UNIVERSITY OF APPLIED SCIENCES UPPER AUSTRIA

MASTER'S DEGREE PROGRAMME

Automation Engineering

Fault Management Design and Evaluation using a Surrogate Control Framework

submitted as a DIPLOMA THESIS

to obtain the academic degree of

Diplom-Ingenieur für technisch-wissenschaftliche Berufe

by

Christopher Neuwirt, BSc

September 2018

Thesis supervisors

FH-Prof. Dr. techn. Jean D. Hallewell Haslwanter MSc BS

Dr. Michael Steurer

Acknowledgements

First of all, I would like to thank my thesis supervisor, Jean Hallewell, for her support. I really appreciated the discussions with her and her constructive comments that steered me in the right direction.
I also want to thank everybody at the Center for Advanced Power Systems for their support and hospitality during my time at their facilities and the productive discussions we had. Especially I'd like to thank Michael Steurer for inviting me and giving me the opportunity to work with him and his team, Mark Stanovich for his extraordinary support with my thesis and Karl Schoder and Harsha Ravindra for our discussions, support and input to my research.
Finally, and most importantly, I want to thank my loving family that supports me in every way I can wish for and always believes in me.

UNIVERSITY
OF APPLIED SCIENCES
UPPER AUSTRIA

Wels Campus

Sworn declaration

I hereby declare that I prepared this work independently and without help from third parties, that I did not use sources other than the ones referenced and that I have indicated passages taken from those sources.

This thesis was not previously submitted in identical or similar form to any other examination board, nor was it published.

Christopher Neuwirt, BSc
Wels, September 2018

Kurzfassung

Die U.S. Navy, das Office of Naval Research und das Electric Ship Research and Development Consortium (ESRDC) arbeiten daran, das Design einer neuen Generation von rein elektrischen Schiffen zu entwickeln. Ein Teil dieser Vision, der weiter erforscht werden muss, ist das Fehlermanagement. Nachdem ein Fehler aufgetreten ist und im elektrischen System isoliert wurde, kann das Bordnetz in einem unerwünschten Zustand verblieben, in dem kritische Verbraucher ohne ausreichende Stromversorgung sind. Wenn das der Fall ist, sollte eine automatisierte Rekonfiguration eine Lösung identifizieren und umsetzen, die eine bessere Energieversorgung für essentielle Lasten ermöglicht.
Das Electric Ship Research and Development Consortium hat ein zonales DC–Schiffsmodell für die nächste Schiffsgeneration vorgeschlagen. Die Rekonfiguration von Bordnetzsystemen hat sich noch nicht mit Modellen dieser Komplexität beschäftigt und es muss daher analysiert werden, ob bestehende Ansätze anwendbar sind. Ein Vergleich der bisher verwendeten Lösungen zeigt, dass ein neuer Ansatz entwickelt werden muss, um die zeitlichen Anforderungen trotz erhöhter Komplexität besser zu erfüllen. Das Center for Advanced Power Systems hat sich zum Ziel gesetzt, ein automatisiertes Fehlermanagement innerhalb von 8 ms zu erreichen. Wenn die Rekonfiguration auch in dieser Zeitspanne erreicht werden kann, kann eine zusätzliche Ausfallzeit, die rein zum Zweck der Implementierung einer neuen Konfiguration nötig wäre, vermieden werden.
Das Ziel dieser Arbeit ist es, eine Lösung für das Rekonfigurationsproblem zu finden, die diese zeitlichen Beschränkungen auch bei der komplexeren vorgeschlagenen Architektur erfüllt. Ein graphenbasierter Rekonfigurationsalgorithmus, der mit vergleichsweise wenigen Einschränkungen die bestmögliche Lösung nach definierten Kriterien findet, wird entwickelt und getestet, um die zugrunde liegenden elektrischen Systemeigenschaften effizient zu optimieren. Die Korrektheit dieses Ansatzes wird durch die Aufzeichnung und Analyse experimenteller Testergebnisse bestätigt. Die Testumgebung– und Algorithmus–Implementierung für diese Validierung wird beschrieben. Die Ergebnisse zeigen, dass der entwickelte Algorithmus ein vergleichsweise schneller Ansatz ist, der eine optimale Lösung innerhalb weniger definierter Restriktionen, basierend auf mehreren Kriterien, findet. Schließlich werden diese Ergebnisse mit alternativen Ansätzen verglichen und ein Ausblick auf die zukünftige Arbeit gegeben.

Abstract

The U.S. Navy, Office of Naval Research and the Electric Ship Research and Development Consortium (ESRDC) are developing the design of a new generation of all–electric ships. One part of that vision that has to be further researched is fault management. After a fault occurred and has been isolated in the electrical system, the shipboard power system might be left in an undesirable state, in which vital loads are without sufficient power supply. If this is the case, automated reconfiguration should identify and implement a solution that better provides power to vital loads.

The Electric Ship Research and Development Consortium proposed a zonal, medium voltage DC shipboard model for the next generation of ships. Reconfiguration of shipboard power systems has not yet dealt with models of such complexity. Therefore, it has to be analyzed if existing approaches are applicable. A comparison of solutions shows that a new approach has to be developed to better fulfill timing requirements despite increased complexity. The Center for Advanced Power Systems set their goal to achieve automated fault management within 8 ms. If the reconfiguration can be achieved in this timespan too, an additional downtime solely for the purpose of implementing a new configuration can be avoided.

The objective of this work is to identify a solution to the reconfiguration problem that meets these timing constraints even with the more complex proposed architecture. A graph–based reconfiguration algorithm that is able to find the best possible solution according to defined criteria with comparably few restrictions is developed and tested in order to allow efficient optimization of underlying electrical system attributes. Correctness of this approach is validated by obtaining and analyzing experimental test results. The testbed and algorithm implementation for this validation are described. Results show, that the developed algorithm is a comparably fast approach which finds an optimal solution within a few defined restrictions, based on multiple criteria. Finally, those results are compared with alternative approaches and an outlook on future work is given.

Contents

1 Introduction

1.1 Motivation

The U.S. Navy, Office of Naval Research and the Electric Ship Research and Development Consortium (ESRDC) are developing the design of a new generation of all–electric ships. The Center for Advanced Power Systems (CAPS) at Florida State University is a multidisciplinary research center conducting basic and applied research in the field of electrical energy and power technology and is one of the main members of the ESRDC. Power system simulation is a vital tool and CAPS' digital real–time simulation capabilities and controller hardware–in–the–loop (CHIL) capabilities are integral in establishing advanced research in the field of integrated system controls.
As Doerry et al. described in [1], the goal is to realize a Medium Voltage Direct Current (MVDC) Shipboard Power System (SPS), which is considered to provide increased power distribution efficiency and power density when compared to traditional Medium Voltage Alternating Current (MVAC) systems. However, the benefits of MVDC–based systems are often curbed by the experience and readiness of MVAC–based systems. The switch to a MVDC–based system brings with it different hardware requiring different approaches and designs for control systems and software.
Several (SPS) architectures have been designed and modeled by the ESRDC. Additionally, real–time capable model implementations have been developed for multiple platforms, which can serve as a baseline for implementing test cases. These test cases can in turn be used to develop and verify new approaches and solutions to the unexplored challenges in MVDC SPS.
In particular, Fault Management (FM) is one such control system that is affected by this switch, creating a number of differences and associated challenges that must be fully understood and decidedly addressed. FM encompasses a broad spectrum of possible research topics. Previous works dealt with the fault detection, localization and isolation [2–5]. One topic that has not been fully addressed in combination with the proposed MVDC zonal architecture is reconfiguration of a SPS after a fault occurred and has been isolated. To understand the aspects and complexity, and to find a suitable approach to this problem leads to a multidisciplinary research question merging different aspects of electrical engineering and computer science. This work proposes a novel approach for reconfiguration, specifically geared towards zonal, MVDC–based SPS and in doing so takes a first step in solving this research question.

1.2 Thesis Statement

This thesis supports the following statement:

Finding a tree rooted at an under–supplied section such that its generation capacity meets its power demand is an approach for MVDC SPS reconfiguration that allows efficiently optimizing underlying electrical system attributes, such as connectivity or mission–based fitness value.

The following research questions should be answered:

- What are reconfiguration requirements specific to the notional zonal model?
- Which other algorithms have been used for SPS reconfiguration purposes?
- How suitable is a graph–based algorithm regarding performance and scalability in that environment?
- What options are there to improve the online performance of the reconfiguration process?

1.3 Scope of the Thesis

This thesis aims to describe the requirements a reconfiguration algorithm has to fulfill, in order to be applicable on the next generation of all–electric ships. The goal is to develop a solution that reconfigures a shipboard power system, optimized to perform under user–specified fitness based criteria, in a timely manner. This is done specifically with the focus on reconfiguration after a fault occurred in the SPS. In order to achieve this goal, not only the algorithm itself has to be chosen, but also the interaction with other vital pieces of the SPS should be explored. This includes the overall process of fault management to detect, locate and isolate a fault. Then, if necessary, a new configuration that utilizes available power and energy for maximizing operational performance of the ship has to be calculated. To be able to define this process and the reconfiguration algorithm itself, the underlying SPS model has to be analyzed and salient attributes which determine requirements for the reconfiguration have to be identified. The process of fault management has to be examined with enough detail to identify necessary information and the available timespan. An overview of approaches that have been applied to other systems will be examined, to be able to compare these techniques and determine their applicability to the envisioned model. Definitions that aid the definition and execution of the FM process are proposed and the computational complexity is derived. Finally, a testbed to validate the developed algorithm is used and results of evaluating the approach are discussed along with possible future work.

1.4 Thesis Outline

The next chapter gives an overview of the notional zonal SPS architecture and its specific requirements relating to reconfiguration. Recommended and envisioned architectures are reviewed, and their characteristics are highlighted. In the second part fault management on shipboard power systems is discussed and the sequence of FM processes and the interaction with reconfiguration is described. Chapter 3 details related work on the topic of reconfiguration, and how suitable those solutions are to be applied on the ZEDS architecture. Chapter 4 describes the methodology used in the scope of this thesis. In Chapter 5 the main contribution of this work is presented, more specifically, the developed graph–based approach and algorithm for shipboard power system reconfiguration, its objective, advantages and limitations. Moreover, options to improve the runtime and computational complexity of the worst case are proposed and integrated into the solution. Chapter 6 discusses an implementation of the proposed algorithm and presents test cases to verify the approach along with results of the testing. This is followed by a chapter presenting and analyzing these findings and proposes future topics. The final chapter concludes the outcome of this thesis.

2 Theory

This chapter presents the special characteristics of shipboard power systems (SPS) with focus on the notional zonal MVDC model, additionally requirements specific to its reconfiguration are discussed. In the second part fault management on shipboard power systems and the sequence of actions that are taken to handle faults efficiently are described.

2.1 Shipboard Power System

This section focuses on the characteristics of shipboard power systems and special architecture recommended for future all–electric ship designs.

2.1.1 General

The power system on a ship supplies energy to various loads, such as propulsion, navigation, communications or other mission and miscellaneous loads. These loads and service systems can either be integrated or designed to operate as separate subsystems. If they are electrically integrated, the SPS is comparable with an islanded micro grid, as long as it is not connected to the main grid when docked. The SPS as a micro grid has couple of specific characteristics, which are partly determined by the systems objective and task to provide power to vital loads like propulsion. These vital loads should perform as uninterrupted and with as good quality of service as possible. The other part that is of importance when specifying some of the characteristics are the conditions under which it must be able to perform, namely being isolated from the main grid, with tightly limited space and weight constraints. The main characteristics include:

- high required survivability and robustness [6–8]
- high required reliability [6–8]
- no significant excess generation capacity when powering all main loads is available [6–8]
- a minority of loads contribute the majority of the power demand, i. e. propulsion motors [6, 7]
- reconfiguration has to be accomplished much faster as damage to the ship or physical injuries of the crew might be the direct or indirect result if a vital load is without supply for too long [7, 8]
- geographically, i. e. physically smaller than utility systems [7, 8]
    - system level measurement is possible [7]
    - short line segments with low impedance [7]

> high required energy density [6]

These are the salient characteristics a SPS has to provide and support in order to be suitable for employment on a ship. The following subsection goes into further detail about the advantages of a medium voltage direct current distribution system when implemented on a SPS.

2.1.2 Medium Voltage Direct Current Shipboard Power System

As described in [1] and [9] the primary reasons for employing MVDC technology rather than medium voltage AC or high frequency AC on ships are:

- better performance of propulsion;
- smaller transformers due to higher operating frequencies of power conversion equipment;
- less weight of cables because there is no skin effect;
- better fault management capabilities through power electronics;
- lesser acoustic signature of the ship because of no common frequency in vibrating equipment;
- simplified and faster process for starting generators because no time–critical phase matching is required; and
- MVDC enables high power demanding electric mission loads in a much more compact and power dense architecture.

These provide a number of potential advantages. The first point mentioned in the above list is better performance of propulsion. Decoupling main propulsion motor speed from the power quality and frequency of the bus enables optimization of the generator for each type of prime mover without having to integrate transmission gears. Moreover, generators are not restricted to a given number of poles with MVDC. Another important reason is, as mentioned, that there is no skin effect present like in AC power transmission, therefore a DC conductor is more effective in transmission of power with the same cross section. Cable weights could therefore decrease for a given power level, depending on the voltage selection. Power electronics are able to control fault currents to considerably lower levels than AC systems. This results in reduced damage during faults due to lower fault currents. [1] and [9] also state that there is no common frequency of vibrating equipment like in AC systems, hence the acoustic signature of the ship has a broader spectrum with fewer tones that can be observed. This makes ships utilizing DC technology harder to detect by acoustic sensors. Another advantage of MVDC systems is that operating multiple generators in parallel only requires voltage matching, and no time–critical phase matching. This should reduce the amount of time needed to start a new generator and bringing it online, which in turn reduces the energy storage that would be necessary for

bridging that period.
All these aspects suggest that MVDC provides better quality of power as well as space and weight reductions and higher energy density. Therefore, meeting all the previously mentioned vital requirements for power systems in oder to be favorable for implementation on all–electric ships. The reason why MVDC has not yet been applied to SPS and a near–term implementation is not planned is that there are many challenges that still need to be overcome before such an architecture can be implemented with sufficient trust, quality and safety [1].
One of those challenges is that current FM techniques associated with AC systems are not desirable in MVDC. Regular AC breakers can be opened without problems by waiting and detecting zero crossings of current flowing through them. Since there exists no zero crossings in direct current systems, DC breakers have to be designed more robustly. This in turn makes them more expensive and heavier, which is undesirable on ships. The mentioned power electronics, especially fast switching devices, provide cheaper and lighter alternative options to face these problems and moreover open up possibilities to reconfigure the system in a much more complex and sophisticated way. This results in a more optimal configuration, especially if the SPS is left in an undesirable state after a fault occurred in the system. Therefore, enabling utilization of such fast switching power electronics would make the use of MVDC architecture on ships even more advantageous.

2.1.3 Zonal Electrical Distribution System

Rather than a monolithic architecture, zonal designs provide additional advantages. Zonal Electrical Distribution Systems (ZEDS) are defined in the IEEE Standard 1826:2012 [10]:

> A zone is a logical and physical grouping of generation, storage, or consumption assets arranged in a common neighborhood. Zones with generation or storage may be operated for periods of time independently from the power system.

Moreover, it is stated that a zone is characterized by a distinct set of attributes:

- Contains one or more independent power device, i.e., converter, load, storage, generator;
- under regular conditions operates as an integral part of a larger system; and
- may operate independently of the zonal system for limited periods of time under special operating conditions.

An electric block diagram for an example zone as described in [10] is depicted in Figure 2.1 and a general architecture using zones and connecting to external systems is shown in Figure 2.2.
Zonal architecture is recommended for shipboard power systems rather than a ring or radial layout for distribution, because it offers increased accessibility to loads and sources, and therefore more flexibility and reliability. Another advantage of the zonal architecture is that it supports AC as well as DC systems. The general layout envisions multiple

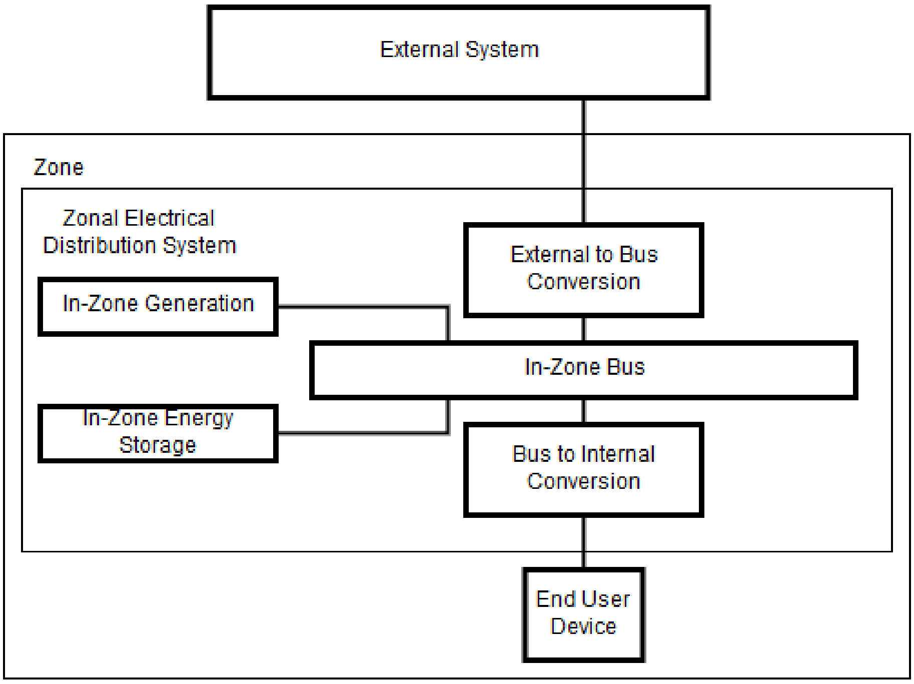

Figure 2.1: Electric block diagram of an example zone

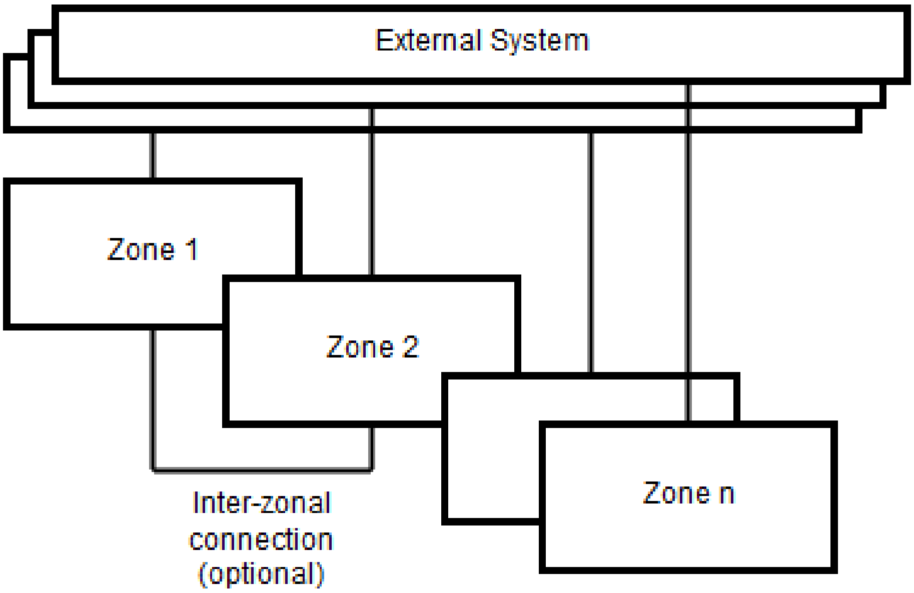

Figure 2.2: General zonal architecture as proposed in [10]

sources connected to the main loop and with a distinction of two main buses, port and starboard, survivability is increased, and sufficient redundancy is added even though the level of redundancy is less than 100% [6].

To ensure a full 100% of redundancy two dedicated sources would be needed for each load. Clearly this is not desirable as that would add unnecessary weight of mostly unused generators and furthermore take up large portions of space. Both of these disadvantages are not acceptable, as already described in Section 2.1. With the recommended zonal architecture, all loads can be fed from two sides, while each bus in turn can be supplied by multiple sources and, if necessary, they can even be interconnected. This increases redundancy of power supply to acceptable levels, without any unnecessary weight or used space. Figure 2.3 shows the SPS architecture initially proposed to incorporate ZEDS.

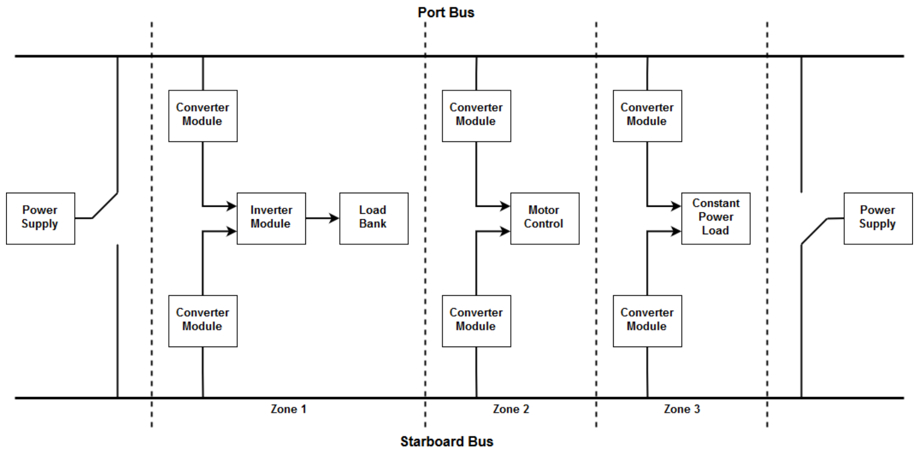

Figure 2.3: Zonal architecture as first proposed in [6]

By using switches and breakers at strategic points throughout the system it is possible to feed loads from different sides and by different sources, which adds reliability. Especially in case of a fault, as another path through the system, even to other zones can be calculated and implemented. This provides increased survivability of pathway through the availability of multiple reconfigurable paths and therefore ensures sustainable power supply to loads. Another advantage of the zonal architecture in combination with switches is that zones can easily be separated to isolate faults or reduce interference of individual parts with each other.

All these aspects increase the number of options for configuration and reconfiguration of the SPS significantly. On the other hand, this also means that the computational complexity of finding such a configuration drastically increases. This complexity is further increased if it has to be verified whether a calculated state is valid and how optimal it is compared to other reconfiguration options. For every possible new path to a load a new option for that specific load is added, and this option might interfere with options for other loads. This interrelationship has to be considered for all combinations in order to find the best overall system configuration based on chosen criteria.

To illustrate the relative complexity of different architectures an example system with two loads and two sources is examined. The three different architectures that are compared are radial, ring and ZEDS shown in Figures 2.4, 2.5 and 2.6 respectively. Switches in the system are represented as either black or white squares, depending on the initial state where white would represent opened switches and black squares visualize closed switches.

The radial architecture allows no reconfiguration should a fault occur on any of the cables. With the ring architecture depicted in Figure 2.5 a single fault can be isolated and reconfiguration would have to close the previously open cable section. As soon as two faults occur, no reconfiguration is possible, only isolation. Therefore, reconfiguration–options in ring architecture are very limited. The zonal architecture in Figure 2.6 on the other hand

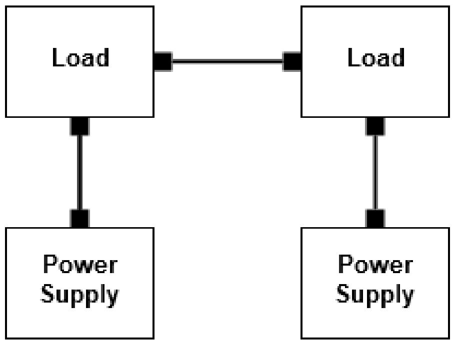

Figure 2.4: Example system with radial architecture

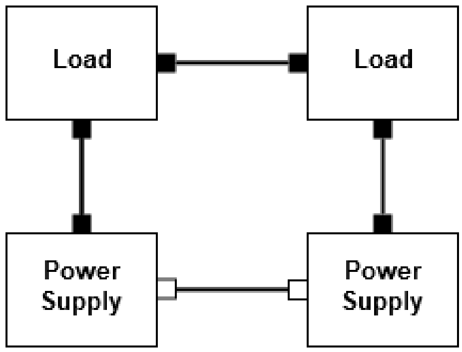

Figure 2.5: Example system with ring architecture

provides a multitude of options to connect the four devices in this small example system. Through adding switches and connections at strategic points in the system many paths can be created, and the number of possible solutions is obviously drastically higher than with radial or ring architecture. This effect is even stronger if the number of loads or sources is further increased.

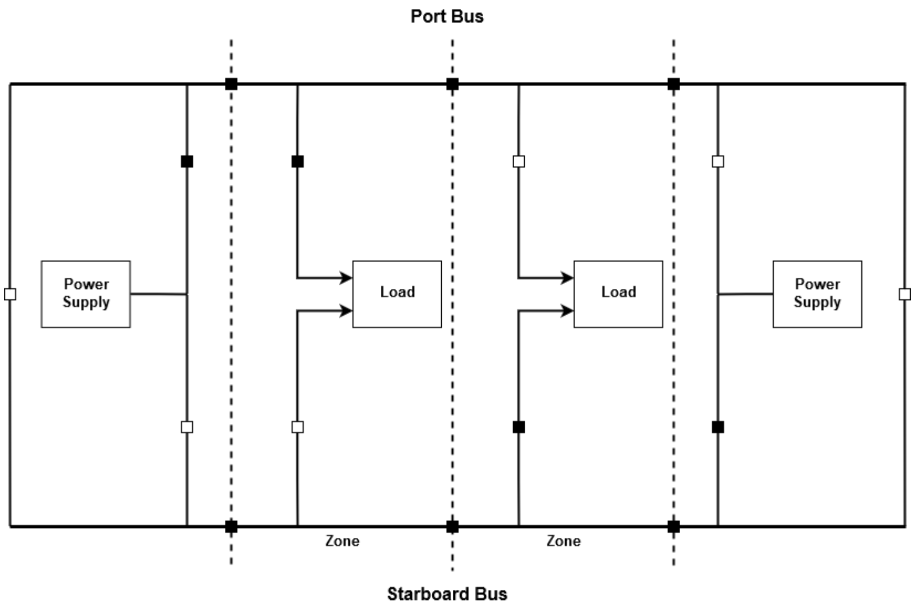

Figure 2.6: Example system with zonal architecture

2.1.4 Notional Zonal Medium Voltage Direct Current Shipboard Power System

The ESRDC is working towards the development of the next generation of all–electric ships. As described in [1], the goal is to realize a MVDC SPS in a combination with a ZEDS. FM on MVDC SPS faces challenges that are not present in the traditional MVAC counterpart and also considerations regarding the ZEDS architecture have not been fully explored. Both [9] and [11] propose a concept for a MVDC ZEDS SPS and the ESRDC developed notional models for different sizes of such a system. More specifically, the three main models incorporate two, four and six zones in order to enable testing of different controls which are being developed on a simulation model that provides all crucial attributes, without needlessly increasing occupancy of the RTDS™ racks and cores that run the electrical simulation. These models incorporate aspects described in Sections 2.1, 2.1.2 and 2.1.3. The two models that are most likely to accurately represent the size and structure of a future ship are the four and six zone models, although the differences in terms of devices that are present in the medium voltage direct current part of the SPS are only marginal.

According to current information from the ESRDC, the six–zone model, as shown in Figure 2.7 is the most favored one. Each zone has port and starboard side switchboards. The

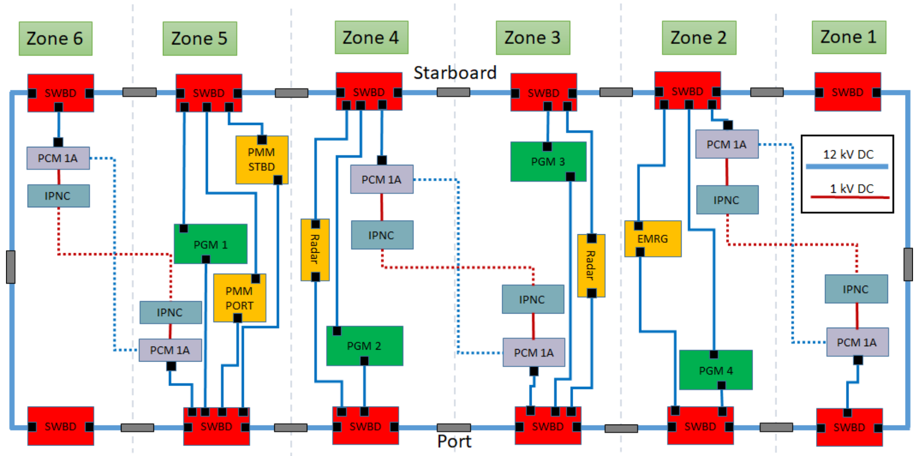

Figure 2.7: Notional zonal medium voltage direct current shipboard power system with six zones as proposed by the ESRDC

zones are electrically connected on either side through cable sections and can be connected to each other if necessary. The general modules present in the model are:

- Power Generation Modules (PGM);
- Power Conversion Modules (PCM);
- Integrated Power Node Center (IPNC);
- Propulsion Motor Modules (PMM);
- Mission loads; and
- Switchboards.

The four–zone model and the central four zones of the six–zone system are similar, the two additional zones in the larger model only accommodate IPNC's and associated PCM's. These integrated power node centers are envisioned to be connected to either starboard or port bus alternatingly, with internal cross–zone connections between pairs.

Both models contain the same number of mission loads and generators, as well as connecting cable sections between starboard and port buses in the first and last zone. Regular

loads are designed with the ability to be supplied by both buses and PGM's are also capable of feeding either bus. This provides increased reliability and survivability for the entire system, as multiple options are provided in case of line or switchboard faults.
When comparing reconfiguration options,a number of parameters may differ:

- Number of necessary switching actions
- Number and occupancy rate of generation modules
- Number and priority of loads supplied
- Survivability and robustness

To utilize these newly gained options of the zonal architecture fully, a methodology to compare them and choose the best one has to be developed. This is responsibility of the reconfiguration algorithm that will be applied to the system. The new, more complex model might render approaches used until now suboptimal or intractable. Therefore, thorough research on algorithms used for reconfiguration has to be conducted, to determine if a new solution is necessary, before an approach can be selected for implementation. Furthermore, it must be defined which aspects this new approach has to consider, and which restrictions have to be met.

2.2 Fault Management

This section deals with definitions that are useful for the fault management (FM) of SPS and the overall process of detecting, locating and isolating faults in such systems. Also, timing sequences and details are explored to reduce fault handling times.

2.2.1 Graph Representation of Electrical System

Although not widespread in classical electrical engineering, a transformation into a graph representation is useful to be able to verify connectivity of devices in an electrical system. This transformation also allows to apply established graph–based software algorithms to the system, which in turn provides efficient methods to identify connected and isolated parts and possible paths form one device to another. The system graph representation is also useful for other parts of the fault management like fault detection and fault isolation, as presented in [2].
With respect to the selected system a system graph is defined as:

Definition 2.1. *System Graph:*
A system graph is an undirected graph representing an electrical system. Electrical devices are represented as vertices, which may have application–dependent attributes assigned to them. If an electrical connection exists between two devices, there also exists an edge between their vertices in the system graph

The system graph representation of an electrical system provides means to apply graph–based algorithms, as well as options to store and access the electrical system. The vertices represent electrical devices like loads, generators, converters or switches, while the edges of the graph represent electrical connection between devices, e.g. cables. An example of an electrical system and its graph representation is shown in Figures 2.8 and 2.9 respectively. Every vertex may have a set of attributes based on the physical attributes, capabilities or

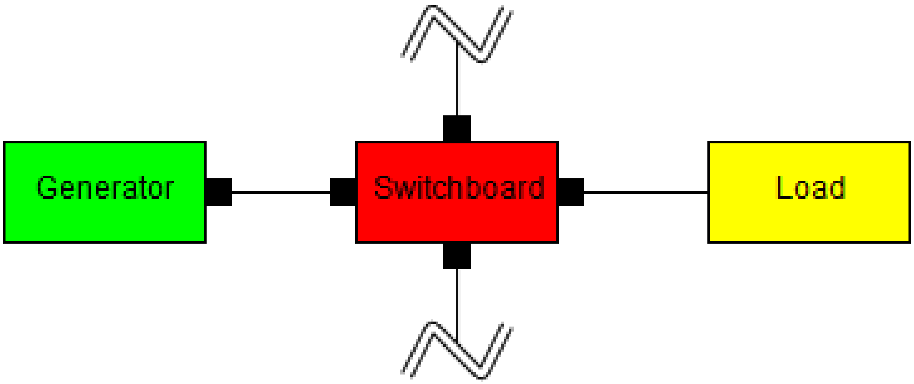

Figure 2.8: Example of an electrical system

state of the represented electrical device. These attributes may be fixed and determinable offline or dependent of the current state.

For use in FM every device needs to provide a set of attributes for the initial transfor-

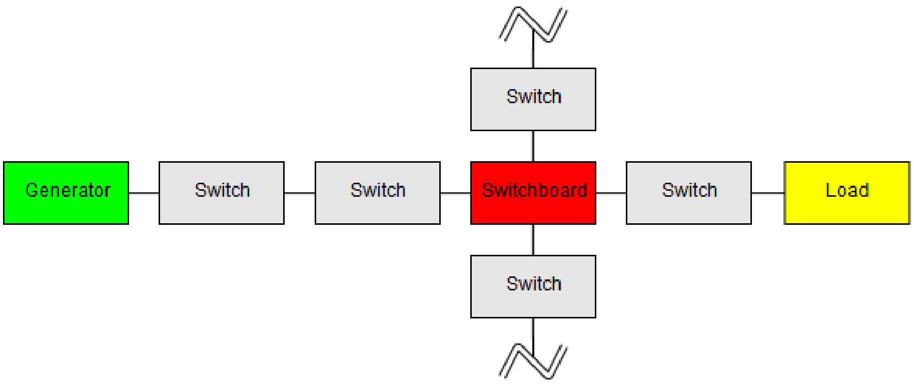

Figure 2.9: System graph representation of the example system

mation into the system graph:

- Can isolate;
- can measure current; and
- disables current measurement.

Based on these attributes, the system can be broken down into sections, both general sections and fault detection sections. A section in general is defined as follows:

Definition 2.2. *Section:*
Connected subgraph of the system graph such that:

- *Order* ≥ 2 *i. e. the subgraph has 2 or more vertices*
- *All vertices within the subgraph that are adjacent to vertices that are not in the subgraph have the attribute "can isolate"*
- *No two or more vertices share an adjacent vertex that is not in the subgraph*

These sections can vary in size, which is useful in combination with online information about current switch states. Examples for such sections are depicted in Figures 2.10 and 2.11. Figure 2.10 shows the smallest possible section, containing only two vertices, one of them is a load module, the other a disconnect switch, that connects the section to the rest of the system. In Figure 2.11 a section representing a switchboard is shown. For FM

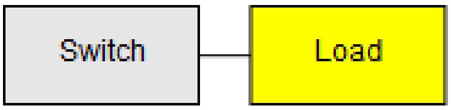

Figure 2.10: The section containing the load module from the system graph in Figure 2.9

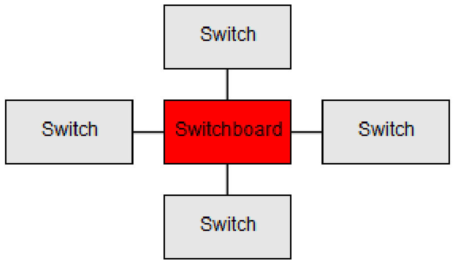

Figure 2.11: The section containing the switchboard from the system graph in Figure 2.9

using the smallest possible set of sections enables fault isolation with the least impact on the remaining system. Minimal sections are defined as:

Definition 2.3. *Minimal Section:*
A subgraph of a given graph is minimal for a particular property if the subgraph itself has that property but no proper subgraph of it also has the same property i. e. the smallest subgraph that still fulfills all requirements of a section, and if any element is removed the subgraph does not fulfill all requirements.

This definition of minimal sections is visualized in Figures 2.12, 2.13 and 2.14. Figure 2.12 shows a valid section, but it is not minimal, as this section can be split up into two valid sections. These on the other hand then are minimal, as they cannot be split without violating the requirements for sections as defined in Definition 2.2.

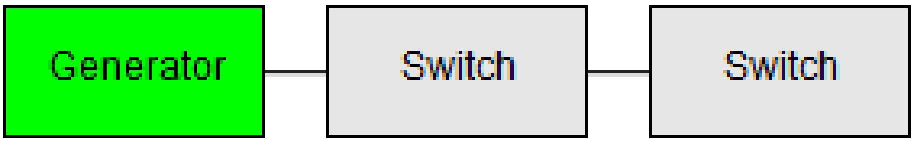

Figure 2.12: The section containing the generation module and the cable connecting it to the switchboard, from the system graph in Figure 2.9

For fault detection minimal sections must meet additional criteria that enable protection schemes as discussed in [2–5]. These fault detection sections are defined in Definition 2.4:

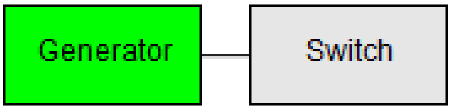

Figure 2.13: The minimal section containing only the generation module and the switch

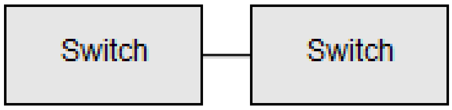

Figure 2.14: The cable section between two switches which is a minimal section

Definition 2.4. *Fault Detection Section:*
Minimal section with the following additional attributes:

- *All vertices in the subgraph with edges to the remaining graph have the attribute "can measure current" in addition to "can isolate"*
- *No vertex within the subgraph has an attribute that disables current measurement according to Kirchhoff's law, e.g. is a source/sink for current and cannot provide that information*

To enable the FM to perform fault detection, localization and isolation, additional attributes that are determined online have to be defined. For the fault detection current measurement has to be provided, in order to be able to run the protection schemes. For the isolation of the fault, the current status of all switches has to be known, to enable calculation of an isolation set. This isolation set provides information about which generators supply power to the faulted section and which loads are affected and can be determined by applying graph search algorithms to the system graph. With necessary and useful definitions proposed in this section, fault management in MVDC SPS can be performed based on this information.

2.2.2 MVDC Fault Management

In this section the fault management of medium voltage DC SPS is explained in further detail, with focus on the sequence of actions that are necessary to handle faults efficiently. In the USA the common frequency for AC power is 60 Hz, which means one cycle has approximately 16.6 ms. Regular AC breakers wait for a zero crossing of current when trying to open. In the worst case this could be up to half a cycle. CAPS therefore set their goal to achieve fault management in less than 8 ms, to prove that fault management in DC systems can be achieved at least as fast as in AC systems by utilizing fast disconnect switches and special converter modules. The general fault management process in MVDC SPS has four main steps:

1. Fault detection and localization
2. De–energizing the system
3. Isolating the fault
4. Re–energizing the system

The sequence of these four steps along with the corresponding voltage and current levels is shown in Figure 2.15. Fault detection and localization is performed by a dedicated system

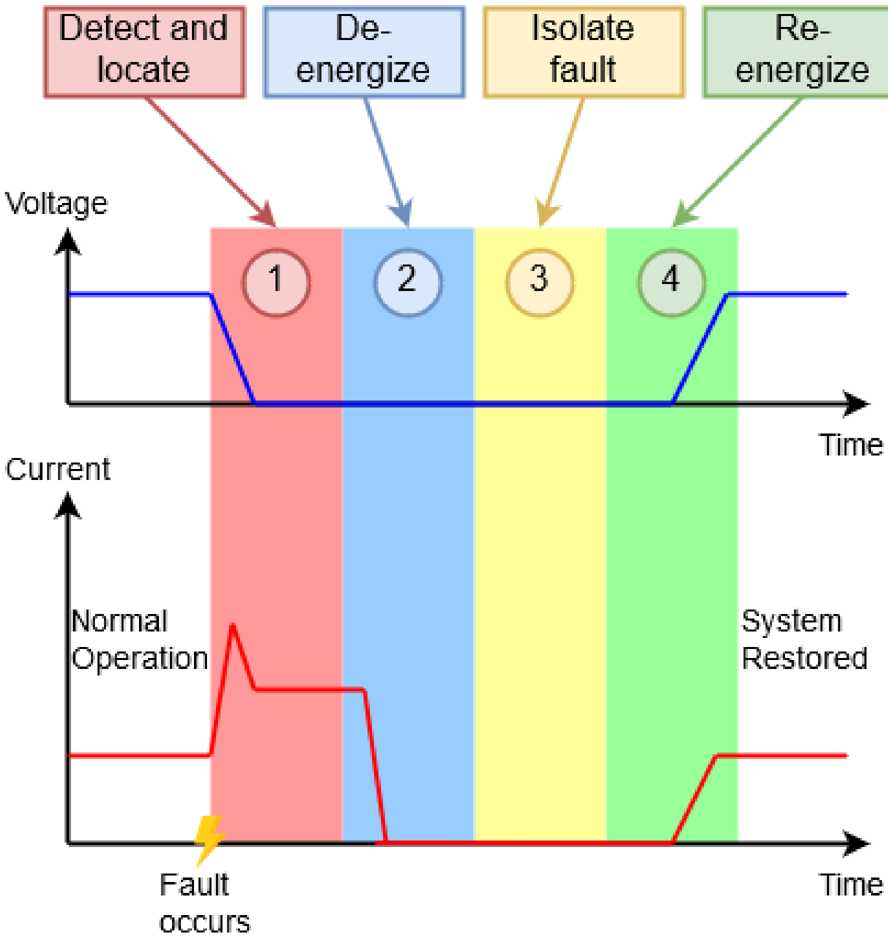

Figure 2.15: Fault management sequence in MVDC SPS with corresponding voltage and current levels

monitoring current flows through different points throughout the SPS. The previously defined fault management sections can be utilized for this purpose. After the fault was detected and located, the system has to be de–energized to enable isolation of the faulty section by opening disconnect switches. For this all devices in the same section as the fault have to be identified, and all sources feeding power to this section have to be powered down to achieve negligible current flow through switches. The generators are not directly connected to the system, but rather feed their power through converters, which have the ability to ramp their output up or down in a matter of milliseconds. Once the faulted section is de–energized, the command to open all necessary switches to isolate the fault can be sent. Finally, the system can be re–energized and resume operation.

Once the fault was successfully isolated, the system might be left in an undesirable state, where one or more loads might be without sufficient power supply. To prevent significant impact on a mission by under–supplying vital mission loads, probably load shedding, reconfiguration of the system or a combination of both might be necessary. This work focuses on the reconfiguration, a new component in the control architecture, where it is assumed that load–shedding is taken care of beforehand. The load shedding will most likely be a third separate component, but it might also be included within other controls. The calculation of a new configuration can be executed in parallel to parts of the fault management process. This reduces the time required for reconfiguration and thereby impact on the system. To ensure optimal operation of the combined fault management and reconfiguration, certain information needs to be exchanged between those systems. More specifically, the reconfiguration part needs information about:

– which section is affected by the fault, i. e. switch states of the system;
– which parts of the system will be at which voltage levels to isolate the fault;
– which loads, and which sources will be operational or shed;
– the operating power level of these devices; and
– priority of these devices, depending on the current mission.

This information is available after a fault has been detected, located and a load–shedding scheme determined if any loads have to be shed, and if so, which ones. Afterwards a new configuration can be calculated based on expected system state and provided online information. The timing sequence of this parallel processes is depicted in Figure 2.16. Once the fault management detects a fault, it starts with locating the fault and identifying the section that has to be isolated. Once this information is available it is passed on to both the load–shedding and the reconfiguration logics. The load shedding can then perform a search through the system graph to identify isolated or islanded parts in the system and calculate which loads have to be shed to ensure that the reconfiguration problem can be solved. As soon as the load–shedding scheme finishes, the active device set is published to the reconfiguration algorithm, so the calculation of a new configuration can be started. At the same time the fault management identifies which sources feed power to the faulty section. Once all such generation modules have been identified, the command to power them down is sent to the electrical system. After the faulted section has been de–energized it can be isolated and once the new configuration is known, the corresponding signals to implement the new state can be sent. Finally, when the new configuration is active, all generation modules can begin restoring their power to regular operation. With the goal of achieving fault handling in less than 8 ms, the timing constraints on any chosen reconfiguration algorithm are strict. On the other the zonal architecture discussed in Section 2.1.3 adds a lot of complexity to finding an optimal or close to optimal configuration. To identify possible approaches that meet both of these criteria, a literature review has to be done on previous solutions to reconfiguration of shipboard power systems.

2.3 Conclusion

In this chapter the different aspects of future all–electric shipboard power systems were reviewed. First a general overview and a comparison to terrestrial utility systems was

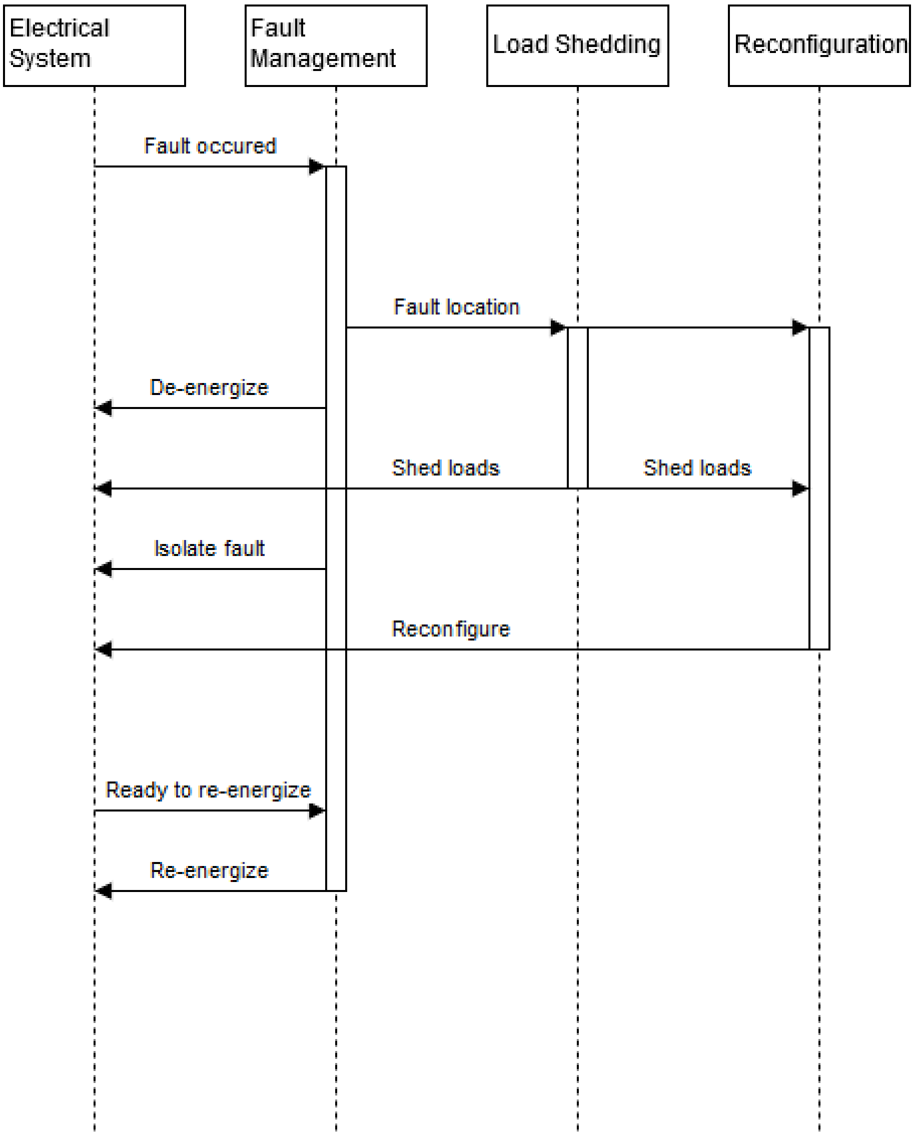

Figure 2.16: Sequence diagram of the parallel processes of fault management, load–shedding and reconfiguration

given. Then the advantages and disadvantages of applying the proposed medium voltage direct current systems were discussed. The idea behind the envisioned zonal electrical distribution system was explained, as well as the opportunities and challenges that come with its implementation in a SPS. Finally, the notional zonal medium voltage direct current shipboard power system model developed by the ESRDC, that incorporates all of the previously discussed designs was presented. This model will then be used in this thesis to verify an approach for the reconfiguration of a SPS merging MVDC and ZEDS aspects. The second part discussed the fault management in MVDC shipboard power systems. First a graph representation of an electrical system was introduced along with definitions that aid the fault management process. Then the different steps of the process and timing details were presented. The next chapter goes into more detail on previous work that has been done on the reconfiguration of shipboard power systems, the advantages and disadvantages of various approaches found in literature, as well as their applicability to the proposed architecture.

3 Literature Review

This chapter discusses approaches found in literature and gives an overview of previous works on the topic of shipboard power system (SPS) reconfiguration, as well as individual advantages and disadvantages of specific approaches and implementations.

3.1 Literature Review of Reconfiguration Algorithms

This section gives an overview of approaches to the reconfiguration of a shipboard power system found in literature. When a fault occurs on the SPS it has to be detected, located and isolated. Once the faulty part is disconnected from the remaining electrical system, the next immediate step is to restore power to loads that might be left under supplied through the change in system topology. Possible scenarios for this could be a feeding line that is no longer available, loss of a generator or one or more faults at various other points throughout the system. Primary objective of the reconfiguration is then to find a configuration that supplies power to as many loads of as high priority as possible, where priority categories could range from a coarse distinction between vital, semi–vital or non–vital loads, to precisely defined rankings of individual loads. Secondary goals for the reconfiguration process could be:

- minimizing switching actions;
- reducing impacts of faults after the reconfiguration;
- ensuring radiality of resulting configuration;
- minimizing power loss; and
- ensuring equal utilization of generation modules.

This optimization problem can be stated as a mixed integer nonlinear programming (MINLP) problem. The term MINLP refers to optimization problems with both continuous and discrete variables, hence mixed integer, and nonlinear functions in either or both the constraints and the objective function. The general form of an MINLP is shown in Equation 3.1.

$$\begin{array}{llcll} \text{minimize} & f(x,y) & & & \\ \text{subject to} & c_i(x,y) & = & 0 & \forall i \in E \\ & c_i(x,y) & \leq & 0 & \forall i \in I \\ & x & \in & X & \\ & y & \in & Y & \text{integer} \end{array} \tag{3.1}$$

where $c_i(x, y)$ is a mapping from R^n to R, and E and I are index sets for equality and inequality constraints, respectively. As such, MINLP are categorized as NP–hard problems,

as they include mixed integer linear programming (MILP) and integer linear programming (ILP) problems, which in turn are NP–hard. NP–hardness will not be discussed in further detail in this work. In the context of this thesis it is only necessary to know that there is no known algorithm that can solve such problems in polynomial time. This means, that the growth of complexity is larger than a given polynomial function of an input. A polynomial time function would be $\mathcal{O}(n^{\alpha})$ with the input n and the constant $\alpha > 1$.
Since there is no known algorithm that can provide global optimum polynomial time solution for NP–hard problems, the reconfiguration as such a NP–hard problem, is intractable if the system size exceeds a certain limit while still aiming for a global optimum in all considerations without any restrictions. This means that there is a broad variety of possible approaches for the reconfiguration, all of them with their own advantages and disadvantages.
One popular approach often found in literature on the reconfiguration of a SPS is to apply different heuristic search algorithms [7, 8, 12–16], like ”Modified Plant Growth Simulation Algorithm”, ”Genetic Algorithm” (GA) or ”Particle Swarm Organization” (PSO). Other approaches that have been applied to SPS reconfiguration include network flow [17, 18], multi–agent systems [19, 20], expert systems [21] or hybrid solutions [22, 23].

3.1.1 Heuristics

A heuristic is any approach to solving a problem that employs a practical method that is not guaranteed to be optimal, perfect or logical, but instead is sufficient to achieve a prompt goal. Where an optimal solution is not possible or impractical, heuristic methods can be used to speed up the solution finding process. More specifically, in computer science and mathematical optimization, heuristics is a technique used to solve a problem faster when classical methods are too slow, or to find an approximate solution when classic methods fail to identify an exact solution. This is achieved by trading one or more of the following for speed:

- Optimality; a heuristic might not find the single best solution
- Completeness; a heuristic method might only find a single solution, instead of finding all of them
- Accuracy; heuristic approaches might only approximate a solution within a confidence interval

All of these trade–off criteria have to be evaluated, if heuristic approaches can be deemed useful. The goal of heuristics is to find a solution within a reasonable time frame that is good enough to solve the problem at hand. This solution may not be the best of all solutions to this problem, or it may simply approximate the exact solution, but it’s still valuable because it doesn’t take an unaffordable amount of time.
Heuristic approaches have the advantage of reducing the runtime drastically by not having to do an exhaustive search of all possible options to find the optimum. They approximate

the global optimum or settling for a local one that can be found faster. It is one of the more common approaches found in literature regarding the reconfiguration of shipboard power systems, e.g. [7,8,12–16], because an exhaustive search is usually too time consuming. This is especially true with the goal to find a reconfiguration option online and in real–time for a large system. Almost all of the papers presenting heuristic search methods to find a reconfiguration focus more on the load shedding aspect of recovering a functional system state.

Moreover, the used topology was often simple enough to be able to neglect concerns of connectivity. The only constraints addressed in [12] are radiality and capacity limits and then load shedding is optimized with a genetic algorithm approach. Connectivity restraints are not discussed as the layout is plain enough.

One layout that is used in some publications applying heuristics is the 8 bus system, which is significantly simpler than the notional zonal model, shown in Figure 3.1. This architecture only requires basic connectivity checks, [13] and [8] do not mention any form of validation for a solution regarding electrical connection, but focus on load shedding with PSO and GA approaches. In [8] the slightly larger 13 bus system is also mentioned, but the approach is still the same. GA or PSO are used to find a solution of which loads to supply, then this solution is checked for power flow constraints, line overloads, and voltage limits. If any of these criteria are not met the solution is rejected and PSO/GA is used to calculate the next solution. The 8–bus system is also utilized in [7], where graph search

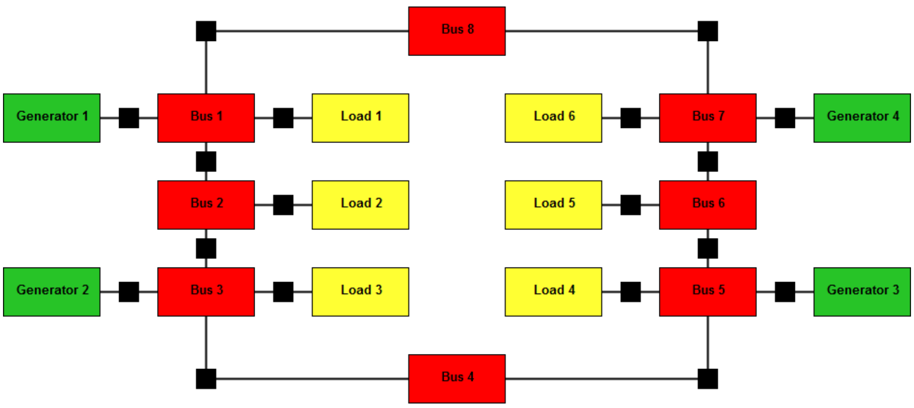

Figure 3.1: 8 bus system

is used to identify all negative sections in the system. Then these sections are regarded for the solution and PSO is applied to the load shedding problem. A similar approach is used in [14], even though it is applied to a terrestrial distribution system. First a random set of switch positions are generated, then a depth first search is applied to the system to determine if radiality and connectivity is maintained. Finally, Discrete Artificial Bee Colony (DABC) is used to calculate the maximum loadability of the system. In [16] a heuristic approach is applied for the reconfiguration of the general zonal SPS architecture proposed in Figure 2.3 in Chapter 2.1.3. However, this system layout does not consider bus faults; hence connectivity is always provided, and only load shedding is optimized.

Other systems that utilize heuristic search techniques frequently are terrestrial distribution systems [15], but as discussed in Chapter 2.1, these focus on radiality and loss reduction in systems that are often simpler than the notional zonal SPS and are not designed to perform under real–time constraints.

3.1.2 Multi Agent System

Multi–agent systems (MAS) are composed of multiple interacting intelligent agents. They have various advantages over individual agents or monolithic systems. They might be able solve problems that are difficult or impossible to solve for an individual agent or a monolithic system, can achieve better performance and runtime, and add redundancy and survivability to system controls. Due to the redundancy of components, the systems also tend to prevent propagation of faults, self–recover and be fault tolerant.
MAS may be useful with reconfiguration approaches as the increased redundancy would support the requirements of shipboard power system controls as described in Section 2.1. Publications presenting multi agent system approaches for the reconfiguration of shipboard power systems focus less on the actual configuration that the algorithm will result in, but more on the communication architecture and the how information is shared and exchanged between agents. The MAS in [20] utilizes the average–consensus theorem to discover global information, and after the location of unfaulted loads and sources is known, the load shedding problem is solved as NP–hard optimization problem, but no remarks regarding connectivity, radiality or number of switching actions are stated. Moreover, no specific approach to solve the optimization problem is given in the publication. In [20] a system layout somewhat similar to the notional zonal model presented in Section 2.1.4 would be considered, but again the paper focuses mostly on the information exchange between agents and states no specific algorithm for calculating a new configuration. Therefore, these approaches provide no solution to find a new configuration, but rather can be combined with other approaches to increase system reliability, execution speed and fault tolerance of control.

3.1.3 Network Flow and Graph Based

Both [17] and [18] utilize a graph representation of the system that has to be reconfigured, to formulate constraints and objective functions for an optimization problem. By doing so, the service restoration problem is reformulated as a variation of the fixed charge network flow problem. The network flow method restores a maximum amount of load demand while satisfying line capacity voltage and radiality constraints, but load priorities are not considered. This reduces the problem to a mixed integer linear form, which is still NP–hard. Therefore, solving the optimization problem cannot be performed in polynomial time. This fact, in combination with the lack of enforcing load priority, renders this approach sub–optimal for the reconfiguration of a zonal shipboard power system.

3.1.4 Expert Systems

A knowledge based expert system as described in [21] is fast while still maintaining the optimality as designed. In general, those systems would be a very good solution, the main problem is although that such systems have to be manually programmed and are therefore very prone to human error during implementation. Another disadvantage is that minor changes in the system topology or operating profiles would require the system to be redesigned to certain extent, where more generic approach can be used for a broader range of architectures. Moreover, with the system become large and more complex, the various options that have to be regarded in the final control might become intractable, which would render the implementation and testing of controls to time–consuming and error prone.

3.2 Decision Criteria

In this section the main decision criteria for reconfiguration approaches and their suitability for application to a MVDC ZEDS architecture are stated and discussed, followed by a comparison of reviewed techniques from the previous section.
The three main criteria a reconfiguration approach has to fulfill that are considered in this work are:

- Time;
- applicability; and
- implementation effort

As described in Section 2.2.2, the goal of CAPS is to achieve recovery of a fault within 8 ms. Therefore, time is the main criterion for reconfiguration approaches that are to be applied on future shipboard power systems. The second criterion is applicability of the chosen approach to a system as complex as the zonal SPS. The approach should incorporate means to verify attributes of solutions like connectivity and generation capacities. The last main criterion is effort needed to implement the chosen algorithm. The main reason for this criterion is that the more effort is necessary, the higher is the probability that errors are made accidentally. Moreover, an approach that requires a lot of effort to implement increases the costs and may also complicate testing and validation.
Now that the main criteria for the decision of approaches have been defined, the reviewed solution can be compared using these criteria. The first approach that was reviewed were heuristic search algorithms. Their main advantage was reduced runtime in exchange for a trade–off in optimality, completeness or accuracy. Therefore, they would meet the first criterion of runtime. Also, the implementation effort is reasonable. Their main disadvantage is that without lots of effort they would either not be applicable to the more complex system, as there is no way to identify connectivity outright by simply identifying a solution. Therefore, heuristics are not optimal for the reconfiguration of a zonal shipboard

power system.
Another approach associated with SPS in literature war the employ multi–agent systems. While the use of MAS might be useful in combination with other approaches, no specific reconfiguration solution was stated in publications on SPS reconfiguration, therefore no approach found was applicable to the zonal SPS either.
Network flow was used in some works to identify possible paths throughout an electrical system. While they would be applicable in the sense of connectivity and also could be implemented with reasonable effort, the approach is reduced to a MILP problem which cannot be solved polynomial time and is therefore intractable with the system size of the model developed by the ESRDC. Furthermore, they lack means to take the current system state into account, which reduces the initial applicability.
Finally, expert systems were reviewed. Their main advantage is that since they rely on very little generic procedures but rather on processes tailored to the system at hand, they are easily able to meet timing constraints imposed on online fault handling. Also, the system specific development ensures applicability of this approach to even most complex systems. The overwhelming downside of this approach is the development and implementation effort. As an intractable amount of solutions has to be considered at development time, the effort to even design such an expert system is significantly higher than other approaches discussed in this chapter. Furthermore, implementing this solution is highly error prone as an enormous number of lines of code has to be written, increasing the probability of human error drastically, especially for such large–scale projects.
This comparison has shown that none of the approaches previously used for SPS reconfiguration are applicable to the new, more complex system within desired constraints. Therefore, a new approach has to be found that makes use of known attributes and requirements of the system that is to be reconfigured.

3.3 Conclusion

Many different approaches on the problem of reconfiguring a shipboard power system have been proposed in literature. One of the most commonly found solutions is to apply heuristics in order to be able to drastically reduce the time consumption of calculating a solution compared to an exhaustive search. These techniques allow to find a good approximation of the global optimum in comparably short times but have not been used on more complex systems that require additional checks of connectivity. Adding these constraints would increase the computational complexity of these approaches significantly. Multi agent systems have been proposed in some papers, but these focus on the information discovery between agents and provide no algorithm that would be applicable to the reconfiguration of a SPS in real–time. They might be useful in combination with other approaches though, to increase quality of controls. Another option is to utilize network flow, but this approach provides no possibility to incorporate load priority. Furthermore, the current switch states are not taken into account, to reduce the number of switching actions that have to be carried out to implement the new configuration, which might affect loads unnecessarily. Finally, expert systems that are based on available information about the system were reviewed, but this approach is less flexible and more error prone during implementation than an automated reconfiguration algorithm. A comparison of approaches showed that a new approach has to be developed that better utilizes known attributes to fulfill requirements imposed on SPS reconfiguration. The next chapter presents methods utilized to approach the problem of reconfiguration of a shipboard power system.

4 Methods

This chapter explains the utilized approaches for the development of reconfiguration solution for a shipboard power system and approaches to generating test data to verify the new solution.
The decision criteria stated in Section 3.2 and the following comparison of existing approaches showed that no solution currently found in literature is optimal for the reconfiguration of a shipboard power system with zonal architecture. A good solution may be able to make use of known or desired characteristics of the system, like connectivity and current state of switches. Moreover, it should also meet the aforementioned criteria. This means the entire fault management process as described in Figure 2.16 should be done in less than a few milliseconds to meet the timing goal set by the ESRDC. Also, the solution should be able to find an optimal or close to optimal solution for the reconfiguration problem of a zonal SPS, where optimality might depend on various aspects such as current mission profile, load priority or number of switching actions. Finally, the process of developing, testing and implementing the approach has to be possible with reasonable effort while minimizing susceptibility to errors.
A graph–based reconfiguration approach will be defined. This allows to easily include such limitations, while enabling use of well–known algorithms for graph traversal.
By defining a representation that is better tailored for this goal, an efficient and well–defined goal of a reconfiguration approach can be formulated. Then an algorithm that solves this problem is designed and improved to better meet the real–time constraints. The theoretical performance of this approach will be evaluated.
To test this new algorithm empirically a testbed is designed, that allows verification by introducing faults on a simulated shipboard power system in different locations.
The logic controlling this process is connected to a simulation testbed, representing an actual shipboard power system. The simulation model used is a simplified version of the notional zonal four–zone MVDC SPS proposed by the ESRDC in [24].
To guarantee realistic conditions, fault identification, localization and isolation is included in the control architecture, disabling affected sections and providing information for the reconfiguration. To demonstrate this integration of reconfiguration into the overall control architecture, fault management as described in [2] is integrated with existing components. After a fault is detected and handled properly, a rudimentary, priority–based load–shedding algorithm component identifies devices that have to be shed so high priority load can still operate at the required power levels. After this, a reconfiguration algorithm proposed in the next chapter calculates if reconfiguration is necessary, and if so, which actions have to be taken to implement a better system state.
There are some limitations to the empirical testbed used. In this stage of the development, the three different control components of fault management, load shedding and reconfiguration have not been separated, and run on the same processor for reasons of

simplicity. Compared to the parallelized architecture, this neglects delay and errors that would be present in actual communication. On the other hand, runtime is slowed down, as the processor executes these 3 logic programs in sequence instead of the parallel runtime shown in Figure 2.16. Therefore, the reconfiguration part of the entire fault management process has to be considered separately when analyzing the produced test data for runtime limitations.

The next chapter will discuss the development of a graph based approach that focuses on the reconfiguration of a SPS according to the notional zonal architecture presented in Section 2.1.4.

5 Reconfiguration Approach

This chapter proposes a new approach for the reconfiguration of a shipboard power system, describes the theory behind the graph–based approach and deals with necessary pre–calculations that can be done beforehand to reduce the runtime during online reconfiguration.

5.1 Section Capacity and Section Graph

In this section definitions regarding the design of an approach to reconfiguration of a zonal shipboard power system are introduced. For the reconfiguration of the SPS, after fault has been handled, a number of online parameters have to be provided, dependent of what aspects shall be considered when calculating a new configuration. One essential piece of information is the operating power level of loads and the generation capacity of generators, as these factors determine if any given system state is even a valid solution. In this thesis a valid solution for the reconfiguration problem is defined as:

Definition 5.1. *Valid Solution:*
Any given solution, defined by a vector containing all switch states, is considered valid if, and only if every active load is supplied with its required level of power, via an electrically conducting path through the system from one or more source modules.

With this definition it is possible to categorize any given combination of switch states into either valid or invalid solutions. This categorization can be done with a graph search through the system graph and identifying the sum of required and demanded power levels in connected sections. To enable a more efficient way of keeping track of this power level of a section, a metric is defined:

Definition 5.2. *Section Capacity:*
Difference between sum of power generation capacity and load power demand of all devices contained in that section as shown in equation 5.1.

$$SectionCapacity = \sum CapacityGeneration - \sum LoadPresent \tag{5.1}$$

If the result of Equation 5.1 for a section is positive, that section has excess capacity available after all loads are supplied, if it is negative the section would need more power than currently available on its own, which means not all loads can be powered to the desired extent. Therefore, according to Definition 5.1, a system state is only a valid solution if all section capacities in the power system are positive. This information can then be used as an input for the reconfiguration algorithm, as all sections with negative section capacity have to be connected to ones with sufficient positive capacity. By transforming the system graph into another representation that includes said information about capacities, a reduction of the graph size can be achieved, and search algorithms can be applied faster and calculate additional information during execution. For this reason, the following definition describes a graph composed of sections:

Definition 5.3. *Section Graph:*
The section graph is composed of sections as vertices, and if two sections share one device capable of isolation, an edge between those two vertices exists. Every section may have attributes dependent of the application of the graph. Moreover, no two sections in the same graph should have more than one electrical device in common.

The definition of a Section Graph allows to transform a regular system graph into a section graph and vice versa, while still enabling application of graph algorithms on a representation with reduced complexity. An example of a section graph is shown in Figure 5.1. Another advantage of this newly introduced graph is that already connected parts

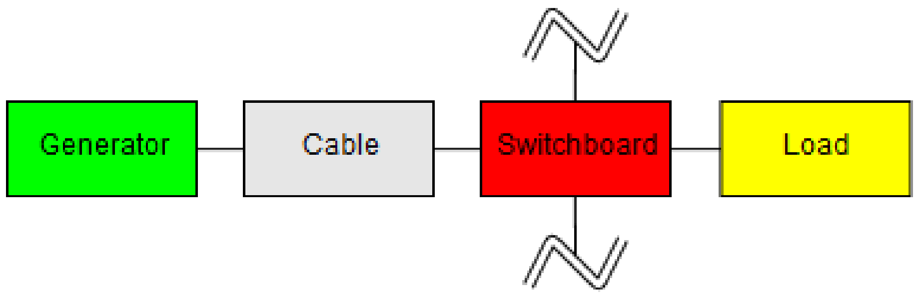

Figure 5.1: Section graph representation of the system graph in Figure 2.9

can be merged into single vertices and further simplifications can be made without losing information, as described in the following sections.

5.2 Calculation of Sections

This section deals with the calculation of the minimal sections as described in Definition 2.3. One approach that would ensure the calculation of all such sections would be to simply generate all connected subgraphs, then remove all that do not fulfill the properties of minimal sections, and for the fault detection reduce this set of subgraphs even further with additional restrictions. This approach obviously becomes intractable as the graph size grows, as the upper bound of connected subgraphs is of the order $\mathcal{O}(2^{|V|})$. The Big–O notation is used to describe the upper bound for growth in complexity of an algorithm. In this case the growth of complexity of calculating all connected subgraphs is bound by two to the power of the number of vertices. By utilizing the known characteristics these sections have to fulfill and the system layout Algorithm 5.1 with better performance and scalability was developed to calculate the basic minimal sections. The algorithm

Algorithm 5.1 Calculate Minimal Sections

INPUT: System Graph G
OUTPUT: SectionList

```
for each Vertex in G with the attribute can isolate do
    for each Neighbor of Vertex do
        DeviceSet ← Vertex
        VerticesToVisit ← Neighbor
        while VerticesToVisit is not empty do
            CurrentVertex ← ArbitraryElement of VerticesToVisit
            remove CurrentVertex from VerticesToVisit
            add CurrentVertex to DeviceSet
            if CurrentVertex does not have the attribute 'can isolate' then
                for each Neighbor of CurrentVertex not in DeviceSet do
                    add Neighbor to VerticesToVisit
        add DeviceSet to SectionList
return SectionList
```

considers every vertex that is a valid border vertex for a section, i. e. has the attribute *can isolate*. Then it considers the neighbor and if that neighbor *cannot isolate* it keeps adding its neighbors until all branches either terminate or encounter valid border vertices. The worst case runtime bound for the basic algorithm is $\mathcal{O}(|V|^2)$ but this can be further reduced by implementing a check if the start vertex and its neighbor are contained in one of the already calculated sections, then the bound is $\mathcal{O}(|V|)$, i. e. linear with t he amount of vertices. Algorithm 5.1 assumes that no electrical device has a degree $d > 2$ in the system graph, which can be implemented by introducing simple bus–vertices, that have no attributes and connect multiple otherwise interconnected devices, e.g. the center of a switchboard. This restriction can be removed by including an additional check if no two vertices in the current section share a neighbor that is not included. This reduces Definition 2.2 to the first two points:

– $Order \geq 2$

– All cross–vertices have the attribute *can isolate*

The algorithm will find only valid sections according to this definition. The restriction of $Order \geq 2$ is naturally fulfilled, as the algorithm only starts constructing sections with a start vertex and a neighboring vertex. Hence, even if the neighbor of the start vertex is a device capable of isolating current flow, the minimal order of two vertices is already reached. The second requirement is always fulfilled as every path from the start vertex outwards is fully explored until either a) no more vertices can be added to the path, or b) a vertex with the ability to isolate current flow is encountered. Case b) is exactly the definition of the requirement, having the attribute *can isolate*. In the case of a) the last vertex added has no connection to the rest of the system graph, and therefore does not need to have this attribute.

The completeness of the algorithm can also be easily verified, as every device with the attribute *can isolate* is considered in combination with each of its neighbors. Hence, if the

algorithm is correct, it is also complete.
Algorithm 5.1 terminates definitely, as the number of vertices in the graph that can isolate is finite and for every one of those vertices their degree is limited, as is the amount of vertices that can be added to a section.
For the calculation of the fault management sections two possibilities exist. Either taking the result of Algorithm 5.1, iterating through all sections and checking for each one, if this section fulfills all requirements for a fault detection section as in Definition 2.4, or modifying Algorithm 5.1, such that it only returns valid fault detection sections and run it separately.

5.3 Graph–Based Approach for Reconfiguration

This section first discusses what the objective of the reconfiguration is and the general procedure of how the proposed algorithm works. Then the advantages and limitations of the chosen approach are presented, followed by additional improvements to the general procedure to improve the runtime. Finally, the theoretical worst–case complexity is compared with the actual model.

5.3.1 A General Graph–Based Reconfiguration Approach

By using the earlier defined section graph it is possible to define a fact on the reconfiguration. Given a section graph G and a section S with $S \subseteq G$ $SectionCapacity(S) < 0$ following statement can be made:

Lemma 5.1. *A tree T, with $T \subseteq G$, $S \subseteq T$ and $SectionCapacity(T) \geq 0$, provides a valid reconfiguration solution for S.*

Proof. Suppose $SectionCapacity(T) < 0$ and T is still a valid option to reconfigure the system in such a way that all loads in S and T are sufficiently supplied. This means if all necessary switches are brought to a state that implements T as a connected section, the sum of power demand of all loads in that section exceeds the sum of generation capacity in T. This contradicts the assumption that that T is a valid reconfiguration solution for S. Therefore, the initial assumption was wrong and $SectionCapacity(T)$ has to be ≥ 0 for T to be a valid reconfiguration option. □

The remaining requirements for T come naturally, as T has to be a subset of the section graph, as otherwise it would contain elements not in the system and S clearly has to be a part of T in order for S to be reconfigured by T. Additionally, the definition of a tree ensures that electrical connectivity is provided and radiality of the resulting system configuration is maintained. With Lemma 5.1 an objective for the reconfiguration can be defined as:

Definition 5.4. *Objective of the Reconfiguration Algorithm:*
Find a cycle free subgraph of the section graph, such that it contains a section with negative capacity, the combined section capacities are positive, and a fitness value based on predefined criteria is minimized.

Using this approach has several advantages, as the tree ensures that the resulting layout is radial, and that all loads in the tree are connected to generation devices that supply them sufficiently. A brute force solution would be to simple calculate all trees containing the section with negative capacity and then compare them all and find the one with the best fitness value. The algorithm that calculates all trees is displayed in Algorithm 5.2. Algorithm 5.2 calculates all possible tree combinations that contain an initially specified

Algorithm 5.2 Compute All Trees with Capacity, Fitness and Path

INPUT: Section Graph G, Next Sections, Capacity, Fitness, Path
OUTPUT: List of valid solutions with respective paths and fitness values

```
function ComputeAllTrees(G, NextSections, Capacity, Fitness, Path)
    for each Section in NextSections do
        update Capacity, Fitness and Path
        for each Neighbor of Section not in Path do
            add Neighbor to NeighborList
    for each Combination in powerset(NeighborList) do
        ComputeAllTrees(G, Combination, Capacity, Fitness, Path)
    return Capacity, Fitness and Path as Tree
```

section. This is done by storing all neighboring vertices of active leaves of the current tree in a list and then using all combinations in a powerset of this list of leaves as the next level of leaves for new trees. A powerset of any set S is defined as the set of all subsets of S, including the empty set and S itself, where for our case the empty set can be neglected, as this would result in the same tree that served as a basis for the current iteration. This way the algorithm calculates every possible combination containing the initial vertex, guaranteeing correctness of the algorithm. It has to terminate as the number of combinations of vertices and edges is limited.
For each tree the algorithm keeps track of section capacity and mission–based fitness value. After the calculation they all have to be compared in terms of fitness value and it has to be verified that they also provide a valid solution in terms of a positive section capacity.

5.3.2 Strengths

The main advantages of the proposed algorithm, compared to other solutions discussed earlier, are:

- Tree calculation always ensures radiality;
- tree guarantees connectivity between vertices; and
- fitness value and capacity are calculated along with the solution.

That the algorithm result is a radial layout is naturally provided by the definition of a tree as a cycle free subgraph, hence the electrical network will also be cycle free. This holds true, even if multiple sections require reconfiguration, assumed those sections are reconfigured sequentially and the section graph is updated after a solution has been found for one or more negative sections. Another attribute that is naturally enforced through the tree calculation is that negative sections are electrically connected to sources that provide sufficient power supply. If only switch states are considered, every combination has to be checked concerning the connectivity of electrical devices, and their combined power capacity. To determine these attributes would require searching through the system graph, which has the runtime bound of $\mathcal{O}(|V|+|E|)$, which impairs scalability if necessary for every combination.

The fitness value has to be determined in the same way. In this thesis the fitness value is based upon multiple criteria:

- Size of the resulting section;
- priority of affected loads; and
- the priority of the utilized sources

The size of a solution that will be implemented is important, because once the system has been restored to regular operation, a larger section in terms of more devices and longer cable sections is more prone to faults while affecting more loads when faulted. A smaller section has a lower probability of an error occurring, be it through material failure or external factors. Hence, inclusion of high priority loads like the propulsion motors in the reconfiguration path of other sections, worsens the fitness value of a solution compared to solutions that find other ways of providing the necessary power. The actual effect on the system and the loads depends on the final system design, especially the ride–through capabilities of power storage equipment dedicated for certain vital loads.

Once a reconfiguration path has been found, the implementation of the desired state might require still functional buses to be brought to voltage levels that allow closing of switches or currents to be negligible to enable opening of power electronic devices like switches. The use of converters enables rapid control of these values, which reduces the effect on loads and sources to a minimum. Again, by using energy storage units for high priority loads, these might not experience any effects of the reconfiguration at all, assuming this process happens fast enough.

Another value that can be incorporated within the fitness is the desire to utilize certain generation devices more than others, to evenly distribute wear and required maintenance between generators. All of these are only a few examples of what values can be merged into the fitness value. Once defined, they need to be weighted to be able to compare them, e.g. the size of the resulting section might be the least important of these, while the priority of affected loads might have the most impact on the fitness value of a given solution. Equation 5.2 shows how the fitness value of a section is calculated.

$$f_{Section} = w_j * (1 - \frac{s}{s_{max}}) + w_k * (1 - \sum p_{load}) + w_l * [1 - \sum p_{source}] \tag{5.2}$$

In this formula $f_{Section}$ is the fitness value of the section a reconfiguration would result in, ranging from 0 to 1. A higher fitness value indicates that one solution better fits the defined criteria than solutions with lower fitness values. The expression $(1 - \frac{s}{s_{max}})$ represents the size of the resulting section, s being its relative size, normalized by s_{max}, the maximum size of the system. This term is 0 when the section includes the entire system and is therefore the largest possible section and converges towards 1 as the system size decreases. In this thesis the size is assumed to be the number of devices in that section. The variables p_{load} and p_{source} are load and source priorities of modules included in the resulting section respectively, which are normalized such that the sum of all p_{load} is 1 and the sum of all p_{source} is 1. To prevent inclusion of unnecessary source modules, while still preferring high priority sources, source–priority is inverted. A high priority source therefore has a lower value. w_j, w_k and w_l are the assigned weight values for size, source priority and affected load priority, with the condition that: $w_j + w_k + w_l = 1$. Equation 5.2 can be rephrased as Equation 5.3.

$$f_{Section} = 1 - (w_j * \frac{s}{s_{max}} + w_k * \sum p_{load} + w_l * \sum p_{source}) \tag{5.3}$$

5.3.3 Limitations

One of the limitations the proposed algorithm has, is that multiple negative sections have to be considered sequentially. One recursion of the algorithm might provide reconfiguration options for more than a single negative section but does not necessarily do so. By setting the fitness value of already affected sections to zero, optimality in terms of load and source priority can still be ensured, merely the result might connect two sections unnecessarily, depending on the actual constellation of the system and the sequence in which negative sections are reconfigured. To consider all sequences or compare all valid solution combinations of every negative section would increase the complexity and runtime too much to still be applicable in real–time. A better solution would be to examine the resulting section and look for switches that could be opened while still maintaining positive capacity on both of its sides. This can be accomplished faster than to compare all combinations, especially if the number of sections that require reconfiguration grows.
The second limitation of the proposed algorithm is that it has no inherent way to decide which loads to shed in the case of a so–called islanding scenario or a generator being lost due to a fault. This restriction can be solved by applying a search algorithm on the system graph once a fault has been detected, and identify all device sets that theoretically be connected. Then any load shedding algorithm may be applied to these sets, either plain and uncomplicated ones that are simply based on priority and stop once positive capacity can be achieved, or more complex approaches as presented in [7, 8, 12–16, 20].

5.3.4 Runtime Improvements and Computational Complexity

This part deals with the computational complexity of the chosen approach, as well as possible improvements to reduce this complexity and enhance the runtime performance of the reconfiguration algorithm. As the algorithm is envisioned to run in real–time during a fault recovery sequence, additional improvements regarding runtime and computational complexity are useful to achieve better execution times. One reduction that has great effect on both runtime and worst–case complexity is to collapse sections with specific attributes. This enhancement can be executed beforehand to reduce the size of the section graph. Further improvements can be gained by adding checks to the implementation of the algorithm.

5.3.4.1 Computational Complexity

The computational complexity of the proposed reconfiguration algorithm is dependent of the number of trees that can be found in the graph representation of the electrical system. When the number of vertices n (=|V|) and the number of edges e (=|E|) of a graph are known the theoretic maximum number of trees F can be calculated by choosing up to $n-1$ edges and sum the number of combinations up. So, without loss of generality, the maximum number of trees can be calculated with Equation 5.4:

$$F(n,e) \leq \sum_{k=0}^{n-1} \binom{e}{k} \tag{5.4}$$

Relevant for the upper bound on the reconfiguration algorithm are only the trees containing the negative section, i. e. trees with the negative section as root, and also Equation 5.4 does not take into account that some of those combinations are actually not connected and therefore not trees but forests. Unfortunately, there is no simple way to formulate the problem such that these additional restrictions are incorporated. To find such a closer bound based on more detailed information about the system is not in the scope of this thesis. One option is to calculate the actual number of trees once the system layout is defined, as the topology will not change afterwards.

5.3.4.2 Collapsing Sections

Certain sections are of no specific use for the reconfiguration and can therefore either be merged with other elements of the graph or disregarded entirely. This reduces the number of both vertices and edges of the section graph, and therefore in turn the average and worst–case execution time of the algorithm. Every section that can be collapsed or merged reduces both $|V|$ and $|E|$ by one. Reducible sections are

- Every section that has only 2 neighbors and a capacity of 0 can be collapsed into an edge between its 2 neighbors

– Every section with only a single neighbor can either be merged with its neighbor if it has active loads/sources, or disregarded otherwise

If a section has no own capacity, it adds nothing to a possible solution other than the option of connecting its neighbors. Therefore, such a section can be disregarded entirely if it has only a single neighbor, as it would at best worsen the solution. If the sections degree is two, i. e. it has two neighbors, it can, along with its two connected edges, be transformed into a new edge connecting its two neighbors. To ensure that the configuration path and fitness value are still calculated correctly, the connecting switches and the sections fitness value have to be assigned to that new generated edge. Sections with a degree greater than 2 cannot be collapsed, as they provide options for trees form a fork.
Another possible reduction for the section graph is to merge sections with degree one and a $SectionCapacity < 0$ with its only neighboring section. If the section has a negative capacity it needs to be reconfigured in any case, and the only vertex that could be added initially to the tree is the single neighboring vertex this section has. Sections with only a single neighbor can also be collapsed if they have one or more generation modules but contain no loads. This is because of their positive capacity and no loads being affected by their inclusion into a solution, it could only result in better generator utilization distribution.

5.3.4.3 Returning Valid Solutions Only

By implementing a simple check if the tree is in fact a valid solution option according to definition 5.1, before returning a it to the main control, the search space after solutions have been calculated can be reduced drastically. This can be implemented with a single **if** statement that validates if the capacity of the tree is ≥ 0 before the **return** statement

5.3.4.4 Stop Growing Valid Trees

One check that reduces the average runtime of the reconfiguration algorithm is to stop growing a particular tree, once it represents a valid solution. The only scenario in which this check would not also reduce the worst–case execution time is if the only valid solutions are spanning trees, i. e. all solutions contains every reachable vertex in the graph.

5.3.4.5 Global Best Solution

Further reduction of runtime can be obtained by introducing a global variable that keeps track of the best fitness value and corresponding path of all valid solutions so far. Once any tree exceeds the current optimal fitness value, it cannot bring any improvements compared to the current solution and can therefore be discarded. Moreover, this approach renders the necessity to search for the best solution out of all found ones obsolete.

5.3.4.6 Parallelization

An enormous improvement can be achieved by designing the algorithm in a way that allows parallelization. This enables comparably fast and cheap solutions to speed up the execution of code. With the general algorithm to calculate all possible trees rooted at a specific section one trees needs no knowledge of other trees, only of its current path and the overall system graph layout. Therefore, additions to each tree can be calculated on a different processor. With the simple algorithm, after all processors finished, a search for the tree with best fitness value has to be done, which is directly bound by the number of trees. In combination with the improvement of keeping track of the best global solution found so far, this search can be omitted. The new problem that would arise in its stead is to ensure communication of said solution. It has to be guaranteed that no processor identifies its found solution as the best one, while another processor is in the process of submitting its solution as best. One solution to this problem could be a queue that accepts new solutions and compares them to the current optimum, to ensure that they still bring an improvement before replacing the current solution.

5.4 Improved Graph Based Reconfiguration

This section is dedicated to the final version of the developed approach. By incorporating most of the improvements proposed in Section 5.3.4, an algorithm with better average and worst–case runtime can be designed. Collapsing sections as described in Section 5.3.4.2 cannot be included in the algorithm itself, but has to be applied to the section graph beforehand. Also the parallelization is not included in this version, for reasons of readability and simplicity. The pseudocode for the refined graph based approach is presented in Algorithm 5.3. This algorithm allows to find an optimal new network configuration after

Algorithm 5.3 Compute Reconfiguration Path

INPUT: Section Graph G, Next Sections, Capacity, Fitness, Path
OUTPUT: Solution path with the best fitness value

```
function ComputeSolution(G, NextSections, Capacity, Fitness, Path)
    for each Section in NextSections do
        update Capacity, Fitness and Path
        for each Neighbor of Section not in Path do
            add Neighbor to NeighborList
    if Capacity ≥ 0 and Fitness better than GlobalBestFitness then
        update GlobalBestFitness and SolutionPath
    else if Fitness better than GlobalBestFitness then
        for each Combination in powerset(NeighborList) do
            ComputeSolution(G, Combination, Capacity, Fitness, Path)
```

a fault occurred on SPS based on predefined fitness value criteria. The advantages presented in Section 5.3.2 still hold true after incorporating all improvements, and optimality of solution in terms of connectivity, load and source priority is ensured even if multiple sections require reconfiguration.

5.5 Conclusion

This chapter introduced a series of definitions regarding solution validity and section attributes. A section graph representation of the system graph based on online data was defined. Then an algorithm with improved runtime based on system layout was proposed for the calculation of FM sections. With this information a graph–based objective for the reconfiguration was formulated, that incorporated a combination of secondary goals for the reconfiguration process. After proposing a general algorithm that solves the stated objective, multiple improvements to the algorithm itself and the overall process were presented that reduce both average and worst–case runtime. Finally, the algorithm incorporating the proposed enhancements was presented. This algorithm provides a good basis for reconfiguration of a realistic shipboard power system. The next chapter will describe a testbed using a model of the notional zonal model proposed in Section 2.1.4 and afterwards the verification of an implementation of the new algorithm is discussed.

6 Results

In this chapter the specifics of an implementation to test the proposed algorithm from Chapter 5.3 are discussed. First the SPS model, hardware and control logic of the testbed is described and then results obtained by the implementation of the graph–based approach are presented.

6.1 Algorithm Implementation

This section first describes the specific implementation of the proposed approach used to obtain test data. The second part discusses improvements to the pre–calculation that were necessary to enable testing of larger systems.

6.1.1 Test Implementation

The proposed algorithm that solves the reconfiguration problem was implemented in Python™according to the pseudo–code shown in Algorithm 5.3. This was integrated with the other components shown in Figure 2.16. To imitate the real environment of a SPS, fault management as described in [2] was implemented alongside the reconfiguration. This includes a calculation of fault isolation sections and fault detection. After the fault is detected, located and isolated, a simple graph traversal is executed to identify islanded part in the system. The result of this traversal is then used as a basis for a simple priority–based load–shedding algorithm. For testing the implementation of the developed algorithm, a simple load–shedding is sufficient. This is sufficient for testing the reconfiguration algorithm, as it has no specific impact on the reconfiguration process. Afterwards the generation of a section graph representing the current state of the system is started.
The reconfiguration takes into account the state of disconnect switches at that time and disabled sections that are not available, accessible or relevant due to faults. The resulting graph is then checked for reductions and simplification as described in Section 5.3.4.
The algorithm implementation includes the presented improvements to keep track of the overall best solution so far and stopping the calculation of trees that already exceed this limit. Not implemented are the proposed improvements through parallelization, for reasons of simplicity and reduced implementation effort. So, unlike the final system, the different control parts of fault management were not separated on different machines, but run in sequence on a single controller using a single processor. The serialization worsens timing measures for the overall process of FM, but the individual part of the controls can still be evaluated for runtime independently. For deployment on ships a more sophisticated algorithm might be desirable. The timing sequence of external control for fault management to test the reconfiguration algorithm is shown in Figure 6.1

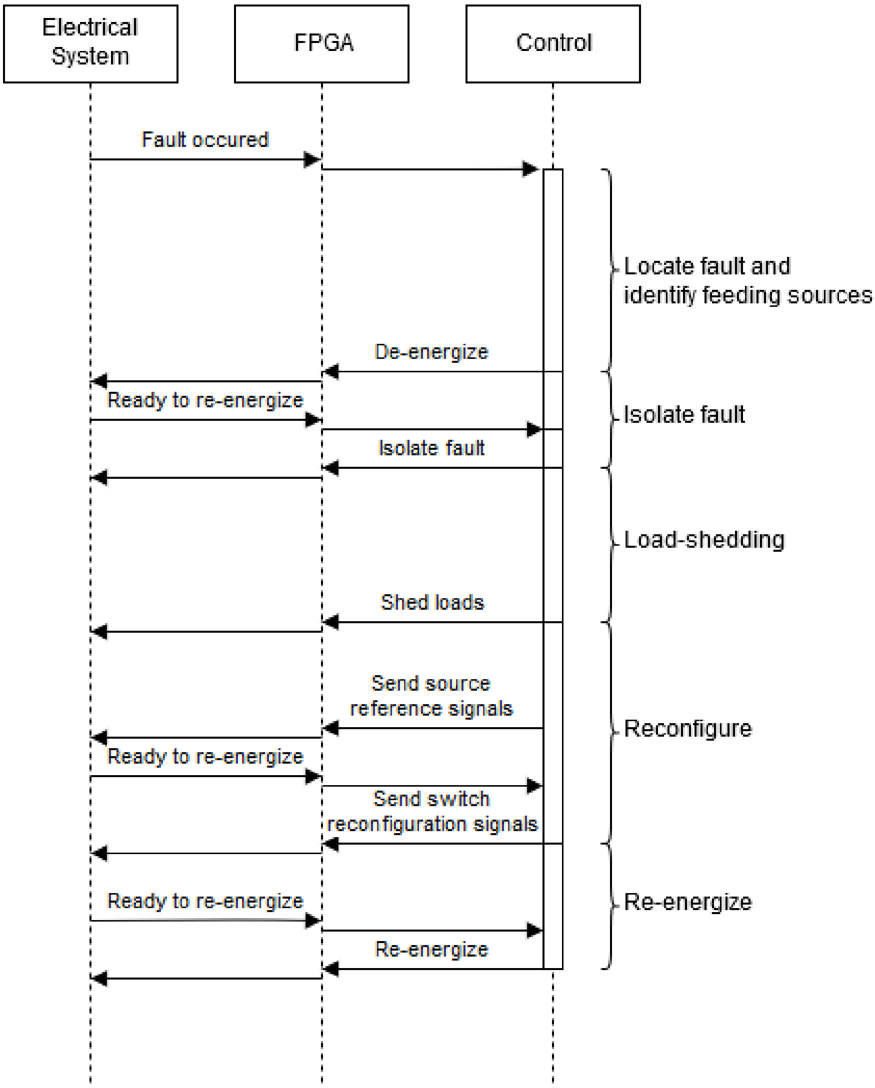

Figure 6.1: Layout of the different pieces of the testbed

6.1.2 Improvements to Pre–Calculation

The used fault detection, localization and isolation control was described in [2]. This approach was only applied to a small system, the corresponding system graph is shown in Figure 6.2. Due to the small system size the applied approach was not checked for scalability. Therefore the approach to generate fault detection sections as stated in Definition 2.4 was done in a top–down manner. At first all connected subgraphs of the system are generated. Then all sections that do not allow current measurement for the protection scheme are discarded. Finally, all of the remaining sections are compared against each other, to determine the minimal sections among them. This approach is clearly intractable, as the number of connected subgraphs is bound by $\mathcal{O}(2^{|V|})$. This was acceptable for a small system, but takes significantly longer as soon as the system size increases.
The system in Figure 6.2 is sufficient to prove fault detection, localization and isolation, but is not complex enough to realistically imitate the final shipboard power system size, especially in regards of the reconfiguration problem. To enable validation of reconfigura-

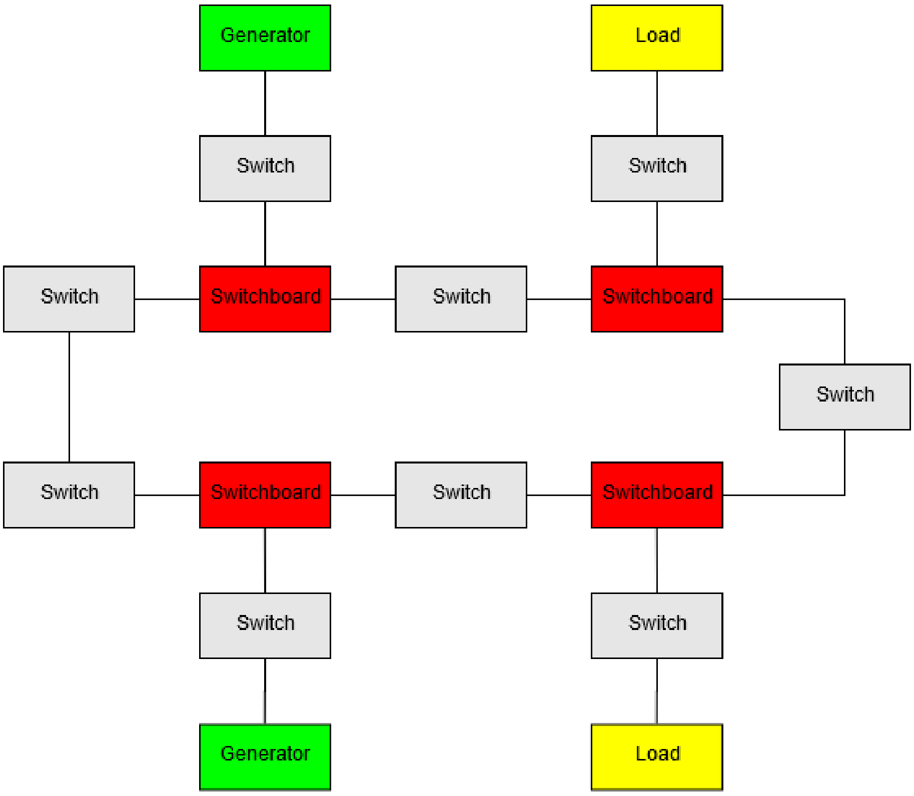

Figure 6.2: System graph of the model used in [2]

tion approaches a larger model has to be used. When changing from the initial system to a two–zone model as a first step, significant runtime increase for the pre–calculation can be observed. For the simple system the pre–calculations took 150 ms, for the two–zone system a runtime of approximately 4 hours was measured. The reason for this was the increased size and complexity of the system.

By developing Algorithm 5.1, utilizing it to identify all minimal sections and afterwards discarding all that do not meet the criteria for fault detection sections, enormous improvements to the runtime of pre–calculations were possible. This was especially vital for the even larger four or six–zone models.

Table 6.1 shows a comparison of execution times for calculating the complete set of fault detection section for the different system sizes and the chosen approach. All of these

Table 6.1: Pre–calculation runtime comparison

Model	Runtime with previous approach	Runtime with developed approach
System shown in Figure 6.2	150 ms	0.3 ms
Two–zone model	4 hours	0.8 ms
Four–zone model	>100 years	2 ms

runtime values were obtained by running the pre–calculation on a Dell™OptiPlex™990 desktop computer with a Intel® Core™i5–2400CPU@3.10 GHz processor. With the estimated runtime of more than 100 years, a test of reconfiguration in combination with fault management would not be possible, and neither would be employment on a ship with the size of a four or even six–zone model. This improvement was vital to testing and implementing a reconfiguration approach within a realistic control architecture on a realistically sized model.

6.2 Testbed

In this section the specifications of the testbed are described in further detail. First the used hardware and the setup of a test environment is presented. Then the simulation model is presented with specifics regarding the complexity of the chosen model.

6.2.1 Used Hardware

As simulation hardware a digital real–time simulator from Real–Time Digital Simulator (RTDS™) Technologies is used. It has the capability to calculate the entire electrical system to a certain detail at the same pace as real–world time progresses. This allows to test and verify developed controls on models with similar effects and latencies as when applying the controls to a real power system. The time–step for tests in this project was set to 50 μs.
The fault management controls are deployed to a Dell™OptiPlex™990 desktop computer with a Intel® Core™i5–2400CPU@3.10 GHz processor that runs Ubuntu 18.04.1 64–bit as operating system. The desktop computer is connected to a Xilinx® ML507 board, which has fiber protocol capability of up to 2 Gbps. The data exchange between the FPGA and the desktop computer is realized via PCI Express. This setup allows for fast deployment and easy implementation of control algorithms, as they can be written in higher programming languages and do not have to be implemented on a programmable logic controller. The RTDS™ and Xilinx® FPGA are connected via optical fiber. An overview of the communication layout is shown in Figure 6.3.

6.2.2 Shipboard Power System Model

In order to test the proposed algorithm, a test case was set up with the envisioned notional zonal shipboard power system model. To reduce the occupancy of simulation hardware and implementation works, a slightly simplified version of the four–zone model originally proposed by the ESRDC in [24] was used for testing. A system overview of this model is shown in Figure 6.4. The depicted model contains Integrated Power Node Centers (IPNC's), but their internal cross connections are not part of the 12 kV MVDC system, and

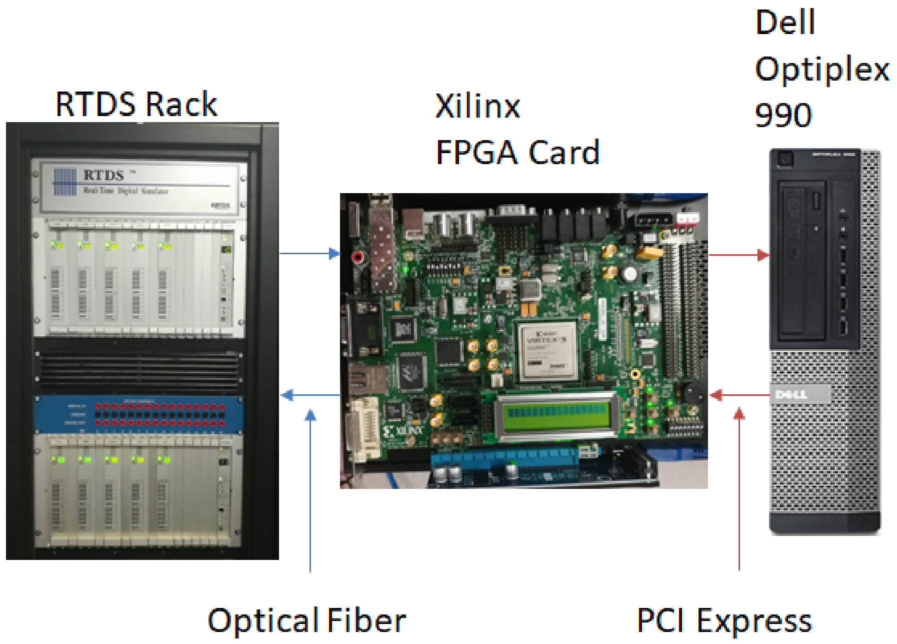

Figure 6.3: Layout of the different pieces of the testbed

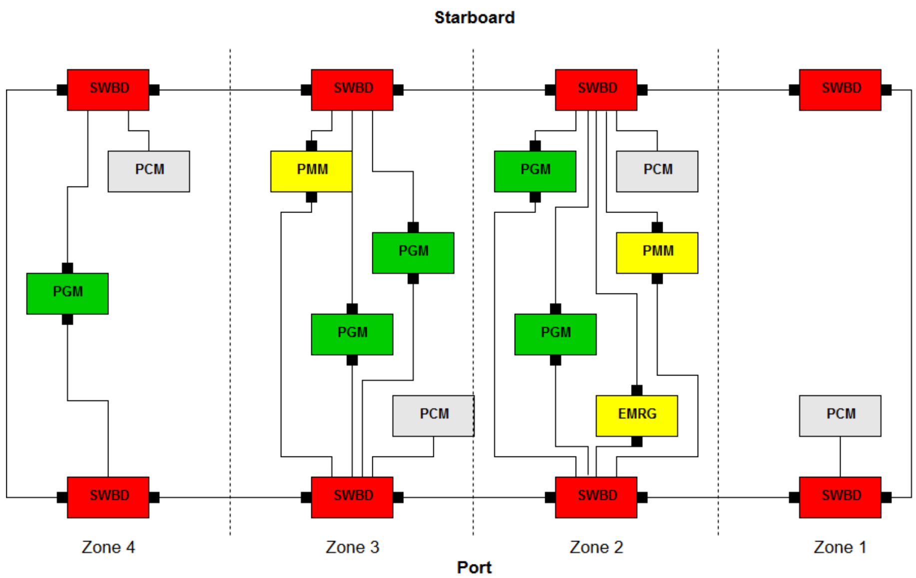

Figure 6.4: Electrical system of the simplified 12 kV ESRDC four–zone model

therefore have to be considered separately. Moreover, the amount of disconnect switches in the switchboards was reduced, as they add no additional reconfiguration options. The resulting simplified model provides all connectivity and equipment necessary for validating the proposed algorithm, while reducing occupancy and engineering effort to a minimum.
With the determined model a tighter bound on the number of trees can be calculated by examining the specified system layout. With the reduced number of switches in the system, the size of the system graph can be determined.
Based on Equation 5.4, the model shown in Figure 6.4 would have $1.065 * 10^9$ trees. By

generating the according system graph and calculating the maximum number of trees for every node, a much tighter bound of $1.58 * 10^7$ trees can be achieved. This is a drastic improvement compared to most other solutions as this bound already contains connectivity and validity checks of the calculated solution, which would worsen the complexity of any approach not inherently considering these aspects by $\mathcal{O}(|V|+|E|)$. The number of switch state combinations for example would be $1.07 * 10^{10}$ and the connectivity check would add another $5 * 10^2$ actions.

This model provides a good estimate for the actual size and complexity of a MVDC shipboard power system that might be deployed on future all–electric ships. The PGMs's and mission loads are designed in such a way, that their power rating is split between two parts of equal size, that are connected through a switch.

6.3 Test Results

This section presents the results of testing the developed algorithm with the testbed that was described in the previous section. First a couple of example scenarios are described. Then the individual results of each of them are presented.

6.3.1 Scenarios

For the test scenarios used in this thesis a model with initial switch states is defined. Then example ratings and fitness weights and priorities are chosen depending on the scenario. The initial switch states of the used simulation model before a fault is applied are shown in Figure 6.5. The initial configuration assumes that both starboard and port buses are

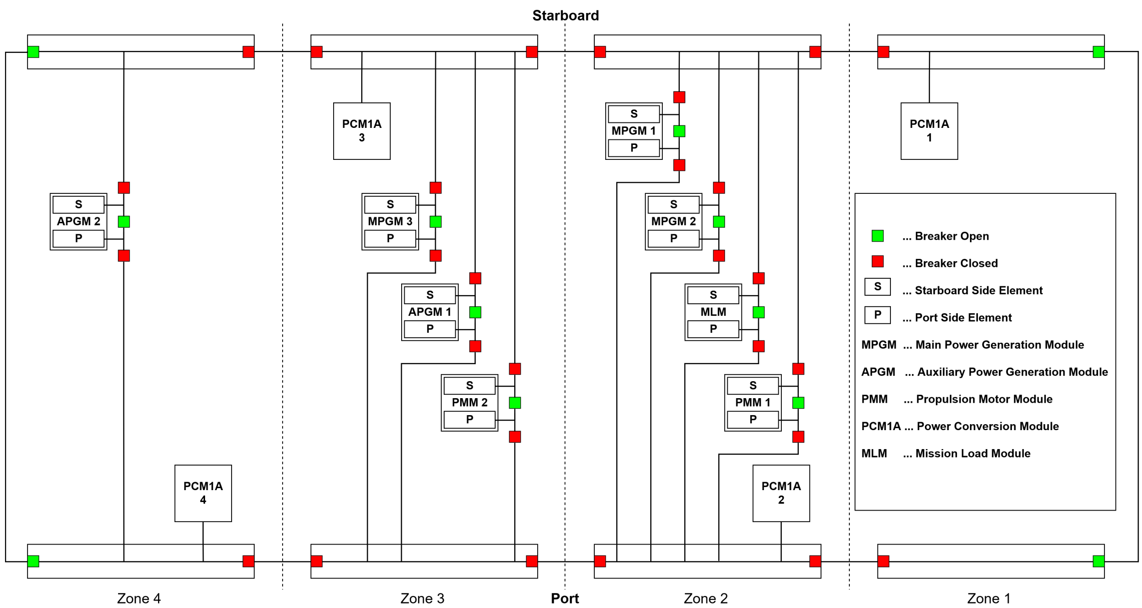

Figure 6.5: Initial configuration of the RTDS™ simulation model

functional and connected, and the respective parts of each module are connected to the

designated bus. All internal cross–connection switches of modules and the connecting cable sections at bow and stern are open, which can also be seen in Figure 6.5.
For power ratings some example values are used. The Main Power Generation Modules (MPGM) are rated with 15 MW on starboard and port side each. The Auxiliary Power Generation Modules (APGM) are set to a generation capacity of 2.5 MW on either bus side. Both Propulsion Motor Modules (PMM) are set to a reference value of 15 MW on each side. The Mission Load Module is rated with 0.4 MW total, distributed on the port and starboard modules. Finally, the Power Conversion Modules (PCM1A) are set to 4 MW each. These ratings result in a constellation in which the system has generation capacities of 100 MW and power demand of 76.4 MW. Therefore, in this configuration the system is not operating close to power limits. This allows to show a variety of reconfiguration scenarios depending on the location of an introduced fault.
As a first scenario, a fault is introduced in the switchboard in zone 4, closest to the bow on starboard side. In the second scenario it is assumed that the switchboard on starboard side in zone 2 is faulted.
Both scenarios assume faults in switchboards. The reason for this is, that a faulty switchboard has the most impact on the current system configuration with a single fault. Lost loads or sources would lead to load shedding, but hardly ever require reconfiguration. Moreover, faults in cable sections might either not need reconfiguration at all, or just require one side to connect to another section with enough positive capacity, whereas a faulted switchboard might render multiple sections with a negative section capacity.

6.3.2 Scenario A

For the first scenario a fault is introduced in the starboard side switchboard in zone 4, detaching APGM 2 from the rest of the starboard bus. The resulting configuration after the fault was isolated and the reconfiguration algorithm terminated is shown in Figure 6.6. The fault in the switchboard is isolated as expected, apart from that no switch states are changed. After the APGM 2 is lost on the starboard bus after the fault isolation, the generation capacity is 47.5 MW and the demand is 38.2 MW, whereas the port side bus still operates with 50 MW generation and 38.2 MW power demand. This system configuration is therefore valid and provides sufficient power supply to all loads.
The corresponding starboard and reference voltage signals for this scenario are shown in Figure 6.7. The port voltage values are not affected during this scenario at all. Moreover, no remarks regarding fitness calculation are made, as to do not effect the reconfiguration process in this scenario.

6.3.3 Scenario B

In the second scenario a fault is introduced in the switchboard on starboard side in zone 2. After the starboard side modules of both loads in zone 2 lose their feeding lines, those have to be reconfigured. Also the PCM1A in zone 1 is isolated from the rest of the system.

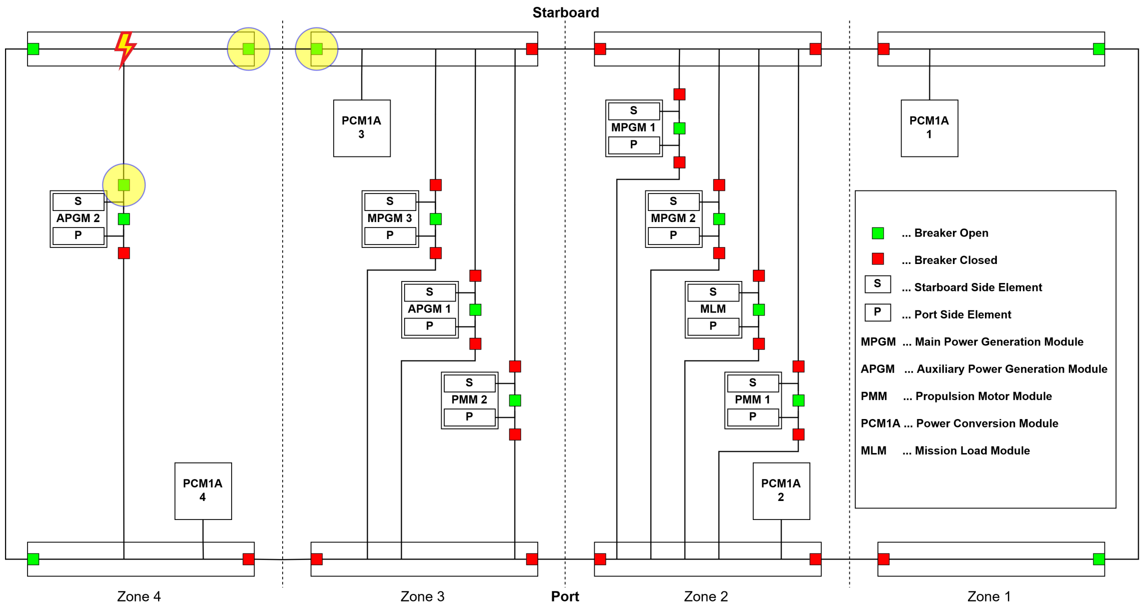

Figure 6.6: Resulting configuration in Scenario A

For fitness calculation according to Equation 5.3 weights and priorities are defined. The relevant values are shown in Table 6.2. The priorities for all sources are equal for starboard

Table 6.2: Values for fitness calculation in scenario B

Variable	Value
w_j	0.5
w_k	0.3
w_l	0.2
p_{MPGM1}	0.05
p_{MPGM2}	0.1
p_{MPGM3}	0.1
p_{APGM1}	0.1
p_{APGM2}	0.15

and port side modules, and load priorities are equally distributed. The resulting switch states after fault handling are depicted in Figure 6.8, and the corresponding signals for the reconfiguration of scenario B is shown in Figure 6.9. In this new configuration, the faulted switchboard on starboard side in zone 2 is isolated and connecting switches for the two loads and MPGM 1 in zone 2, as well as the bus connecting cable section in zone 1

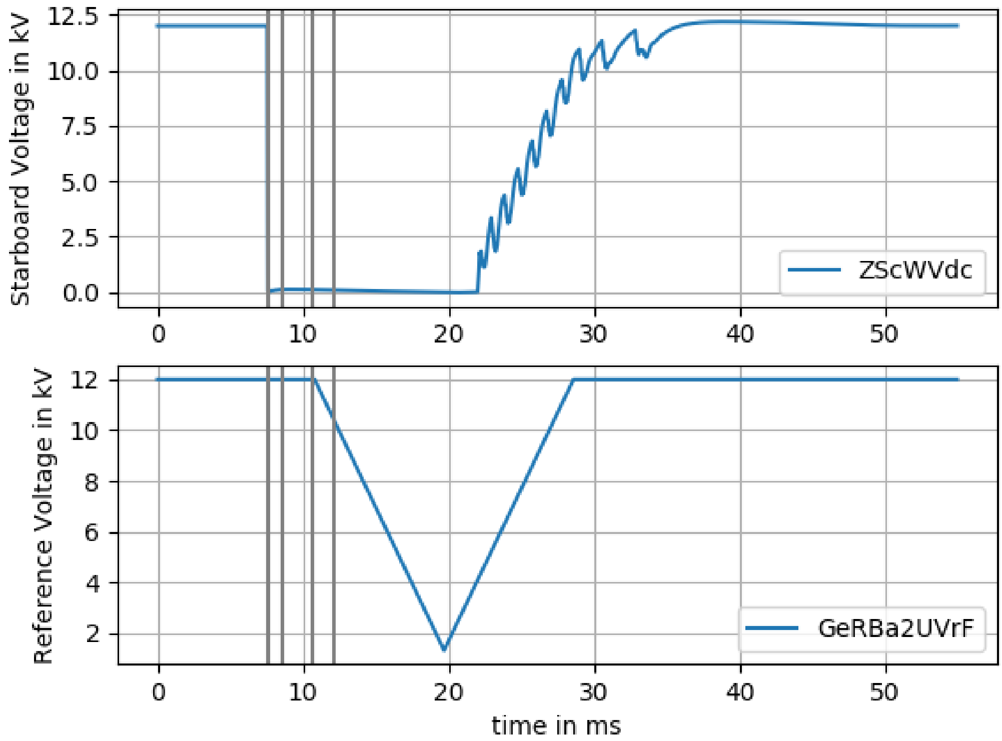

Figure 6.7: Voltage signals for Scenario A

are closed. The fitness value of this new section can be calculated to:

$$
\begin{aligned}
f_{Port} &= 1 - [0.5 * \frac{13}{20} + 0.3 * \frac{8}{10} + 0.2 * (0.05 * 2 + 0.1 + 0.1 + 0.1 + 0.15)] \\
&= 1 - [0.325 + 0.24 + 0.11] \\
&\Rightarrow f_{Port} = 0.325
\end{aligned}
\tag{6.1}
$$

The generation capacity on the port side bus increases to 65 MW, while the demand increases by 19.2 MW from 38.2 MW to 57.4 MW. The remaining starboard bus generation capacity decreases to 20 MW while the power demand falls from 38.2 MW to 19 MW. Therefore, both buses have positive section capacities and the new configuration is valid and provides sufficient power supply to all loads.
The corresponding voltage signals for this scenario are shown in Figure 6.7.

6.4 Conclusion

In this chapter the algorithm implementation, testbed used for the verification and testing of the algorithm that was developed and presented in Chapter 5.3 and results of chosen scenarios were explained. First the test implementation of the algorithm itself was discussed, followed by improvements made to the pre–calculation process. Then the model

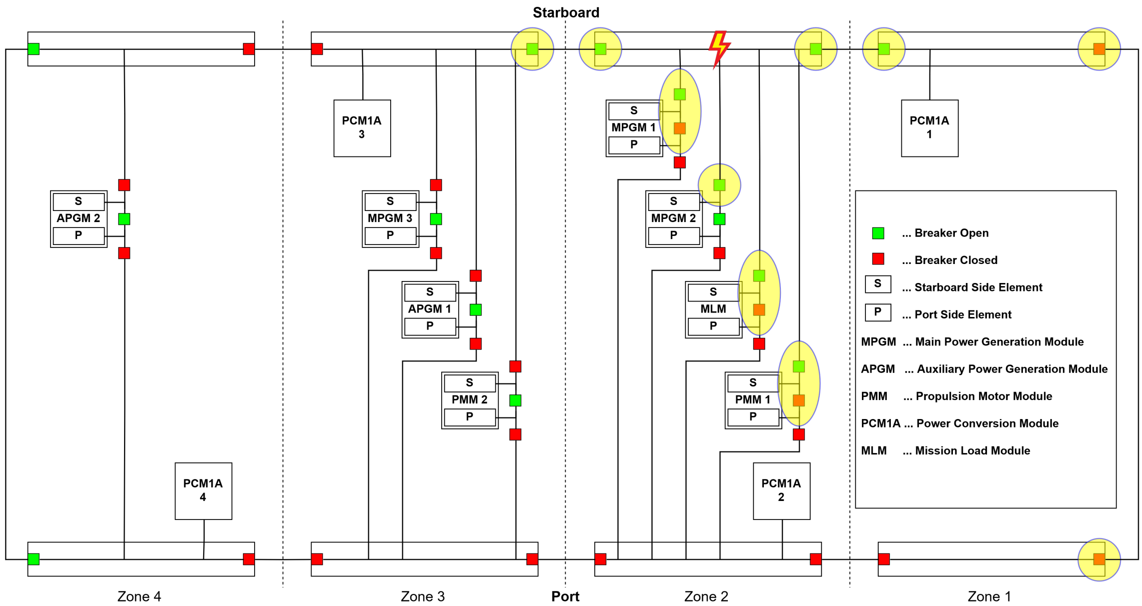

Figure 6.8: Resulting configuration in Scenario B

implementation of the shipboard power system and the hardware used to run the model and control algorithm were described. Finally, the results of testing the algorithm were presented. The two scenarios shown assumed faulted switchboards, for both scenarios the resulting switch configuration and the corresponding voltage signals were shown.
The next chapter will discuss these findings and compare them with other options for the reconfiguration, followed by an outlook of future work that has to be done on the topic of reconfiguration of shipboard power systems using zonal architecture.

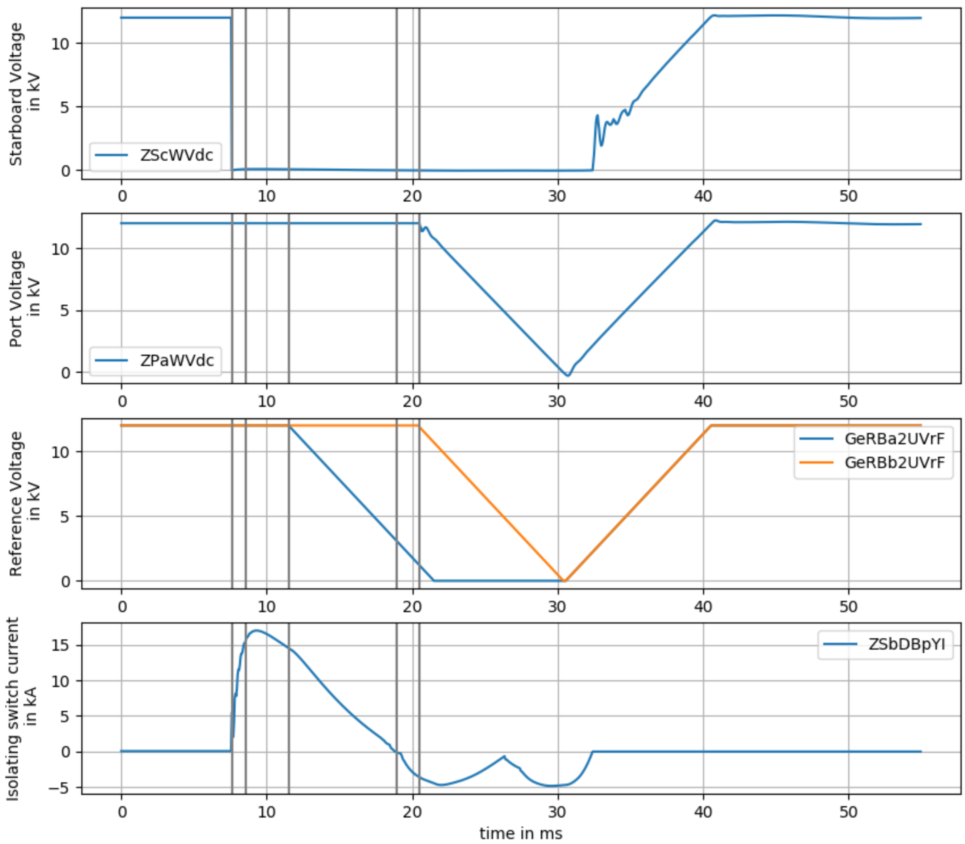

Figure 6.9: Voltage signals for Scenario B

7 Discussion

The main goal of this thesis was to develop an algorithm for the reconfiguration of a notional zonal shipboard power system and show that it enables to efficiently optimize underlying electrical system attributes like connectivity and mission–based fitness value. The previous chapter presented the testbed used to verify if the approach is useful. This chapter will discuss findings of the results and present a comparison with other options for reconfiguration that might be used. Finally, topics that require more attention and work in the future are discussed.

7.1 Limitations

The algorithm works as expected and provides a promising basis for reconfiguration of a zonal shipboard power systems. It allows flexible calculation of different network topologies and application dependent fitness values. The implemented algorithm operates under a number of certain limitations though. The three main restrictions are:

- Currently no inherent functionality for load–shedding is implemented;
- line limits or current limitations of switches are not considered; and
- the current implementation only identifies switches that have to be closed, neglecting additional options that could be explored by opening switches.

These limitations reduced the development and implementation effort for a reasonable trade–off in optimality. They still allowed verification of the proposed approach and have to be addressed in future works dealing with the reconfiguration of shipboard power systems.

7.2 Comparison of Results

The main part of this section presents the data obtained from testing the proposed approach on a simulated SPS. Different fault locations are analyzed. Afterwards a comparison against other elementary approaches is given.

7.2.1 Automated Tree–based Reconfiguration

The first observation that can be seen from the chosen test–cases is that the algorithm always finds a valid new configuration of the system after a fault was introduced. This holds true for different fault locations. The optimality of each solution regarding fitness value can also be determined by further analysis.

7.2.1.1 Scenario A

For the first scenario of reconfiguration, a fault was introduced in the switchboard on starboard side in zone 4. Figure 6.6 shows that, apart from the fault isolation, no reconfiguration happened. Due to the amount of excess power generation capacity, no reconfiguration is necessary. Therefore, the resulting system configuration only changes through the isolation of the fault in the affected switchboard.
The voltage signals shown in Figure 6.7 allow further analysis of this behavior. The first vertical line at approximately 7.5 ms marks when the fault is introduced. Approximately 1 ms later, where the second vertical line is drawn, the fault is detected by the fault management control that handles fault detection, localization and isolation. After another 2 ms, at approximately 10.5 ms the fault location has been determined and feeding sources have been identified. At this point, the commands to ramp feeding source modules down are sent. One millisecond later, the algorithm has already finished with the conclusion that no reconfiguration is necessary. After the ramp down of all feeding sources is finished, the switches bordering the fault are opened to isolate it. Finally, the commands to re–energize the system are sent, after which the system continues operation.
Since no reconfiguration was necessary, no assumptions for fitness calculations had to be made, as they would have no effect on the fault handling sequence. This scenario showed that reconfiguration might not always be necessary, and the developed reconfiguration algorithm is able to determine if changes are required. The entire sequence, including ramp–down and ramp–up of voltage sources took 30 ms to finish. Here it has to be noted that the control sequence terminated after 4 ms with the conclusion that no reconfiguration is required.

7.2.1.2 Scenario B

For the second scenario of reconfiguration, a fault was introduced in the switchboard on starboard side in zone 2. Figure 6.8 shows the resulting switch states after successful reconfiguration.
As already presented in the previous chapter, the new configuration is a valid solution. Optimality in terms of the chosen fitness values can be verified by comparing the calculated fitness value against other reconfiguration options. One other valid solution would be to connect MPGM 2 instead of MPGM 1 to the port bus. Due to the lower priority of MPGM 2, this would result in a fitness value of $f_{Port} = 0.315$. Connecting both MPGM 1 and MPGM 2 would also solve the reconfiguration problem, but with a fitness value of $f_{Port} = 0.28$ this solution is significantly worse than the one chosen by the algorithm.
The voltage and current signals shown in Figure 6.9 provide a better overview of the individual steps taken by the external control to handle the fault. The first vertical line at 7.6 ms marks when the fault is introduced. 1 ms later, where the second vertical line is drawn, the fault is detected by the fault management control that handles fault detection, localization and isolation, and the controller starts the isolation sequence. After 3.8 ms, at approximately 11.5 ms the location of the fault has been determined and feeding sources

have been identified. Commands to ramp feeding source modules down are sent immediately. Then the system waits until the ramp–down has almost finished and the current through the isolating switches reaches zero.
This happens at 19 ms where the current has a zero–crossing, which is not correct as the has not reached zero yet. The reason for this is an error in the simulation model, which should check both current and voltage for a switch before changing its state. The fix for this error was pending at the time this work was written. This error does not influence the execution time of fault management controls, but rather the total time necessary for the fault handling scenario. Therefore, the reconfiguration runtime measurement is still correct and can be used to verify the developed approach.
After the switches are closed at 19 ms, the calculation of reconfiguration solution is started. 1.6 ms later, optimal solutions for all negative sections have been found, and the control issues commands to ramp down relevant source modules to allow switches to close. At 31 ms all sources of the new section reach equal voltages of zero V and switches are closed. Finally, the commands to re–energize the system are sent, after which the system continues operation with the new configuration.
This scenario showed that the algorithm is capable of finding a solution for multiple sections that require reconfiguration. It also showed that the chosen solution is optimal with regards to the defined fitness value requirements. The set goal of staying within 8 ms for fault handling was not met for the entire process, as the ramp–down and ramp–up of sources takes significantly longer than that timespan. The combined runtime of reconfiguration controls adds up to approximately 6.4 ms, which would be within the set goal. As the simulation model is not completely finished, the final results are still pending.

7.2.2 Manual Operator

One reconfiguration option that the developed automated process could be compared to is to have a manual operator decide a new configuration once a fault disables parts of the shipboard power system. This individual would supposedly have detailed knowledge about priority of loads, system layout and other effects that might impact reconfiguration. But even with this information and years of experience the optimality of a new configuration cannot be guaranteed, as even with significant amounts of time a human could not go through all possible configurations of a system as vast as an entire SPS. Additionally, a human operator would introduce another point of global failure, as human error is always possible as soon as humans are involved.
Another main disadvantage of a manual operator is the amount of time needed to find and implement a valid solution. With automated reconfiguration a solution can be found and implemented faster than the average reaction time of humans. With reaction time, time to analyze the new state, identify possible solutions, figure out the best one and finally identify and implement the necessary steps to realize a new configuration, the entire process would most likely take multiple seconds. Even simple changes that a basic solution with several restrictions could solve would take significant amounts of time. During this time span one

or more vital loads might be left without sufficient power supply. Even tough vital loads will be provided with dedicated energy storage, the ride–trough capacity of these storages might either be insufficient for this long a time span, or significantly more expensive and heavier than necessary compared to other reconfiguration approaches.
Overall, it can be easily seen that an automated approach is preferable compared to manual operator, both in regards of time and optimality of a new solution.

7.2.3 Simple Solution

Another option for reconfiguration of a SPS is to implement a simple search algorithm that operates under a certain number of restrictions. A simple algorithm would have a significantly shorter runtime, as the restrictions reduce the complexity. One example for such an algorithm would be to simply search for any source that has enough spare capacity for every under–supplied load in the system. This way, the reconfiguration logic would only have to conduct a simple graph search of the order $\mathcal{O}(|V|+|E|)$. This obviously decreases worst–case runtime significantly, compared to both the manual operator and the developed automated, tree–based approach. A new system state could be calculated in $\leq$1 ms. The critical downside of such approaches is, that the restrictions are too limiting if they are kept really basic. A simple search might not be able to find an optimal or even close to optimal solution, and in some cases might even fail to find a valid solution to the reconfiguration problem at all. Therefore, the approach cannot be designed too simply if a satisfying system state has to be found in all cases.

7.2.4 Exhaustive Search

The final alternative that will be considered in this thesis would be to conduct an exhaustive search of all possible system states. No restrictions have to be applied and all desired aspects can be included and considered. Therefore, this approach always yields the optimal solution that can be found for the system. The advantages of this compared to other approaches are obvious, but so are the disadvantages. An exhaustive search of a system this size would be of order $\mathcal{O}(2^{|n|})$, where n is the number of switches in the system. This becomes intractable with systems as large as the four or six–zone models proposed by the ESRDC. In those systems there are 70 and 75 switches respectively, resulting in $1.18 * 10^{21}$ and $3.78 * 10^{22}$ combinations respectively. Even if it would be possible to determine a fitness value and verify connectivity within a nanosecond per combination, the exhaustive search of the six–zone system would still take a million years. The complexity can be reduced by adding rules to eliminate irrelevant combinations, e.g. a combination that would open both feeds of an active load can never be a valid solution, or opening one end of a cable section while closing the other end cannot bring any advantages. With such rules the number of combinations can be reduced to approximately 10^{10}, which is still significant and does not include connection verification. Moreover, an exhaustive search approach would be highly parallelizable, as no communication during the search is necessary. Even with the use of rules combined with parallelization calculation of all combinations is too complex to be done efficiently. Therefore, this approach is not applicable if the set goal of fault management within 8 ms should be met, but it would rather take at least several seconds to iterate through all relevant solutions, even with drastic reductions and improvements.

7.3 Further Work

This section describes the topics that further work has to be conducted on in the future. The main subjects are the improvement of current limitations and restrictions of the developed algorithm, to further improve performance for real–time online application of reconfiguration and the interfacing of the fault management with other part of the shipboard power system controls.

7.3.1 Reduce Impact of Limitations

One important topic that needs to be explored further in order to make the developed algorithm more useful for implementation on ships is to lessen or remove the impact of current restrictions. The three main limitations were described in Section 7.1. As described in Section 5.3.3, the algorithm has no functionality that would allow load–shedding along with the calculation of an improved network layout. For this reason, the algorithm either has to be combined with a more sophisticated load–shedding algorithm or modified in such a way that a better on which loads to power and which to shed can be made. A separate execution does not slow the reconfiguration down, and significant performance improvements are possible if the load–shedding is executed on a separate processor as shown in Figure 2.16.
Another limitation that will require additional work in the future is that in the current version no line limits are considered. This restriction has to be solved, as otherwise a reconfiguration might overload lines or devices, that are not rated for higher currents. If this problem is not addressed, this might lead to decreased lifespan of elements or damages to equipment. The problem with line limits or current limits in switches is that they too might affect the number of loads that can be served and would have to be considered when load–shedding takes place. Currently load shedding is executed before the reconfiguration algorithm, hence possible reconfiguration paths are not known and can therefore not be checked if they would violate line limits.
The last restriction that the algorithm proposed in this work faces is that it only considers closing switches. This might leave the system in a state with unwanted and unnecessarily large sections. One solution to this problem would be to apply the algorithm to the section graph with only the basic sections, without merging currently connected sections. The disadvantage of this solution is that the average execution time would drastically increase, as the basic sections calculated by Algorithm 5.1 result in the largest possible section graph size, and therefore the largest possible number of trees. The other option to

reduce the negative impact would be another algorithm that checks if any switches could be opened without impacting the power supply of any loads. Ideally this algorithm would run instantly after the reconfiguration path is defined, so the power system requires no additional ramp down and ramp up sequence to be able to close and open switches. By combining both of those solutions the size of individual sections is always as minimal as possible while ensuring highest possible fitness value of the new configuration.

7.3.2 Performance Optimization

The goal of fault management on shipboard power systems is to reduce the impact of faults on the functionality of the system to a minimum. To accomplish this the execution time for the entirety of FM has to be shortened as much as possible. Hence all runtime improvements and reductions of computational complexity are helpful. The improvements presented in Section 5.3.4 are a first step towards reducing the execution time of the reconfiguration. While the collapsing and merging of sections directly effects the worst–case runtime, the introduction of a variable keeping track of the global optimum reduces the average runtime drastically.
One option that is worth looking into would be a lookup table for reconfiguration options. This would improve the execution time for the reconfiguration to $\mathcal{O}(1)$, i. e. to identify the optimal solution is no longer dependent of the system size, but of some other constant growing slower than that, which is a significant improvement. The downsides of a lookup table are the creation and even more so the storage. The calculation of a lookup table is very computationally expensive, but as the generation can be done offline and the system layout of a ship will stay fixed once determined, even computation times of weeks or months would not matter. The main problem is the amount of storage space necessary to save all potential system configurations. The number of possible initial situations before the reconfiguration is intractable, as the combinations of states of all switches alone are $n = 2^{NumberOfSwitches}$. This has to be multiplied with different fault locations and combinations, and if the power demand of loads is variable that might require an additional check that has to be executed for every entry in the lookup table. The number of combinations could be reduced by introducing equivalence classes. Some of the solutions could be grouped together using symmetry aspects of the SPS layout or by splitting the system into smaller parts that can be considered separately.

7.3.3 Integration with Overall Control Architecture

Another part that needs additional work in the future is the integration of the proposed recovery sequence with other parts of fault management and overall shipboard power system controls like power and energy management. The information that has to be exchanged has to be identified and defined in thorough detail. The localization and isolation of a fault already needs information about the system and devices, like if a device can isolate or if they can provide or disable current measurements, in order to enable transformation into the system graph. Online current measurements are necessary for the detection of faults. A piece of information that is needed both for isolating a fault and system reconfiguration is the current state of each switch in the system. For the transformation of the system graph to the section graph even more online information is necessary, like if certain loads or generators are operational, their current power level reference and the priority of devices based on the current mission the ship is set to. All of this information directly effects the section graph layout or the fitness value of specific reconfiguration paths. Table 7.1 shows a detailed description of what pieces of information the reconfiguration algorithm as presented in this thesis needs at which instances. This information comes from

Table 7.1: Information that is necessary for reconfiguration with respective points in time

Necessary Information	Point in Time
Switch states	Before the creation of the system graph
Which loads and sources will be operational	After the system graph was generated, before simplifying it
Operating power level of loads and sources	After the system graph was simplified, before the section capacities get calculated
Priorities of loads and sources according to current mission profile	After the system graph was simplified, before the section capacities get calculated

different pieces of the power system control. The switch states have to be reported by one or more agents monitoring the system. The information about operational loads will be determined by a load–shedding control, which in turn needs information about which loads, and which generators were online before the fault and which parts are connected or disconnected from the rest of the system. The operating power levels will most likely be determined by the power and energy management control, while the priorities of individual loads or sources could either also come from that control part or simply be based on the current mission of the ship.

Another important aspect of the integration of reconfiguration within the overall control architecture is the information that is available to other parts of controls and the earliest point in time at which this information can be provided. This data is presented in Table 7.2. This table just gives an overview of available information, which parts of the overall control architecture needs what information is a topic that has not been fully explored yet. Most of this information will be useful for power and energy management to be able to take a new configuration into account as early as possible, but other controls might benefit from more available information too. Hence, defining this interface is an important topic that future research has to be conducted on.

Table 7.2: Information that is available after the reconfiguration with respective earliest points in time

<table>
<tr><th>Available Information</th><th>Point in Time</th></tr>
<tr><td>Which parts are disabled by the fault</td><td>After the fault has been detected, before the isolating algorithm</td></tr>
<tr><td>Which parts are affected by the isolation</td><td>After the isolating algorithm ran before the load shedding algorithm</td></tr>
<tr><td>Which loads will be shed</td><td>After the load shedding algorithm, before the reconfiguration algorithm runs</td></tr>
<tr><td>Which loads and sources are affected by the reconfiguration</td><td rowspan="3">After the reconfiguration algorithm ran, before the solution is applied</td></tr>
<tr><td>Which loads and which sources are interconnected</td></tr>
<tr><td>Switch states for the new configuration</td></tr>
</table>

7.3.4 Signal Sequence

Future work also has to deal with possible dependencies that have to be considered when sending reconfiguration signals. In breaker–less DC systems fast disconnect switches are envisioned to be used instead of expensive DC–breakers. however, these switches can only operate under certain conditions. More specifically, the current flow has to be negligible and voltage levels on both sides have to match within a certain margin. This requires more intelligence when implementing a new configuration. Assume one side of a switch had to be powered down to isolate a section, and the other side was brought to a certain voltage to close another switch. Then the switch between those two sections cannot connect them before at least of one them adapt their voltages again, which might slow down the implementation process significantly. The resulting delay depends mostly on the required time for ramping converters up or down. To reduce the impact of those dependencies between switches, further work has to be conducted on identifying optimal converter controls and signal sequence, once a reconfiguration has been identified and compared to the current system state.

7.4 Conclusion

In this chapter the results of testing the developed, automated, tree–based approach to reconfiguration of a shipboard power system in a defined testbed were presented. It was shown that the designed algorithm found valid solutions for different fault locations. First tests showed that the set goal of 8 ms was not met for the entire process, but rather only for the control sequence. Final results are still pending until remaining errors in the simulation are fixed. A comparison with alternative options for reconfiguration showed

that some might fail to find a valid solution, while others would take significantly longer to bring the system back to an operational state.
In the second part future work that has to be conducted on the topic of SPS reconfiguration was proposed. The first topic that was discussed is to improve the current constraints and restrictions of the algorithm. Three main limitations were identified and debated. Firstly, no load–shedding is included in the algorithm itself, secondly, line limits are not taken into account when calculating a reconfiguration path and finally, the current implementation does not account for the opening of disconnect switches, which might provide even better reconfiguration choices. The next section examined the options for optimizing the performance of the proposed algorithm. Some improvements were presented in this thesis and implemented in the algorithm that was tested. One option for further improvement would be a lookup table, but some obstacles tied to this approach still have to be solved. The following section explained possibilities of integrating the presented reconfiguration algorithm in the control architecture of the ship. The information that is necessary as well as the data that can be provided were explained with their respective points in time. Finally, dependencies between switches and the resulting necessity to find an optimal sequence of signals was discussed.

8 Conclusion

This work dealt with the reconfiguration of a zonal shipboard power system. After a fault occurred and was isolated in the electrical system, the SPS might be left in an undesirable state where vital loads are without sufficient power supply. If this is the case, automated reconfiguration should identify and implement a solution that provides power to a maximum number of vital loads.
Some of the most important requirements are timing limitations, validity of solution and scalability for realistic SPS sizes. An analysis of related approaches showed, that heuristics, graph–based and expert system solutions have been used. Closer study showed that a graph–based approach fulfilled these requirements and allowed for efficient optimization of electrical system attributes. To find a close to optimal solution, the system was transformed into a section graph representation. With this representation storing system data and calculating a solution is possible with known approaches for graphs. Also, it allowed the development of a tree–based reconfiguration algorithm that is able to find the best possible solution according to defined criteria with comparably few restrictions. This reduces the required number of human inputs and therefore the likelihood of implementation errors while improving real–time capabilities.
To enable testing on realistically large systems, improvements to the pre–calculation process were necessary. The algorithm was deployed in a testbed which included a simulated SPS and an external controller for fault handling and reconfiguration. With the described testbed the approach was verified against a number of different fault locations.
Test results showed that the fault management process including reconfiguration was completed within 35 ms, and therefore failed to meet the goal of 8 milliseconds. Because of sequential instead of parallel execution of fault handling reconfiguration calculation on the external controller, these times have to be separated for exact measurements, leaving the runtime of reconfiguration itself to 1.6 milliseconds for the larger scenario and 6.4 ms for the entire fault handling control sequence. Once minor issues with the simulation model are dealt with, final results can be obtained. To further improve performance, a parallelization of FM, load–shedding and reconfiguration would be possible. Moreover, the calculation of individual paths can be parallelized too. Hence it was proven that a tree–based reconfiguration algorithm is indeed an approach that allows efficiently optimizing underlying electrical system attributes for systems as complex as envisioned zonal shipboard power systems.

9 List of Figures

10 Bibliography

[1] N. Doerry and K. McCoy. Next generation integrated power system: NGIPS technology development roadmap. Technical Report Ser 05D / 349, NAVSEA, 11 2007.

[2] C. Diendorfer. Graph traversal-based fault management for medium voltage DC shipboard power systems. Master's thesis, University of Applied Sciences Upper Austria, 07 2016.

[3] IEEE guide for protecting power transformers. *IEEE Std C37.91:*, 2008.

[4] S. Miao, P. Liu, and X. Lin. An adaptive operating characteristic to improve the operation stability of percentage differential protection. *IEEE Transactions on Power Delivery*, 25(3):1410–1417, 07 2010.

[5] M. J. Thompson. Percentage restrained differential, percentage of what? In *2011 64th Annual Conference for Protective Relay Engineers*, pages 278–289, 04 2011.

[6] R. Jayabalan and B. Fahimi. Naval shipboard power system. In *2005 IEEE Vehicle Power and Propulsion Conference*, 09 2005.

[7] N. Kumar, A. K. Srivastava, and N. N. Schulz. Shipboard power system restoration using binary particle swarm optimization. In *2007 39th North American Power Symposium*, pages 164–169, 09 2007.

[8] F. Shariatzadeh, N. Kumar, and A. K. Srivastava. Optimal control algorithms for reconfiguration of shipboard microgrid distribution system using intelligent techniques. *IEEE Transactions on Industry Applications*, 53(1):474–482, 01 2017.

[9] N. Doerry and J. V. Amy. The road to MVDC. In *ASNE Intelligent Ships Symposium 2015*, Philadelphia, PA, 05 2015.

[10] IEEE standard for power electronics open system interfaces in zonal electrical distribution systems rated above 100 kW. *IEEE Std. 1826:*, 2012.

[11] N. Doerry and J. V. Amy. MVDC shipboard power system considerations for electromagnetic railguns. Technical report, NAVSEA, 09 2015.

[12] X. Yang and Y. Zhang. Intelligent real-time fault restoration of the large shipboard power system based on genetic algorithm. *International Journal of Information Technology*, 11(12):129–135, 2005.

[13] P. Mitra and G. K. Venayagamoorthy. Implementation of an intelligent reconfiguration algorithm for an electric ship power system. In *2009 IEEE Industry Applications Society Annual Meeting*, 10 2009.

[14] M. M. Aman, G. B. Jasmon, A. H. A. Bakar, and H. Mokhlis. Optimum network reconfiguration based on maximization of system loadability using continuation power

flow theorem. *International Journal of Electrical Power and Energy Systems*, 54:123–133, 2014.

[15] R. Rajaram, K. Sathish Kumar, and N. Rajasekar. Power system reconfiguration in a radial distribution network for reducing losses and to improve voltage profile using modified plant growth simulation algorithm with distributed generation (DG). *Energy Reports*, 1:116–122, 2015.

[16] M. Babaei, R. Jafari-Marandi, S. Abdelwahed, and J. Kluss. A novel approach for real-time implementation of MVDC shipboard power system reconfiguration. *International Journal of Electrical Power and Energy Systems*, 100:240–252, 2018.

[17] K. L. Butler, N. D. R. Sarma, and V. R. Prasad. A new method of network reconfiguration for service restoration in shipboard power systems. In *1999 IEEE Transmission and Distribution Conference (Cat. No. 99CH36333)*, volume 2, pages 658–662, 04 1999.

[18] K. L. Butler, N. D. R. Sarma, and V. Ragendra Prasad. Network reconfiguration for service restoration in shipboard power distribution systems. *IEEE Transactions on Power Systems*, 16(4):653–661, 11 2001.

[19] Y. Xu and W. Liu. Novel multiagent based load restoration algorithm for microgrids. *IEEE Transactions on Smart Grid*, 2(1):152–161, 03 2011.

[20] C. L. Su, C. K. Lan, T. C. Chou, and C. J. Chen. Performance evaluation of multiagent systems for Navy shipboard power system restoration. *IEEE Transactions on Industry Applications*, 51(4):2769–2779, 07 2015.

[21] T. Sakaguchi and K. Matsumoto. Development of a knowledge based system for power system restoration. *IEEE Transactions on Power Apparatus and Systems*, 102(2):320–329, 02 1983.

[22] A. J. G. Mena and J. A. M. García. An efficient heuristic algorithm for reconfiguration based on branch power flows direction. *International Journal of Electrical Power and Energy Systems*, 41(1):71–75, 2012.

[23] K. L. Butler and N. D. R. Sarma. General reconfiguration methodology for AC radial shipboard power systems. In *2000 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No.00CH37077)*, volume 2, pages 1226–1230, 2000.

[24] ESRDC Team. Model description document notional four zone MVDC shipboard power system model. https://www.esrdc.com/media/1007/notional-four-zone-sps-model-desciption-v10.pdf. Accessed: 2018-08-27.