-connected multiterminal hvdc systems," *IEEE Transactions on Power Delivery*, vol. 27, no. 4, pp. 2286–2294, 2012. DOI: 10.1109/TPWRD.2012.2202405.

[40] S. Zhang, G. Zou, C. Wang, J. Li, and B. Xu, "A non-unit boundary protection of dc line for mmc-mtdc grids," *International Journal of Electrical Power & Energy Systems*, vol. 116, p. 105 538, 2020, ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2019.105538. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S014206151931275X.

[41] J. Fan, J. Huang, and J. Pan, "An adaptive multi-step levenberg-marquardt method," *Journal of Scientific Computing*, vol. 78, no. 1, pp. 531–548, Jun. 2018. DOI: 10.1007/s10915-018-0777-8.

[42] L. Van Der Sluis, *Transients in power systems*. Jun. 18, 2001. DOI: 10.1002/0470846186. [Online]. Available: https://doi.org/10.1002/0470846186.

[43] L. Tang, X. Dong, S. Shi, and B. Wang, "Analysis of the characteristics of fault-induced travelling waves in mmc-hvdc grid," *The Journal of Engineering*, vol. 2018, no. 15, pp. 1349–1353, 2018. DOI: https://doi.org/10.1049/joe.2018.0256.

[44] C. L. Fortescue, "Method of symmetrical co-ordinates applied to the solution of polyphase networks," *Transactions of the American Institute of Electrical Engineers*, vol. XXXVII, no. 2, pp. 1027–1140, 1918. DOI: 10.1109/T-AIEE.1918.4765570.

[45] Y. Gu, W. Li, and X. He, "Analysis and control of bipolar lvdc grid with dc symmetrical component method," *IEEE Transactions on Power Systems*, vol. 31, no. 1, pp. 685–694, 2016. DOI: 10.1109/TPWRS.2015.2403310.

[46] L. Liu, A. Lekić, and M. Popov, "Robust traveling wave-based protection scheme for multiterminal dc grids," *IEEE Transactions on Power Delivery*, vol. 38, no. 5, pp. 3117–3129, 2023. DOI: 10.1109/TPWRD.2023.3265748.

[47] Y. Zhang, N. Tai, and B. Xu, "Fault analysis and traveling-wave protection scheme for bipolar hvdc lines," *IEEE Transactions on Power Delivery*, vol. 27, no. 3, pp. 1583–1591, 2012. DOI: 10.1109/TPWRD.2012.2190528.

[48] J. J. Grainger, W. D. Stevenson, and G. W. Chang, *Power System Analysis*. Feb. 1, 2016.

[49] S. Yang, W. Xiang, and J. Wen, "An improved dc fault protection scheme independent of boundary components for mmc based hvdc grids," *IEEE Transactions on Power Delivery*, vol. 36, no. 4, pp. 2520–2531, 2021. DOI: 10.1109/TPWRD.2020.3029308.

[50] C. Zhang, G. Song, T. Wang, L. Wu, and L. Yang, "Non-unit traveling wave protection of hvdc grids using levenberg–marquart optimal approximation," *IEEE Transactions on Power Delivery*, vol. 35, no. 5, pp. 2260–2271, 2020. DOI: 10.1109/TPWRD.2020.2964717.

[51] L. Liu, A. Shetgaonkar, and A. Lekić, "Interoperability of classical and advanced controllers in mmc based mtdc power system," *International Journal of Electrical Power & Energy Systems*, vol. 148, p. 108 980, 2023, ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2023.108980. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061523000376.

[52] A. Shetgaonkar, T. Karmokar, M. Popov, and A. Lekić, "Enhanced real-time multi-terminal hvdc power system benchmark models with perfor-mance evaluation strategies," *CIGRE Science & Engineering*, vol. 32, pp. 1–29, 2024. [Online]. Available: https://cse.cigre.org/cse-n032/enhanced-real-time-multi-terminal-hvdc-power-system-benchmark-models-with-performance-evaluation-strategies.html.

[53] L. Liu, *The amlm algorithm for parameter fitting of discrete data*, 2023. DOI: 10.4121/9ba76916-6abe-4d75-bfaf-a22bc87787e1.v1.

A

Adaptive Multi-step Levenberg-Marquardt Algorithm

This appendix introduces the parameter fitting method using the Adaptive Multi-step Levenberg-Marquardt (AMLM) algorithm presented in [41].

A.1. The Mathematics of the AMLM algorithm

A basic error function is considered for the backwards traveling line-mode voltage wave V_{b1} based on Equation 2.28

$$F(A_0, \tau_0) = v_{b1} - A_0\left(1 - e^{(-\frac{t-T_{d0}}{\tau_0})}\right) = 0. \tag{A.1}$$

A two-dimensional matrix is defined as $x = [A_0, \tau_0]_{1\times 2}$. Next, the merit function of Equation A.1

$$\min_{x\in R^n} ||F(x)||^2. \tag{A.2}$$

The AMLM algorithm computes the trial step d_k at the k-th iteration by solving

$$(G_k^T G_k + \alpha_k I)d_k = -G_k^T F_k, \tag{A.3}$$

where G_k is the Jacobian. I is the identity matrix. The ratio of actual changes R_k compared to the predicted changes P_k of the merit function $||F(x)||^2$ is defined as

$$r_k = \frac{R_k}{P_k} = \frac{||F_k||^2 - ||F(x_k + d_k)||^2}{||F_k||^2 - ||F_k + G_k d_k||^2}. \tag{A.4}$$

The trial step x_k is updated by

$$x_{k+1} = \begin{cases} x_k + d_k, & \text{if } r_k \geq p_0, \\ x_k, & \text{otherwise,} \end{cases} \tag{A.5}$$

where p_0 is a small constant. The Jacobian G_k is updated by

$$G_{k+1} = \begin{cases} G_k, & \text{if } r_k \geq p_1 \text{ and } s < t, \\ J_{k+1}, & \text{otherwise,} \end{cases} \tag{A.6}$$

where $0 < p_0 < p_1 < 1$. The Levenberg-Marquardt parameter α_k is updated as

$$\alpha_{k+1} = \begin{cases} \alpha_k, & \text{if } r_k \geq p_1 \text{ and } s < t, \\ \beta_{k+1}||F_{k+1}||^{\delta}, & \text{otherwise,} \end{cases} \tag{A.7}$$

where

$$\beta_{k+1} = \begin{cases} c1\beta_k, & \text{if } r_k < p_2, \\ \beta_k, & \text{if } p_2 \leq r_k \leq p_r, \\ \max(c_2\beta_k, \beta_{min}), & \text{if } r_k > p_3. \end{cases} \tag{A.8}$$

For the equations above it holds that $0 < c_2 < 1 < c_1, 0 < p_0 < p_2 < p_1 < p_3 < 1, 1 \leq \delta \leq 2$, and $\beta_{min} > 0$. Based on the equations the AMLM algorithm is established, which is shown in 1.

Algorithm 1 The adaptive multi-step Levenberg-Marquardt algorithm

Require: $x_1 \in R^n, c_1 > 1 > c_2 > 0, 0 < p_0 < p_2 < p_1 < p_3 < 1, 1 \leq \delta \leq 2, t \geq 1, \mu_1 > \mu_{min} > 0$.
Ensure: Set $G_1 = J_1, \alpha_1 = \beta_1||F_1||^{\delta}, k := 1, s := 1, i := 1, k_i = 1$.
1: **while** $||G_{k_i}^T F_{k_i}|| \neq 0$ **do**
2: Compute d_k by solving (A.3).
3: Compute $r_k = R_k/P_k$ by solving (A.4) and set x_{k+1} by (A.5).
4: Update G_{k+1}, α_{k+1}, and β_{k+1} by (A.6), (A.7), and (A.8), respectively.
5: Set $k = k + 1$. If G_k is the Jacobian J_k at x_k, set: $s = 1, i = i + 1, k_i = k$, Otherwise, set: $s = s + 1$.
6: **end while**