Generative Design of Robust Modular System Architectures

Giota Paparistodimou

A thesis submitted to the University of Strathclyde
For the degree of
Doctor of Philosophy

Department of Design, Manufacturing and Engineering Management
University of Strathclyde
Glasgow, UK

August 2020

Declaration of authenticity and author’s rights

This thesis is the results of the author’s original research. It has been composed by the author and has not been previously submitted for examination, which has led to the award of a degree.

The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.50.

Due acknowledgement must always be made of the use of any material contained in, or derived from, this thesis.

Signed:

Date:4th August 2020

Giota Paparistodimou

to Dave McLean

Keep Ithaka always in your mind.
Arriving there is what you're destined for.
But don't hurry the journey at all.
Better if it lasts for years,
so you're old by the time you reach the island,
wealthy with all you've gained on the way,
not expecting Ithaka to make you rich.

Ithaka gave you the marvelous journey.
Without her you wouldn't have set out.
She has nothing left to give you now.

And if you find her poor, Ithaka won't have fooled you.
Wise as you will have become, so full of experience,
you'll have understood by then what these Ithakas mean.

Ithaka by Constantine P. Cavafy

Acknowledgement

I would like to acknowledge and thank my industrial supervisor, Dr Robb, for the continued support he has given me over the last four years. I am also grateful to Willi Galloway for the insightful discussions and help to gather the data, his support was essential to the development of this study.

I would also like to express my gratitude to Dr Voong for encouraging me to investigate network science and Dr Knight 's guidance helps me explore the field of network science, and I would not be able to grasp the field 's potential without his continuous help, which I am grateful for.

I am also grateful to my colleague’s Dr Trivyza and Dr Anderson for being key supporters during my studies.

I would also like to thank Dr Whitfield for his support for the journal publications and Professor Corney for his kind help during my studies. I would like to thank Professor Duffy for his supervision.

I want to thank my mother who has always set the bar high for me, stressing that a woman can do anything she is dreaming of. I also thank my dad who encourages me to enjoy the little pleasures of life.

Finally, I am grateful to my rock, Dave McLean, who helped me conclude this study with his love.

Abstract

Modern engineering systems are frequently formed from complex networks of interwoven technological solutions, whose functions combine to enable key functions in the society: the defence of nations, the transport of people, the transmission of energy. The creation of such systems requires a detailed understanding of how the components interact both physically and functionally as a whole. However, while physical and functional connections (e.g. lines of power and control) can be defined, characterising a system's level of robustness when subjected to different disruptions is more challenging. Essentially although many system architectures aim to create a degree of redundancy, that ensures robust operation under disruptions, it's difficult to quantify the level of robustness and so to evaluate and select the right system architecture option during the initial stage of the design.

This problem can be observed in many different domains of distributed engineering systems i.e. systems dictated by their configuration of source and sink components, structured in such a way as to provide a specific set of functions. Because of this, architects often do not consider, how the definition of the modular configuration (i.e. the degree of modularity) affects the overall robustness of the system. Indeed, the choice of new system architectures is often dictated by subject matter experts and previous designs during the initial design phase. This leads to limited exploration, analysis and evaluation of potential system architecture design options, and selection of the system architecture that proves to be balanced between robustness and modularity at the beginning of system development.

Motivated by this need this thesis proposes a 'Robust Modular Generation and Assessment' (RoMoGA) methodology that combines a network tool (used to create alternate system architecture options) with a robustness and modularity evaluation metrics (that quantifies the robustness and modularity of each candidate system architecture option). The methodology has been used in case studies of three naval ship distributed systems and an explorative application was performed though experiments. The industrial evaluation was based on semi-structured interviews and industrial design practices. The findings of the evaluation highlighted the existence of trade-offs between redundancy, modularity and robustness, the influence of the type of redundancy, the effects of the type of disruptions and effects of patterns and topological features. The evaluation part of the study led to redesign and update proposals for the original naval system designs, which the experts have assessed as rational and improved design solutions. The thesis reflects on the limitations of the proposed methodology and makes recommendations for future work.

Table of Contents

List of Figures

List of Tables

Research Outputs

The following publications form part of this thesis.

Journal papers

Chapter 6, parts of Chapter 8 and 12 have been published:

1. Paparistodimou, G., Duffy A., Whitfield, I.R., Knight, P., Robb, M., 2020 "A network science-based assessment methodology for robust modular system architectures during early conceptual design", *Journal of Engineering Design*, 31 (4), 179–218; doi:10.1080/09544828.2019.1686469

Chapter 7, Chapter 9, and Appendix III have been published:

2. Paparistodimou, G., Duffy A., Whitfield, I.R., Knight, P., & Robb, M. (2020) "A network tool to analyse and improve robustness of system architectures", *Design Science,* 6, E8. doi:10.1017/dsj.2020.6

Conference papers

3. Paparistodimou, G., Duffy A., Voong, C., Robb, M.**,** 2017 "System architectures assessment based on network metrics" *Understand, Innovate, and Manage your Complex System! - Proceedings of the 19th International DSM Conference* pp. 117-126 Published by The Design Society.

4. Paparistodimou, G., Duffy A., Knight, P., Whitfield, I.R., Robb, M., Voong, C. 2018 "Network-based metrics for assessment of naval distributed system architectures", *14th International Naval Engineering Conference and Exhibition (INEC),* Glasgow, UK, 2 – 4 October 2018, doi: 10.24868/issn.2515-818X.2018.030.

The research mentioned in each publication was performed by the author of this thesis as an independent PhD student.

Nomenclature

Networks	
A	Directed network adjacency matrix
A'	Disrupted directed network adjacency matrix
adj	Undirected network adjacency matrix
S	Exponential matrix
Robustness	
$R_{s,\,t}$	Robustness evaluation metric
RTC	Redundancy threshold criterion
s	Sources
t	Sinks
Modularity	
µ	Stability resolution parameter
Q	Modularity metric
Ship terms	
CWP	Chilled water plant
DG	Diesel generator
EDC	Electrical distribution centre
Fwd.	Forward
GT	Gas turbine
HV	High voltage
LCG	Longitudinal centre of gravity
LPSW	Low pressure seawater
LV	Low voltage
SWBD	Switchboard
TCG	Transverse centre of gravity
VCG	Vertical centre of gravity
Others	
BAE Systems	BAE Systems Surface Ships Ltd.
DOE	Design of Experiments
MBSE	Model Based System Engineering
MoD	UK Ministry of Defence
R&T	Research and Technology
SME	Subject matter experts
Sparx EA	Sparx Systems Enterprise Architect
SURVIVE	QinetiQ SURVIVE®

Chapter 1: Introduction

> ***The best-laid schemes o' Mice an' Men***
> ***Gang aft agley***
>
> ***Robert Burns***

The architecture of a system[1] is developed at the initial stages, moving from concept to the selected system architectural option, and the architectural decisions affect the system throughout its life cycle. Although the architecture of the system is more concrete than the concept, it takes place during the initial phases of system design and is therefore of critical importance (Crawley et al., 2015). The decisions made during these early stages have a crucial effect on both the system's cost and capabilities. Duffy et al. (1993) stated, for example, that nearly 80% of costs were committed at the conceptual stage. Elias and Jain (2017) advised that "a system's architecture profoundly influences the cost and success of a system throughout the entire lifecycle and the business processes it supports". It is, therefore, critical during the conceptual stage to "proceed systematically and resist their preconceptions" (Wynn and Clarkson, 2018). The research study reported in the thesis is motivated by a desire to improve how system architecture is formulated by replacing subjective and qualitatively based decision-making processes with objective and quantitatively based methods that support the intentional design of system architectures.

Modularity and robustness are two desire and strategic attributes[2] of the system architecture that support the successful development and operational performance (Baldwin and Clark, 2000; Fricke and Schulz, 2005; Griffin, 2010; Hölttä-Otto et al., 2012). This study found that decisions on the appropriate level of redundancy, modularity and robustness of the system were not taken together at the initial design stage. Redundancy is a design approach that improves robustness (Turner et al., 2017). However, it is difficult to decide the right type and level of redundancy in the system architecture to improve robustness, and it is even more

---

[1] **System – "**a set of entities and their relationships; whose functionality is greater than the sum of the individual entities" [Crawley et al. 2015]

[2] **Attribute – "**inherent property or characteristic of an entity that can be distinguished quantitatively or qualitatively by human or automated means" [ISO/IEC/IEEE 15939:2017 Systems and software engineering]

challenging to successfully modularise a highly redundant architecture. Moreover, the design definition of the modular boundaries in the architecture can affect the expected robustness of the system (Walsh et al., 2019). This points to the need to evaluate alternative system architecture options of different level of redundancy and modularity and to select a system architecture design that improves robustness by balancing the different attributes. However, it is identified in practice and literature that there is a tendency to repeat earlier designs in which different architectural alternative options have only been explored and analysed in a limited way. This reinforces the need for generative methods to support the development of robust modular systems during the initial stages of design.

1.1 Research context

The context of the research is outlined in the following sections. The main areas discussed are system architecture, robustness, and modularity in the design of the system and the relationship between modularity and robustness.

1.1.1 System architecture design

Numerous definitions of systems architecture can be found in the literature. Crawley et al. (2015) elaborated by stating that “system architecture is the embodiment of the concept, the allocation of physical/informational function to elements of form, and the definition of relationships among the elements and with the surrounding context”. Wyatt et al. (2012) defined it as “a model of engineering artefact in terms of components linked by relations”.

Chen and Crilly (2016) defined system architecture as “characterisation of a system in terms of compositional relationships between its elements”. They continued by clarifying that the “characterisation a system’s structure distinct from the mapping relationships between its structure and function”, implying the definition provided by Ulrich and Eppinger (1997) that stated, “arrangement of the functional elements into physical blocks”. A concise definition provided by Crawley et al. (2004) stated that system architecture is “an abstract explanation for a system’s entities and the relationships between them’. This is the definition adopted in this study as a focus on the characterisation of the structure or topology of engineering systems and is suitable for a quantitative operationalisation. An example of a naval engineering system architecture is illustrated in Figure 1 that depicts the key power and propulsion components of a ship and the relationships between them.

Maier and Rechtin (1997) defined system architecting process as “creating and building systems. It strives for fit, balance, and compromise among the tensions of client needs and resources, technology, and multiple stakeholders interests" and affirmed that is "an art and science – both synthesis and analysis, induction and deduction, and conceptualisation and certification- using guidelines from its art and methods from its science". INCOSE (2015) stated that during architecture definition process "alternative system architectures are defined, and one is selected ... and that architecture definition activities include optimisation to obtain a balance among architectural characteristics and acceptable risks". This includes choices on the trade-offs between contradictory requirements and demands. Moullec, Jankovic, and Eckert (2016) highlighted that it is necessary to identify early the concepts, and their underlying architectures, that are most likely to provide the best trade-offs. Different system architectures may be designed to satisfy the same desired function, but they are characterised by different attributes.

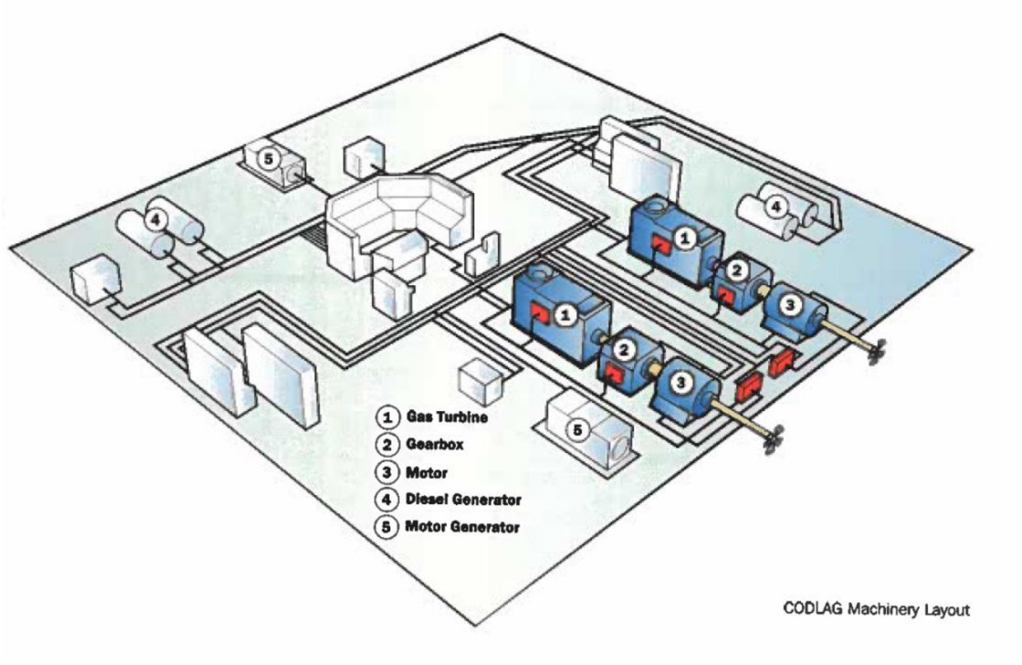


Figure 1: Naval engineering system architecture (copyrights BAE Systems)

System architecture assessment is defined as an activity where the architecture will "be reviewed, measured, appraised, analysed, and evaluated by system stakeholders or experts to identify attributes and as to whether they aid in the accomplishment of business goals" (Elias and Jain, 2017). The architecture decides the characteristics of the system termed also as its attributes, lifecycle properties, or ilities such as flexibility, robustness, reliability, and modularity are discussed in the engineering-system literature (De Weck et al., 2012), and are important assessment criteria for the competitive assessment of different architectures during the initial design phase (Sinha et al., 2019).

There are various challenges in developing scientific approaches that are objective, explorative, quantitative, and systematic to design and assess the complex system in the initial stages. Examples of these challenges are the limited time, low fidelity of available information or system details; the development of models to represent and analyse complex systems. Whitfield et al. (1998) highlighted that one of the difficulties in the initial design stage is that time is typically limited and especially for large complex products "this shortage of time causes a further complication since models which accurately represent the design and its behaviour or performance are, by necessity, large and complex". These led architects to adopt methods for carrying out analysis at the initial stages of complex systems, which are not typically quantitative (Elias and Jain, 2017). Other methods available in the engineering design literature typically examine individual attributes of a system in isolation (De Weck et al., 2012) causing them to be insufficient to examine the attributes of the system in combination. Also, the explorative design of the system architecture is often limited due to previous designs and intuitive expert opinions referred to as design fixation effects (Purcell and Gero, 1996) and the "discontinuous and qualitative nature of the 'design space' of product architectures ….[that] make it difficult to construct new product architectures, or even to be aware that there are alternatives to a particular architecture" (Wyatt et al., 2012).

Models are widely used in the field of system engineering for system architecture description. The model used addresses the most critical issues of stakeholders[3]. It has been common to use logical and physical models in the definition of architecture (IEEE and ISO/IEC, 2015). In the system engineering area, architectural modelling languages such as system modelling language (SysML) (Friedenthal et al., 2014) and object process methodology (OPM) (Dori, 2002) are used to enable the system architecture modelling. However, as Buede (2000) argued that the challenge is to develop "modelling techniques that have significantly more grounding in mathematics while maintaining the quality of communication among the stakeholders and the engineers of the various disciplines".

Design structure matrix (Browning, 2015) or coupling matrix (N^2 diagrams) (INCOSE, 2015) are proposed for modelling and analysing engineering systems, that are also useful in representing networks and facilitating employment network approaches in the field of engineering design. Network-based modelling and analysis is grounded in the graph theory mathematics, and grew in various research disciplines, after the attacks on the World Trade Centre on 11 September 2001, when it became apparent that a targeted attack on one or more major nodes of a complex engineering system could disable not just one system, but multiple

---

[3] **Stakeholder** - a group or a person influenced or accountable in some way for the system.

systems at once (De Weck et al., 2011). Network modelling and analysis is recommended in the design research (Chen et al., 2018). Its application is an area of research that is continuously expanding in the literature (Baldwin et al., 2014; Braha and Bar-Yam, 2007, 2004a; Parraguez et al., 2015; Piccolo et al., 2018) contributing on the development of new solutions that address engineering design problems.

Robustness and modularity are two attributes that can contribute to the success of the system and can be used as assessment criteria during the analysis and evaluation of the system architecture options at the initial stage of the system design. The following sections examine robustness and modularity separately, and then present a discussion on the relationship between modularity and robustness, offering research context for the study reported in this thesis.

1.1.2 Robustness

The industry has highlighted that robustness is critical in modern engineering systems and that designing robust system architecture is one of the most vital design tasks. Ross et al. (2008) stated that “the desire for “robustness” stems from the fact that change is inevitable, both in reality and perception”. Whitfield et al. (1998) highlighted that during the initial stages of the design process “it is essential that many alternative proposals are examined to identify those designs which are robust”.

The US Department of Defense (2011) identified robustness as a key characteristic of today’s engineering systems, recognising it as an architectural element of a system’s resilience. The US Navy ship USS Yorktown and the UK Royal Navy’s Type 45 destroyer are examples of complex systems which experienced discontinuity in their functionality during operations (Goodrum et al., 2018). Tellingly, in the US space industry, the Office of the Assistant Secretary of Defense for Homeland Defense and Global Security (2015) acknowledged that “today’s space architectures designed and deployed under conditions more reflective of nuclear warfighting deterrence than conventional warfighting sustainability, lack, in general, the robustness that would normally be considered mandatory in such vital warfighting services”. Latora and Marchiori (2004) highlighted the importance of robustness to safeguard critical infrastructure from malicious attacks. This infrastructure includes information-communication systems; electrical, oil and gas systems; and physical distribution systems, such as transportation and water supply systems that are vital to human services. Carlson and Doyle (2002) claimed that highly optimised tolerance (HOT) systems may be robust concerning the variants for which they are intended, but they are very vulnerable to unknown or emerging changes. A single event may have catastrophic consequences that challenge a system's behaviour immediately after an attack.

Various robustness definitions are found in the literature. Robustness is perceived as an ability of a system or the degree to which a system is able, thereby defining it as an attribute or property of a system. Another aspect of robustness is that this ability of the system arises when something happens, such as a change, a fluctuation or attack or disruption. Fricke and Schulz (2005) defined robustness as "a system's ability to be insensitive towards changing environments". This definition includes the word insensitive, which is means that the expected behaviour should remain unchanged. However, Carlson and Doyle's (2002) suggests the system should maintain some desired functionality or, as they stated, "the maintenance of some desired system characteristic despite fluctuations in the behaviour of its component parts or its environment". Griffin (2010) affirmed that a robust system should "degrade gradually and gracefully in response to component failures, changes in its operating environment, or when design loads are exceeded". Graceful degradation, performance insensitivity, and post-disruption functionality continuity are aspects of robustness discussed in the literature.

Elias and Jain (2017) provided a definition which also considers "to what extent the instantiated system(s)' functions(s) as architected can correctly operate in stressful conditions or with incorrect inputs". This definition is aligned with the study of this thesis, as robustness is perceived as an attribute of the system architecture and the ability of the instantiated architecture (the structure of the system) to satisfy its function by exhibiting the correct behaviour. A further definition is that of Yazdani and Jeffrey (2012) that is motivated by a topological (architecture/structure) perspective on robustness within network theory, describing robustness as: "a topological quality (based on patterns of connectivity) of a network, which makes its performance more resistant to the disruption of operation after failure of components". Finally, the Department of Defense (2011) also gave attention to robustness from an architectural viewpoint, because the defence industry is particularly interested in related attributes of systems, such as resilience and survivability. For the Department of Defense (2011) robust systems have "architectural properties and system of systems design features to enhance survivability and resist functional degradation". Most of the definitions consider robustness to be either a system's ability or capacity to maintain its function (performance) after a change or disruption. Three of the above definitions (Department of Defense, 2011; Elias and Jain, 2017; Yazdani and Jeffrey, 2012) also included in the definition of robustness the following related expressions: architecture/ instantiated system as architected/ topology. In the research study, the definition of robustness is the ability of the instantiated system architecture to support sufficient functional continuity after a disruption.

Redundancy is the primary strategy used in system design to improve robustness. Turner et al. (2017) acknowledged redundancy as a principle to achieve robustness of a system. Fricke and Schulz (2005) acknowledged that redundancy “is key to flexibility and robustness since it enables capacity, functionality, and performance options as well as fault-tolerance”. However, designing redundancy in the system may be improving robustness but reducing modularity as will be following also discussed.

There is a distinction between robust design methods and robustness from a system design viewpoint. In general, robust design methods are more tailored to the detailed design stages, where information on the parameters of the design is better understood. However, from a system design viewpoint, robustness also needs to be addressed during the conceptual stages, since it focuses on how the system behaves as a whole after a disruption. Adams (2015) stated that “there is no one universally accepted method for assessing robustness in systems design endeavours”. At the same time, the application of network science to engineering design is an area of research that is continuously expanding the field of engineering design which views robustness in a system-level: Walsh et al. (2018) stated that “the use of complex networks in engineering design has perhaps been one of the more prominent developments in recent years”, and Braha and Bar-Yam (2004) have introduced this field into engineering design research. Most importantly in the current context, the robustness of engineering systems has begun to be studied using network-based metrics (Haley et al., 2016; Haley and Dong, 2014; Mehrpouyan et al., 2014; Walsh et al., 2019).

1.1.3 Modularity

Projects involving complex engineering systems have frequently experienced increases in system development time and costs. This has resulted in the “abandonment of many expensive projects and led to highly impaired implementations in other cases” (Bar-Yam, 2003). For instance, in the development of the Boeing 787 (Tang et al., 2009), the aircraft industry reported delays and increases in costs, while other aerospace industry projects have also been subject to rising costs (Tamaskar et al., 2014). A common denominator in these examples is their high-level of complexity. Aspects of complexity include multiple components, high interconnectivity, and the heterogeneity of patterns in the architecture (Sheard, 2012).

Modularity is “building of complex product or process from smaller subsystems that can be designed independently yet function together as a whole”(Baldwin and Clark, 1997) and is regarded as a strategy which helps to resolve complexity (Baldwin and Clark, 2000). Three core aspects of modularity are suggested to be structural encapsulation, one to one mapping of function to structure mapping and well-defined interfaces (Chen and Crilly, 2016). The common denominator of the different definitions is that modularity in systems is equal to that of systems consisting of modules. Miraglia (2014) stated that modularity is “the systems’ property of being made up of modules. A module is a system’s element that presents a high, albeit not complete, independence of other elements”. Modular architecture compromises modules, which are defined as “relatively independent chunk of a system that is loosely coupled to the rest of the system” (Hölttä-Otto et al., 2012).This is the definition adopted in this research specifically because it highlights, the "loosely coupled", indicating that modularity is not absolute in design of systems.

Modularity is argued to have a wide range of advantages (Ulrich, 1994), which explains its use in the design of various types of products and systems. A product or system with a modular design also allows the reuse of components (Meehan et al., 2007), concurrent design and production, and easy maintenance, repair, upgrades, and retirement through its lifecycle (Dahmus et al., 2001; Jose and Tollenaere, 2005). The many advantages mean that modularity may also have a positive influence on the cost of the system (Hölttä-Otto and de Weck, 2007). In practice, modern large-scale supply chains and the various construction and manufacturing locations associated with complex projects have led to an increasing demand for systems to be modularised to facilitate their parallel development and the engagement of subcontractors. Moreover, modularity is also pursued political and societal reasons. For example, Laurence et al. (2010) pointed out that there are political reasons for pursuing modularity in the construction of complex systems, such as ships, where there is a need to divide up the construction to support different regions or countries. As such, modularising a design is driven by several factors, from the desire to obtain competitive benefits in terms of cost and development time to other practical and societal purposes.

In order to gain these benefits and successfully modularise a design, the literature suggests various modular design methodologies to support the grouping components into modules to formulate modular configurations of the system architecture. Whitfield et al. (2002) defined the criteria for the optimal structure of a modular product as the clustering of components so that the degree of interaction/dependency is maximised internally within groups (modules) but minimised externally between groups (modules). This wide range of modular design methods also helps to identify which components should be grouped into modules depending on the modularisation drivers (Bonvoisin et al., 2016). The idealisation of modularity in engineering design research has led to different modularity metrics and methods which seek to find an "ideal modular architecture". (Sanaei et al., 2015).

However, the search for the "ideal modular architecture" has been challenged in both industry and academia. Hölttä et al. (2005) investigated the trade-offs between modularity and performance, advising that an integral architecture, rather than a modular one, is more suitable for designs that have technical constraints. Bar-Yam (2003) explained that "as the systems become more complex, the design of interfaces between parts occupies increasing attention, and eventually the process breaks down". This means that substantial resources need to be used to design these interfaces, reducing any modular design benefits (Walsh et al., 2019). The industry also indicates that modularity may introduce drawbacks, as companies responsible for complex engineering systems have experience challenges in successfully implementing modularity and in realising the anticipated benefits in practice.

Hvam et al. (2017) explained that the maximum degree of modularisation is perhaps not most beneficial for engineer-to-order companies, which "usually attempt to remodel their products to become completely modularised and with detailed descriptions of the modules – although such companies might experience greater benefits from a lesser degree of detail and/or modularization". This suggests that a lower (than a maximum) degree of modularisation is advantageous in such situations. These observations from industry and the academic literature indicate that modularising the system may need to consider trade-offs and constraints with other aspects of the design. Research works have been identified in the literature that shares similar perspective (Bayrak et al., 2018; Otto et al., 2020; Sanaei et al., 2016; Sinha et al., 2019). In this way, modular approaches are essential, as they help to decide how the system architecture can be divided into modules based on multi-design goals and constraints. The decisions affecting the formulation of the modular architecture will subsequently affect different aspects of the development of the system, such as its integration, construction, delivery, operation, and retirement. Moreover, the modular boundaries of the architecture potentially affect the system's robustness.

1.1.4 The relationship between modularity and robustness

The perception in the industry is that it is desirable to modularise the system for cost, development and maintenance reasons and that the robustness of systems, that is, the system's ability to resist suddenly catastrophic interruption, is also a vital quality that modern engineering systems have to satisfy. This aligns with the findings of the review of the literature on modularity and robustness and leads to the conclusion that today's engineering system are required to be both robust and modular. Walsh et al. (2018) highlighted the importance of studying modularity and robustness together as "many of the same systems that would benefit from modularity are the same systems that are robustness-critical". However, a clear understanding of the relationship between modularity and robustness was absent in both the research literature and industry. Walsh et al. (2018) defined this as the "paradox of modularity", the question of whether modularity benefits or inhibits robustness.

Previous scholarly works asserted that modularity carries advantages for robustness because it enables manufacturing faults to be identified early and these and other mistakes to be isolated (Gershenson et al., 2003). But other scholars have argued the opposite about modularity, suggesting that greater modularity is associated with reduced robustness (Walsh et al., 2018). For example, Mehrpouyan et al. (2014) recognised that modularity is helpful to build and maintain complex systems, but that mistakes into a single module usually makes a system less robust. As it has noted, robustness in complex engineering systems is typically enhanced by designing for redundancy, and this can be achieved in the systems through duplication and substitution (Chen and Crilly, 2014). However, redundancy and modularity can be in tension, as implementing redundancy decreases the modularity of a system, because the mapping between the function and elements of the system becomes more integrated (Chen and Crilly, 2014).

The industrial and academic findings indicate that there is a need to better design robust modular system architectures. Mehrpouyan et al. (2014) asserted that "to design a robust system and to recommend or oppose the modular physical system architecture, it is utterly important to understand the architectural properties of complex engineered systems". Chang (1996) discussed robustness concerning quality, stating that "products with higher modularity may be subjected to greater noise effects and/or higher direct costs, yet have better readiness ... How to balance these conflicting interests or how to integrate these interest into the modular design objective function and/or constraints, is a research topic that needs careful review". Walsh et al. (2018) concluded that "the overall benefits of increasing a system's modularity are situation-dependent rather than universal. Constant re-evaluation of established design principles is necessary given their usefulness". The present study recognises that specific attention is needed to examine the relationship between modularity and robustness in line with these industrial and academic findings. In turn, the study defines the term robust modular as the ability of the instantiated system architecture to support the sufficient functional continuity after disruption of a module

System architecture design involves attempts to place functions and components in different modules (Otto and Wood, 2001), while developments in systems engineering have also enhanced the effectiveness of the robust design, pushing the entry point of robust design methods further upstream in the process of product development (Slagle, 2007). Designing robust modular systems may involve architects making trade-offs between robustness and modularity. Notwithstanding the broad range of methods and metrics developed in the literature to improve systems ' modularity (Bonvoisin et al., 2016) or robustness (Göhler et al., 2016), it has been found that engineering design deal separately with modularity and robustness. Crawley et al. (2015) stated metrics are “a way in which decisions can be linked to each other”, and this study suggests that approaches that combined application of metrics of robustness and modularity can support early architectural decisions.

According to the current practice in industry modularity and robustness are not examined together in the initial stages, even though formulating a modular configuration might affect the system’s robustness. However, modern engineering systems are required to be robust and modular, to take advantage of the benefits of modularity, while also ensuring that they are robust concerning disruptions. It is both academic and industrial interest to incorporate approaches at the earliest stages to support the design of robust modular system architecture.

1.2 Research question

In light of the above discussion, the research question that guides this research is:

> How can the design of a robust modular system architecture be supported during its initial stage?

1.3 Research aim

The research aim is to develop a methodology to support the design of robust modular system architecture.

1.4 Research objectives

Research to the academic literature and industrial practice suggested that this aim could be associated with the following objectives:

Objective 1

Define an approach to the representation, generation, and analyses of system architectures.

Objective 2

Develop a methodology to combine the assessment of modularity and robustness in system architecture.

Objective 3

Identify a method for quantifying the level of modularity in the system architecture.

Objective 4

Identify a method for quantifying the level of robustness in the system architecture.

Objective 5

Develop a method that generates alternatives system architectures.

Objective 6

Apply the methodology in technical and theoretical system architectures.

Objective 7

Evaluate the methodology in an industrial context.

1.5 Thesis structure

This thesis has three parts: 1) research scope; 2) methodology development; and 3) evaluation (Figure 2). Before focusing on the research, this introductory Chapter 1 prefaces the thesis. Chapter 2 outlines the research strategy; research philosophy; methods and design. The rest of the thesis is structured as follows:

Part 1: Research Scope (Chapter 3 and 4)

Chapter 3 outlines the focus group discussion, the process, and the analysis. The findings from the analysis directed the researcher to establish the research direction. The emerged industrial challenges were relating to the need for systematic system architecture approaches, finding the right level of modularity and balancing modularity and robustness. Chapter 4 presents a literature review focus on the emerged challenges raised through the focus group.

Part 2: Methodology Development (Chapter 5, 6 and 7)

Chapter 5 provides an overview of the Robust Modular Generation And Assessment (RoMoGA) methodology. Chapter 6 presents the descriptive implementation of RoMoGA methodology, providing information on the selection and development of the modularity and robustness metrics and methods, and discussion on the different stages of the methodology. Chapter 7 presents the explorative implementation of RoMoGA focusing on explaining the development of the network tool that populates alternative system architecture options.

Part 3: Evaluation (Chapter 8, 9, 10 and 11)

Chapter 8 presents the industrial application: the proposed methodology is applied in three technical case studies of existing naval engineering systems. Chapter 9 presents the explorative application: various experiments are performed using the network tool. Chapter 10 and 11 present industrial evaluation. Chapter 10 presents the semi-structured interviews that the researcher conducted with the experts to evaluate the usefulness, appropriateness, and applicability of the methodology. Chapter 11 outlines the industrial design practices: original and updated designs informed by the outcomes of the application of the methodology in Chapter 8 are compared using ship design vulnerability software. Chapter 12 discuss the research findings of the four evaluation methods and summaries the key limitations of the methodology and recommends future work. Chapter 13 finish with an overview of the study, discusses the achievement of the research aim and objectives concluding the study reported in the thesis.

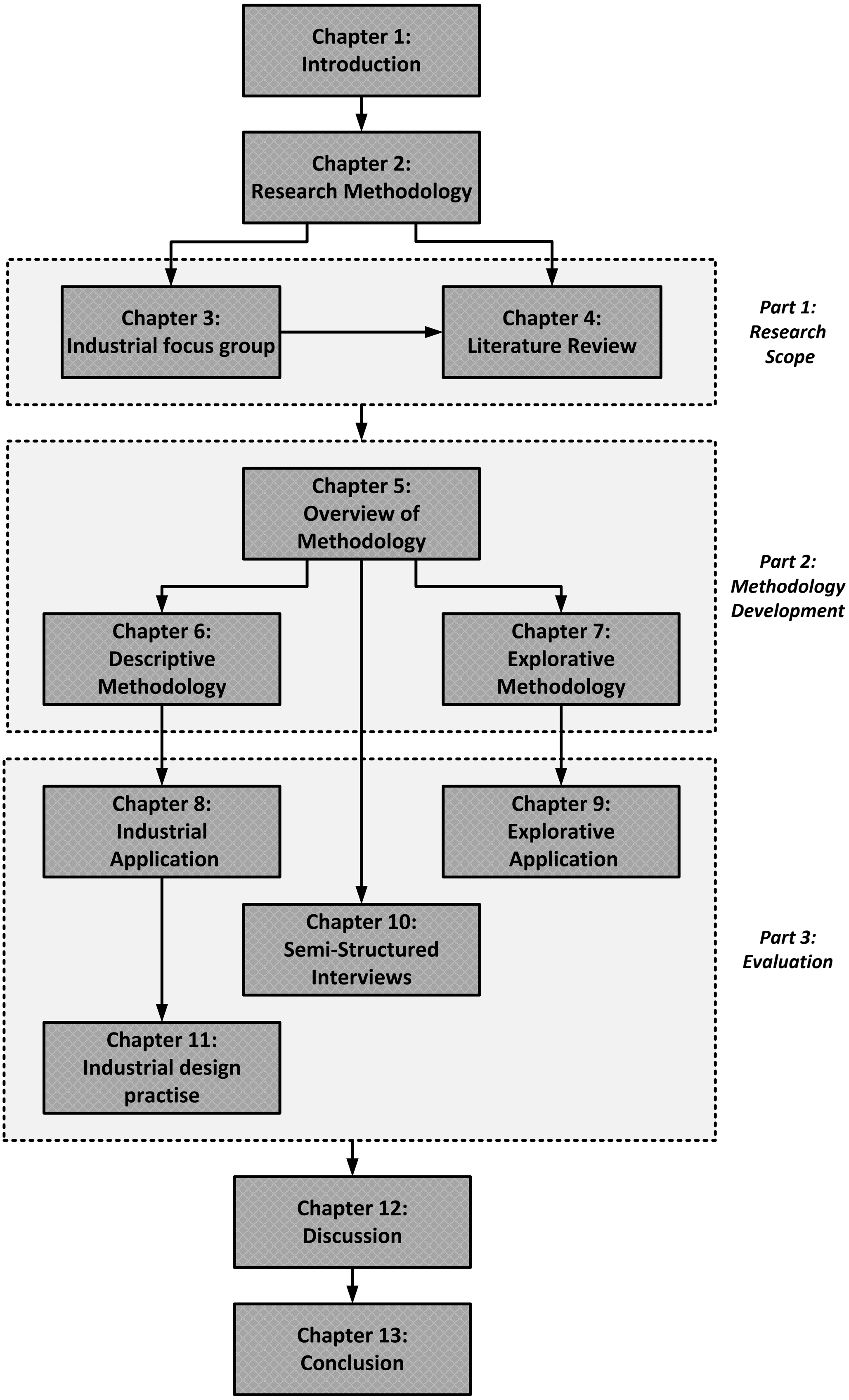


Figure 2: Thesis structure

1.6 Chapter summary

Chapter 1 introduced the research study reported in this thesis. The research context based on academia and industry was outlined. The research question was formulated and based on this; the research aim and objectives were derived in Section 1.3 and 1.4. The structure of the thesis was presented in Section 1.5 outlining the thesis main parts: research scope, methodology development and evaluation. The following Chapter 2 introduces the research strategy discussing the research philosophy, methods, and approach.

Chapter 2: Research strategy

Chapter 2 presents the research strategy that includes the philosophical assumptions, methods and approach used in the research study. The core of the research process is the determination of a research strategy to ensure the rationale and a systematic approach to the research study. Elements of the research strategy define the philosophical research assumptions that are driven by the researcher’s beliefs and the research methods are based on research aim. This chapter provides an overview of this research philosophy, design and the methods used therein.

2.1 Philosophy

Easterby-Smith et al. (2012) explained that comprehending the philosophical assumptions and selecting the correct philosophy, which corresponds to the research aim and objectives, helps the researcher to select from the possible ontologies, epistemologies and methodologies available in the literature. The way, in which the research approach is developed, is shaped by the philosophy, embraced by the researcher, and his/her assumptions about human information and how reality is experienced. Saunders and Tosey (2013) argued that a research philosophy is the researcher’s “personal view of what constitutes acceptable knowledge and the process by which this is developed.” The following paragraph presents a debate amongst three possible research philosophies, namely, positivist, critical realism, and pragmatism that, potentially, could apply to this research study. This discussion outcomes find that pragmatism philosophical assumptions are the applicable to this research based on the research aim and objectives. It provides, also, the background to the pragmatism philosophical movement and its relationships with engineering and architecture.

According to Johnson and Onwuegbuzie (2004), during the research procedure, positivists ignore the fact that several subjective decisions are made and that often researchers belong to a variety of different social groups. Some instances of subjectivism and inter-subjectivism within quantitative research are relevant to the research study. These are: developing construct; selecting particular factors and examinations for evaluation; interpreting quantitative scores; making reflections and reaching conclusions regarding the gathered data; and establishing which results are significant and realistic (Johnson and Onwuegbuzie, 2004). In this research, it would be misleading to claim that these research philosophy assumptions are positivist because they are subjective decisions that influenced the research study.

Saunders and Tosey (2013) explained that “a researcher reflecting a critical realist position argues that what is initially experienced through senses is subsequently processed subjectively by the mind." The research study aims to systematically support the architect to design a robust modular system by developing an objective and analytical methodology. However, while this research study accepts that the existence of subjectivity can impact on the interpretation of any quantitative results, it aims to manage this subjectivity of human minds. Saunders and Tosey (2013) argued that, for a critical realist, "there is a need to find out both what is immediately experienced and the structures and relationships that lie beneath this.". Therefore, this research study does not accept the critical realism philosophical standpoint.

It is argued that the research study's philosophical assumption is one of pragmatism. The research aim of the study is to support the architect in designing robust modular system architecture. The study embraces a pragmatic viewpoint on deciding to develop a methodology that generates and assesses system architecture options during initial system design stages of design to support the architect to more objectively make decisions. Saunders et al. (2012) explained that pragmatics "recognise that there are many different ways of interpreting the world and undertaking research, that no single point of view can ever give the entire picture and that there may be multiple realities." The pragmatist view accepts that the driver of the research is the question and the researchers can bring together different philosophies in their research to achieve their research objectives. "Reality matters to pragmatists as practical effects of ideas and knowledge are valued for enabling actions to be carried out successfully" (Saunders et al., 2015).

Bulleit (2017) stated that "pragmatism and engineering have some significant similarities, both in the way the world is viewed and the way that beliefs are proven to be true. The two very different fields also consider the practice to be more important than theory, and experience is critical in each of their approaches". Similarly, the research study uses pragmatism since it embraces "philosophical issues in a practical way" (Bulleit, 2017). There are two ways in which an essential congruence exists between engineering and pragmatism. These are, namely: how the world is perceived; and how the truth of beliefs is established. Furthermore, these two distinct areas regard the practice as having greater importance than theory. Additionally, experience is crucial in both types of method. Sometimes, engineers are mentioned directly in pragmatism literature, whereas, at other times, some of the literature is perceived as being relevant to engineering. From most perspectives, the pragmatic movement is a system of analysing philosophical matters practically. Therefore, it is essential that the pragmatists regard philosophy in the same way as engineers regard science and technology.

Goldman (2004) suggested that pragmatism offers the most significant possibility for the rational evaluation of technological action, with engineering as its facilitator. Saunders and Tosey (2013) argued that researchers, who accept pragmatic philosophy, discover the significance of research through the practical consequences of the results. They believe that any single perspective can never describe a complete portrayal and that multiple realities are possible. Nevertheless, this does not imply necessarily that a pragmatic researcher would apply consistently a variety of analytical processes and data-gathering methods, but rather that research design ought to allow “credible, reliable and relevant data to be collected that support subsequent action” (Saunders and Tosey, 2013). The practical consequence of the results of this research study is how the methodology can in practice contribute to the redesign or new designs of robust modular system architectures.

2.2 Ontology and epistemology

Easterby-Smith et al. (2012) considered that ontology relates to “philosophical assumptions about the nature of reality” while epistemology correlates with the various ways in which it is possible to analyse the world given the researcher’s philosophical assumptions.

The ontological position of pragmatism is that reality is discussed continually, clarified and renegotiated given its value within novel unforeseeable circumstances. Two main ontologies, discussed in the literature (Easterby-Smith et al., 2012) are objective and subjective. The objective ontology posits that the truth exists irrespective of who is the viewer; the aim is to uncover what is this truth. On the contrary, subjective ontology is the basis of the truth. The epistemological position is that the optimal technique is the one that resolves difficulty and facilitates the discovery. This is the method and changes are the fundamental objectives.

The researchers who follow the theory of pragmatism, like architects and engineers, solve problems and obtain outcomes. Table 1 encapsulates the pragmatism philosophy approach, ontology, and axiology and research strategy. Essentially for pragmatism, the research questions and objectives are the drivers of the research. Different epistemologies and ontologies are to be adopted if they can provide replies to the research questions or help on accomplishing the research objectives (Saunders et al., 2008). The following Table 1 is adapted by Saunders et al. (2015) and Wilson (2010).

Table 1: Pragmatism: ontology-epistemology-axiology-methods

Ontology	**Epistemology**	**Axiology**	**Methods**
Deductive/Inductive	***Objective or Subjective***	***Value-free/Biased***	***Quantitative and/or Qualitative***
Managing and organising are: -the fluid of processes, experiences, and practices -complex, rich - "Reality" is the practical consequences of ideas.	Solve problems and inform future practice through: -Searching for the practical meaning of knowledge in specific contexts -Theories that enable successful actions -Focusing on problems, practices, and relevance	Value-driven research: -Research initiated and sustained by researchers' doubts and beliefs. -Researcher reflexive.	The range of methods to fit the research problem. Often mixed and/or action research. Emphasis on practical solutions and outcomes.

2.3 Research methods

Corresponding to the research study, the researcher used throughout the research approach pragmatic philosophical assumptions and mixed-method approaches. According to Brannen (2005), pragmatic rationality adopts mixed methods more easily if they are suggested by the practicalities and questions of the research context. Exponents of pragmatism combine more than one strategy in their studies, and this allows the use of multiplied methods such as combining qualitative and quantitative. As explained by Johnson and Onwuegbuzie (2004), the formal definition of mixed methods is a classification in which the researcher amalgamates or merges into one unified study qualitative and quantitative research methods, concepts and approaches. The rational inquiry involves the utilisation of deduction (theory and hypothesis testing), induction (discovery patterns), and abduction (revealing and depending upon the optimal series of interpretations to understand results). When responding to questions, mixed methods research endeavours to legitimise the application of multiple methods rather than limiting the researcher's options and, thereby, rejecting dogmatism. This research method is inventive and extensive, non-restrictive, pluralistic, comprehensive, and complimentary. Creswell et al. (2003) stated that a mixed-methods study: "involves the collection or analysis of both quantitative and/or qualitative data in a single study in which the data are collected concurrently or sequentially, are given a priority and involve the integration of the data at one or more stages in the process of research". In the research study, the researcher used not only qualitative methods, such as a literature review, explorative focus group discussions, conceptual modelling, case study methods, but, also, quantitative methods such as mathematical, computational modelling and, experiments. Consequently, the research study uses both quantitative and qualitative methods and adopts a mixed-method approach in alignment with the pragmatic philosophical assumptions.

2.4 Research approach

Figure 3 illustrates the research approach and methods which the author employed for collection and analysis to address the research objectives guided by Creswell (2014).

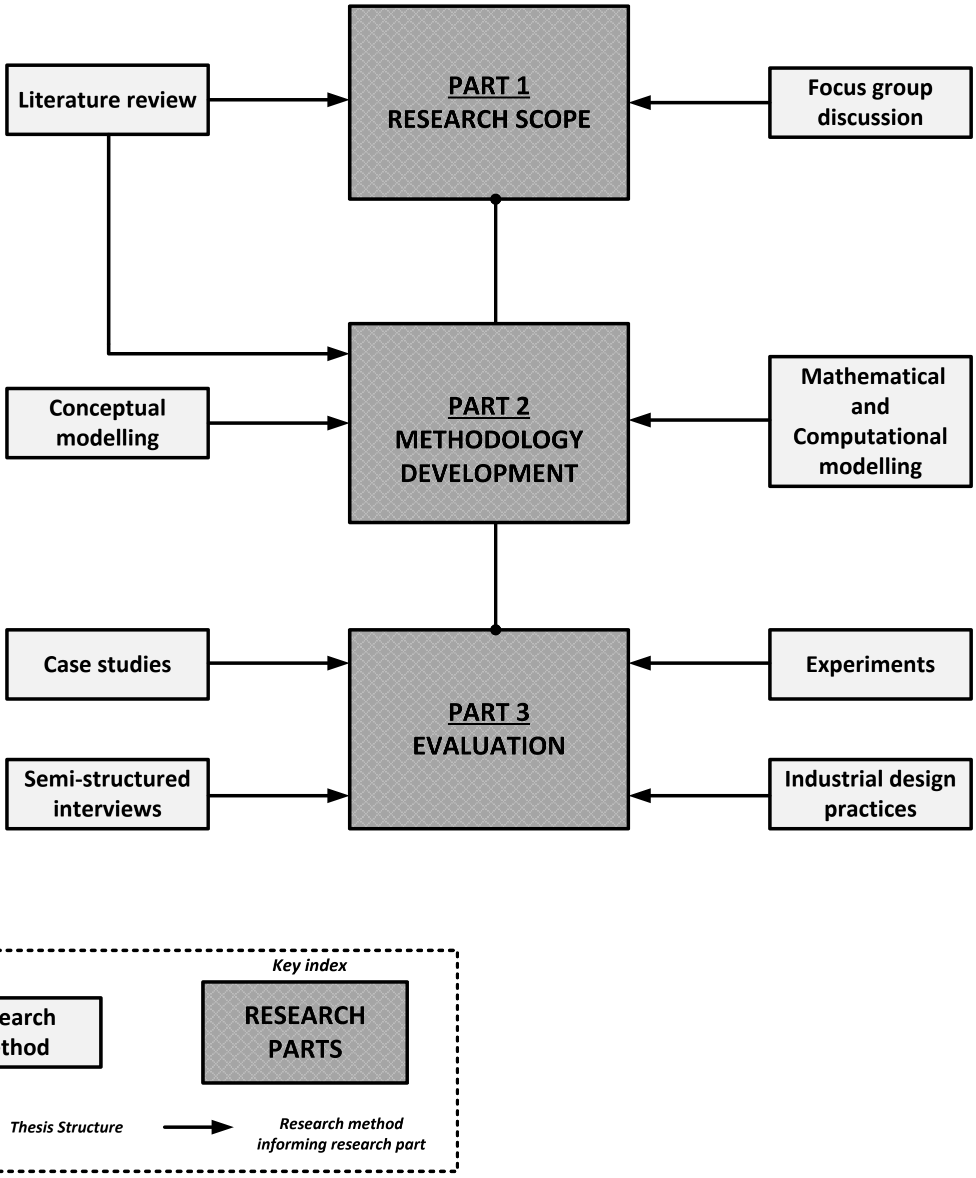


Figure 3: Research approach

2.5 Mixed methods

In this research, both qualitative and quantitative techniques for data collection and analysis were employed. This mixing of methods and techniques for data collection and analysis ensured research triangulation. In this regard, Eisenhardt (1989) stated, “a combination of quantitative and qualitative data types can be highly synergistic”.

The following sections discuss the research methods used in the different parts of the study.

2.5.1 Research scope

For the research direction, the two qualitative methods employed were: 1) focus group discussion; and 2) literature review.

2.5.1.1 Explorative focus group discussion

In the focus group discussion, the researcher adopted a qualitative method to explore the research scope and direction. According to Saunders et al. (2007), a focus group discussion is an efficient method since, in a limited time, the researcher can gain significant understandings and insights of various viewpoints because of the interactions between the participants. In the focus group discussion, the participants were engineers and managers from different departments and positions of BAE Systems. This mix of participants was highly beneficial in allowing an informative discussion to be held that, critically, helped the researcher to identify the research scope and direction.

A unique feature of a focus group framework analysis is that while using a thematic approach, the themes can be developed from research questions and research participants' narratives (Rabiee, 2004). In Section 3, the challenges emerged first through familiarisation with the transcript, then through the writing of short sentences, ideas or concepts from the transcript and the preliminary development of categories, and then the quotations were re-arranged in the newly developed categories (Rabiee, 2004). Data reduction was also used to ignore non-essential information, which helps to focus on the most important quotations and ideas.

2.5.1.2 Literature review

Saunders et al. (2007) stated “reviewing the literature will provide the foundations on which your research is built.” In the research study, the researcher used the literature review for two purposes. The literature review focused on the three challenges that emerged from the focus group discussion. These were modularity optimality, robustness and its relationship with modularity and the assessment approaches to system architecture. Moreover, the literature review was used to inform the development of the methodology.

2.5.2 Development methodology

For the central part of this research (that developed the methodology), mathematical, conceptual and computational modelling was employed. The findings of the literature review also informed the key aspects of the development of the methodology.

2.5.2.1 Conceptual modelling

The conceptual modelling, used in the research study, ensured that the developed methodology was based on the combinations of theories and knowledge from engineering systems and mathematical network science. The researcher's viewpoint is that the intersections of these two different areas of knowledge enable the generation of new knowledge. Therefore, the research study accepts an analogy between the engineering system's architecture and a network. Analogy signifies that the engineering system and the network are comparable and have admitted similarities. This allows for a methodology, which can generate outcomes useful in engineering systems, to be developed and inspired by concepts in network science. The research study does not try to adjust network science to the context of engineering system context but extracts the engineering systems architecture at a generically high-level and then, through network analysis deduces correlations amongst their behaviours. This is the reason that herein the development of the methodology is termed conceptual modelling. The researcher achieved the method, used to create the methodology, through the utilisation of network science concepts and giving them engineering system significance. The research study incorporates network science-based metrics, method, terminology and classifications into a methodology that supports architects when designing and assessing system architectures. The solution was generated by mixing ideas from the domains and was the basis on which the analogies were accepted. The network science metrics, method, and classification were used in the methodology that applies to the distributed engineering systems.

In the methodology, a high-level abstraction of the engineering system is modelled as a flow network that, in combination with the definition of specific nodes as source and sink, encapsulates an abstract representation of the system's functionality. The set of sources and sink components (nodes) are defined because of the proposition that the function is satisfied when sufficient connectivity exists specifically between the source and sink components. The methodology is created by incorporating mathematical network science methods and metrics such as the stability algorithm, the modularity metric, and the development of the novel robustness metric, which is a hybrid mathematical and engineering-driven metric. The conceptual modelling method for the development of this novel methodology, which incorporates network science concepts combined with the concepts of engineering systems, is justified through the researcher’s pragmatism philosophy paradigms.

2.5.3 Evaluation methods

Finally, for the evaluation of the methodology, the researcher used four methods: namely, case studies; experiments; industrial design practice and semi-structured interviews.

2.5.3.1 Industrial application

The industrial application of the methodology proposed was conducted through case studies based on three naval designs. According to Yin (2003), case studies can involve single or multiple cases as well as numerous analytical levels. In the research study, the researcher presents three technical case studies. Concerning the selection of case studies, Eisenhardt (1989) advised that polar type case studies are “transparently observable.” In the research study, the researcher’s objective is to use different case studies from the same technical system so that the results can be compared. To develop the case studies and to develop the Design Structure Matrix (DSM), the researcher extracted secondary data from such as archived records, technical documentation, and drawings. This was done with the help of experts in the specific technical systems.

In cooperation with the industrial supervisors, the researcher selected three technical system architectures. The Type A system architecture is in operation and has been demonstrated to be successful. While the Type B system architecture has a similar function, a different design is in operation and this has experienced updates during its lifespan. The third system architecture, Type C, is a modern system architecture design. The researcher paid attention to replication between the case studies to ensure that the systems’ boundaries were similar and that the results could be compared. The purpose of this research was not only to select case studies that were appropriate for establishing diversity but, also, to allow comparisons to be made.

2.5.3.2 Modern and legacy system architectures case study

The age gap between Type A and Type B system architectures is more than 15 years. Type C is the modern system architecture and the differences between Type A’s and Type B’s architecture are 15 and 25 years, respectively. Over this period, many different technical solutions and requirements emerged. The investigation and comparisons between the architectures provided insights to the impact of factors including the operational experience of Type A and B over the years, had influenced the design of the system architectures.

2.5.3.3 Explorative application

The explorative application employed using the network tool based on different experiments. The setup of the experiments was informed from the findings of the case studies. The experiments demonstrated an explorative implementation of the methodology. The generator was used to simulate networks and inform the analysis of the effects of patterns and parameters of hubs on modularity and robustness.

2.5.3.4 Industrial evaluation

The researcher conducted semi-structured interviews with engineering managers and systems architects to evaluate the usefulness, appropriateness, and applicability of the proposed methodology. Semi-structured interviews permit a versatile and flexibility on collecting data that allows altering the style of the questions during the interviews given the flow of the discussion or the expertise of the interviewees (Saunders et al., 2008). The semi-structured interviews were decided as a suitable research method to evaluate the developed methodology, because they allow the researcher to discuss with various stakeholders of the complex development project in a flexible approach, with a set of basic questions, setting the scene of the conversation, but simultaneously providing the freedom to the interviewees to express their opinions based on their specific expertise and knowledge.

The researcher also evaluated the proposed methodology in an industrial design practice developed in BAE Systems. Industrial design practices support the assessment of the methodology in an industrial context. Duffy and Donnell (1999) described industrial studies as actual design processes and proposed them as methods to verify and analyse design research. The industrial design practice concerns SURVIVE software, the UK MoD approved approach to the assessment of ship design vulnerabilities.

2.6 Analysis framework

According to Saunders et al. (2009), the two ways to gain knowledge are through deduction and induction. Typically, quantitative methods adopt deduction and qualitative methods embrace induction. An induction approach is a problem based whereas a deduction approach is theory-based. In the research study, the researcher employed both deduction and induction approaches to answer the research aim and objectives. The pragmatism philosophical assumptions of the research study allow this combination of induction and deductive approaches and provide the researcher with the flexibility to move cyclically through previous studies. The development of the solution partially followed an inductive approach because the researcher based the formulation of the novel robustness metric and the methodology on the acceptance of the analogies between network science and engineering system concepts. In contrast, the researcher adopted a deductive approach for the application and evaluation parts of the research study. This was done in the sense that the outcomes of the methodology were compared and evaluated against existing knowledge about the established technical systems and deliberated.

2.7 Validity of the research

In the research study, the researcher used the triangulation of data and methods. The data triangulation relates to the various types of data collected through different research methods. The data were obtained from engineering documentation, archival documents, and design documents as well as holding meetings with experts and discussed any contradictions in the sources of the data with the relevant experts. In order to ensure the accuracy of the gathered technical data, the researcher had technical meetings with various experts from different domains in the company. The triangulation of the methods was accomplished using various methods such as literature review, focus group, conceptual modelling, case studies, experiments, industrial design practices and semi-structured interviews. This variety of qualitative and quantitative methods ensured that the researcher used different approaches to accomplish the research aim. Triangulation by methods was adopted throughout the study, during the research scope, development and evaluation parts. The research scope part used two different methods: exploration focus group discussion and literature review. The development part reports the methodology that was accomplished through the literature and conceptual, mathematical and computational modelling and was based on requirements deduced by focus group discussion combined with the literature. The evaluation of the methodology was made through its application in the case studies, experiments, industrial design practice and semi-structured interviews. The variety of research methods involved in this research into compliance with the research triangulation principles.

Duffy and Donnell (1999) explained that the establishment that there are levels of reality for research hypotheses or results is the main aim of a validation process. The methodology is regarded as viable as it can capture reality as evident by case studies (Chapter 8) and in the industrial design practice (Chapter 11). In particular, in each technical case study, we were able to offer a level of realism on the results of the methodology, as, the disruptions of strategies of components offered an exhaustive summary of results that could be tested against the reality. Additionally, the industrial design practice that entails a practical instance of design reinforced the viability of work reported in this thesis, giving examples of the use of the outcomes of the proposed methodology in combination with existing design software.

A limitation of the evaluation part for this work is that the data were collected from a single company. However, the data were related to different research methods, different projects (different types of ships) that happened in distinct chronological times (legacy and modern systems). Also, the participants that were involved in the focus group discussion (Chapter 3) were different people that had different positions in the company than the semi-structured interview' participants (Chapter 11). In conclusion, the study has adopted principles for enhancing research validity based on triangulation principles.

2.8 Chapter summary

Chapter 2 presents the research philosophy, strategy and design of the study reported in this thesis. The philosophical assumptions were accepted as pragmatism and the researcher used mixed methods in conducting the research study. To reach the point of developing and evaluating the proposed methodology, a research approach was followed as discussed in Section 2.4. Next Chapters 3 presents the industrial exploration focus group and Chapter 4 the review of the literature.

PART 1: RESEARCH SCOPE

Chapter 3:

Industrial exploration focus group

This Chapter reports the exploration focus group discussion that was elaborated with the industry that aid in finding the research scope and direction. The researcher was based in BAE Systems a company that designs and manufactures naval ships and aimed to position the research within the challenges of the industry. In this regard, the researcher established an inductive, explorative in nature, focus group discussion. The focus group discussion led to the emergence of three key challenges that helped guide the literature review reported in Chapter 4. The explorative and inductive nature of the focus group discussion had a key influence on the direction of the research reported in this thesis.

3.1 Agenda and questions

A qualitative research method, termed focus group discussion, was used that involved participation of the subject matter experts (SMEs) from BAE Systems. The reason for selecting the focus group discussion as a research method for exploration was that it enables the discovery of different opinions, experiences and understanding of a subject by exposing the dynamics between the participants. The development of a complex system is a design activity involving multiple stakeholders, such as designers from different disciplines, the supply chain, production, etc. It was therefore decided that a research method that could help to provide multiple perspectives and not a single view of the subject would be useful.

The focus group discussion method is a type of interview analogous to a joint interview with a panel of experts. The focus group discussion was used as an exploration research instrument for qualitative data collection and analysis. Langford and McDonagh (2003) stated that: “focus groups are group interviews and involve gathering people with knowledge about a specific topic or issue for a relatively informal discussion.” The researcher presented to the participants an agenda for the meeting along with a short PowerPoint initial presentation. The researcher was the moderator, who facilitates the discussion, and led the meeting. In conducting the focus group discussion, Krueger and Casey's (2000) guidelines were followed. The questions asked were open-ended such as what do you think are effects of modularity on cost, cycle time, quality? The questions prompt discussions on participant’s previous experiences, thinking back on previous projects and providing examples. There were different types of questions to trigger the participants to become involved and reflect on their previous design choices and project' outcomes. The participants were asked questions designed to motivate the conversation about the role of system architecture; their perceived effects of modularity; and the design system architecture assessment process. Drawn from their experiences in working for the company, the participants were encouraged to give opinions on the various discussions. The main themes, introduced by the moderator, were revolved around system architecture, attributes of system architecture, modularity effects and the assessment of system architecture. It is noteworthy that these were the only themes raised and that the moderator did not introduce the robustness, survivability, resilience, or vulnerability of systems. The questions were simply formulated and practical examples were discussed. The focus group agenda is shown in below Table 2.

Table 2: Focus group agenda

Focus group agenda	**Location:** **BAE System** **Glasgow**	**DATE:** **15TH August 2017** **1330-1630PM**
1. Welcome **2. Ground Rules** **3. Introductions** **(15 minutes)**	Introduction, of the researcher and the topic. Then rules of focus group discussion and an introduction of the attendee's names and positions.	
4. Group Discussion - Topic 1 **(20 minutes)**	**Topic #1: Introduction** Question 1: What do you think about the system architecture term? Question2: What do you think about the are key attributes of system architecture? Question 3: What do you think about the process of assessment of system architecture? Question 4: What do you think about the effects of system architecture on system development life cycle competitive performance?	
5. Group Discussion - Topic 2 **(40 minutes)**	**Topic #2: Effects of Modularity on Cost, Cycle time, & Quality** Now, we would like to discuss the effect of modularity Question 5: What do you think are the effects of modularity on cost, cycle time, quality?	
6. Group Discussion - Topic 3 **(50 minutes)**	**Topic #3: Effects of Attributes** Question 6: What are the attributes of the system architecture you think are most influential? Question 7: What do you think about the interplay of other attributes on modularity?	
7. Group Discussion - Topic 4 **(40minutes)**	**Topic #4: System Architecting Assessment** Question 8: What do you think are the pros and cons of system architecting? Question 9: What do you think regarding analysis during the system architectures process? Question10: What objectives a system architecting evaluation process should be focused on?	
8. Final thoughts **(10 minutes)**	Question 11: Does anyone have any final thoughts that they have not yet expressed?	
9. Review and Wrap-up **(5 minutes)**	FINALE	

3.2 Location and participants

The focus group discussion constitutes the views of seven subject matter experts. The meeting took place on 15 August 2017 in the premises of the BAE Systems in the Glasgow offices.

The participants are stakeholders of a complex engineering system's development project. The researcher pre-selected the participants who shared experiences related to system engineering, and that were influential to the system development process. They included the Head of System Engineering, the Design Authority for Electrical and Power systems, and the Platform Technical Lead. These three participants were involved in the focus group discussion since they were influential in the design and decision making of the company's complex engineering systems. Their opinions on the complex engineering systems provided significant input to the research study. Moreover, the participants from the company's Research and Technology department were, also, in attendance since they were involved in international research programs such as, for example, NATO's investigations of modularity concepts. Also, the Research and Technology Manager attended the focus group discussion is the company's expertise concerning survivability and vulnerability design. In conclusion, the attendees of the focus group discussion (Table 3) provided a significant pool of knowledge and their fruitful debates helped to provide the researcher with different viewpoints.

Table 3: List of participant's positions

Participants names	Position (on 15th of August 2017)
MG	Type 26 Platform Technical Lead
NH	Senior Technologist Research and Technology
AM	Head of Systems Engineering BAE Naval Ship
MR	Engineering Manager of Research & Technology
DL	Senior Principal Engineer Research & Development Afloat Capability
GR	Delegated Design Authority T45 Electrical and Power Management Systems
CV	Principal Engineer Research & Development Readiness & Sustainability

A scanned signed copy of the participant's attendance list is included in Appendix I of this thesis.

3.3 Industrial challenges

In order to analyse the focus group discussion, the researcher's first steps were to transcribe the discussion from the voice recorder. The framework for the analysis of the focus group discussion follows key steps discussed in the literature (Rabiee, 2004) as discussed in Section 2.6.2. Firstly, the most noteworthy quotes "well said statements that illustrate an important point of view" (Krueger and Casey, 2000) were identified in the transcription. These notable quotes were included in an initial summary quotes Table. Then, the main opinions and ideas identified from the questions were qualitatively derived. For each part of the discussion, the main ideas that emerged from the participants' answers were recognised. It is noteworthy that the researcher ignored some comments and discussions that were not directly relevant to the research study. Next, the notable quotes were re-categorised under the new developed main themes. This was an iterative process that allowed to narrow on the three challenges that are following discussed. In addition, it should be noted that additional challenges have been identified during the focus group discussion which has not been further investigated in the study reported in this thesis. For example, the complexity of the systems that were designed and developed was cited as a challenge, but the aim and objectives of this study do not explicitly address complexity. Consequently, the challenges discussed in this Section are those that are intended to be addressed in this study.

The main challenge, emphasised in the discussion, was the company's purpose is to design survivable design, with minimum design vulnerabilities through redundancy. Even so, the only property discussed by the moderator (researcher) was modularity, the participants repeatedly brought to the discussion terms such as resilience, vulnerability and robustness of systems. The other recurring idea was that modularity was not perceived by the participants as an entirely positive attribute and that the implementation of modularity in the previous project had experience challenges to materialise competitive benefits. These two basic ideas, regarding the importance to design robust systems and the challenges with modularity, recurred through various questions and discussions. A narrative of the discussion is that, during the design of such a system, there are conflicting requirements; trade-offs are necessary, and the main driver of the design of naval ships is to be robust. The discussion moved away from modularity and its potential benefits, even though that was the starting point. The focus group discussion enlightened the researcher to position modularity amongst many other attributes of design that need to be pursued. A third challenge discussed was the subjectivity and over influence on opinions and previous designs during the conceptual stage. In previous and current projects, the approach to assessment is through workshops that are meetings of experts. Following sections discuss the findings from the focus group discussion and outlines the three emerged challenges.

3.3.1 Emerged challenge 1: Importance of system architecture design

Table 4 contains the participants' quotations that indicate how they perceive the system architecture assessment process, and what is the current practice. The Head of System Engineering highlighted that quantitative approaches within the system engineering domain are not used and that the main approaches are qualitative. The discussion shows that the system architecture process is significantly influenced by previous designs and by the subjective based opinions of stakeholders. It is stressed that there is not an explorative and quantitative analytical approach used to study the system architecture at a high-level of abstraction.

Table 4: Focus group quotes on system architecture approaches

	Assessment Approach to System Architecture
1	NH: can we declare that we are assessing the system architecture, without having this explicit model of describing the system architecture. Historically we do our assessment in a more in social and judgement-based engineering interaction.
2	AM: simple model, which a company used, is the system to design, the system of interest and its environment in which operation, we spend a lot of time to look at the systems of interest, and we spend little time to see the system we system design as a whole.
3	MR: There is an obsession in previous designs, that may not in truth have been the most successfully, the design starting point is these previous designs…. our new design are evolutions of previous…
4	CW: In early stages that are not many detail information to perform analysis….

Comment (1) indicates the social and judgement-based engineering interactions of a typical assessment process in the company. Typically, in the company the system engineering department hosts workshops with domain experts to discuss structural and behavioural diagrams developed in Sparx EA software that is based on Systems Modelling Language (SysML). The expert's knowledge and previous experience are valuable in the early design conceptual decision making, however, there is the impression that there is a need for the expert's decision to be supported by systematic and analytical based approaches, to avoid biases and introduce traceability and evidence-based reasoning to the design decisions taken in the early stages. Comment (2) highlights the attention that is typically given to subsystems and not to the system as a whole. Comment (3) identifies there is a habit to start a new design based on previous existing designs while Comment (4) argues that this is because there is not much information available, thus previous designs provide a foundation to start developing a new concept. It was evident that the process of designing new concepts was important for the company however typically was predominantly based on the previous design. The challenge which emerges from the participants ' statements was that approaches to help system architects during the initial system design phases were needed.

3.3.2 Emerged challenge 2: Finding the right level of modularity

As shown by the participants' quotations in Table 5, in general, the dynamics of their interactions indicate that modularity is not perceived to be in all cases and situations beneficial. The participants have experienced challenges in implementing modularity in practice and realising the claimed benefits during the design and build of the naval engineering systems.

Table 5: Focus group quotes on an optimum level of modularity

	Optimality of modularity
1	AM: There is a requirement for balance, increasing modularity reduce flexible, which kind of flexibility? The flexibility of design! Robustness by which meet the requirements.
2	AM: Compartment has a function in a ship, so if you consider modularity within a ship, you need to consider building blocks also there is functionality that contributes to ship/platform performance, but it is different than plug and play weapon systems modularity which is brilliant/but different.
3	DL: Modularity let say you buy a generator from a supplier which is a module…but is not that you could get to take out/and replace that generator so easy even is a module. There are still many interfaces interplay that makes that difficult.
4	GR: There is an interesting example whereas modularity fails…. Supplier A….is supplying with a propulsion system for Type B part required UPS we said you should supply UPS, they said no…why not? Because they always fail…and then we could always pin the failure to UPS… because they do not want to take the responsibility of the UPS. After all, then they can claim the failure to UPS and do not get any responsibility from their equipment, so we cannot get them to supply a total package….
5	AM: it comes down to us plays into to what we as a business… you can increase modularity if you accept, we are a systems integrator, or are we shipbuilder with integration problems? Simplistic model, multidimensional to all of that…

Comment (1) suggests that a modular design does not allow revisions during the more detail design. AM suggests that in his experience modularity limits the flexibility in the design. Comment (2) argues the need for the functional allocation to modules, implying that there is a need for successful identification of function-based modules. However, the previous modularisation approaches adopted in the company, focus on modularising during the construction stage, thus failing to successfully modularise the design. Comment (3) highlights the contractual disputes and the module integration challenges introduced. GR noted that developing modules requires a supplier's willingness and cooperation. To establish modular contracts there is a need, very early in the initial system design stage, to develop a basic modular architecture to be able to inform the decisions taken during assignation of contracts to suppliers. However, in the previous project, the concept of modularity was not purist during the contractual stages, thus the suppliers were not aware that their components were allocated in specific modules and not willing to deal with problems arising during to the efforts to modularise the design. Also, GR exemplifies the reliability issues facing the integration of modules. There is a problem to identify the stakeholder responsible in resolving errors at the integration level, given that such errors may be related to faults within the modules. For example, if a failure occurs within the module during the integration phase the integrator may not be able to recognise and identify the error within the module. This will increase the amount of time to resolve the overall problem. It may require the supplier that provided the module to attend the integration location, as there may be a lack of knowledge on the part of the integrators, to deal with the problems. The arguments posited by the participants construct the proposition that modularity is not an all-around positive principle in engineering systems specifically that have a complex integration construction phase. The emerged challenged was to find the right level of modularity that corresponds to a respective design that is sufficient to realise advantages, and good enough not to introduce additional problems.

3.3.3 Emerged challenge 3: Balancing modularity and robustness

Highlighted in grey, in Table 6 below, are some of the discussed arguments, which demonstrate that robustness (also relates to survivability that is a common term in naval design) is the main driver of the design of the complex systems. Modularity is positioned as a desirable attributed if it brings benefits. However, robustness is the most important requirement of the design. In the study reported in this thesis, survivability is not discussed directly, as it is a naval specific term. Robustness is the term that is adopted, accepting that robustness relates to survivability, as designing a robust system is a precondition of designing a survivable system. During the discussions, the terms survivability, vulnerability, resilience, and robustness were found to be frequently mentioned by participants.

Comment (1) is an important statement that has contributed to identifying this research direction and focus. The modularity and robustness trade-offs had arisen during the focus group discussion. A modular design entails the grouping of components into modules, the architect needs to consider how this affects the robustness of the system. It was observed that an important requirement of the design is to view and examine aspects of the design together, because they may be conflicting. Comment (2) motivated the researcher to conceive the robust modular term. AM suggested that the ideal modularity should be based on robustness. To achieve the architecture should be developed based on modules that can be removed without impeding the functional continuity of the whole system. Comment (5) elaborates on a balance design AD states “almost like the graphic equaliser type of view when you are pushing up you are influencing something further down”. In short, the third challenge emerged was that modularity and robustness are two attributes need to be balanced in the design of engineering systems. More significantly, the main attribute of the system was the robustness and whereas modularity was perceived advantageous if it could bring competitive benefits. This challenge was important in establishing the research scope and direction of study. The robust modular concept envision was to be able to remove a module from the system, without functional discontinuity. This idea arose from the explorative focus group, inform the development of the methodology.

Table 6: Focus group quotes on a balance of modularity and robustness

	Robustness (resilience, survivability, robustness) and its relationship with modularity
1	DL: "Taking MG's view in simplifying things, if everything it was very dense all in one module but then the survivability could be reduced. You could then take this down by two or three".
2	AM: "Partly this is simply is about resilience if you could walk out without the module and pull something out the bag and keep your system one going with what remains available then yes you have consequently made simpler the system and more robust".
3	MG: "I guess modularity comes to break down complexity so that the whole ship is extremely complex, but you then modularise one element of it and take that module away then the focus is on what works and then bring it back and plug it in"
4	AD: we try to make modules so you end up with many interfaces, focus on the package to be more resilience…the subsystem itself is more resilience…start working… from down… you introduce more interfaces start increasing your integration phase, so the increasing interfaces lower resilience…
5	AD: We have reliability and maintainability requirements. We could apply those to a supplier and there has a property in the system that goes to it how modular you can make something you know, and commonality is but one part of that. What is your prime focus of the design is its resilience then you would have question modularity you know you would have to reduce the level modularity to increase resilience and so again what is the context in which are designing your product. What is your prime driver or design driver and, so you have understood at what point these come into play and how do fit within the design life cycle? [……] almost like the graphic equaliser type of view when you pushing up you are influencing something further down, and each one of those attributes will go to influence you to know what you are looking at you know when you're looking at centrally in particularly terms of resilience you don't what quite centrally you want distributed networks.

3.4 Key findings

The level of the participants' expertise in complex engineering system and development projects and the fact that this focus group discussion's primary purposes were to explore and gain insights of the subject, justified the researcher's use of this research method. The interactions and dynamics amongst the participants and the fruitful discussion helped the researcher to direct and, find the research scope.

The findings of the focus group bring to light new aspects of the system architecture design problems. The first aspect is the reluctance to view modularity as an all-inclusive positive feature of the complex engineering system. Regarding the nature of the company's design systems, there have been challenges to implementing modularity successfully especially during the integration phase of the development project. Although modularity is desirable, there is deduced to be an appropriate right level that requires more careful consideration given that the system is designed with other more important requirements and the expensive integration and verification stage such systems experience.

In designing these complex systems, the important driver is their robustness. In the focus group discussion, the participants used terms such as survivability, resilience, and vulnerability. Redundancy is the main enabler of robustness in the design of such complex systems. From that viewpoint, the researched deduces that robustness is a critical aspect that must be included in the research study. The focus group discussion raises, also, the subject of the design of system architecture. The findings suggest that the company’s decisions on the system architectures are influenced greatly by the previous designs and subjective judgements rather objective and systematic methods. Moreover, a highlighted point is the lack of analysing the system at an abstract and high-level since the detailed analysis happens mainly at sub-system levels and by domain experts (i.e. naval architects or electrical engineers). Also, the existing process introduces persistence to using previous designs rather a wider exploration of more potential solutions. Consolidation of the design problems identified from the focus group discussion is that:

- Modularity may conflict with other essential requirements.
- Robustness is an essential requirement of design complex systems.
- The assessment of approaches to system architecture requires more attention since such assessments are sometimes subjective; focused on subsystems rather than the whole abstract system architecture; and are influenced greatly by previous knowledge and designs.

3.5 Chapter summary

Chapter 3 provided an exploration focus group discussion to help identify the direction and focus of research. Three challenges were identified: the need for systematic approaches, finding the right modularity, balancing modularity, and robustness. Next, Chapter 4 presents a literature review (Chapter 4) on system design, robustness, and modularity, which leads to the development of the research gap and provides background for the development of the methodology reported in this thesis.

Chapter 4: Literature review

This chapter presents a literature review that addresses the main challenges raised in the focus group discussion. The chapter provides an overview of the attributes of design methodologies, the conceptual stage and the definitions and assessment of system architectures. This is followed by a literature review on the robustness of the system design and modularity effects on competitive indicators, modular design methods and metrics. The relationship between modularity and robustness is discussed. These reviews lead to the identification of the research gap and inform the requirements for the development of the methodology.

4.1 Approach to literature review

The literature review, reported in this thesis, happened in two chronological stages. Firstly, an initial exploration literature review on modularity and system architecture was conducted that informed the focus group discussion. After the focus group discussion, a literature review was performed that included system design, robustness and robustness and modularity relationship. In this thesis, the literature review is framed based on the main challenges emanating from the analysis of the focus group discussion. This Section combines the findings of the literature reviews conducted before and after the focus group. Sections 4.2, 4.3 and 4.5 is based on findings from the review performed after the focus group discussion. Section 4.4 is based on findings from the initial literature review and was updated after the focus group emerged challenges (Section 3.3.3 and 3.3.3).

The objectives of this study, the narrative literature review, is to recognise the prominent and recent research works in the different research streams, to identify the intersections of these streams, leading to the establishment the research gap. The researcher adopted a narrative integrative approach to the literature review (Cooper, 1989). Grant and Booth (2009) explained that “a literature review involves some process for identifying materials for potential inclusion— whether or not requiring a formal literature search—for selecting included materials, for synthesizing them in textual, tabular or graphical form and for making some analysis of their contribution or value”. The benefit of a literature review is that attempts to recognise what has already been achieved, to refine, expand on previous works, to sum up, to avoid repetition and to identify the research gap. Furthermore, the disadvantages of a literature review are that “conclusions they may reach are therefore open to bias from the potential to omit, perhaps inadvertently, significant sections of the literature” (Grant and Booth, 2009).

The literature data was initially found through the SCOPUS database. The inclusive limits of the search were bounded by engineering design, system engineering, engineering systems. The initial literature review strategy on the terms of modularity definitions, benefits, metrics, and methods. The findings were reviewed empirically, the focus was given in the highly cited and newest papers. Thereafter, based on the study's bibliography, authors' names, and the "related" articles, they conducted a citation search and retrieving citation (backward and forward search). Iteration of this process of searching the literature, until adequate saturation of the prevalence of a collection of research works, provided the researcher with sufficient overview to inform the organisation of the focus group discussion with the experts.

Using the same narrative approach, the literature review was performed after the focus group based on the emerged main three challenges for focus group discussion (Section 3.3.1-3.3.3). This stage of the literature review addressed robustness as a new theme emerging from the focus group (Section 3.3.3) from a system design viewpoint, robustness relationship with modularity. Additionally, a review on system architecture design, and an overview of the conceptual design stage, and the characteristics of design methodology was included. As shown in Figure 4, the literature review is positioned at the intersection of the literature of system architecture design, modularity, robustness, and network science.

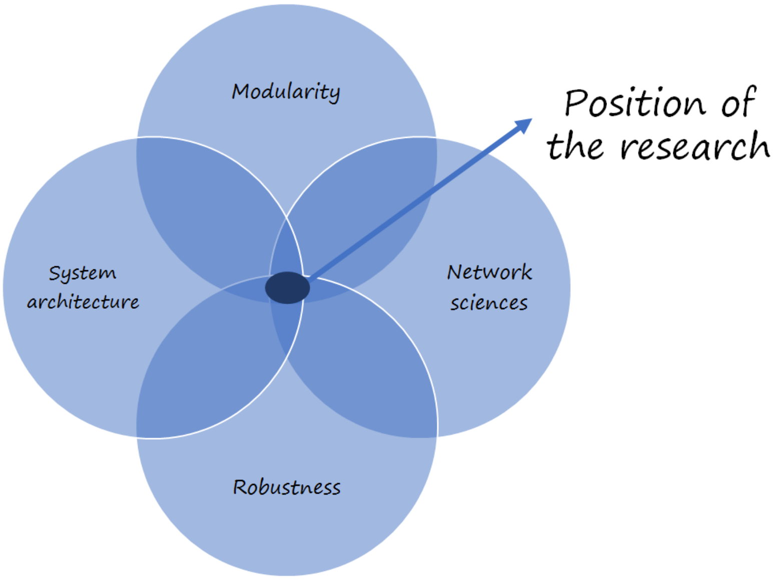


Figure 4: Position of the research in the literature

The outcome of the literature review defines the research gap and establishes the requirement for the development of the methodology. The following section of the literature review is divided into three parts firstly the literature on system design is presented, then robustness and modularity are discussed. The literature review concludes with the research gap and the requirements for the methodology.

4.2 System design

This section discusses the literature on attributes of design methodologies, the conceptual stages, the system architecture design, the phenomenon of design fixation and the DSM and network-based modelling, analysis, and generation methods.

4.2.1 Design methodologies

Design methodologies are the foundation of the domain of design science. Design methodology "is prescriptive as it indicates how to do design" (Evbuomwan et al., 1996). Roozenburg and Eekels (1995) emphasised that the limited ability of human decision-making to comprehend complex problems drives the requirements of formalised and structured methods for the design decision process. Design methodology should "foster and guide the abilities of designers, encourage creativity, and at the same time drive home the need for objective evaluation of the results" and these systematic procedures simple "try to steer the efforts of designers from the unconscious into conscious and more purposeful paths" (Pahl and Beitz, 1996). ISO/IEC/IEEE (2017) definition is that design methodology is a "systematic approach to creating a design consisting of the ordered application of a specific collection of tools, techniques, and guidelines".

Design methodologies include "plans of action to link working steps and design phases according to content and organisation; strategies, rules and principles to achieve general and specific goals; and methods to solve individual design problems or partial tasks" (Pahl and Beitz, 1996). As they described design methodology requirements include the following: a problem-directed approach; facilitating the optimum search; being compatible with concepts, methods and findings from other disciplines; not responding to finding a solution by chance; facilitating the application of known solutions to related tasks; being compatible with electronic data processing; being easily taught and learned; and reducing workload, saving time and preventing human error. Evbuomwan et al. (1996) proposed ten desirable features for design methodologies. They should be explorative, rational, investigative, creative, opportunistic, incremental, decision-making, iterative, trans-disciplinary, and interactive. In specific relation to concept selection methods, King and Sivaloganathan (1999) indicated four desirable requirements: systematic nature; multiple attributes; coupled decisions; and simplicity and user-friendliness. Ullman (2002) suggested that "an ideal decision support tool should manage incomplete alternatives and criteria generation and allow their addition throughout the decision-making". Therefore, a design methodology to achieve the purpose of aiding in decision-making should be systematic, explorative, investigative, iterative, multi-attribute, user-friendly and time efficient. A design methodology is also a "prerequisite for flexible and continuous computer support of the design process" (Pahl and Beitz, 1996). The development of design methodologies can provide a systematic approach to help the architect design system architecture intentionally during the initial phases and provide the basis for the development of computer-based design support tools.

4.2.2 Conceptual stage

Conceptual design in the engineering design domain is "the process by which ideas are generated or configurations are created or selected to meet the specifications and constraints of an identified technological need" (Jansson and Smith, 1991). In conceptual design, designers are "concerned with establishing the mode of action (way of working) of the technical system and the morphological structure (component structure)" (Andersson, 1997). Wynn and Clarkson (2018) advised that many scholars recommend certain general stages during the conceptual design. They discovered that scholars typically prescribe three major stages: analysis, synthesis, and evaluation. The analysis includes concentrating on a problem and structuring it into goals. Synthesis includes generating various candidate solutions. The evaluation includes a critical assessment of those alternatives against goals, rational selection between them, and/or driving iterative enhancement.

The generation or synthesis phases is whereas alternative structural solutions are developed. Gero and Kannengiesser (2004) proposed a Function-Behaviour-Structure (FBS) framework that states that structure drives behaviour that satisfies function. In that respect, a synthesis process converts the anticipated behaviour into a solution structure (S) that displays the required behaviour. Roozenburg and Eekels (1995) explained that a design decision-making process entails alternatives that can allow the evaluation and selection of alternative options. The choices are different possible solutions that are developed and considered during the process. Such options can be either different variations of the same solution or entirely distinct solutions. The reason for having some alternatives to evaluate is that these options lead to distinctive results. The decision-maker prefers specific results; therefore, the evaluation process aims to match the correct solution to the desired results. Pahl and Beitz (1996) also discussed selection and evaluations methods: "for the systematic approach, the solution field should be as wide as possible". A large number of options can be found if all the potential criteria and characteristics are taken into consideration. The large number of solutions considered in a systematic approach as both positive and perhaps negative. It is recommended that technically appropriate, yet functionally impossible alternatives should be dismissed the soonest.

The analysis and evaluation stage are concerned with assessing alternative solutions. Gero and Kannengiesser (2004) explained that the analysis process derives real behaviour (Bs) from the synthesised structure (S). The evaluation process contrasts the structural driven behaviour (Bs) with the anticipated behaviour to guide the acceptable choices for the design solution. In this study context, the solution structure (S) reflects on the synthesis (generation) of the system architecture and the analysis is the assessment of the structural driven behaviour and the evaluation on the comparison between the expected behaviour and the structural driven behaviour. Roozenburg and Eekels (1995) explained that the evaluation process is different from the decision-making process. The evaluation stage includes the collection and comparison of alternative options and, in order to ensure the objectivity of the process, aims to do that in a systematic and structured manner. Pahl and Beitz (1996) stated that the evaluation and decision making are vital phases of product development owing to their impact on the cost, quality, and end-product performance of all the subsequent stages. The significance of these stages is self-evident as a poor choice of design concept can rarely be compensated for at subsequent design phases and entails a high redesign cost. Moreover, Okudan and Tauhid (2008) argued that the emphasis in conceptual selection methods is placed on function rather than form (structure): “most methods focus on the function of the design concept rather than form”. They advise that there is a need for more structure-oriented conceptual selection methods. The review of the literature identifies the key stages of conceptual design: generation, analysis and evaluation and the importance of generating alternative structural solutions that in this study is equal to alterative system architecture options.

4.2.3 System architecture design

The development of system architecture is expected to take place at the initial stage of system design, moving from the needs of stakeholders to the definition of requirements and the generation of options for system architecture, and before the final selection and definition of design (de Weck, 2015). Crawley et al. (2015) explained that concept is “notional mapping between function and instruments of form, whereas architecture is a fairly comprehensive description of the relationships between internal functions and instruments of the form” and explained that architectural decisions “relate to form-function mapping, they determine the performance envelope, the encode the key trade-offs in the eventual product, and they often strongly determine cost”. Although the system architecture is more concrete than the concept, it is suggested that it should take place during the initial stages before the selection and detail definition of the design. Decisions on architecture have the greatest impact on the success of the system.

The architecture definition process translates the specific needs of stakeholders into critical characteristics that, as a result, drive architectural decisions, including architecture assessment. (IEEE and ISO/IEC, 2015). As INCOSE (2015) states “nonfunctional requirements and/or architectural characterises are used as criteria to analyse, assess, and select”. In alignment, Elias and Jain (2017) suggested that assessing system architecture is grounded in determining a system’s architectural attributes. The authors proposed a qualitative explorative framework to assess system architecture, which groups the attributes into six categories aligned with generic system goals. One of the categories is the architectural design attributes that describe the nature of the overall structure and the underlying concepts of the system architecture. Attributes, such as modularity, coupling, encapsulation and interconnection are included within their category of architecture design attributes. Quality attributes, such as robustness, survivability and dependability, are also relevant from a design perspective. Such quality attributes, which are defined as the “extent that the instantiated systems as architected”, can “correctly operate in stressful conditions”, “accomplish its intended purpose” and “avoid failures” in different conditions and environments (Elias and Jain, 2017). Sinha (2014) also developed a structural complexity quantitative metric, highlighting the need for early quantification metrics to support objective decision making, and stated “for architecture evaluation and optimization ... lends objectivity to the process of system architecture selection and design”. This means that key attributes act as assessment criteria during the system architecture design.

INCOSE (2015) states that “the goal of the architecture definition process is to provide the “best” possible architecture made of suitable system elements and interfaces” and continues that the “preferred way to do this is by producing several candidates’ architectures; analysing, assessing, and comparing them; and then selecting the most suitable one”. Moullec et al. (2016) clarified that the system architecture design involves the generation and comparison of alternative solutions in terms of performance, compromises, effects, and risks. During the system architecture assessment, all fields must be considered simultaneously, and gains and losses must be traded against each other. The ideas and their fundamental architectures which will most probably offer the best possible trade-offs must thus be identified at the conceptual stage. This choice generally takes place with a restricted number of criteria, deriving either from the system requirements or business goals (Moullec et al., 2016). This involves methods of modelling and analysis that regard system architecture as a whole at a high-level of abstraction. Wyatt et al. (2012) explained that the understanding of possible solution architectures allows the generation of specific solutions. The architectural choices of system attributes and patterns generate effects through the life of system development. Selva et al. (2016) developed a pattern-based approach to support the system architectural decision-making by presenting six patterns that appear in the structure of architectural decisions. The patterns were descriptively and prescriptively discussed providing models, examples, algorithms and heuristics to support the architect's decision-making process. Architecture rationale documents clarifications, justification or reasoning of architecture decisions taken. "The rationale for a decision can include the basis for a decision, alternatives and trade-offs considered, potential consequences of the decision and citations to sources of additional information" (IEEE and ISO/IEC, 2011). Chen and Crilly (2016) highlighted the need for novel approaches arguing that "when a large number of architectural configurations are possible, being able to simulate them and analyse the functional implications of certain family groupings would provide more solid justification for making architectural decisions". The generation and evaluation of different and alternative options facilitate the development of logical and unbiased architectural rationale to support decision-making throughout the design of the system architecture.

4.2.4 Design fixation phenomenon

The phenomenon of design fixation reported in the literature relates to the findings of the focus group on the repetition of previous designs. It also relates to the lack of exploration of alternative potential solutions in new ways that do not limit architects and engineers to thinking in traditional and repetitive ways. This section provides literature, without attempting to examine in depth the creative nature of the problem. The term design fixation encapsulates situations that "limit their creative output because of an overreliance on features of pre-existing designs, or more generally, an overreliance on a specific body of knowledge directly associated with a problem" (Youmans and Arciszewski, 2014). The phenomenon of design fixation has been identified in the literature as having different aspects, the fidelity of the representation of stimuli and the disciplinary background of engineers are inferred to be relevant to the industrial challenge 1 referred to in Section 3.3.1. For example, Cardoso and Badke-Schaub (2011) found that "exposure to a line-drawing illustration resulted in potentially newer concepts, while a photographic representation seemed to have led participants to develop less original ideas". Thus, abstract or low-fidelity models allow for more original ideas than concrete or high-fidelity models during the initial design stage. Inspiration and fixation also relate to people's previous experiences and disciplinary backgrounds. Agogué et al. (2014) suggested the context of a person may affect the depth or width of solution space exploration. Typically, engineers offered more complex and detailed approaches to the working process behind the concept. Prior knowledge and expertise of individual engineers and architects may lead to a fixation on the previous idea. Crilly (2019) pointed out that more case studies are needed to research design fixation to understand what is or may occur in "real world" design work. Identifying the design fixation phenomenon in literature in conjunction with the findings of the explorative focus group discussion (Chapter 3) has also informed the need to support architects to analysis and explore alternative system architecture options during the initial phase of system design.

4.2.5 DSM and network-based modelling and analysis

DSM modelling approaches are widely applied in engineering systems to represent the structures of products and systems architectures and are valuable for modelling and analysis purposes (Browning, 2015; Eppinger and Browning, 2012). The DSM is simple to understand as a square matrix illustrating the relationships between matching elements positioned in the rows and columns of the matrix and offers an efficient and effective way to model and analyse systems. The relationships included in a component-based DSM relate with design dependencies such as spatial, energy, material and information (Pimmler and Eppinger, 1994), and can be symmetrical or asymmetrical (Sosa et al., 2003). The majority of component-based DSMs are built based on the symmetric premise (non-directional) since the configuration of symmetric DSMs is simpler (Helmer et al., 2010). Most DSM analysis studies to date have focussed on clustering components for modular architecture determination (Browning, 2015). Researchers have also employed product DSMs for determining architecture patterns such as modules and cycles (Fixson, 2005; Sharman and Yassine, 2004; Sosa et al., 2013). DSM is a useful way to represent a network, facilitating the application of network-based approaches (Carliss Baldwin et al., 2014; Sarkar et al., 2014; Sosa et al., 2007).

Network theory is a mathematical field of studying networks represented as the nodes (vertices) actors or elements and edges (arcs) as the connections between the actors or elements. It builds on graph theory of the study of graphs originated by Leonhard Euler. There are the past and emerging literature that is suggesting methodologies in relation with classification of system architectures and design changes propagation based on network theory (Giffin et al., 2009; Pasqual and Weck, 2011; Baldwin et al., 2014; Dong et al., 2016). Moreover, the network theory establishment combined with its mathematical background has reinforced the use of network metrics as quantifiable apparatus for measuring system architectural properties. Even so, has been recommended, “researchers should continue to draw upon the advances in closely-related areas such as graph theory, network analysis, complexity, and other types of architectural models” (Browning, 2015).

Parraguez and Maier (2016) provided an overview of the challenges and key decisions when using network science to support design research. They advised that the first decision is to define the overall research purpose (explorative or explanatory) and second to select relevant network features. A pivotal work in the literature of engineering design (Braha and Bar-Yam, 2004) applied network-theory approaches to study the topology of large-scale engineering problems. Other works applying network theory include Braha and Bar-Yam (2007, 2004), which examined product-development networks and contributed significantly to introducing network-science approaches in the engineering design domain.

The use of network science approaches in the engineering design domain has been eminent in recent years. Given the increasing challenges and complexity of modern engineering systems, network modelling and analysis is recommended in the design research (Chen et al., 2018). Its application is an area of research that is continuously expanding (Baldwin et al., 2014; Braha and Bar-Yam, 2007, 2004a; Parraguez et al., 2015; Piccolo et al., 2018) contributing on the development of new solutions that address engineering design problems. The network science literature offers strong, effective analytical algorithms that can aid in the development of the solution. Walsh et al., (2018) explained also that “The reason for the success of network analysis is that it relies on a simple representation using nodes and edges and includes a number of powerful, efficient algorithms for analysis”. Parraguez and Maier (2016) advised that there are “strong evidence for the usefulness and benefits of network science to support design research and increasing uptake in engineering design studies”.

Network science requires low technical information accuracy, which is good for the initial design stages. Rigterink (2014) claimed that low fidelity is important because the preliminary design process is iterative. This is because, if many potential designs are created, it is time-consuming to create detailed models for all the alternative potential solutions and, if there are time and resources, there may be insufficient information to support the development of detailed models due to the early stages.

However, alongside compelling reasons to conduct network-based analyses, there are also conceptual and methodological challenges. Limitations of the network-based approaches relate to the assumption of equality among nodes, and the inference of robustness and vulnerability from the connectivity of the system regardless of the physical laws governing the different components. In conclusion, network science provides tools and mathematical foundations that can be used to support system architecture design that can support the development of systematic approaches.

4.2.6 Network-based generation methods

Methods for generating system architecture options in engineering systems, that are network-based, have been proposed in the field of engineering research. Wyatt et al. (2012) proposed a computational support method for generating new concepts through design synthesis with network structure constraints. The method establishes a set of solution architectures using four network structure constraints and employs a synthesis algorithm to form the space of architecture through four operations: removing or adding a component or connection. The generated architectures are then graphically represented and evaluated through multiple quantitative graph-based metrics. The limitations of this approach are that the designer may over-constrain the design problem, thus limiting wider and more counterintuitive space design exploration.

Moullec et al. (2013) suggested generating system architecture using Bayesian Networks (BN). The Bayesian network is a direct acyclic graph consisting of nodes representing variables, and the edges are dependent. Product architecture modelling is represented by different nodes: decision node (represents all possible components), characteristic node (represents design parameters and are linked to decision nodes), performance node, constraint nodes (represents compatibility constraints) and global confidence node (overall degree of designer confidence in the architecture solution). The design problem of product architecture is modelled in this way. The generation and exploration phase of the product architecture includes the analysis of the constraint and decision variables leading to the calculation of the global confidence node, which is then compared to a predefined threshold. If the global confidence node exceeds the threshold, all decision variable nodes will find a design solution. This method's unit of analysis is not the architecture of the system but concerns the design problem. A limitation of this approach is that does not consider possible trade-offs within the system architectures.

De Vos and Stapersma (2018) have developed an automatic topology generator for the early design of on-board energy distribution by combing network theory and marine heuristics engineering. The method includes a robustness metric that evaluates the reconfigurability of the system and a “claim” objective that assess weight, space and cost requirements The limitation of this method is that is specific to the naval distribution system, the advantage is that the method includes two specific metrics, allowing trade-off analysis to be performed and visualising the outcomes (optimal design solutions) in a Pareto front.

In the extended network science literature, network generation models allow simulations and experimentations, aiding understanding and analysis of the properties of the networks, by enabling the population of alternative topologies for evaluation and comparison purposes. Chakrabarti and Faloutsos (2006) explained that graph generators allow “to create synthetic graphs which can then be used, for instance, for simulation studies” and these synthetic graphs are realistic when they mimic all or at least several patterns found in real-world graphs. In short, network tools allows evaluation of new network designs and the study network resilience, survivability and disruption tolerance (Çetinkaya et al., 2011; Jin et al., 2000; Sterbenz et al., 2013).

Chakrabarti and Faloutsos (2006) classified network tool to five categories: random graph generators (linking nodes using random probabilities), preferential attachment generators (linking nodes with preference to nodes with many edges), geographical models, optimization-based generators (minimize risk under limited resources), and internet-specific generator (tailor for special internet features, for example, hierarchical structure).

The first category is the random graph generator example is the Erdos-Renyi random graph model. The second category is the degree-based or preferential attachment generators model the power or other degree distribution of real-world systems. Albert and Barabási (2002) proposed a preferential attachment network algorithm that generates scale-free networks that represent various real-world system properties which are based on growth (the model begins to generate small networks and adds nodes and edges over time, increasing the size of the network) and preferential attachment (the likelihood of connectivity to a node relates to the degree of the node, i.e. a more highly connected node is the more likely to receive new connections). However, the preferential attachment generators are applicable for generating network bigger than 1000 nodes in size, and in smaller networks (less than 100 nodes) that are not appropriate. Tangmunarunkit et al. (2002) stated “power-law distribution is almost meaningless if the number of nodes is small. With only a few nodes, it is unlikely that the degree distribution will be able to create the implicit hierarchy necessary for modelling networks”.

The third category is the geographical model generator is and an example is the small-word model (Watts and Strogatz, 1998) that generates synthetic network that has a high clustering coefficient and low diameter. Scholar in the engineering design community has claimed that engineering design processes exhibited properties of the small-word networks (Braha and Bar-Yam, 2004a; Piccolo et al., 2018).

Example in the fourth category is the highly optimized tolerance (HOT) model that is an optimization-based generator developed by Carlson and Doyle (1999), that generates power-law behaviour systems which incorporate global optimization. The HOT model is intended to “reflect systems designed for high performance in an uncertain environment and operated at densities well above a standard critical point” (Carlson and Doyle, 1999). Lastly, the fifth category is the internet specific or structural generator create networks with a deliberate structure (Doar, 1996) specifically exploiting the hierarchical structure of the internet. These specific structural generators attempt to match the hierarchical structure however the degree distributions of the generated graphs need not be power laws. In that respect, in the realm of engineering systems that have sizes smaller than the internet topology, and are not always exhibit power laws (Paparistodimou et al., 2020), structural generators may be more appropriate.

4.3 Robustness

System engineering literature points out that the primary responsibility of the system engineer is to develop an elegant design that produces the intended result and “is both robust and efficient, and generates a minimum of unintended consequences” (Griffin, 2010). In the following paragraphs, section 4.3.1 discusses the literature on robust design methods and metrics. Sections 4.3.2 provide a review of the robustness of system architecture methods and metrics. Section 4.3.3 discusses robustness-related attributes and methods. Finally, Section 4.3.4 presents literature on redundancy as it is a key approach for achieving robustness in the design of the system.

4.3.1 Robust design methods and metrics for concept design

Robust design in the detailed design stage is most often used to increase the efficiency of an existing product or solution (Ulrich and Eppinger, 1997). The robust design uses parameters and tolerance, together with statistical analysis, to reduce the sensitivity of the product’s performance sensitivity (measured by a response variable) to noise factors. Control, noise and signal factors are categorised in a robust design as factors that impact the product’s performance (Taguchi, 1987). The hypothesis is that the factors that work with a process/product quality features are both controllable (control variables) and uncontrollable/difficult to monitor (noise variables). The aim is to select the control variables to optimise a specified quality feature and minimize the noise variability imposed on the method (Robinson et al., 2004). Although these methods are widely proven to be valuable, they target more detailed stages of the design of the product and process, where technical information about the parameters is known; they are not applicable for the initial stages of design. Moreover, as Andersson (1997) stated they “cannot be made independent of system design. System design and above all, conceptual design, sets the stage for parameter design. The inherent robustness of a design brought forward for parameter design determines the attainable level of robustness that can be reached on that stage”.

Robustness in conceptual design is primarily restricted to evaluating the robustness of various concept variants. That is, a range of solutions are contrasted with a set of design requirements, including robustness (Andersson, 1997). Melvin and Deo (2002) proposed to axiomatically designed robustness based on the independence and information axioms proposed by Suh (1990). The independent axiom (Axiom 1) states that the “independence of FRs [Functional requirements] must be maintained during the design process. This means that the choice of physical embodiment must be made so as not the couple FRs when the DPS [Design Parameters] is changed” and the information axiom (Axiom 2) refers to the “minimisation of some parameter called information. More specifically, it states that among all designs that satisfy functional independence (Axiom 1), the one that possesses the least information is the best” (Suh, 1990). Melvin and Deo (2002) asserted that “by nature of the two design axioms, robustness is improved” and suggested a framework for the axiomatic design to address functional requirements early in the conceptual phase with corresponding design parameters and affirmed, “that the system designed for conceptual robustness will show a better response to the optimization and therefore will have superior robustness to the noise factors that have been targeted during conceptual design”.

Andersson (1996) discussed conceptual robustness as “conceptual solution that is robust to variations in specified input and to design variables' deviations from nominal values”, and proposed a semi-analytical approach to determine the robustness of conceptual solutions based on the concepts of error transmission formulas, which could help to identify critical design variables. Whitfield et al. (1998) proposed a robust design methodology that is suggested as appropriate for one-off products such as ships, that incorporates techniques of design of experiments (DOE), response surface methods, analysis of variance (ANOVA) and optimisation genetic algorithm. Du and Chen (2000) proposed methodology for design feasibility robustness that is the basis of probabilistic feasibility, but this approach requires a description of the distribution of uncertainty and the distribution of constraint performance that may not be available in initial design stages. As another viewpoint for assessing robust systems, Malak et al. (2015) proposed the use of normative decision theory for robustness in the conceptual stages of designing a system.

Göhler et al. (2016) presented a systematic review of the literature on robustness metrics in product development and engineering design. Their findings identify the following four classifications for robustness metrics: sensitivity robustness metrics that assess the robustness of a concept; the size of the feasible design space metrics assessing the robustness of design; functional expectancy and dispersion robustness metrics assessing the robustness of a function; and the probability of functional compliance metrics are assessing the robustness of a product. Specifically, metrics categorised in the sensitivity classification assess the robustness of the concept regardless of specified functional requirements and anticipated variation, as they measure the capability of a design solution to reduce or increase variation and are more appropriate in the initial design stage. However, these metrics relate to the variation of the design parameters rather than to the structure of the system, and therefore, in this systematic review by Göhler et al. (2016), the metrics relating to the system architecture have not been identified.

The research studies mentioned above relate to robust design methods and metrics that can apply to the conceptual phase but do not focus on assessing the robustness of the system architecture.

4.3.2 Robustness of system architecture

Crawley et al. (2004) explained that robustness can be accomplished in different ways, such as by focusing on the quality of the components and also the dependability of the interconnections between them. Concerning connectivity, this means that robustness can be improved by better design of the system architecture. From a system engineering viewpoint, Griffin (2010) explained that a robust system means that “the system should not produce radical departures from its expected behaviour in response to small changes to its operating input, internal state, or external environment. It should degrade gradually and gracefully in response to component failures, changes in its operating environment, or when design loads are exceeded”. As a result, a robust system in response to change does not generate any drastic deviations from its intended behaviour and is steadily deteriorating.

Chen and Crilly (2016) acknowledged that there are two different types of robustness: namely, architectural robustness and behavioural robustness. The difference between them is that for “architectural robustness and flexibility, it is the relationship between architecture (which might be the architecture of a system, system type, state or behaviour) and function that we are concerned with. By contrast, in the case of behavioural robustness and flexibility, we are concerned only with the architecture itself (characterised as regularities in behaviour)” (Chen and Crilly, 2016). Following Gero and Kannengiesser's (2004) views, this study accepts that architecture – structure – drives behaviour. Thus, in alignment Chen and Crilly (2016), the study is focused on architecture robustness as the behavioural property of engineering systems driven by architecture.

Slagle (2007) focused on architectural robustness and recommends that robustness should be determined during the conceptual design process, where the system's architecture is also decided. In this way, Slagle (2007) highlighted the need to address robustness earlier in the design process and proposes a set of principles to address robustness during architecture design. These principles are stability, modularity, feedback, standardisation, redundancy, simplicity, independence, autonomy and scalability. Slagle does not propose a quantitative method or metric to determine this architectural robustness. Fricke and Schulz (2005), meanwhile, argued that the principles, which support robustness, are simplicity, independence, modularity/encapsulation and redundancy. These principles are decided based on the system architecture (components and the relationship between them).

While it is acknowledged that robustness is an essential property of engineering systems, engineers typically assess that a system is robust through experience and intuition, rather than by using quantifiable and objective measures. Griffin (2010) argued that "while confidence in the robustness of certain systems can be gained through experience and intuition, and while fragile systems can be identified after the fact of failure, quantifiable measures of robustness do not exist". There is a need for a systematic approach to assessing the robustness of system architecture, since in system engineering "it remains true that some system designs are more robust than others, and some system designers know how to enhance this property while others do not. Work is needed to understand how this can be done more purposefully" (Griffin, 2010). Also, as discussed in Section 4.3.1 the robust design methods and metrics are not appropriate for the early stages. As Slagle (2007) explained, "addressing robustness and the 'ilities' at the architecture level may be more effective because it is the earliest and highest leverage point in the product development process". Some system architectures are intrinsically more robust than others.

4.3.2.1 Robustness methods for system architecture

Methods identified to relate to the robustness of the system architecture are discussed in this section. Zakarian et al. (2007) proposed a framework for robust systems based on "system modelling, integration analysis and quality engineering". The modelling approach identifies functional connections between parts, creating a DSM which corresponds to a directed graph. Then, clustering analysis is conducted by dividing parts into subsystems and identifying subsystem functions. At this point, experimental design methods are used to discover controls that are preferable for developing a robust design development from a quality engineering point of view. However, Zakarian et al.'s research did not include a specific robustness metric and focuses all its attention on inter-module connectivity without expanding on module size and types.

Another recent system-level approach found in the literature is that of Pumpuni-Lenss et al. (2017), whose focus is on resilience. However, it is mentioned here as it demonstrates similarity in terms of its system-level approach. Pumpuni-Lenss et al. (2017) suggested a "combination of network analysis and agent-based modelling" to assess the system's performance after disruptions and recovery attempts. This approach uses a popular network science optimisation approach, max flow problem, that was first proposed Edmonds and Karp (1972) consist of the source node, a sink node, and a set of other nodes. Edges linking these nodes are enabled, and the objective is to discover a viable route from origin to minimise costs. Sources have favourable (supply) stream, while sinks have adverse (request) stream. Each edge has associated ability, so the flow through that edge cannot reach that capacity. Each edge also has connected costs. The network flow problem is aimed at transporting the entire stream from the source nodes to the sink nodes at minimum cost (Selva et al., 2016).

de Vos and Stapersma (2018) acknowledged that it is difficult to determine robustness quantitatively and, during the early stages of design, which motivated the formulation of a metric which aim to help to design robust energy ship systems. The proposed robustness metric assesses on-board ship energy distribution systems in respect of system reconfigurability. The concept of the proposed metric is that "maximises the flow between hubs ... using the max flow function the number of disjoint paths between hubs is increased by the system reconfigurability objective function". The limitation of this work is the use of undirected networks; these lead the authors to adopt this new matrix that includes only the hub connectivity. de Vos and Stapersma (2018) noted that "that the necessity of setting supplier-hub and hub-user connections to zero is a consequence of the choice to model the networks as undirected graphs, with directed graphs this would not have been the case", acknowledging this limitation of the work.

4.3.2.2 Robustness network-based metrics

Network science metrics are concerned with the robustness of the topology and are most closely related to the assessment of the robustness of the system architecture in engineering design. The most popular network-science metrics discussed in the engineering-system literature relate to the largest connected component, algebraic connectivity, average path length and global efficiency.

Robustness is the property of a network to withstand failures and is studied through percolation theory. Percolation theory explains the behaviour of the network if a proportion of the network's nodes or edges are removed. Newman (2010) stated that "the presence of a giant cluster is an indicator of a network that is at least partly performing its intended function, while the size of the giant cluster tells us exactly how much of the network is working". Dynamic robustness has been studied by (Braha and Bar-Yam, 2007, 2004b; Piccolo et al., 2018) using largest connected component demonstrating that design networks are highly robust under random disruption but are highly vulnerable under targeted disruption. The problem with the use of the largest connected component for engineering systems is that it fails to capture which key components are required to remain connected to achieve post-disruption functionality.

Algebraic connectivity is a network metric related to robustness. According to Mehrpouyan et al. (2014) "because of the close correlation between complex networks and complex engineered systems, algebraic connectivity is a good metric to be considered to determine the resilience of architecture of complex systems". Yazdani and Jeffrey (2012) recommended an algebraic connectivity metric to measure the structural robustness of water distribution systems. However, algebraic connectivity is calculated while the network is intact, as a structural metric, and does not capture behaviour after a disruption.

Other robustness-related metrics are the average shortest path-length metric and a global efficiency metric, which are two interrelated metrics. Wash, Dong, and Tumer (2017) used the network average shortest path-length metric to study the function of bridging components in engineering systems. Global efficiency is the average inverse shortest path length in the network. In the power-grid literature, global efficiency is used to assess the vulnerability of power systems (Arianos et al., 2009; Bompard et al., 2011). The global efficiency metric for vulnerability analysis is calculated before and after attacks, and the change in value indicates the vulnerability. The metric is adjusted in the field to capture electrical power-system features. The problem with the average short path and global efficiency metrics is that the key to the engineering system is that the correct node communicates to the corresponding node, not that all nodes communicate to all nodes.

In summary, despite the advancement in the use of the various network metrics in engineering design literature, a robustness network metric that meets the specific engineering system conceptual requirements has not been found in the literature.

4.3.3 Robustness-related attributes

Robustness relates to the systems' survivability, changeability and resilience (Department of Defense, 2011; Fricke and Schulz, 2005; Kott and Abdelzaher, 2016). De Weck et al. (2012) researched the relationships between fifteen system lifecycle properties. They employed twelve experts and divided them into four groups, questioning them about the relationships among the ilities and how they form hierarchies amongst them. Robustness is included in their investigation and is defined as the ability of the system "to maintain its level and/or set of specified parameters in the context of changing system external and internal forces". Their findings note that "system value is heavily driven by the ability of a system to be robust (despite internal and exogenous disturbances), flexible or changeable and resilient or survivable over time". De Weck et al. (2012) stated that the experts "were unable to clearly differentiate between survivability and robustness, for example, as their meaning was determined to be related, but distinct; imperfect synonyms". This finding in the literature is consistent with the findings of the explorative focus groups reported in Section 3.3.3, which stated that the experts used interchangeable survivability, resilience, and robustness.

Fricke and Schulz (2005) also suggested that robustness supports changeability, and robustness is identified as to be enabled by redundancy, modularity, independence and simplicity. Kott and Abdelzaher (2016) suggested that the link between robustness and resilience must be studied together as "a system that lacks robustness will often fail beyond recovery, hence offering little resiliency". Robustness also relates to adaptability, whereas adaptability "characterizes a systems capability to adapt itself towards changing environments to deliver its intended functionality" (Schulz and Fricke, 1999). It is suggested that robustness is "a prerequisite to achieve adaptability, i.e., adaptability is the evolutionary level of robustness" (Schulz and Fricke, 1999). In summary, robustness relates as a precondition and an enabler of other attributes of the system such as reliability, resilience, survivability, changeability, and adaptability. This is also one of the reasons why it is regarded as such a decisive attribute of complex engineering system architectures.

4.3.3.1 Reliability

The capacity of a technical system to meet its operational conditions within specific limits and for the period of its service life is described as the reliability of the system (Pahl and Beitz, 1996). The likelihood that a component, product, or system will function without any failure during the prescribed lifetime is associated with reliability. Reliability methods are widely cited in the literature and used to improve the design in the industry by offering approaches to identify potential failures that the system may experience during operation.

Failure modes and effects analysis (FMEA) is a “systematic method of evaluating an item or process to identify the ways in which it might potentially fail, and the effects of the mode of failure upon the performance of the item or process and on the surrounding environment and personnel ” (International Electrotechnical Commision, 2018). FMEA utilises a team's knowledge and skills to assess the criticality of future issues. FMEA is a systematic approach however is subjective, method of identifying causes and failure modes and assessing their relevant risks (Arabian-Hoseynabadi et al., 2010). FMEA also can include criticality analysis (FMECA) failure modes whereas failure modes are prioritised according to their importance. IEC 60812 explains the “the prioritization can be based on a ranking of the severity alone, or this can be combined with other measures of importance” (International Electrotechnical Commision, 2018).

Another method is the Fault Tree Analysis (FTA) which also examines methodically the effects of faults. Fault Tree Analysis (FTA) is concerned “with the identification and analysis of conditions and factors that cause or may potentially cause or contribute to the occurrence of a defined top event” (International Electrotechnical Commission, 2006). Typically, the FTA top is the loss of system functionality or deterioration of system performance. The outcome is depicted in a tree diagram, and this method provides graphical representations that use different Boolean combinations (AND/OR) of faults and events that may lead to top failure (Pahl and Beitz, 1996). IEC 61025 outlines that they are two approaches to the application of FTA: qualitative and quantitative. The qualitative approach does not consider the likelihood of occurrence, whereas the quantitative FTA approach faults or events have a likelihood of occurrence, which leads to the calculation of the likelihood of occurrence of a top which represents reliability or probability of fault or failure (International Electrotechnical Commission, 2006). The main disadvantages of the FTA are that the user is expected to recognise the connection between the components and the degree of loss of connection without further analysis.

Reliability methods related to the architecture of the system (structure) are also identified in the literature. Mhenni et al. (2014) suggested an automatic generation technique of fault trees based on SysML structural diagrams. The methodology represents SysML structural diagram as a directed multi-graph and generates generic fault trees with logic gates and events via a graph traversal algorithm and a set of defined fault tree patterns for example fault tree for redundant pattern, fault tree for feedback pattern, fail tree for entry and exit pattern. This method allows combines an architectural (structural) and behavioural analysis of a system.

However, this method needs information, which may not be accessible during the initial stages of the design. Roth et al. (2015) also developed an approach for creating and evaluating fault trees based on Design Structure Matrix (DSM) and Multiple Domain Matrix (MDM) relevant to early phases of system design, but this approach results in fault trees consisting of a large number of elements and difficult to interpret for a human. In general, the limitation of reliability methods relates to the great manual effort, need for experience, lack of a method to analysis connectivity and need for applicable and sufficient data (Fussell, 1975; Roth et al., 2015).

4.3.4 Redundancy

From the system design viewpoint, redundancy is a pivotal approach that is adapted to enhance the system's robustness. Chen and Crilly (2014) stated that redundancy "relates to the provision of additional capacity in a system, so that system performance is maintained despite partial system failure". Redundancy design requires the identification of the right components and connections that will benefit the robustness of the system, enabling the system to continue to operate despite disruptions. On an architectural level, additional redundancy translates into extra components, extra connections, and spare capacity.

ISO/IEC/IEEE (2017) defined redundancy as "the presence of auxiliary components in a system to perform the same or similar functions as other elements to prevent or recover from failures". Redundancy is also defined as the presence of "independent alternative paths between source and demand nodes which can be used to satisfy supply requirements during disruption or failure of the main paths" (Goulter, 1987). The first definition refers to redundancy as auxiliary components that are added as supplementary provisions, above what is required to achieve the functioning of the system. The second definition is more specific, as it mentions alternative paths and, thus, connectivity to satisfy demand.

ISO/IEC/IEEE (2017) also sub-divided redundancy into four categories: active, stand-by, homogeneous and diversity redundancy. The following are the definitions in ISO/IEC/IEEE (2017): 1) Active redundancy is defined as "the use of redundant elements operating simultaneously to prevent, or permit recovery from, failures"; 2) Standby redundancy is defined as "the use of redundant elements that are left inoperative until a failure occurs in a primary element"; 3) Homogeneous redundancy is defined as "realisation of the same function with identical means"; 4) Diversity is defined as "realisation of the same function by different means." More generally, redundancy is included in two broader categories. The first category relates to the duplication of the same components, and the second category relates to the substitution of alternative ways of achieving the same functions (Chen and Crilly, 2014).

In terms of design purpose, both types of redundancy provide additional capacity to the system so that it can continue to operate after a disruption.

Chen and Crilly (2014) indicated the main questions relating to redundancy are asked from a system engineering perspective. These are also the same questions that relate to redundancy and system architecture. They are:

1) Which components should be made redundant?
2) How much redundancy should there be?
3) What form of redundancy should there be?

Redundancy is desirable if it contributes positively to robustness, but an excessive degree of redundancy brings complications during system development: it increases the overall development costs and increases the operational likelihood of errors and accidents.

This research seeks to define redundancy in terms of how it manifests itself at the architectural level. The most apparent structural characteristic of redundancy is that it adds components (duplication or substitution) resulting in an increase in the size (in total components) of the architecture. The addition of components also means the addition of connections. Therefore, redundancy design is defined here as architectural (components and their connections) options in the instantiated system architecture that can satisfy the same function. It is expected that redundancy is incorporated in the architecture to enhance its robustness and that it relates to the level of the system's functional requirements.

In summary, the findings of Section 4.3 indicated that the methods and metrics mentioned in the literature were not capable of assessing the robustness of the system architecture options at the initial design stage while taking into account redundancy decisions and considering module-based disruptions. The findings prompted the definition of research objective 3: to define a method and metric for quantifying the level of robustness of system architecture. The following Section discusses modularity in system design.

4.4 Modularity

This section reviews modularity related literature: Section 4.4.1 review of modularity effects on competitive indicators of system development such as development time, cost, and quality. Section 4.4.2 and 4.4.3 provides an overview of the literature on modular design methods and methods. Section 4.4.4 concludes with findings on the modularity literature review.

4.4.1 Modularity effect on competitive indicators

In the literature, there is a wide discussion of the benefits of modularity in the context of product architecture and product platform. The benefits of modularity are argued to be economies of scale and cost-saving through high standardisation and commonality. However, there are differences between product platforms and complex engineering system architectures, and there are differences between the industries contexts, that modularity is designed to be implemented. One main distinction is that typically the complex engineering systems are bespoke, unique, and small in number. The following review focuses on the modularity effects on competitive indicators in the context of complex engineering systems.

4.4.1.1 Modularity effect on development time

In complex engineering systems, which make modularity a beneficial property, there are, namely, parallel development, subcontractors or experts' involvement, early testing of modules before the final integration process, and easy maintenance and upgradability of complex systems. The literature agrees broadly on the positive influence on cyclical times throughout the various phases of system development because of the parallelism of modules manufacturing and assembly. This is done by enabling modules to be designed and manufactured concurrently and, then, assembled (Jacobs et al., 2007; Lau et al., 2007).

Research performed in the machinery, electronic, and transportation equipment industries (Danese and Filippini, 2013) found that product modularity can help improve new product development (NPD) time and product performance, however, the modularity and NPD time relationship is partially mediated by supplier involvement during product development. That means that the participation of suppliers in the development of products is necessary for the modularity to be advantageous. Vickery et al.'s (2015) findings showed that "the positive effect of product modularity on launch speed, but it is delivered through the mediating effects of product platforms and manufacturing flexibility". In comparison to previous work, the findings show that modularity alone is not a sufficient condition for the timely and regular introduction of new products.

In the context of construction and shipbuilding, Pero et al. (2015) proposed that product modularity has positive effects on lead time and provides benefits to on-time delivery and the cycle time. Piran et al.'s (2016) comparison of modularised and non-modularised product designs confirms the positive effects of product modularity on-time delivery.

However, according to Vickery et al. (2016), there is a "negative interaction between product modularity and complexity" and they claim that this may be driven by "over modularity". Vickery et al. (2016) stated that "in essence, when the situation is rather complex the number of cross-module interdependencies grows …. may be offset by the time consumed to test and integrate the complex system”. This is supported by Ethiraj and Levinthal (2004) who stated that “the speed and efficiency gains from modularization will be offset by the increased time spent in the testing and integration phase, where the consequences of ignored dependencies will come to fore”. In this regard, Ryan and Efatmaneshnik (2014) carried out simulation work and considered the probabilistic failure of the processing time for any number of modules and different link failure probabilities. They concluded that an optimum degree of modularity could improve process time significantly but, if there were too many modules, this could lead to the opposite result and have a negative influence on the process time.

4.4.1.2 Modularity effect on the cost

The literature acknowledges that to implement modularity correctly, there is a need for significant additional effort during the early development phase when the design rules and modularity in system architecture are established (Baldwin and Clark, 1997). This means that there is an additional cost. There are, also, arguments in the literature about the declining relationship between the positive benefits of modularity and cost. Guo and Gershenson (2007) conducted an experimental study on the relationship between modularity and the cost of four off-the-shelf products. The results show that the only modularity benefits on cost were produced during the end of life phase. Otherwise, their findings showed that, except for large changes in the degree of modularity, there is no relationship between life-cycle modularity and lifecycle cost.

Efatmaneshnik and Ryan (2015) argued that, from a construction viewpoint, there is an optimal modular which can be achieved “through balanced modularisation of structural symmetry in the distribution of the sizes of modules”. They present an argument to support that “system construction cost is highly sensitive to both the number of modules and the modularisation structure”. Also, they stated that “the assumption that a modular design process will lead to a modular product or system is relatively naïve. Iterative processes, for example, can create high-levels of integrality”.

From their study of modularity and innovation in complex systems, Ethiraj and Levinthal (2004) suggested that there is a “trade-off between the destabilising effects of overly refined modularisation and the modest levels of research and a premature fixation of inferior designs that can result from excessive levels of integration”. The study shows an imbalance in this relationship with increasingly sophisticated modules that lead to cyclic behaviour and lack of efficiency.

Ethiraj and Levinthal (2004) claimed that “a degree of modularity above an optimum [level] leads to a higher performance penalty than [it would] below the optimum [level].” The authors consider that mistakes on the side of integration are either safer or preferable to errors on the side of modularity. Also, they discuss the phenomenon of ‘chaotic behaviour’, which arises when modularity approaches extreme limits; this is akin to complexity theory (Kaufman, 1995). They conclude that modularity has a significantly positive effect on business performance but only when a few high-performance modules are recombined.

4.4.1.3 Modularity effect on quality

In respect of the modularity-quality relationship, Mikkola and Gassmann (2003) argued that maintenance synergies lead to positive gains. In commenting on the earlier identification of failures and errors before modules enter the main flow of manufacturing, Erixon et al. (1996) said that the parallel development of modules allows simplification of testing and troubleshooting and isolation of failures. Kusiak and Huang (1997) advised that testing is simplified through high modularity “by a reduced amount of I/O [Input/Output] simulation required to test these functions.” However, from their investigations of the effects of modularity on reliability from a theoretical management viewpoint, D’Adderio and Pollock’s (2014) findings showed that, at the high levels of modularity, there are tensions between reliability and cost. D’Adderio and Pollock (2014) argued that an imbalance between the cost reduction and efficiency, gained by a large degree of modularity in design and organisation because of losses in reliability and quality, causes a lack of traceability of errors in the modules. The reason for this is that the supplier is unable to perform system-testing of their components and the integrator’s inability to find the mistakes and fix them in the components outsourced by suppliers. Moreover, Lau et al. (2007) rejected the hypothesis that product modularity has a positive relationship with product quality. Their findings show that there is no significant association between product modularity and product quality. Ethiraj and Levinthal (2004) concluded that the cost of mistakes in over-modularisation is more expensive than the cost of errors of over-integration. Vickery et al. (2016) suggested that, when there is high complexity, product modularity may bring unexpected problems concerning interdependencies and integration of modules. Vickery et al. (2016) advised that “modularisation may be fruitful in a complex environment but too much of it might produce adverse effects. Thus, we caution against the excessive level of product modularisation when complexity is high”.

The arguments, presented in this section, provide a valid justification for seeking the right level of modularity as opposed to a maximal level. This is contingent upon the individual objectives of the system and business. The following is an overview of the modular design methods discussed in the engineering design literature.

4.4.2 Modular design methods

Published literature reviews on modular design methods provide a comprehensive overview of the subject. In this regard, Allen and Carlson-Skalak (1998) classified modular design methods into two broad groups: namely, function-based methods; and matrix-based methods. Jose and Tollenaere (2005) identified five categories of modularity methods. These are, namely: clustering methods; graphs and matrix partitioning methods; mathematical programming methods; artificial intelligence methods; and genetic algorithms& heuristics methods. Daniilidis et al. (2011) grouped them into three categories: namely, DSM methods; modular function development; and function structure heuristics.

The following section presents an overview of the literature on the published-on modularity methods. The different classes of modularity methods are identified as following: matrix-based -based methods, functional and lifecycle methods, and multi-objectives and constrained methods.

4.4.2.1 Matrix-based methods

In the literature review, the first category of modular design methods is suggested which classified together matrix, graph, and metric-based methods. For the design of a modular product or system architecture, highly interactive groups of elements are identified and arranged in modules. To find potential modules, a broad range of approaches incorporating the Design Structure Matrix (DSM) is proposed. DSM is a tool for depicting and assessing systems for decomposing and integrating them. DSM is popular because it is a simple, clean and solid instrument that provides a clear, visual representation of complex systems. In practice, DSM is a square matrix that illustrates the links between matching components in the rows and matrix columns. The DSM objective is to provide insight and support the design or redesign of the systems, products, processes or organisations, through a matrix representation. DSM is employed in system engineering and design, organizational management, project management, product development and project planning fields and used in the industries of construction, semiconductor, automotive, photographic and aerospace, semiconductors, telecoms, small scale production, plant equipment and electronics from an industrial perspective (Browning, 2015).

A plethora of scholar works exist employing the DSM in combination with a clustering algorithm which aims to maximise the connectivity within a module and minimise the connectivity amongst modules that were first proposed by Eppinger et al. (1994).

A variety of clustering algorithms are found that are the basis for genetic algorithms (GA). Examples are the methods developed by Whitfield et al. (2002) and Yu et al. (2007). Whitfield et al. (2002) present a DSM based two-step method to identify the modular structures. The first step determines a value for the clustering criterion to group component dependencies together. A GA algorithm is applied to minimize the clustering criterion value and to recognize the most significant component modules within the product structure. The second step applies a Module Strength Indicator (MSI) function to determine the value of the degree of modularity of the component's cluster. The maximization of the MSI objective function to the DSM produces a Module Structure Matrix (MSM) representing the modularity of the available component grouping within it. But Whitfield et al. (2002)' approach drawback is that it is difficult to implement the proposed algorithm in complex, high-number and connectivity system architectures.

Yu et al. (2007) proposed a modular design method that uses the minimum description length as a metric for the suggested clustering objective function. Then, the MDL-based objective function is used with a genetic algorithm (GA) to cluster weighted graphs or corresponding DSMs. The objective of a GA is to discover the clustering arrangement that minimizes the minimum description length objective function for a specific DSM. A weakness of the approach was that "the larger the DSM is, the larger the discrepancy between the GA results and the expert's arrangement"(Yu et al., 2007). The explanation for this was assumed to be the human mind's failure to be precise in choices with largest DSMs.

AlGeddawy and ElMaraghy (2013) incorporated cladistics analysis into the DSM-based and clustering method to develop a method that defines the hierarchical relationship between modules. The proposed method identifies "the best product modular granularity level which minimizes the overall interactions between modules and sub-assemblies at all granularity levels". This method is developed based on the finding that "degree of modularity can vary for the same system when the system is represented at the two different levels of granularity" (Chiriac et al., 2011b).

In general, DSM-methods objective is to maximise modularity metric, offering automated methods to support the design, however, the underline assumption accepted is that modularity is an all-inclusive positive attributed of system architecture, thus these methods purpose is to maximise modularity.

A network science approach in modularity methods in the engineering design field is suggested also by Sosa et al. (2007) to develop component modularity metrics. These are based on centrality measures which are used commonly by graph theory and network analysis when studying social networks. This research views the complex products as a network of components that share technical interfaces so that they can function as a whole. Sosa et al. (2007) view modularity at the component level through evaluating the components’ design interface behaviours along with the other components in the product and did not use a clustering approach into identify grouping of components into modules.

Network theoretic methods such as Newman and Girvan (Newman and Girvan, 2004) and community detection algorithm (Blondel et al., 2008) are also found to be adopted (Sinha et al., 2018, 2017) in engineering design literature. Further discussion on network science-based methods is provided in Section 6.3.1.2.

4.4.2.2 Functional, strategic driven and lifecycle-based methods

In the literature review second category of modularity methods groups together functional, strategic driven and lifecycle-based methods. The reason that these methods are categorised together is that they are not found to adopt a maximisation approach to the identification of modules in the system architecture and therefore do not accept the ideal modularity concept.

A prominent functional-based modularity method is the function structure heuristics approach that was introduced by Stone et al. (2000) basis on Pahl and Beitz (1996). The method’s first step employs a functional decomposition block diagram of the product’s function, material, energy, and information. In order to identify the dependencies among the modules, (Stone et al., 2000) employed functional structures as dominant flows, branching flows and conversion-transmission flows for identifying. The heuristics define modules; it is up to the designers to select the suitable module. The heuristics express the maximum function structure. The method is manual allowing the decision making of the designers to be considered, who may choose to modularize groups of function structure smaller than the heuristics definitions suggest. The function structure heuristics approach does not necessarily maximise the degree of modularity, and it is found a flexible functional-based method to identify modules. This approach helps the user to decide on the formulation of modules, but users usually need more assistance in deciding on large-scale engineering systems’ modules, as the increase in the size and complexity of the systems limits the human mind's ability to process information. Another drawback of this approach is not resolving the issues of physical system modularisation. Moreover, Otto and Sudjianto (2001) advance the functional structure heuristic approach to support multiple brand platforms, whereas Dahmus et al. (2001) also proposed the “the idea of the modularity matrix, which lists the functions in the family versus the products in the family”.

Ericsson and Erixon (1999) proposed Modular Function Deployment (MFD) modularity method that is based on strategic drivers, and grouping components into modules is the basis on identifying similarities that relate to component’s strategies. Sanaei et al. (2017) commented on MFD methods stating that “modularity drivers are selected based on the product-related company-specific strategic requirements which imply there is no single ‘ideal’ architecture”.

Borjesson and Hölttä-Otto (2014) developed an advance hybrid module algorithm based on component interactions and strategic drivers by combining Modular Function Deployment (MFD), DSM and the IGTA (Idicula-Gutierrez-Thebeau Algorithm) clustering algorithm. The authors achieve by integrating these methods a balance between the interaction of the components and the strategic objectives of a company. Zhang et al. (2006) suggested a behaviour-driven function-environment-structure (B-FES) modelling framework method linking functions to behaviours advocating that it provides the designers with the freedom to innovate, parallel with modularising by not strictly linking the function to structure. The method introduces a behavioural modularity matrix to identify behavioural modules.

Lifecycle modularity methods view modularity as the mean to achieve lifecycle objectives. Nepal et al. (2005) presented a method that optimises the modular structure based on life cycle cost, quality, reliability and maintainability. Modules are based upon the maximization of the similarities between the components. For each objective (life cycle cost, quality, and reliability and maintainability) a set of three performance metrics are used to evaluate each pair of components (module candidates). Via fuzzy logic, and overall ‘performance index’ is calculated for each pair of components according to how the three metrics are evaluated.

4.4.2.3 Multi-objective and constrain-based methods

A third category is dedicated to the proposed multi-objective and constraints modularity methods found in the literature. Multi-objectives methods accept that there trade-offs between modularity and other aspects of design. Sinha and Suh (2017) proposed a multi-objective method that minimising the alternation of complexity across the various modules and maximising modularity. The findings of combining examining complexity and modularity were that “the modularity maximizing decomposition induced a large variation in module-level complexity variation, while more balanced complexity distribution among modules leads to erosion of modularity” (Sinha and Suh, 2017). The proposition adopted is that is beneficial to minimising the variation in complexity allocation to individual modules, and thus to achieve this complexity objective the modularity requires to be declined. Bayrak et al. (2018) proposed a multi-objective optimization framework based on high-level system requirements and life cycle objectives. The findings of the application of Bayrak's approach on a fleet of military vehicles indicated that "a large number of smaller modules reduces the overall fleet weight and increases required personnel resources" (Bayrak et al., 2018). Hvam et al. (2017) recognised that engineer-to-order companies lack the tools to support the identification of the right level of modularity. Hvam et al. (2017) proposed a Modularity Application Matrix which is a conceptual tool that aid on improving the comprehension of the idea of a partial modularization.

Sanaie et al (2017) acknowledged that existing methods predominantly view ideal modularity, besides that modularity can conflict with each other criteria. Sanaie et al (2017) proposed a multi-objective optimisation employing a set of metrics together. Sinha et al. (2019) also identified that modular design methods do not explicitly consider design constraints, and proposed a modularization method that is based on the modification of two existing methods: Blondel et al. (2008) community detection method and Idicula-Gutierrez-Thebeau-Algorithm.

Otto et al. (2020) stated that "DSM methods search for an optimal clustering of the system into modules and do so without the benefit of considering field constraints" and thus proposed new design guidelines for modular design approaches related to field constraints. They suggested two sets of guidelines: field separation and concept generation. Specifically, the field separation guidelines include the proposition that "a maximum-sized module should be defined the zonal boundaries", in that respect it is suggested that the "largest modules possible are defined by the set of product-bisecting field separation lines" (Otto et al., 2020).

4.4.3 Modularity metrics

Various modularity metrics are found in the literature that is used in combination with modularity methods operating as objective function; employ to evaluate the potential of modularisation in the architecture, or to assess the benefits of modularity. Hölttä-Otto et al. (2012) classified modularity metrics into similarity, combination and coupling categories. The focus of the review is on coupling modularity metrics that are aligned with the modularity definition accepted in this study, i.e. the module is a "relatively independent chunk of a system that is loosely coupled to the rest of the system" (Hölttä-Otto et al., 2012).

The degree of modules independency relates to metrics that measure, the level of coupling in the system architecture. A prevalent metric found in literature is the Singular Value Modularity Index (SMI) developed by Hölttä-Otto and Weck (2007) that is based on the rate of singular value decay. SMI measures “the decay rate of the sorted, normalized singular values in the system” (Hölttä-Otto and Weck, 2007), noting that there is a more gradual decay in the singular values, for modular architectures. This metric characterises the inherent modularity of the architecture, and one of the limitations of this metric is that it does not explicitly consider the density of the connectivity within modules.

Guo and Gershenson (2004); Jung and Simpson (2017); Whitfield et al. (2002) have developed metrics to determine the coupling and have applied the metrics on examples of technical systems. Whitfield et al. (2002) proposed a ‘Module Strength Indicator’ (MSI) as a modularity metric that measures the strength of the modules. The MSI equals the mean value of the internal dependencies within the module minus the mean value for the external dependencies within the module. This is done by finding “modules that have a maximum number of internal dependencies and a minimum number of external dependencies” (Whitfield et al., 2002). However, as Hölttä-Otto et al. (2012) noted, this metric level is specific to the strength of the modules and does not quantify the architecture’s overall modularity. Whitfield’ metric requires to be revised to offer a generic and normalised metric for overall architecture. Similar to Whitfield et al. (2002), Guo and Gershenson (2004) proposed a modularity metric that focuses on intra-module and inter-module connectivity. Guo and Gershenson's (2004) metric is an improved version of the modularity metrics that focus on intra-inter module connectivity. It is “normalised to be sized independent which is a significant evolution from the early coupling metrics” (Hölttä-Otto et al., 2012).

Jung and Simpson (2017) proposed a modularity metric, which has intra-inter module connectivity that considers the four distinct characteristics influencing the level of modularity in system architecture. These four characteristics are connection strength within and between modules; density of connection with and between modules; the proximity of interaction to the diagonal; density of connections between buses and other components. However, this metric is more useful in the cases of redesign purposes when the comparison of redesign solutions is driven by the initial design in which case most information is known rather than during the conceptual stage.

Sosa et al. (2003) developed a metric which is founded on how components share design interfaces. This is translated as the fraction of the number of interactions between the modules divided by the number of interactions in the overall architecture (modelled in DSM). Sosa et al’s interpretation have been used by other scholars, such as Hölttä-Otto et al. (2012); and Jung and Simpson (2017) in quantitatively comparing modularity metrics and evaluating their outcomes.

Yu et al. (2007) proposed modularity metric based on information theory. The adopted Minimum Description Length (MDL) principle is that “among all possible models, choose the model that uses the minimal length for describing a given data set”. The modularity metric has the advantage of being able to measure the level of modularity of bus-type architecture and allows the overlapping of components into different modules. However, it has a significant disadvantage because it is unable to be used for architectures of different sizes. Therefore, this metric is more appropriate for use in redesigns where a single initial design is considered, and the only alternative options are to make comparisons rather than using in the conceptual phases which show that it is possible to use alternative designs of different sizes. From their comparative work, Hölttä-Otto et al. (2012) concluded that Yu’s metric achieves the best results since it is the only metric that can recognise bus architecture. However, it is acknowledged that the metric’s weakness is concerning quantifying systems of different sizes. Allowing the overlapping of components into modules can also be considered as a drawback in terms of robustness as a failure of the overlapping component means that it will have an impact on the multi-modules that belongs.

Whitney index (Dong, 2002), proposed an interaction density index that measures the level of coupling in the architecture. The interaction density is the fraction of the number of interactions divided by the total elements of the architecture (that is modelled in the DSM). Luo (2015) used the average nodal degree as “an indicator of product integrality or reverse indicator of product modularity”.

Sosa et al. (2007) proposed component modularity metrics based on network theory by additionally creating and consolidating existing measures. They defined “complex products as a network of components that share technical interfaces (or connections) to function as a whole and define component modularity based on the lack of connectivity among them”. Modularity is based on the absence of connections among components rather than the connectivity between the modules. This is one of the first network-theory based research works in engineering design literature. The proposed component modularity metrics are based on prevailing centrality metrics from network theory. Such network-based metrics are the degree centrality, closeness centrality and betweenness centrality. These are interpreted correspondingly into, degree modularity, distance modularity and bridge modularity. However, as Van Eikema Hommes (2009) noted, also, such measures can provide insights into central components. While these that can help to improve component modularity, they do not directly capture modularity.

In the broader network science literature, there is the Newman modularity index (Q) (Newman, 2010) as per equation (1):

$$Q = \frac{1}{2m}\sum_{ij}\left(A_{ij} - \frac{k_i k_j}{2m}\right)\delta(ci, cj) \quad (1)$$

Where “m: the number of edges, δ (ci, cj): the Kronecker delta, A_{ij}: the adjacency matrix, k: the degree of nodes”. For example in engineering design literature, Sinha and Suh (2017) used Newman's (2010) modularity metric developed previously in the network science context. Other network-based metrics found to be used in the engineering design literature related to modularity are the clustering coefficient, average nodal degree and the centrality metrics. Braha and Bar-Yam (2004b) proposed the clustering coefficient as a measure of the potential modularity of complex design networks. Braha and Bar-Yam (2007, 2004a, 2004b) used the average nodal degree for large-scale design networks. Parraguez et al. (2015) used in combination centrality metrics and clustering coefficient to determine the level of process modularity in complex engineering design projects, stressing that it is difficult to evaluate global network topology properties such as modularity where either centrality or clustering metrics are calculated in isolation.

4.4.4 Key findings of modularity literature review

Modularity effects on competitive performance indicators such as cost, development time and quality were firstly reviewed in this part of the literature review. The first findings from the literature review indicated the modularity is a beneficial property of engineering systems, however, also highlighted that there are drawbacks on implementing modularity. There is a lack of sufficient evidence to objectively support the perception that modularity is an all-around positive principle in the design. The finding of the review is in agreement with Bonvoisin et al. (2016) “potential benefits have been attributed to modular product design, concrete evidence is rarely provided and potential benefits are not supported by arguments but merely asserted”. The findings of the research are also aligned to Hölttä-Otto and de Weck (2007) concluded that regarding their study “results presented here are in apparent violation of the independence axiom in axiomatic design as well as of the idea that a high degree of modularity is always achievable or is always a virtue”. The literature reviewed confirmed the industrial focus group findings of Chapter 3 that the maximum level of modularity may not always be desirable.

The literature review on modular design methods finds that the DSM based methods are predominantly concern on maximising modularity. Sanaei et al. (2016) have identified this gap in the literature explaining that “it is difficult to simply embed all objectives of modular thinking into one metric to optimize” and adopted a multi-objective optimisation technique which supports to identify trade-offs of modularity against other constraints. Also, Bayrak et al. (2018) proposed a multi-objective optimization framework based on high-level system requirements and life cycle objectives. These research works develop methods that view modularity in position with other requirements and constraints of the design.

Another finding of the literature was to note that there are different automated and computational recent work and older manual methods developed. The automation of methods was needed due to the increasing size of systems and the many possible alternative options of grouping the various components. However, the manual methods allow expert input to be included that automation could not. There were several researchers such (Helmer et al., 2010; Sanaei et al., 2016) that include manual intervention for amending, refining and verifying the automated results in their proposed methods. Within an automated methodology, such manual stages introduce flexibility, allow users to interact with the automated part of the method and have more control over the methodology. Combining computational and manual approaches was deduced as the more reasonable way to support the identification of appropriate modules.

The findings of the literature review in this Chapter agreed with the explorative focus group discussion findings (Chapter 3), highlighting the need that the right level of modularity in a system requires a careful investigation. In the research study, modularity is positioned not as the single requirement of a system architecture methodology. It is viewed in combination with another significant driver of design, namely, robustness. The next stage of literature review considers specifically the relationship between modularity and robustness.

4.5 Modularity and robustness

The summary of the previous literature on modularity, redundancy and robustness provides the definitions and background on the methods and approaches used in the engineering design area. It was identified from the literature review that modularity and robustness are typically found to be studied in isolation in the academic literature. Individual strands of literature have considered modularity in engineering design and, similarly, robustness in a manufacturing context and systems as fields of research. A literature review that discusses specifically the relationship between modularity and robustness and this is presented here.

4.5.1 The relationship between modularity and robustness

In engineering design literature, there is an emerging discussion of the relationship between modularity and robustness. This section focuses on identifying the references in the literature which explore how modularity and robustness are related.

In general, there is no agreement in the literature on the effects of modularity on robustness. They are scholars who argue that modularity favours robustness (Gershenson et al., 2003; Ishii and Yang, 2003). In this analysis, modules have a high quality due to their standardisation and early testing, and thus they contribute towards making a more robust system overall. Slagle (2007) claimed that "modularity can be used to subdivide a system using the control factors to achieve enhanced failure mode avoidance. By using the physics of operation, an architect can segment the complex system so that coupling is minimized between components or modules; this allows each module to adjust independently and more favourably to noise factors. Because of the loose coupling between modules, the system can contain and prevent the transmittal of noise or error states to other parts of the system". Raz and DeLaurentis (2017), meanwhile, established "a proof of concept that more modular architectures are more dependable and survivable. Hence, modularity—which is obtained solely from the SoS [System of System] architecture and before any functional execution—could be used as a decision aid tool by the systems designers".

Mehrpouyan et al. (2014) argued that "to design a robust system and to recommend or oppose the modular physical system architecture, it is utterly important to understand the architectural properties of complex engineered systems". They used a network-based metric of algebraic connectivity to quantify robustness, finding that high modularity has a negative influence on robustness. They acknowledge that "it is commonly known that modularity in complex engineered systems is useful for system construction and maintainability, but the isolation of failures into a single module typically makes the system less robust".

Chang (1996) explained in respect to quality (that is an attribute that relates to robustness) that is a challenge for modular design, claiming that "products with higher modularity may be subjected to greater noise effects and/or higher direct costs, yet have better readiness. ... How to balance these conflicting interests, or how to integrate these interests into the modular design objective function and/or constraints, is a research topic that needs careful review".

From a system design point of view, Adams (2015) acknowledged that reduced connectivity in the design of systems is a consequence of modular design and construction and that, typically, it reduces the system's robustness. Ross et al. (2008) advised that, in modular design, each module follows principles of robust design but because the redundancy has a critical role in achieving robustness at the system level, a high level of modularity may be counterproductive for robustness due to the natural decline in the number of connections.

Elsewhere in the literature, Walsh et al. (2019) ascertained that there is a "trade-off between modularity and robustness in the design of engineering systems". Given this robustness – modularity trade-off, they advise that the "overall benefits of increasing a system's modularity are situation-dependent rather than universal. Constant re-evaluation of established design principles is necessary given their usefulness to designers".

Although the literature discussed the relationship between modularity and robustness, no design methodology in engineering design literature has been established to address the design of robust modular systems.

4.6 Research gap

The research gap identified in the literature is the lack of an explorative and analytical methodology to support the design of robust modular architectures in the initial stage of system development.

4.7 Methodology requirements

Through the literature review and the focus group discussion, the requirements for a methodology to address the gap were established. The following are the challenges driving the requirements that were identified as the key elements in developing a solution to the research problem:

Research challenge 1

The literature highlights the need to generate and evaluate various candidate system architecture options during the initial system design stage and indicates the design fixation on the over-reliance of features of pre-existing designs (Section 4.2).

Methodology requirement 1

A methodology should offer the capability to explore alternative system architectures That is, the methodology should be able to ***generate alternative system architectures.***

Research challenge 2

Redundancy, modularity, and robustness are examined in isolation and the practice was either not assessed or was assessed by different disciplines and in different phases of the development process (Section 4.3 & 4.4 &4.5).

Methodology requirement 2

The solution should be able to assess multi-attributes altogether. The methodology should ***combine analysis and evaluation based on redundancy, modularity, and robustness requirements.***

Research challenge 3

The limited quantitative, computational, and automated analysis in the initial system architecture design stage is identified as a problem (Section 4.2).

Methodology requirement 3

This requires including quantitative evaluation indicators of the system architecture. The methodology should provide ***computational and quantitative evaluation metrics and an analysis stage.***

Research challenge 4

Detail models limit creativity and detail technical information is not available during the concept initial design stage (Section 4.2).

Methodology requirement 4

The methodology should ***consist of high-level modelling and to require the input of simple data.***

These requirements for the methodology are addressed throughout the development of the research methodology.

4.8 Chapter summary

This chapter presented a review of the literature that identifies the gap in the research and requirements for the development of the methodology. The findings of the literature review show that there is a need to support architects to design robust modular systems architectures. No specific methodology was found in the literature to aid the design of robust modular system architectures.

The literature review findings indicate that system architecture design can be supported by design methodologies applicable during the initial stages of design. The importance of the conceptual design phases and the design fixation phenomenon is highlighted in the literature.

The literature review found that modularity is a desirable property, but the various potential challenges it poses mean that it is not desirable always to search a maximum, level of modularity. The second finding from the literature is the identification of the lack of a quantitative metric to assess the robustness of engineering systems. A third finding from the literature review is the identification of tradeoffs between modularity and robustness. This is driven also by the redundancy and modularity interactions. The literature review concludes by identifying the research gap that is the lack of an explorative and analytical methodology to support the design of robust modular architectures in the initial stage of system development.

A summary of the key definition adopted in this study is provided as a foundation for the following chapters. The system architecture is accepted as “an abstract explanation for a system’s entities and the relationships between them” (Crawley et al., 2004). Modularity is a property of a system architecture consisting of modules. The module is a “relatively independent chunk of a system that is loosely coupled to the rest of the system” (Hölttä-Otto et al., 2012). Robustness is defined as the ability of the instantiated system architecture to support sufficient functional continuity after a disruption. Robust modular is defined as the ability of the instantiated system architecture to support the sufficient functional continuity after disruption of a module. Redundancy design is defined as architectural (components and their connections) options in the instantiated system architecture that can satisfy the same function. Next, Chapters 5 presents the robust modular generation and assessment (RoMoGA) methodology.

PART 2: METHODOLOGY DEVELOPMENT

Chapter 5:

Robust modular generation and assessment (RoMoGA) methodology

This Chapter outlines the robust modular generation and assessment (RoMoGA) methodology. RoMoGA has three, iterative, stages: generation, analysis, and evaluation. The following sections summarise these different stages and describe how the implementation of RoMoGA progresses through descriptive, explorative and prescriptive approaches: where descriptive refers to the modelling and analysis of an established technical system, while the explorative employs a network tool to create novel topological analogous system architecture options and lastly the prescriptive suggests redesign system based on the two previous approaches. The iteration variable [4] is the system architecture modelled as a design structure matrix (DSM) that is iteratively modified and assessed until a set of desired system architecture options are satisfactory found. In the following overview of these stages, the initial technical system is termed the “nominal system architecture” and subsequent iterations termed the “system architecture options”. After this overview, the details are presented in Chapter 6 and 7.

5.1 Generation stage

The generation stage of the RoMoGA methodology use network and the matrix representation for the system architecture definition where the system architecture is modelled in a design structure matrix (DSM).

DSM is a popular tool for representing system architectures (Browning, 2015). The DSM is analogous to an adjacency matrix in network theory. The nodes of the network account for the heading of rows and columns of a DSM, and the edges represent the interactions inside the DSM. System architecture as a network is defined as G= (C, E), where C= {c1, c2, c3…} are the nodes forming the system and E= {e1, e2, e3…} edges. An adjacency binary matrix, A_{ij}=1, if the edges (interconnections) exist between nodes (components), and A_{ij}=0 if there are no connections.

---

[4] **Iteration variable -** the variable that changes each time the iteration executes and controls when the iteration finishes.

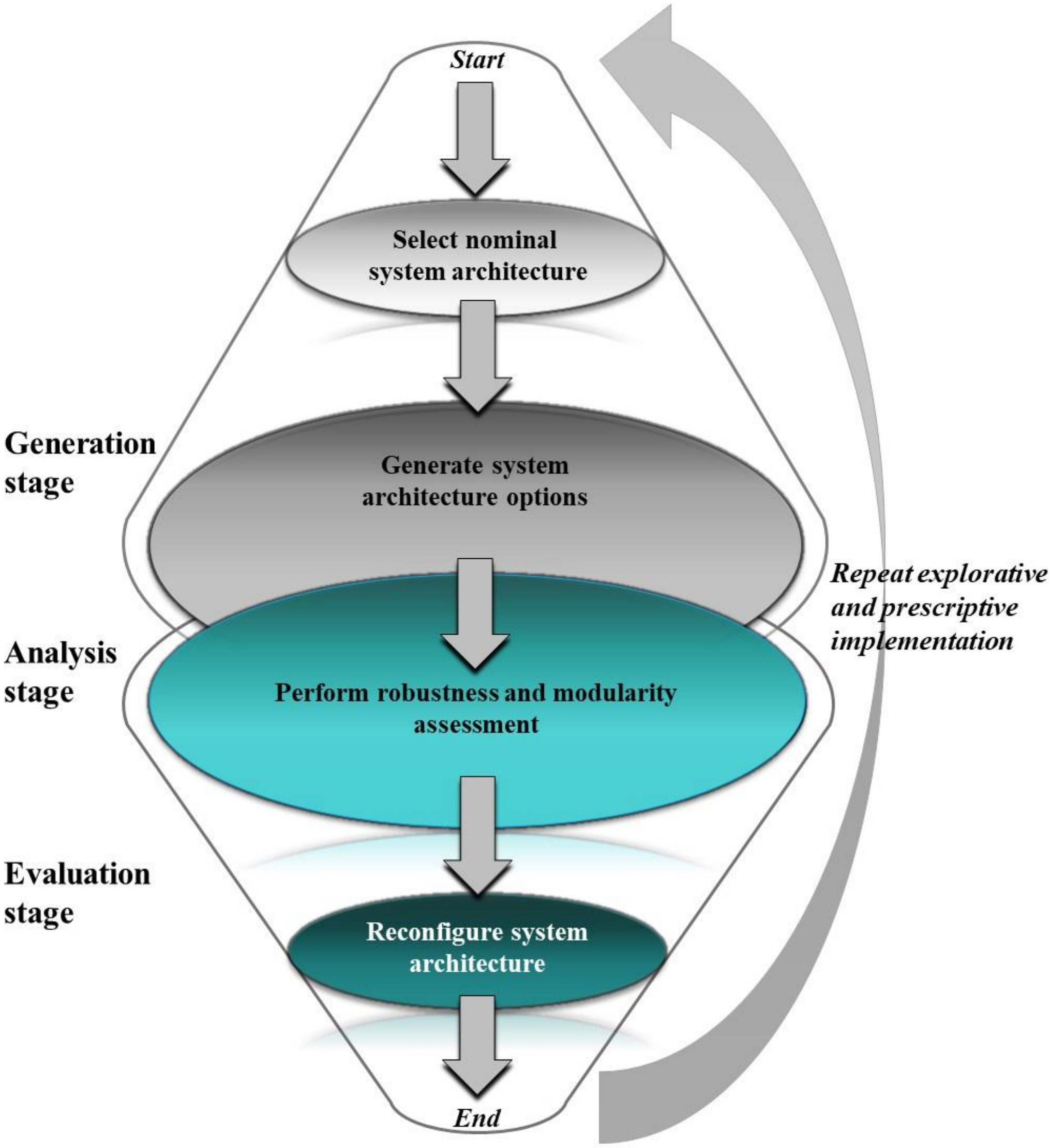


Figure 5: RoMoGA conceptual model

5.1.1 System architecture definition

There are three options for the definition of system architectures: descriptive, explorative and prescriptive. The descriptive system architecture generation is based on the user selecting an initial nominal system architecture, creation of system architecture options based on the user varying the redundancy through redesigning the number source components and connectivity between source and sink components to create new system architecture options (Chapter 6). The explorative system architecture generation is based on an initial nominal system architecture uses a network tool to create system architecture options. The user controls the network tool (Chapter 7) by varying the main structure patterns and parameters of hub components to create system architecture options that are variations of the nominal system architecture. The prescriptive generation the user redesigns the nominal system architecture based on the outcomes of the descriptive and explorative implementation of RoMoGA.

In this way, the system architecture modelled in the DSM, which is analogous to the adjacency matrix and represents a network, is effectively the iteration variable of the methodology and the input to the analysis stage.

5.2 Analysis stage

This stage consists of a nested modularity and robustness assessment.

5.2.1 Modularity assessment

Modularity is varied in the descriptive RoMoGA through a flexible modularity method that incorporates a tunable parameter[5] is employed for identifying different potential modular configuration options. In the explorative RoMoGA modularity is varied by changing the topological pattern and features of the system architecture options. The modularity assessment evaluates the readiness of the nominal and system architecture options to be modularised by calculating a normalised modularity metric. A modularity metric is calculated establishing a quantitative evaluation indicator to assess the level of potential modularity of the system architecture options.

---

[5] **Tuneable parameter** – a parameter that can be adjusted by the user of the methodology

5.2.2 Robustness assessment

The robustness assessment includes a robustness evaluation indicator metric that is calculated for a given system architecture option. A disruption is imposed on the system architecture by removing several nodes and edges from the network. Robustness is calculated post disruption based on the novel robustness evaluation metric.

A set of strategies for disruption are suggested that the architect can employ to assess robustness. There are two levels of disruption strategy. First is the component or module level disruption strategy applicable to descriptive RoMoGA, and the second the target and random disruption strategy applicable to explorative RoMoGA. A robustness assessment is performed by imposing different types of disruptions on the various system architecture options.

For each modularity and redundancy modified system architecture option, robustness assessment is performed. This allows the trade-offs between redundancy, modularity, and robustness to be analysed. At the end of the analysis stage, robustness assessment of the various system architecture options is established that informs the evaluation stage.

5.3 Evaluation stage

In the descriptive implementation of RoMoGA, the system architecture is evaluated based on the analysis phase findings to reconfigure it into a robust modular system architecture and examine the trade-offs between redundancy, modularity and robustness based on the system architecture options identified.

In the explorative implementation of RoMoGA, the objective of the evaluation phase is to identify options for system architecture that have topological features (i.e. patterns or hub parameters) that are desirable for redundancy, modularity, and robustness. The findings of the evaluation phase of the descriptive and explorative implementation of RoMoGA are suggested to supplement each other and subsequently inform the prescriptive definition of the system architecture in which the nominal system architecture is redesigned.

5.4 Chapter summary

This chapter has presented an overview of the developed RoMoGA methodology (Figure 5) that is detail outlined in Chapter 6 and 7. The RoMoGA iteratively generates system architecture options based on the iteration variable that is initialised as a nominal system architecture (modelled as a design structure matrix (DSM) that is analogous to an adjacency matrix representing a network).

The analysis stage incorporates a nested assessment of modularity and robustness. The RoMoGA has a descriptive, explorative, and prescriptive implementation. The following Chapter 6 describes in detail the different stages of the descriptive RoMoGA methodology whereas Chapter 7 presents the explorative RoMoGA that includes the novel network tool.

Chapter 6: Descriptive RoMoGA

Chapter 6 details the descriptive implementation of RoMoGA methodology. An overview of the methodology is given first and then three phases: generation, analysis and evaluation are discussed in detail. Figure 6 depicts the methodology and two simple network examples (Figure 7) are employed to guide the description of the different aspects of the methodology.

6.1 Overview

The generation stage of the methodology requires the following inputs for the nominal system architecture: 1) the description of the nominal system architecture as DSM, 2) the definition of system's main function, sub-functions, 3) the definition of the level of redundancy and 4) a definition of the sets of source and sink components corresponding to sub-functions. The nominal system architecture is modelled as a binary directional DSM that is analogues to a directional network. The directed network in combination with the source and sink represents the function viewpoint of the system architecture, whereas the edges of the network represent flows (energy, material, flow) and nodes of the source, sink and intermediate components. The flow network with its nodes identified as the source and sink components is termed the functional viewpoint.

The generation stage involves populating system architecture options with varying redundancy. The generation stage defines the system architecture that is assessed in terms of its modularity and robustness through the analysis stage. First, the nominal system architecture is assessed in terms of its robustness and modularity during the analysis stage. Then the generation stage is repeated by modifying the nominal system architecture and designing a new system architecture option of an alternative level of redundancy. This occurs through the redundancy loop of the methodology.

The analysis stage of the methodology consists of nested loops: modularity loop and disruption loop that encapsulate the modularity and robustness assessment. These loops implement a novel computational procedure that provides a quantitative approach to simultaneously examining modularity and robustness. This is done using a directed network generated by the transformation from a DSM into a mathematical adjacency matrix. The direct network is transformed into an undirected network that is the basis for the computations in the modularity loop. Given this representation, the computations are performed by the modularity loop and generate the classification of modules.

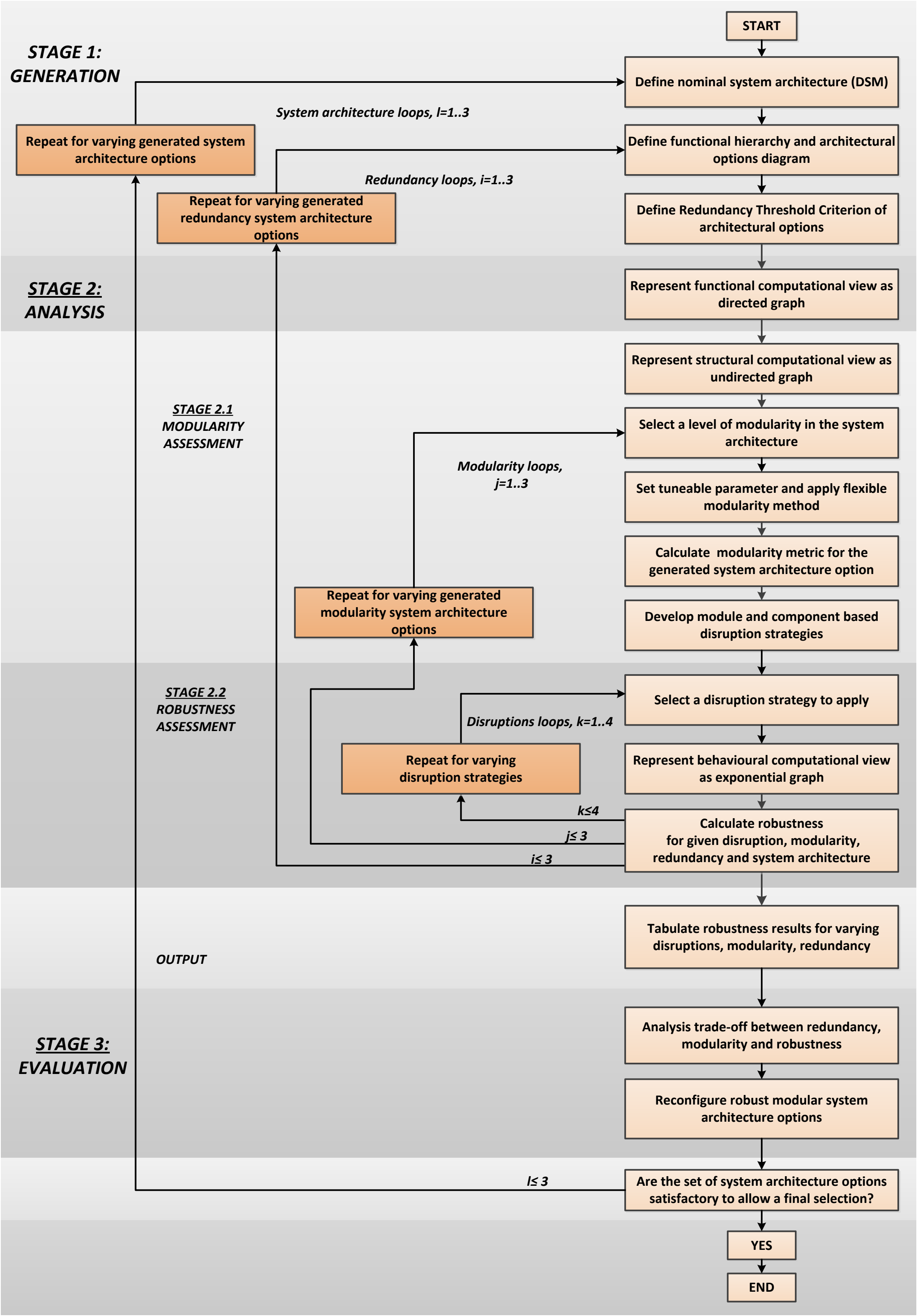


Figure 6: RoMoGA diagram

In the structural undirected network, the edges represent tangible structural connections amongst components and represent the structural viewpoint. The modularity loop uses a network science-based method to identify modules termed stability method, which is tuned by a resolution parameter, allowing the generation of different modular configurations corresponding to different levels of modularity.

The direct network is then transformed into an exponential matrix that is the basis for the robustness calculation. The robustness computations utilised an intact and disrupted exponential matrix with the identifiable nodes as source and sink. The direct network is transformed into an exponential matrix that forms the basis to perform the robustness computation in the disruptive loop. The exponential matrix represents the behavioural network, requires the definition of the source, and sink to enable the robustness calculation. The disruption simulations disrupt modules that correspond at different levels of modularity that are generated through the modularity loop. Component level disruption is also recommended, which can be used as a verification approach to ensure that the robustness calculation is corrected by testing that the inputs lead to rational outputs. Moreover, combined component disruption (i.e. all possible combinations of two or three components to be disrupted at the same time) offer a comprehensive overview of the system robustness and a reference point for comparison of the module' disruptions. Disruptions become a link between the structural and behavioural aspects of the system architecture.

The methodology continues iterating through redundancy loop, modularity loop, and disruption loop as per recommendations of Design of Experiments (DOE) or as an architect (user) prefers. If the user adopts the DOE approach, redundancy and modularity can be viewed as the independent variables and robustness the response variable. For example, redundancy and modularity can have three levels (low-medium-high). For a full factorial, DOE two independent variable for three levels suggests a total nine experimental run. This means the user will iterate three times over the redundancy loop and three types over the modularity loop and will calculate the robustness for all the possible level combinations (low-medium-high) of redundancy and modularity. The tabulated robustness results inform the evaluation stage of the methodology.

In the evaluation stage, a reformulated robust modular configuration of the system architecture for different levels of redundancy is suggested to be developed by the user based on the outcomes of the methodology. At this stage, the methodology recommends reformulating a robust modular configuration for a given a level of redundancy by combining robust modules identified for different levels of modular configuration.

The robust modular reconfiguration principle is that a robust module of higher modularity is preferable only if the module disruption does not penalise the robustness of the architecture. The methodology recommends that modules should be selected and gathered together to form a modular configuration on the basis that a module disruption does not compromise functional continuity, thus ensuring the robustness of architecture, rather than on maximising the degree of modularity of the architecture. One of the novelties of this research is that the robustness drives the selection of modules from the set of the various modular configurations and that different levels of redundancy can be considered.

The evaluation stage of the methodology focuses on supporting a decision-making process of the choices in a trade-off between the desired levels of modularity and redundancy. The evaluation stage allows the experts (user) preferences and input to be considered. The methodology explains the approach to formulate the robust modular configuration based on the analysis stage outputs. Design of Experiments (DOE) mean effects plot is also recommended to be developed to support the user to appreciate the trade-offs between modularity, redundancy, and robustness.

Finally, the user decides whether the results are satisfactory and concludes the descriptive implementation of RoMoGA. The results of the descriptive implementation of RoMoGA informs the explorative implementation of the RoMoGA, where new options for system architecture are then further explored adopting the novel network tool presented in Chapter 7.

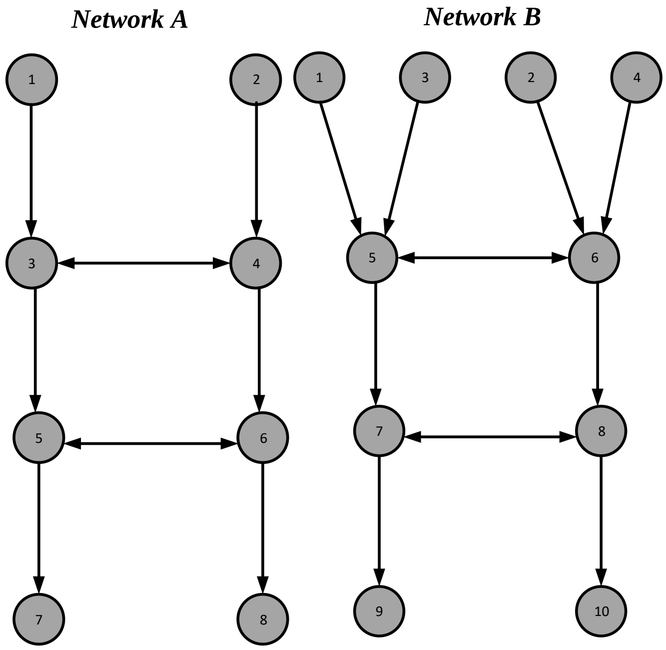


Figure 7: Network A and B examples

Two simple network examples are provided to support the explanation of the methodology as seen in Figure 7. These examples have similarities with the ship power distribution system, and they provide a preliminary exposition of the methodology and its constructs.

6.2 Generation stage

The generation stage of the methodology includes firstly the definition of the nominal system architecture represented as a DSM that is analogous to the adjacency matrix representing the network. The nominal system architecture is defined in a directed network that represents a functional network. The methodology transforms the defined functional network into structural and behavioural networks offering additional system architecture representations that allow for modularity and robustness assessment.

Also, the system architecture options of the nominal system architecture are generated by a redundancy loop that recommends the variation of the redundancy source components and the connectivity of the architecture.

6.2.1 System architecture definition

A binary DSM (Browning, 2015) is used to model the nominal system architecture which is the initial input for the generation stage of the methodology. The proposed methodology models the system architecture using three viewpoints: functional, structural, and behavioural. The functional network is required in the generation stage and forms the initial input. The structural and behavioural networks are required in the analysis stage, the structural network to calculate modularity and the behavioural network to calculate robustness. The structural and behavioural networks are discussed in Section 6.3.1.1. and in Section 6.3.3.1 respectively.

6.2.1.1 Functional network

A system architecture that is conceived during initial system design stages is represented as a schematic synthesis of functional elements and their interconnections modelled as a directed graph. Components of the system architecture correspond to functional elements and are represented as nodes, and functional flows (such as material or energy or flow) are the connections that are represented as directed edges. Figure 6 illustrates the point in the methodology, that the nominal system architecture is represented in a functional view as a directed graph.

Source and sink component identification

An initial input in the methodology is the definition of source and sink components. This is required in combination with the definition of the directed network to define the functional network. This is in line with the notion that a functional continuity is satisfied if sufficient connectivity between source and sink component exists after a disruption. Associating source and sink nodes of a directed graph with a function becomes the criterion that is assessed in the behavioural network (Section 6.3.3.1). Figure 8 the Networks A sources are remarked as the 1 and 2 nodes whereas sinks 7 and 8 nodes and B the sources are 1, 2, 3, 4 nodes and sinks 9 and 10 nodes.

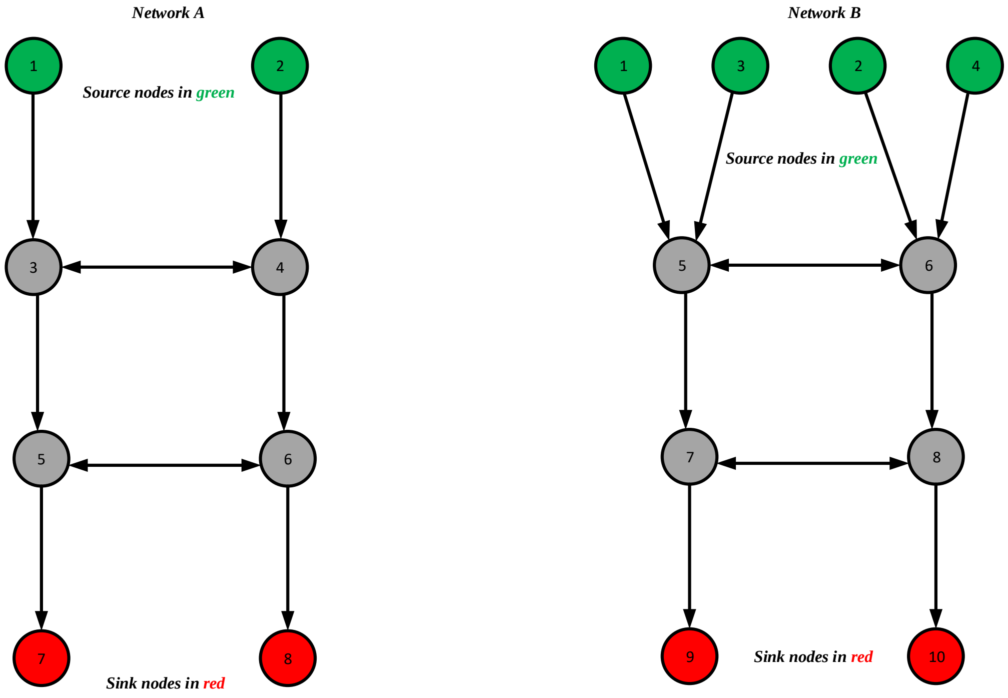


Figure 8: Network A and B examples definition source and sink nodes

6.2.2 Redundancy loop

The redundancy loop lies in the generation stage of the methodology. The redundancy level variations in the system architecture options are considered on the structural view of the architecture thus, varying the redundancy leads to generate new system architecture options. The redundancy in the system architecture is varied by the user using the redundancy threshold criterion that defines the required level of connectivity amongst source and sinks components to satisfy function, and by changing the number of the main supply/source components and number of interconnections which reflect on the architectural options. By varying the level of redundancy, a new system architecture option can be generated, which is then analysed in terms of its modularity and robustness in the analysis stage of the methodology.

6.2.2.1 Identify the source and sink components corresponding to a function

A set of the source (supply) and sink (demand) components that relate to a functional requirement are identified as input for the methodology as shown in Figure 6. This is required for the calculation of the robustness evaluation indicator metric (Section 6.3.3). The proposition is that the source components are required to maintain connectivity with sink components, to satisfy functional continuity after a disruption. With respect of the example presented in Figure 8 for Network Type A, if any of source {1,2} and sinks {7,8} are connected, it is assumed that the function is maintained at acceptable levels, and this is the same for Network Type B if any of the sources {1,2,3,4} and sinks {9,10}.

6.2.2.2 Determining the level of redundancy in the architecture

Two constructs are proposed to determine the redundancy of the architecture. The first is the redundancy threshold criterion (RTC) and the second is the functional hierarchy and architectural options diagram.

6.2.2.3 Redundancy Threshold Criterion (RTC)

The first construct proposed to control the level of redundancy in the system architecture is the Redundancy Threshold Criterion (RTC). The RTC indicates the ratio of connectivity amongst a set of sources and sinks components that are sufficient to satisfy a specific level of functional requirements in the system architecture. The RTC is derived based on the level of redundancy in the connectivity of the architecture and is considered as a design variable during the implementation of the methodology. RTC is, therefore, a way to vary the level of redundancy. The RTC is a threshold, measuring if the connectivity post disruption between a set of sources and sink components is sufficient to satisfy functional continuity. Thus, RTC is defined based on the sufficient level of connectivity which can support function. The RTC requires to be defined by the user and is expected to be different for different operational demands. In the methodology is suggested that the first iteration of the redundancy loop treats RTC as an input, the subsequence redundancy loops treat the level of redundancy as a design variable.

The robustness evaluation indicator discussed in Section 6.3.3. catalogues robustness values that only exceed the pre-defined RTC. For example, if a system has triple redundancy, the RTC is defined as 0.33, and for quadruple redundancy, it is defined as 0.25. By varying the RTC and recalculating the robustness evaluation indicator metric, the effects of redundancy on the robustness of the system can be determined.

For Network Type A, the RTC was defined as 100% amongst the individual set of source and sink, meaning that source {1} should remain connected 100% with sink {7} or source {2} should remain connected 100% with sink {8}). For Network Type B, the RTC was defined as 50% amongst the set of source and sink meaning that for the set of source {1, 2} connected with sinks {9} is required only 50% connectivity to achieve the satisfactory function, respectively RTC is 50% connectivity is required amongst the sources {3, 4} to sinks {10} to satisfy function.

6.2.2.4 Functional hierarchy and architectural options

A second construct proposed to control the redundancy of the system is the functional hierarchy and architectural options diagram. This captures the high-level of redundancy that entails the design of alternative architectural options to satisfy sub-functions. In complex system architecture, there is often a hierarchy of functions that can be satisfied through various architectural options. The architectural option involves a set of sources and sinks components and an RTC that is required to meet a specific level of functional requirements. Alternative architectural options that involve different components and connections may exist in complex engineering systems. This is illustrated in Figure9 that architectural options are characterised as “AND”, “OR”.

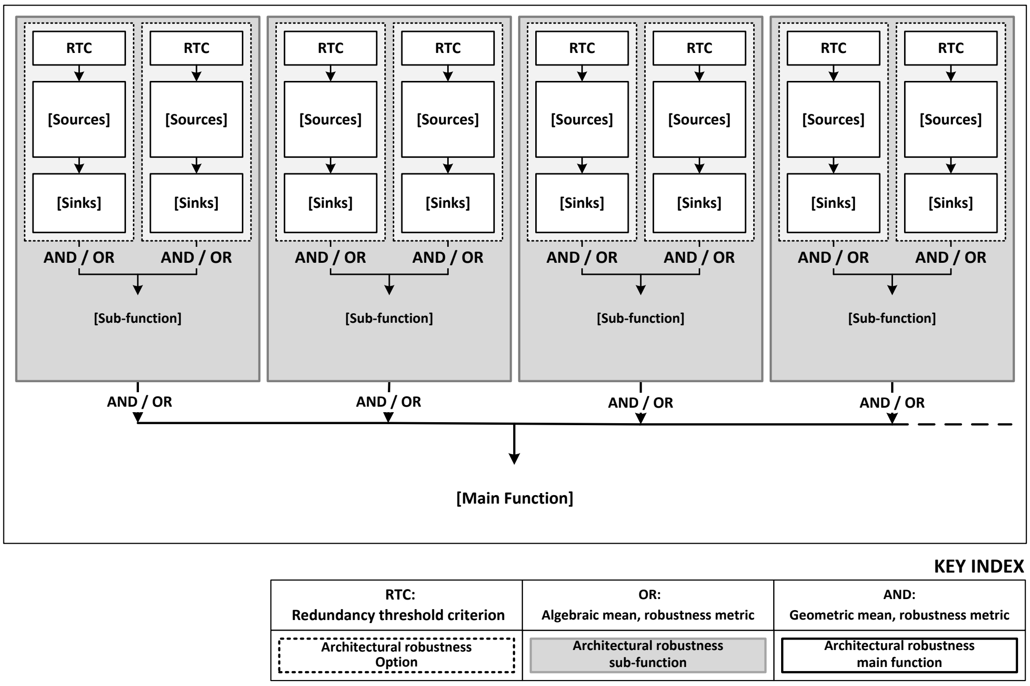


Figure 9: Functional hierarchy, architectural options, and RTC

Architectural options are viewed as a segment of the architecture that individually can satisfy a sub-function. If more than one architectural option is designed to satisfy the same sub-function that constitute as an architecture option (“OR”) designed for redundancy in the architecture.

Figure 9 reflects on the functional hierarchy and architectural options illustrating how they are designed in the system. This hierarchy of main functions and its sub-functions, and the definitions of “AND”, “OR” should be defined by the users. The functional hierarchy and architectural option diagram allow the investigation of the system's high-level redundancy as diagram changes lead to a new system architecture option. The user can, therefore, update the functional hierarchy and the architectural option diagram to change the redundancy and create new system architecture options. In summary, the RTC reflects redundancy concerning additional connectivity between the source and the sink components, while the architectural options reflect redundancy concerning the design of the additional set of sources and sink components in the system.

Network A has two architectural options: option 1 is the set of source and sink {source: 1, sink: 7} and architectural option 2 is {source: 2, sink: 8}. Network A has four options: {source:1, sink:9}, {source:2, sink:9}, {source:3, sink:10}, {source:4, sink:10}.

6.3 Analysis stage

The analysis stage of the methodology constitutes modularity loop and disruption loop as illustrated in the analysis stage of Figure 6. The following sections elaborate on the modularity, and robustness loops of the methodology.

6.3.1 Modularity loop

The following paragraphs outline the modularity loop aspects, i.e. the structural network, the network-science stability method and the modularity metric.

6.3.1.1 Structural network

An undirected network represents the structural connectivity among components and is generated by transforming the flow network (which is constructed based on the DSM) into being undirected. This undirected network represents tangible physical connections such as cables and pipes. In the methodology, the initial input is defined to be a flow-directed network that is mathematically transformed into an undirected and symmetrical structural network in the analysis stage. In case that the flow connections do not correspond to tangible structural connections, such as the flow of information across a wireless network, then a structural network should also be included as an input by the user. In analysis stage Figure 6 depicts the point that the system architecture is represented in a structural undirected network which indicates the start of the modularity loop.

6.3.1.2 Stability method

In network science there two modularity approaches clustering methods and community detection methods. Community detection methods, unlike clustering methods, do not require a given number of clusters as input. Communities are typically undefined, can be of an asymmetric size and density and have hierarchies (Fortunato, 2010; Martelot and Hankin, 2013). The stability method (Delvenne et al., 2008) is a community detection method used in this study because of the inclusion of a resolution parameter. This allows the consideration of modularity as a design variable in this study. This enables the degree of modularity to be varied through the manipulation of this parameter.

Various network science approaches were investigated (Arenas et al., 2008; Blondel et al., 2008; Newman, 2006; Reichardt and Bornholdt, 2006) before selecting the stability method (Delvenne et al., 2008).

Criteria for selecting the method for identifying modules were:

1. Does the approach require an input number of modules?
2. Does the approach include a resolution parameter?
3. Does the approach generate oversize modules?
4. Does the approach identify smaller size modules?
5. Does the approach computational efficient and suitable for the analysis stage of the methodology?

Preliminary experimentation using the above-mentioned methods was performed based on pilot technical examples. Through iterating, and checking the performance of the codes and their suitability to be incorporated into the proposed methodology, the stability method as defined by Delvenne et al. (2008) and computationally implemented by Martelot and Hankin (2013) was selected as it outperformed the other methods. The stability method does not require an input number of modules, includes a resolution parameter, does not generate oversized modules, and can generate smaller modules and is computationally efficient and effective for implementation in the MATLAB code.

The stability method formulates a quality function that captures the persistence of clusters in time (Delvenne et al., 2008). A cluster is persistent about a random walk (a walk in a network is a series of edges, not necessarily distinct) after t time steps if the probability that the walker escapes the cluster before t steps is low. The probability is computed via the clustered auto-covariance matrix R_t, which, for a partition of the network in c clusters, is defined in Equation (2):

$$R_t = H^T(\Pi M^t - \pi^T \pi)H \qquad (2)$$

Where H is an $n \times c$ membership matrix where element H_{ij} equals one, if node i is in cluster j and is zero otherwise; M is the transition matrix of the random walk; Π the diagonal matrix whose elements are the stationary probabilities of the random walk:

$$\Pi_{ii} = \frac{k_i}{2m} \qquad (3)$$

With k_i: the degree of node i; π is the vector whose entries are the diagonal elements of Π. Thus, the quantity $(R_t)_{ij}$ expresses the probability for the walk to start in cluster i and end up in cluster j after t steps, minus the stationary probability that two independent random walkers are in i and j.

In this way, the persistence of a cluster i is related to the diagonal element $(R_t)_{ii}$.

Delvenne et al. (2008) defined the stability of the clustering is defined by equation (4):

$$r(t;H) = \min_{0 \le s \le t} \sum_{i=1}^{c} R_{s\,ii} = \min_{0 \le s \le t} trace[Rs] \qquad (4)$$

For a given t, the aim is to maximise the stability. For $t = 0$, the most stable partition is that in which all vertices are their cluster while for $t = 1$, maximizing stability is equivalent to maximizing Newman–Girvan modularity.

Time can be considered as a resolution parameter, and the advantage of this method is that the resolution can be tuned by treating time as a continuous variable, delivering a multiresolution version of modularity (Fortunato, 2010).

The specific version of the algorithm used in this methodology was the fast multi-scale detection of communities using stability optimisation as described by Martelot and Hankin (2013). This method uses a resolution parameter to tune the algorithm, to identify modules that are not necessarily maximising modularity but correspond to different levels of modularity. Another reason of using this method was that it does not involve defining in advance a specific number of clusters (Fortunato, 2010; Martelot and Hankin, 2013) and that was found in practise to generate logical modules consistent with the expert's opinions.

For the examples of Figure 10, using the stability method for Network A, allowed four modules to be identified: Module 1 (2, 4 nodes); Module 2 (1, 3 nodes); Module 3 (5, 7 nodes); and Module 4 (6, 8 nodes). For Network B also four modules were identified: Module 1 (1, 3, and 5 nodes); Module 2 (2, 4, and 6 nodes); Module 3 (7, 9 nodes); and Module 4 (8, 10 nodes).

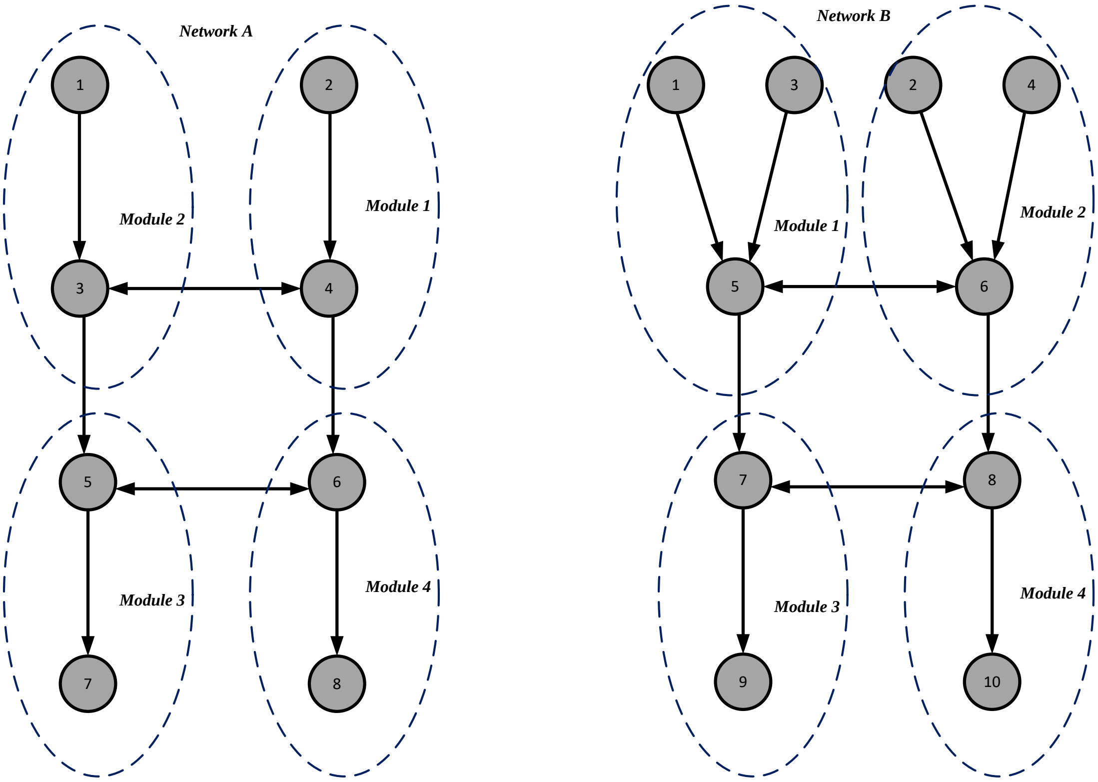


Figure 10: Network A and B modules examples

6.3.1.3 Modularity metric

Magee et al. (2010) suggested a normalised Newman modularity metric to allow fair comparison among different size networks. A parameter f is introduced equation (5):

$$\mathrm{f} = 1 - \frac{(\mathrm{k}-1)}{\mathrm{m}} \qquad (5)$$

where k is the number of modules and m is the total number of edges of the network. The normalised version of the Newman modularity metric is given by equation (6):

$$Q_n = \frac{\sum_{i=1}^{k}(e_{ii} - a_i^2)}{f - \sum_{i=1}^{k}(a_i^2)} \qquad (6)$$

Since the normalisation of the metric allows comparisons among architectures of different sizes it is the one selected to be incorporated in the proposed methodology of this study.

The metric is used to assess the level of modularity of system architecture, and it requires as input the modular configuration and the undirected network (symmetric and binary adjacency matrix). Network A has the degree of modularity calculated as 0.67, whereas for Network B it was 0.77.

6.3.1.4 Structural classification of modules

Structural classification of modules is a concept proposed to categorise generated modules of a modular classification into central, peripheral, or semi-peripheral. The development of the classification is suggested that may allow additional characterisation of the modules.

The network theory concept of eccentricity was adapted to suggest a classification for modules. The node eccentricity v is the greatest distance between node v and any other node. The diameter d of a network is defined to be the maximal value of node eccentricity and the radius r is the minimum value. A central node in a network of radius r is one whose eccentricity is r and a peripheral node is any node with eccentricity equal to the diameter d (Estrada and Knight, 2015).

The concept of eccentricity was adapted to classify modules by determining the smallest and largest diameters of each module, and as shown in Table 7, modules are categorised into central, semi-peripheral and peripheral classes.

Table 7: Classification of modules

Classification of modules		
Central	**Peripheral**	**Semi-peripheral**
Modules' diameter equals the smallest diameter amongst the modules.	Modules' diameter equals the largest diameter amongst the modules.	Modules that are neither central nor peripheral

Due to the simplicity of Network A, the modules were classified as central and periphery. For Network B, Module 1 (1, 3, 5 nodes) and Module 2 (2, 4, 6) were classified as central modules, whilst Module 3 (7, 9 nodes) and Module 4 (8, 10) were classified as periphery modules.

6.3.2 Disruption loop

The objective of the methodology is to modularise the architecture in a way that any single module disruption does not stop the system functionality thus does not jeopardise the robustness of the architecture. The disruption loop aid on identifies how the disruption of the generated modules will affect the robustness to the architecture. In this way, the disruption loop is informed by the modularity loop and includes the robustness computations. The disruptions are required inputs to calculate the robustness. Disruptions involved the removal of components and connections of the system architecture. Disruptions are considered as generic (without knowledge of the cause).

There are two levels of disruption strategy: component or module level disruption strategy. The component and module disruption strategies were conceived based on the industrial and literature motivations. The component level disruption provides an approach to enumerate all the possible combinations of a set of components that can cause a total system loss. It is proposed that component disruption may be used as an approach to verify the calculation of robustness. The module disruption strategy is devised to assess the robust behaviour of modular system architectures.

In this methodology, the focus is the disruptions of modules. Disruption can relate to damage of several or all components within a module due for example to an explosion, fragmentation, fire or any external or internal cause that lead to losing a module. For example, in naval ships, survivability viewpoint is expected that a single disruption event can result in the loss of several components and connections at once. If the system architecture is modularised, then a single disruption event may consequence with the loss of a module. Module disruption is simulated within the methodology by removing the nodes and edges within the exponential matrix. The probability that disruption will occur, or the type of disruption, are not specifically investigated in this study.

The main disruption scenario of the methodology which involves each module within the modular configuration to be consecutively disrupted. The structural classification of modules can inform other potential disruption scenarios such a scenario that only central or peripheral or semi-peripheral modules within the configuration being disrupted. The formulation of disruption scenarios formulated basis on the classification of modules is suggested that may provide additional insight to the role of classes of modules in system architecture and their effects on the robustness of the system architecture.

The disruptions are formulated based on the generated modules from the stability method for the different levels of modularity by the modularity loop. That means that for different modular configuration in the system architecture different disruptions are developed. Disruptions provide a basis for assessing how different modular boundaries affect the robustness of the architecture, helping to select an appropriate robust modular configuration at the subsequent evaluation stage.

In general, there are wide ranges of potential disruptions that must be considered in the design of a system. The effect of any disruption will depend on the architecture, which it is applied. The purpose of the analysis stage of the methodology was to link disruptions with a modular architecture, addressing the research objective 2 that is “develop a methodology to combine the assessment of modularity and robustness in a system architecture”.

6.3.3 Robustness evaluation metric

The engineering systems notion that a function can be satisfied if a set of source and sink components of the flow network maintain sufficient connectivity after a disruption is accepted in order to formulate the robustness evaluation indicator metric. Shanthikumar (1987) stated that: “the system functions if there is a connection from the source to the sink through functioning components”. Goodrum et al. (2018) also employed network science and explained that “the undirected and directed connectivity analyses identify if each component's demand is satisfied by checking if paths exist from the demand source to that component in the presence of damage on a single layer”. If there is a path between the source and the demand (sink), component it is considered that the functions will continue. If there are no paths, there is a loss of functionality.

The robustness evaluation metric was developed with the objective that should be capable of capturing robustness on the basis of the following criteria: capturing robustness after disruption, considering directionality and indirect connections, and identification of the source and sink components.

6.3.3.1 Behavioural network

A behavioural network is modelled as an exponential matrix, which captures direct and indirect connections to assess the robustness of the system architecture. The exponential matrix models the connectivity among components directly and indirectly (at any number of steps of connectivity). This means that the exponential matrix becomes the mathematical basis to assess robustness by determining if the source and sink components remain connected (in any number of steps). The sufficiency of the level of connectivity amongst source and sink components is evaluated through the redundancy threshold criterion (RTC). In the disruption loop shown in Figure 6 denotes the stage that the system architecture is represented in a behavioural network.

6.3.3.2 Measuring connectivity among source and sink nodes before and after disruption

For a system architecture that is represented by a directed network G, and has an adjacency matrix A, a new matrix S is constructed, a binary matrix that catalogues the existence of paths/walks (of any length) between nodes in the network. A mathematical way to compute *S* is to first compute the matrix exponential e^A.

Estrada and Knight (2015) stated that an entry of the exponential matrix is nonzero if and only if there is a path in the network between nodes i and j. We then form a matrix *S* using the identities:

$$S_{ij} = \begin{cases} 1, & (e^A)_{ij} > 0, \\ 0, & \text{otherwise.} \end{cases} \qquad (7)$$

S can be interpreted as determining the structurally driven behavioural robustness of the system architecture.

Sets of e sources $\mathrm{s} = \{\mathrm{s}_1, \mathrm{s}_2, \mathrm{s}_3, \dots, \mathrm{s}_\mathrm{e}\}$ and k sinks $\mathrm{t} = \{\mathrm{t}_1, \mathrm{t}_2, \mathrm{t}_3, \dots, \mathrm{t}_\mathrm{k}\}$ are chosen in system architectures, and the number of 1s in the corresponding intersection of rows and columns of *S* is computed. The proportion of 1s gives a measure of the interconnectivity between the source and sink nodes of the network. This can be recalculated after the network is subject to disruption (i.e. loss of nodes or edges). The measure works equally on directed and undirected networks. More precisely, the robustness R_{st} of the intact system architecture is measured using equation (8):

$$R_{s,t}(G) = \frac{\sum_{i=1}^{e} \sum_{j=1}^{k} S_{ij}(s_i, t_j)}{ek} \qquad (8)$$

A disruption (loss of nodes/edges) generates a damaged system architecture represented by a network G' with adjacency matrix A'. Robustness is recalculated as per equation (9) by evaluating the connectivity among source and sink nodes after disruption to give:

$$R_{s,t}(G') = \frac{\sum_{i=1}^{e} \sum_{j=1}^{k} S'_{ij}\,(s_i, t_j)}{ek} \qquad (9)$$

The equation (9) is computed for all non-empty subsets of S. Equation (9) is compared against the threshold redundancy criterion (RTC) generating the robustness evaluation indicator values, which if greater than the RTC are recorded as successful and values less than RTC are recorded as failures (zero value).

6.3.3.3 Average of a weighted combination of operational sources

If the source components of the system architecture are designed to be functional only in the context of specific operational scenarios, the robustness evaluation metric could be weighted to also consider possible combinations of operational sources.

Thus, for e number of source components, all $2^\mathrm{e} - 1$ combination of operational sources are tested (excluding the case of all sources unavailable as that inevitably leads to loss of functional continuity). For ease of use, the information contained in the individual values of the robustness is condensed into a single term. This is achieved by calculating a weighted average where the robustness of operational sources R_i is weighted by a value that is proportional to the number of states with i sources operational and is represented by the reciprocal of the binomial coefficient $\mathrm{w_i} = \frac{(\mathrm{e-i})!\mathrm{i}!}{\mathrm{e}!}$. Alternative experts' input basis on the design of the system and the expected operational scenarios could also be used to define the weight values (w_i). Therefore, the weighted robustness R_w is calculated as per equation (10):

$$R_w = \frac{1}{e}\sum_{i=1}^{e} w_i R_i. \qquad (10)$$

6.3.3.4 Total robustness evaluation indicator metric

Figure 9 presented in Section 6.2.2.4 illustrates the functional hierarchy and the architectural options diagram which help to calculate the overall robustness of the system architecture. The robustness evaluation indicator metric shown in equation (9) is, therefore, calculated for each architectural option.

At the sub-function level, the robustness is calculated in the following way: If a sub-function is designed for redundancy of architectural options (OR) then the robustness of sub-function is calculated through an algebraic mean. If sub-function is not designed for redundancy the architectural option (AND) the robustness of the sub-function is calculated by the geometric mean.

Finally, the robustness of the below equation (11) relates to the main functionality of the system as a whole. The geometric mean of the sub-function robustness values to generate an overall robustness measure for the whole system is used, since if any individual value is zero (which means the relevant sub-function has failed) then the overall robustness is then automatically equal to zero, given sub-functions are interdepended to satisfy the main function. For a system with interdependent functions enumerated $l = 1, \dots, q$ with corresponding robustness $R(l)$ the total robustness is given by

$$R_{total} = \prod_{l=1}^{q} \sqrt[q]{R_{s,t}(l)} \qquad (11)$$

Equation (11) is applicable for sub-functions that must at the same time be in operation for the main function to be satisfied, thus interdependent functions. It is noted that a component could simultaneously be a source for a Type A sub-function (e.g. source cooling pump for cooling) and a sink for a Type B sub-function (e.g. sink cooling pump for power). By using a directed (flow) network, in combination with the robustness evaluation indicator metric that is capable to classify the same components as a source or sink in the network, interdependencies among the sub-functions can be captured basis on the Equation (11).

6.4 Evaluation stage

At the end of the analysis stage, robustness results are tabulated which provide an overview of the robustness of a set of different system architecture options at different levels of redundancy and modularity. The results of the stage of analysis become inputs of the stage of evaluation. The evaluation stage of the methodology includes two approaches to the reformulation of robust modular configuration and a DOE analysis of the results.

A manual process of reformulating the robust modular configuration, to allow an expert's preferences to be input is suggested. The robust modular configuration is formulated based on the outcome results of the analysis stage, following the principle that a disruption to a module should not stop the functional continuity of the architecture and that maximum modularity is favoured only if is possible. The first step on the reformulation is to select the generated modules from the maximised modular configuration. However, the maximisation of the degree of modularity may generate non-robust modules.

The second step is to search on substituting the non-robust modules with robust modules generated in the medium modular configuration. If a robust module is not found in the medium configuration, then the low-level configuration is investigated. If in the low-level configuration, a robust module is found, then two approaches are suggested to be employed. The first approach is to perform, additional iterations through the modularity loop by manipulating the resolution parameter of the stability method in different values until a substitutable robust module is identified. The second approach is to manually intervene to update a specific non-robust module by removing one by one its components and calculating the robustness until a suitable robust module is identified. Through these approaches, the architect can devise a final robust modular configuration for the system architecture.

Through the application of the proposed methodology on the Networks A and B of Figure 10 following module is being found: for Network A, Module 1 contains 2 and 4 nodes and Module 2 contains 1 and 3 nodes, Module 3 groups 5 and 7 nodes and Module 4 clusters 6 and 8 nodes. For Network B, the methodology proposed module 1 to contain 1,3,5 nodes and Module 2 to contain 2,4,6 nodes, Module 3 to group 7 and 9 nodes and Module 4 to cluster 8 and 9 nodes. The generated modules were found to be robust. Whilst an experienced engineer could find such robust modules in this simple power system without computational tools, this is challenging for increasingly complex systems with interconnected subsystems. The methodology provides both a verifiable means to track decisions made during initial system design stages and an objective and systematic approach to engineering systems design.

The second approach suggested at the evaluation stage is the calculation of mean effects[6] diagram, which is part of the analysis used in the Design of Experiments (DOE) analysis. It is proposed to treat modularity and redundancy as two design variables (independent variables) of the system architecture and to use robustness as a response variable (depending variable). If the methodology is used through a full-factorial DOE all possible combinations of variable levels can be considered e.g. two design variables that have three-level (low, medium, high) leading to nine experimental runs. After collecting the results of robustness for all possible combinations of redundancy and modularity in the system architecture, the main effects of the design variable on the response variable can be calculated, in this case, the effects of modularity on robustness and the effects of redundancy on robustness. The main effects diagram can provide the architect with a visual outcome to better appreciate the relationship between redundancy, modularity, and robustness of the system architecture under review.

This provides insights to guide architects and decision-makers in the design of system architecture. The final stage of the methodology involving selection is not further discussed here, as it is a qualitative process that depends on the decision-making of the users.

6.5 Chapter summary

Chapter 6 outlines in detail the descriptive implementation of RoMoGA methodology, which starts with nominal system architecture and generates, and analysis of the system architecture options for different redundancy, modularity, performing robustness assessment. The descriptive methodology of RoMoGA encourages the architect to consider different system architecture options and to consider the trade-offs of redundancy and modularity on robustness.

The outcome of the evaluation stage is the reformulated robust modular system architecture and insights into redundancy and modularity trade-offs on robustness. The results of the evaluation stage are expected to be qualitatively reflected and interpreted by the users. The results of the evaluation stage are intended to assist the user in designing new system architecture concepts for the nominal system architecture. In addition, the evaluation results

---

[6] **Main effect** - the effect of an independent variable on a dependent variable averaged across the levels of any other independent variables.

are suggested to inform the explorative RoMoGA (Chapter 7) where the user can consider ways to redesign and improve more broadly.

The next Chapter presents the explorative implementation of RoMoGA that integrates a novel network tool that enables to wider the generation of system architecture options in the design space. Then, Chapters 8 and 9 present case studies on naval system architectures and the experiments with network tool, demonstrating the industrial and explorative application of the proposed RoMoGA methodology.

Chapter 7: Explorative RoMoGA

This Chapter outlines the explorative implementation of RoMoGA methodology that incorporates a novel network tool that allows the generation and analysis of system architecture options. The generation stage of the methodology outlines the development of a novel generator that enables the user to develop various potential system architecture options. The generator combines knowledge of network science, engineering systems, and heuristics. Its parameters allow experimentation with patterns to define the main and hub structure, vary the number, size, and connectivity of hub components, define the source, and sink components at the hub level and adapt a redundancy threshold criterion. The evaluation findings of the explorative RoMoGA implementation are suggested to be used by the architect to support improvements, redesigns, or development of novel designs during a perspective implementation of RoMoGA.

7.1 Preliminary exploration

Prior to the RoMoGA explorative application, it is recommended that the user perform a qualitative or quantitative characterisation of the nominal system architecture discussed in Chapter 5, in order to identify dominant topological features that can aid to populate analogous new system architecture options using the network tool. The user can define the size of the network and the main structural pattern in such a way that the newly developed system architecture options share similarities with the nominal system architecture.

Network tool preliminary experiments can also be carried out to help the user set up the boundaries of the experimentation. After the preliminary investigation, the boundaries of the explorative RoMoGA application can be established, and the user can focus on experimenting with aspects of the hubs, i.e. number, connectivity, density, in order to explore the different options of the redundancy in the system architecture. The redundancy investigation in the explorative RoMoGA is different from the descriptive RoMoGA, whereas the focus here is on the aspects of the hub components. These guidelines are provided to guide users on how to use a network tool in the context of RoMoGA's explorative implementation. The user may, however, choose to use the network tool in more initiative-based ways and out of the context of the RoMoGA methodology, as it is a stand-alone tool.

7.2 Generation stage

The generation stage of the explorative implementation of RoMoGA methodology incorporates a novel network tool that is capable to generate alternative system architecture options. A network tool is a computational tool that parameterises a nominal system architecture allowing experimentation with different system architecture options through a variety of topological features. The following sections outline key concepts on the development of network tool, and then the network tool is described.

7.2.1 Key concepts of the network tool

In the following paragraphs, the key concepts that support the development of the network tool in this study are discussed.

7.2.1.1 Patterns

Alexander (1964) notably identified the concept of patterns as "the idea that is possible to create such abstract relationships one at a time, and to create designs which are whole by fusing these relationships". Minai et al. (2006) stated that "pattern formation is an integral part of most complex systems' functionality". A pattern is " reoccurring structure within a design domain" (Maier and Rechtin, 1997). Various studies have adopted and discussed patterns in engineering design literature. Rivkin and Siggelkow (2007) explored patterns for decision-making in complex systems and Selva et al. (2016) proposed using patterns in system-architecture decision-making. Hölttä-Otto et al. (2012) used patterns to study various modularity metrics, given that the degree of modularity of the various patterns could be anticipated. Min et al. (2016) used engineering patterns to study structural complexity, while Sharman and Yassine (2004) discussed various patterns to characterise complex product architectures, and Yassine and Naoum-Sawaya (2016) employed patterns to examine architecture, performance, and investment in product development networks. Baldwin et al. (2014) proposed a methodology to identify hidden patterns in system architectures that was based on a directed network that allows the identification of direct and indirect dependencies. In their approach, they defined three fundamental key patterns: core-periphery, multi-core, and hierarchical (Baldwin et al., 2014). For the engineering design process, Parraguez et al. (2015) used expected patterns characterised by network metrics and compared with empirical data to analyse the flow of information through a complex project. In the field of applied engineering, in the context of early-stage ship design, Chalfant and Chryssostomidis (2017) proposed using patterns to provide "a high-level view of a system or portion of a system, consisting of a connected set of functional blocks plus general design rules and guidelines of creating a system”. For cyber-physical systems resilient design (Tomiyama and Moyen, 2018), it was suggested that controllers, sensors and actuators should form a mesh topology and that subsystems be separated from other subsystems to the greatest extent possible.

Table 8 presents various patterns found in the engineering literature. The generator developed in this study uses a set of theoretical engineering patterns to define the main high-level and hub low-level structure of populated network system-architecture patterns.

Table 8: Patterns in the engineering-design literature

Reference	**Type of patterns**
(Sharman and Yassine, 2004)	Simple bus, multiple-bus, auxiliary or weak or subsidiary buses, planar triangular clusters/tetrahedron of clusters/tree-level design hierarchy.
(Yassine and Naoum-Sawaya, 2016)	Random, diagonal, block-diagonal, local, hierarchical, depend, small world, scale-free patterns
(Hölttä-Otto et al., 2012)	Integral, bus-like, modular
(Min et al., 2016)	Integral pattern/linear-modular architecture/bus modular pattern
(Rivkin and Siggelkow, 2007)	Random, local, small world, block-diagonal, preferential attachment, scale-free, centralised, hierarchical, diagonal, dependent
(Glazier et al., 2015)	Ring, mesh, star, fully connected, line, tree, bus
(Selva et al., 2016)	Combining, assigning, partitioning, down-selecting, connecting (bus and star, ring, mesh, tree), permuting

7.2.1.2 Hub and Source/Sink concepts

This section presents the topological features of system architectures that were developed in the generator. Key components can be characterised as hubs, sources, or sinks, depending on their function in the system architecture. This characterisation of components at the initial stages of design can provide high-level information about the role of the component in the system architecture, without the need for details. By designing an architecture referring, for example, to a hub component instead of a switchboard, or a source component instead of a generator, a generic hub–source-sink architecture can be devised, allowing the details to be decided at later stages. In short, hubs, sources, and sinks are used in the present study to offer a high-level characterisation of the components of system architectures and are therefore incorporated in the development of the novel generator.

Hubs

Across various system-related fields, it is acknowledged that the presence of hubs. Hubs are components that have a high number of connections. Sosa et al. (2011) explain that “complex engineered systems tend to have architectures in which a small subset of components exhibits a disproportional number of linkages”, such components are known as hubs. Studies of hubs in engineering systems based on network science include (Braha, 2016; Braha and Bar-Yam, 2007, 2004a; Mehrpouyan et al, 2014; Sosa et al, 2011; Piccolo et al. 2018). Sosa et al. (2011) studied the effects of hubs on quality (measured as the number of defects) based on empirical data. They found that there is an optimum number of hubs that will give a positive influence on quality, highlighting the importance of the management of hubs during the early design stages.

Various studies (Braha and Bar-Yam, 2004a; Piccolo et al., 2018) found that the engineering design process exhibit right-skewed degree distribution, meaning a network with few high degree nodes while the remaining nodes have low degree nodes. In network science, scale-free networks with right-skewed degree distribution are found to be highly robust against random disruptions, however vulnerable to failure in the event of target disruption (Albert et al., 2000). Similar findings in engineering design literature were confirmed for large-scale engineering design projects involving activities and people (Braha and Bar-Yam, 2007; Piccolo et al., 2018).

The existence of hubs relates also to other network properties such as disassortativity and rich-club. Braha (2016) found that “complex engineering networks tend to be uncorrelated or disassortative”, explaining that this characteristic “decrease the likelihood that the project spirals out of control”. Rich-club phenomenon characterises networks with interconnections among its hubs. Zhou and Mondragón (2004) investigated internet topologies, claiming that “the connectivity between rich nodes can be crucial for network properties, such as network routing efficiency, redundancy, and robustness”. Findings from the literature agree on the focus on hubs for the development of improvement approaches (Piccolo et al., 2018).

The definition of a hub suggested in this study is that a node is a hub if its degree is proportionally greater than the average of its neighbours. The notion of defining hubs as high-degree nodes when compared to neighbours is different from the definition adopted in scale-free networks with a power-law degree distribution that suggests that relatively high-degree nodes are automatic hubs.

It is argued the importance of a hub is due to its position in the neighbour of a network and cannot be directly inferred from the degree of distribution. As the hub is a key feature of systems, it was therefore chosen as a parameter in the development of the network tool described in this study.

Source and sink

A source component supplies energy, material or information to the sink components and is a feature of flow systems. The source and sink concept is used in engineering in reliability, robustness, and resilience work (de Vos and Stapersma, 2018; Pumpuni-Lenss et al., 2017). Max flow optimisation problems use the concept of the source and sink (Pumpuni-Lenss et al., 2017) in improving the resilience of complex engineering systems, as well as studies of reliability (Chao et al., 1995). Goodrum et al. (2018) considered "directed connectivity analyses, as well as sink– source flow analyses". Chalfant and Chryssostomidis (2011) assessed electrical flow by employing the max-flow, min-cost algorithm. In the max-flow problem, sources supply a flow whereas sinks demand flow and that the aim is "to transport all the flow from the supply nodes to the demand nodes at minimum cost" (Selva et al., 2016). In terms of reliability, the literature suggests that "the system is functioning if there is a connection from source to sink through working components" (Chao et al., 1995; Shanthikumar, 1987). Moreover, hubs are noted as components between source and sink in distribution system architectures (de Vos and Stapersma, 2018).

The source and sink concept were found to be present in the distributed system architectures and was adopted to be included in the development of the generator. In summary, theoretical patterns, hubs, sources, and sinks are the key concepts used in the development of the novel generator

7.2.2 Network tool description

The research presents a novel network tool, developed from the relevant literature on engineering design and network sciences and with empirical knowledge of distributed engineering systems. The generator does not aim to produce feasible solutions, but rather, theoretical complex patterns that have similarities with engineering systems based on minimum specific knowledge of the system. The generator can select theoretical patterns of the main structure, hubs, and parameters of the hubs, such as hub size, density, and level of connectivity among hubs while classifying the components as sources or sinks combined with a threshold criterion.

Theoretical patterns are developed in the generator as unweighted networks whereas directionality is included to create source and sink nodes at the hub level. Source nodes feed hubs: in contrast, hubs feed the sink nodes. For distributed engineering systems, the energy, materials and information flow from source to sink components, it is therefore important to identify this directionality of the flows in the architecture to link the structure with behaviour and function and allow a combined analysis of structural and behavioural properties.

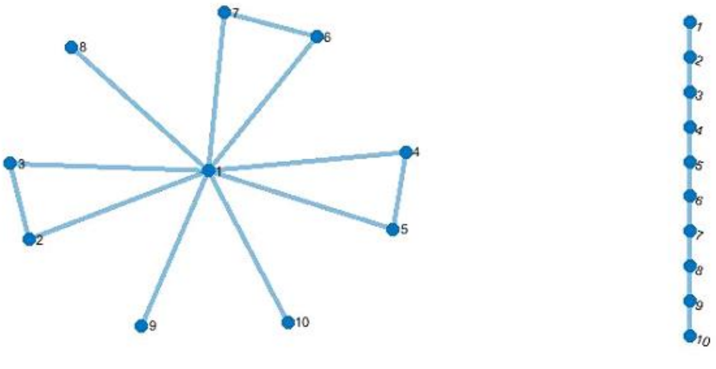
Nine parameters can be chosen by the user to fine-tune the generator to populate tailored simulated networks as shown in Table 10.

The first two parameters (1 and 2 in Table 10) control the number of nodes and the type of main structure of the simulated network. The main structure patterns used in parameter 2 of the generator are presented in Table 9. Min et al. (2016) stated that commonly "system architectures are categorized into three different configurations: integral, linear-modular, and bus-modular architecture ... many engineering systems are configured as or resemble one of these architectural configurations to some degree". The bus-modular, path, hierarchical and integral patterns were selected as options based on the engineering design literature (Hölttä-Otto et al., 2012; Paparistodimou et al., 2017; Sharman and Yassine, 2004). Additionally, the cyclical pattern was included as an option based on the observations from the preliminary empirical investigation (Appendix III) that were consistent with references in the literature (Agarwal et al., 2014; Chalfant and Chryssostomidis, 2017; Whitney, 2003).

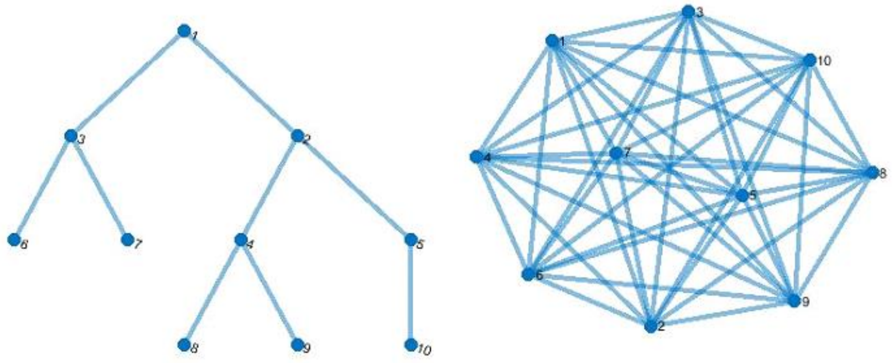
The study suggests that these five main structure pattern options can form the basis for the creation of various and different style simulated networks, establishing a broad range of applicability for the proposed generator.

Table 9: Main structure patterns included in the network tool

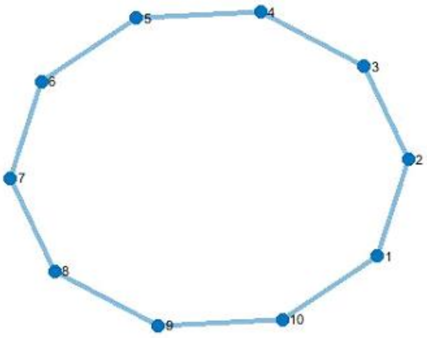
Reference	**Parameter 2** **Main structure pattern**
(Hölttä-Otto et al., 2012; Sharman and Yassine, 2004)	Bus-Modular (BM)
(Hölttä-Otto et al., 2012; Min et al., 2016)	Path (P)
(Baldwin and Clark, 2000; Rivkin and Siggelkow, 2007; Sharman and Yassine, 2004; Yassine and Naoum-Sawaya, 2016)	Hierarchical (HI)
(Hölttä-Otto et al., 2012; Sharman and Yassine, 2004)	Integral (IN)
(Agarwal et al., 2014; Chalfant and Chryssostomidis, 2017; Whitney, 2003)	Cyclical (CY)



Bus modular pattern **Path pattern**



Hierarchical pattern **Integral pattern**



Cyclical pattern

Figure 11: Main structure patterns included in the network tool

The five network main structure patterns shown in Figure 11 were generated based DSMs that is analogous to adjacency matrices. The simulated networks can be further tailored using the additional parameters of the generator. Parameters 3, 4, and 5 in Table 10 determine the number, density, and type of pattern of the hubs on the main structure.

The pattern of the hub can be defined to mimic one of the main structures or can be randomised between pattern options. A level of randomisation is included in the generator for the position of the hubs in the main structure. The next two parameters (6 and 7 in Table 10) define the probability that a hub's node links to sources or sinks.

Nodes with edges pointing to hubs become sources (for example in Figure 12 Node 10); nodes pointed to by hubs are sinks (for example in Figure 12 Node 8). The remaining nodes of hubs are assumed to be intermediate components.

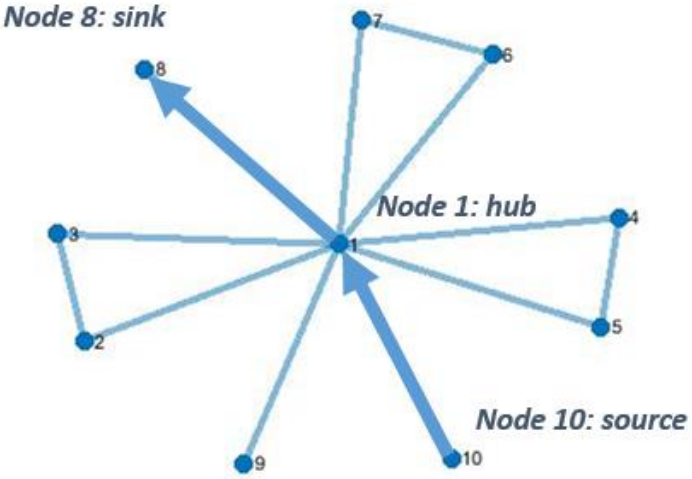


Figure 12: Example of source-hub-sink nodes on the bus-modular pattern

Parameter 8 in Table 10 is a redundancy threshold criterion variable that defines the level of redundancy of the source (e.g. 0.5 indicates double redundancy in the number of sources). The final parameter 9 in Table 10 gives the level of connectivity among hubs (termed pal). The parameters of the generator are summarised in Table 10. The simulated networks populated by the network tool are not expected to follow a power-law degree of distributions. Evidence of this is in the findings of the technical systems previously presented (Appendix III Type A and B); previous studies on climate control and the aircraft engine systems (Sosa et al., 2011) and bicycle drivetrain, automobile drivetrain and aircraft (Walsh et al., 2019), indicating that technical systems do not always exhibit power-law degree distribution. Also, the degree distributions of the design process network studied by (Piccolo et al., 2018) were found not to be power laws, however, they noted that were broad and not similar to the Poisson distribution, allowing the assumption that the behaviour of the network would be similar to the scale-free network.

Table 10: Parameters of network tool

Parameters	
1	Number of nodes in the network (n)
2	Type of pattern of the main structure *Bus modular, Path, Hierarchical, Integral, Cyclical*
3	Number of hubs (k)
4	Density of hubs (mx)
5	Type of pattern of hubs (Ty) *Bus modular, Path, Hierarchical, Integral, Cyclical*
6	Probability of hub's nodes are sources (P_{so})
7	Probability of hub's nodes are sinks (P_{sk})
8	Redundancy threshold criterion (RTC)
9	Level of connectivity amongst hubs (Pal)

The proposed generator can be used to generate system architecture options that have similar features and behaviours to those of the technical systems of interest, in order to carry out an exploratory investigation of the suitability of various potential system architecture designs to meet the desired robustness and modularity.

The application of the generator can be through a variable at a time approach that suggests one at a time instead of modifying several parameters simultaneously or through a DOE approach that also examines interactions between variations of parameters. In addition, Monte Carlo techniques could be used to generate a large number of instances of system architecture options through the generator. Monte Carlo methods can help "gather information about a random object by observing many realizations of it. An example is simulation modelling, where a random process mimics the behaviour of some real-life system"(Kroese et al., 2014).

The value of the parameter (e.g. the size of the system architecture) can be selected randomly for each experiment within the predefined range. By adopting this approach, each experiment may involve a large number of instantiation options for system architecture (e.g. 3000). It could also be used more generally to allow a broader exploration of unprecedented styles of system architectures.

7.3 Analysis stage

The stage of analysis of the explorative RoMoGA involves the performance of various experiments using the network tool. Various experimental set-ups are recommended by selecting different ranges of generator parameters based on the nominal system architecture that the user considers to be the reference point for the population of different system architecture options. The design of different experiments allows the impact of different parameters on robustness and modularity to be assessed. Iteration through various experimental set-ups leads to the creation of various architectural system options. The parameter being examined varies for each experimental establishment, whereas the remaining parameters vary or remain fixed randomly. For the different system architecture options generated for each experimental setup, robustness and modularity are calculated.

7.3.1 Modularity assessment

A state of art modularity optimisation algorithm Louvain community detection method was selected to be incorporated in the analysis stage. In the explorative implementation of RoMoGA, there is no requirement for a resolution parameter as was defined in Section 6.3.1. The analysis stage aims to preliminary assess the modularity of the simulated network to provide an overall evaluation indicator for the potential of a system architecture option to be readily modularised. The Louvain community detection method was therefore found to be appropriate, as was previously established in engineering design literature. (Parraguez et al., 2019; Sinha et al., 2019)

Louvain community detection method is used to identify a modular configuration, in combination with a normalised Newman modularity index proposed by Magee et al. (2010). The Louvain community detection method (Blondel et al., 2008) is used to find the modules and there is no requirement to define the number of modules. A subgraph is considered as a community if the density of interactions is higher than what could be anticipated in a random graph. The communities (modules) are discovered using spectral modularity maximisation established by the Louvain method. This method searches for communities/modules c1, c2...cp that maximise a quality function Q, known as modularity, defined in equation (12) as:

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(ci, cj) \qquad (12)$$

where m is the number of edges, δ (ci, cj) is the Kronecker delta, Aij is entries of the adjacency matrix, and k is the degree of the node. Positive contributions to Q occur when an edge in the network connects nodes assigned to the same community. The quantity $\frac{k_i k_j}{2m}$ is the probability of an edge occurring in a random network with the same degree of distribution; thus, Q is sizeable only if there is a preponderance of like-to-like connections. Q is normalised to give values lying between −1 and 1. This quality function can be used for both unweighted and weighted networks. Parraguez et al. (2019) also used the Louvain method to study process modularity over time, concluding that the method identifies "real modules with a high degree of accuracy". The Louvain community method output is a community (module), which is then used as the input to calculate the modularity metric.

7.3.2 Robustness assessment

The novel robustness evaluation metric presented in Section 6.3.4 is employed for robustness assessment. Two strategies are employed as shown in Table 11. The target and random strategies were selected because of the network science literature, as is the classic approach to investigate network robustness.

Table 11: Strategies of disruptions

Type / Level	Simulated networks	
Component level	Target disruptions	Random disruptions

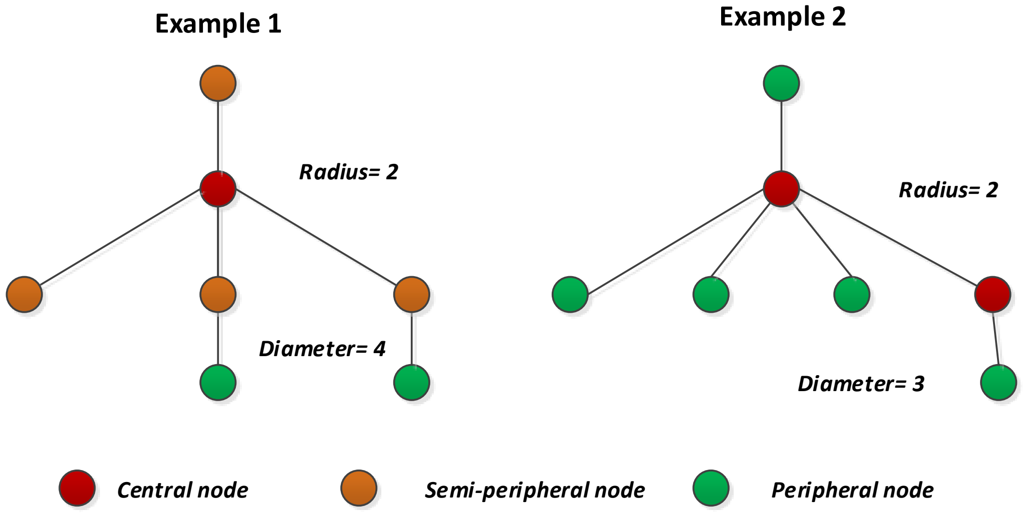


Figure 13: Eccentricity network science concept

Robustness calculation under target disruptions is based on eccentricity, a concept in network theory as shown in Figure 13. The eccentricity of a node is the greatest geodesic distance between node v and any other (the geodesic distance is also known as the shortest path). The diameter d of a network is defined to be the maximal value of node eccentricity, and the radius r is the minimum. In targeted attack scenarios, components with the lowest eccentricity are sequentially attacked because the lowest eccentricity components are the central components of the network and represent the core part of the system's functionality. Compared with random attacks on networks, targeted attacks tend to be more severe.

7.4 Evaluation stage

In the evaluation stage of simulated theoretical system architecture networks, reflections based on the analytical results on the key architectural characteristics of the system architecture, such as the main structure pattern and several hubs is suggested. The network tool included in the first stage of the methodology is expected to be used to investigate novel system architecture options; experiment with main patterns or parameters of hub components. In this case, the evaluation stage is focused on assessing the effects of the patterns and the parameters of hubs on robustness and modularity.

Reflections on the analysis results for various system architecture options can provide insights to help the architect to design improved architectures, by selecting the type of the main architectural pattern of a system (for example cyclical or bus modular) or by deciding on the number, size and connectivity amongst hubs. The architect can then translate the system architecture options into feasible technical systems architectures.

A feasible system architecture satisfies the desired function. The evaluation results could yield a large number of possible system architecture designs, but few could be practical. The expert is suggested to evaluate the functionality of a generated system architecture design, and only those functionally rational designs are suggested to be selected for further analysis. The findings of the evaluation are suggested to steer the architect to perform subsequent redesign and designing new architectures informing the prescriptive RoMoGA.

7.5 Chapter summary

This Chapter presented the explorative RoMoGA exploration, which suggests the use of the novel generator at the generation stage. The generator can alternate patterns of the main structure, hubs, and hub parameters while experimenting with the source and sink components.

Modularity and robustness network metrics and methods were used at the analysis stage to assess the different theoretical patterns. The next Chapter describes the evaluation part of the study that includes the industrial application in Chapter 8 whereas the methodology is applied in technical naval engineering system architecture and the explorative application, Chapter 9, whereas the network tool is applied, to simulate networks that share similarities with the actual naval engineering systems examined in Chapter 8.

PART 3: EVALUATION

Chapter 8: Industrial application - case studies

Chapter 8 presents applications of the proposed descriptive implementation of RoMoGA that answers the research objective 6 that is to apply the methodology in established technical systems.

The three case studies represent three distinctive generic naval distributed system architecture designs of three individual ships and are herein named Type A, Type B and Type C. The three system architectures are designed to satisfy the same main function which is to enable the ship to move and operate. The main function is decomposed to four sub-functions: electrical power, propulsion, chilled water and seawater systems power, propulsion, chilled cooling and seawater systems.

Ministry of Defense (2015) describes that "a distributed system is defined as a system with the redundancy of critical elements which are physically separated in the ship". A distributed system is defined by Brefort et al. (2018) as "a specific type of system that is disbursed through the vessel". The highly interdependent collections of the distributed subsystems that make up a modern naval combatant qualify them as a complex system.

The reason to study three similar systems of the same functionality in the case studies was to allow a comparative investigation to be performed focusing on the level of redundancy, and modularity, avoiding comparing dissimilar systems amongst them that would not provide fruitful comparisons.

The generation stage of the methodology that involves the definition of the DSM that represents the system and the definition of the source and sink components was formulated with the support of experts from BAE Systems. The data collection process for constructing the case study included access to different domain drawings, system descriptions and operational configurations to develop the system architectures of interest. The case studies were developed in close cooperation with BAE System' experts and the data collection and analysis of the case studies that happened in two years that the research was established within the company. An extended amount of time working with experts of the systems was needed to understand the interconnections and operational requirements to develop the basis for the main function and four sub-functions. Data were collected based on the analysis of document and expert' elaboration. A Senior Propulsion and Power engineer with 40 years' experience, the Research and Technology Manager 35 years' experience and Senior Principal Engineer 30 years' experience were directly involved in these case studies.

The robustness behaviour of each of the baseline system architectures, for each ship type, was established based on the established technical documentation and expert input, before the application of the proposed methodology that is presented in this Chapter. A detailed assessment of the system performance following component failures was prepared by the expert and provided to the researcher that function as a verification document. This document was used to evaluate the outcomes of the application.

DOE approach was adopted to apply the methodology. Redundancy and modularity were defined to have three levels of low-medium-high. DOE two independent variable for three levels is a total nine experimental run. Therefore, the researcher iterated (by changing the independent variables) three times over the redundancy loop and three types over the modularity loop and calculated the robustness for all the nine possible level combinations of redundancy and modularity.

The outputs of the methodology were illustrated through the formulation of the Tables and Figures. The computational results generated from MATLAB[7] were collected in a macro-excel sheet were the numbering of the nodes was translated into the names of the components. The translated results inform the Tables presented in this Chapter. Microsoft Visio[8] software was employed to design the input schematics representing the system architectures and the functional hierarchical and architectural options diagrams. The same software was used to represent the colour schematics outcome representing a robust modular system architecture. Minitab[9] was also used to generate the mean effects plots that are included in the evaluation stage of the methodology. The software's mentioned were used due to pre-existing knowledge and availability. Alternative tools may be employed to represents inputs and computationally model the analysis and depict the outcomes of the methodology.

The outcomes were aimed to be presented in a visual fashion that could be readily understandable by the user of the methodology. The outcomes of the proposed methodology on the three case studies are descriptively presented in this Chapter. Reflection and interpretations on the results of the case studies are included in Discussions and Reflections Chapter 12.

---

[7] **MATLAB** was used due its availability and prior knowledge. Any alternative programming language tool may be employed.

[8] **Microsoft Visio** was used due its availability. Any alternative illustration software tool may be employed.

[9] **Minitab** was used due to its availability. Any alternative statistical software tool may be employed.

8.1 Main function of a naval ship distributed system

The naval ship's main function is typically described as float, move, and operate. The float function is achieved by the design of the ship and internal subdivision and is not considered in this study. The distributed systems support the other two main functions which are move and operate.

- Move – The move function was to move and orientate the ship. This was met if one shaft and one rudder were operational.
- Operate – The operate function was met if either the sonar or foremast radar was operational.

The main function for all the three case studies is defined as to move and operate. The architecture of the system can be reconfigured in different ways. In the three case studies, cruising configuration is defined as the state of the architecture to be investigated. The cruising configuration requires a lower number of equipment and connectivity for achieving the functional requirements. The system architecture has a high-level of redundancy, as the functional requirements could be achieved with different alternative options of connectivity among the source and sink components or a different set of architectural options.

The functional assessment was carried out for a cruising configuration which has a low functional performance requirement, with high redundancy. The system architecture configuration state maximised fluid interconnections, with the minimum number of operational sources. This is a low threat environment and operation with a partial electrical failure considered acceptable to satisfy functional continuity. This definition of functional satisfaction informs the creation of the functional hierarchical and architectural options diagram for the three case studies, which are inputs for the robustness calculation.

8.1.1 Key sub-functions

The main function is decomposed in four sub-functions (power, propulsion, chilled water, and seawater cooling). These key sub-functions were defined due to their direct influence on achieving the main functions of the move and operate. The four sub-functions of the system architecture are examined: generating and distributing power to Electrical Distribution Centres (EDCs); provide energy (power or mechanical) for propulsion and steering; provide chilled water-cooling for combat systems and power and propulsion; and, provide seawater cooling for power and propulsion and other consumers. The system architecture provides additional functionalities to the naval ship that are not included within this analysis. Three case studies (referred to as Type A, B and C) are elaborated that examine existing naval distributed system architectures that have similar functional requirements from different ships.

8.1.2 Multi flows and source/sink based DSM

The naval distributed system architectures represented in DSMs for the Type A, B and C case studies compromise systems with different types flow as following discussed:

- Electrical systems – These provided electrical power, that is generated by diesel generators or gas turbine (source components) and distributed by associated switchboards (intermediate components). EDCs are distributed along with the ship and are connected to either the forward or aft LV switchboard. Forward HV Switchboard supplies all the odd-numbered EDCs and the aft all the even-numbered EDCs.
- Fluid systems – These provided water cooling to equipment throughout the ship and were represented by source components (pumps and chilled water plants), sink components (cooling loads), manifolds (intermediate components) and pipework (connections). Each source component of the fluid system (pump or chilled water plant) is associated with an EDC sink component of the electrical system.

8.2 Case study Type-A: Legacy frigate

8.2.1 Generation stage

Type A is a legacy anti-submarine warship with sonar systems and quiet electric propulsion. Power is generated by four diesel generators, which supply HV switchboards which in turn supply two propulsion motors, and the LV switchboards, via Transformers. Each shaft can also be driven by the propulsion motors, mounted on the shaft or by a gas turbine via a dedicated gearbox. Gas turbines lubrication coolers depended on seawater for cooling. The chilled water system only supplies combat and other systems, this does not influence the power and propulsion. The low-pressure seawater system (LPSW) provides cooling for the motor generators, motor converters, propulsion motors and the gas turbines lubrication coolers. Table 12 below summarises Type-A ship spatial layout zones that the components included in the system architecture are located (Zone 1 forward to Zone 4 aft).

Table 12: Overview of spatial separation of Type A design

Aft **Zone 4**	**Zone 3**	**Zone 2**	**Fwd.** **Zone 1**
		Mast (CWManifold2) (EDC2)	Sonar (CWManifold1) (EDC1)
Steering1 & 2	Emergency diesel generator/Emergency switchboard		
	DG1 DG3 HV SWBD1 MG1 LV SWBD1	DG2 DG4 HV SWBD2 MG2 LV SWBD2	
EDC6, EDC5, EDC4		EDC3, EDC2, EDC1	
	Converter1 Motor1 GT1 GT LO Cooler1 Gearbox1	Converter2 Motor2 GT2 GT LO Cooler2 Gearbox2	
Port		**Starboard**	
	CW3(EDC5) CWManifold3	CW2(EDC2) CWManifold2	CW1(EDC1) CWManifold1
LPSW2(EDC4) LPSW Manifold2		LPSW1(EDC3) LPSW Manifold1	

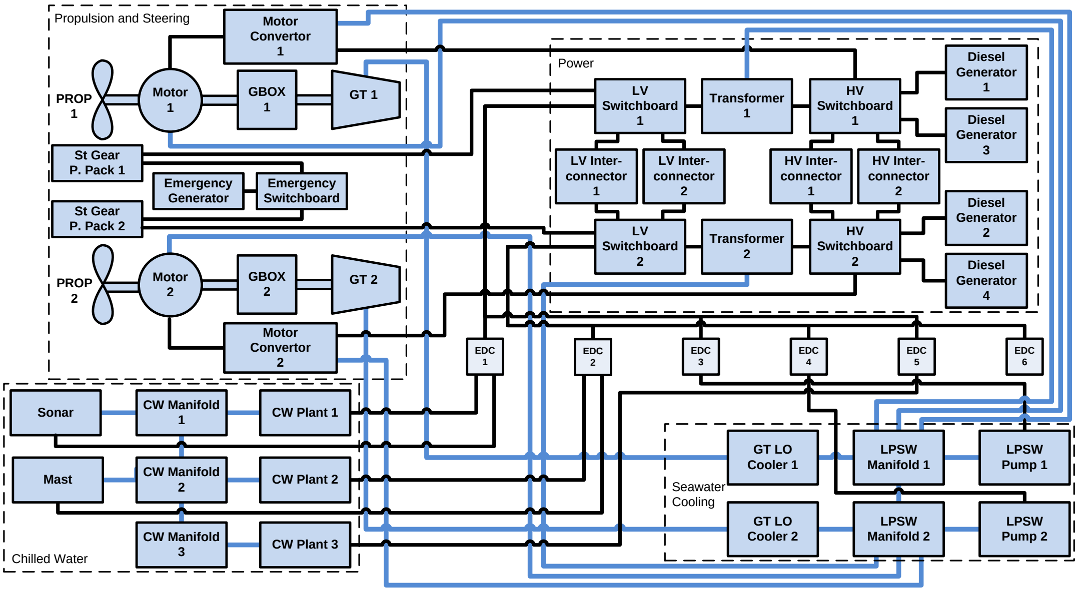
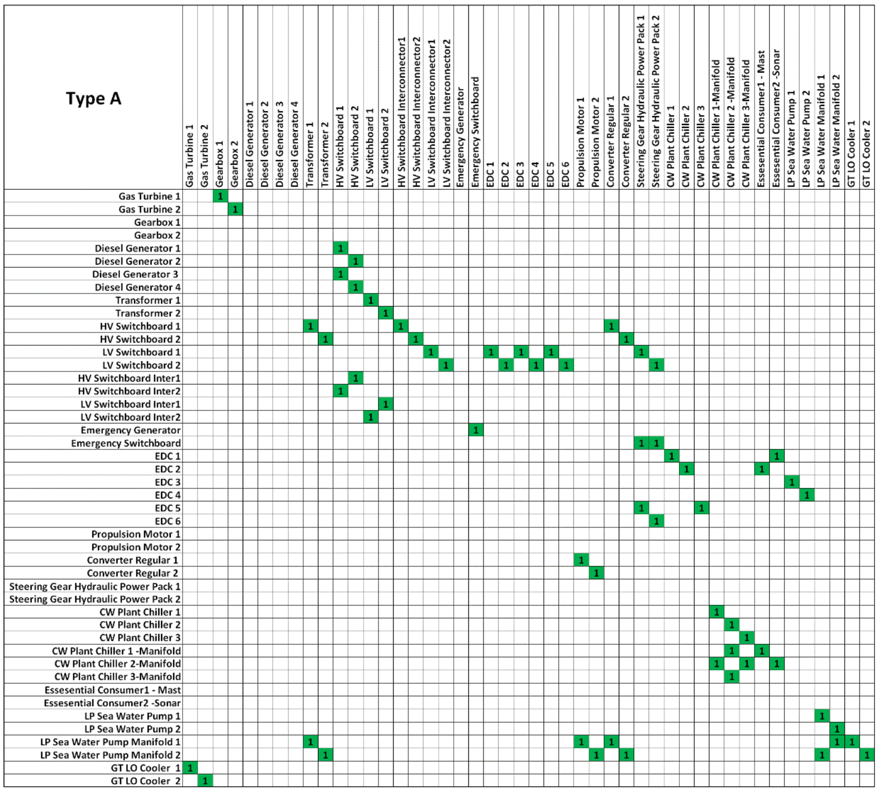


Figure 14: Type A – high redundancy (nominal) system architecture schematic

Figure 14 illustrates the schematic representation of the system architecture which is divided into four parts representing the four primary systems. The upper right part of the schematic illustrates the power system, the upper left illustrates the propulsion system, the lower left part illustrates the chilled water system, and the lower right illustrates the seawater system.

The power sources (diesel generators 1 to 4) are positioned on the upper right side of the schematic, whereas sinks such as propulsion motors 1 and two are positioned on the left of the schematic. Figure 14 captures the interconnectivity amongst the different subsystems that are crucial for the system to satisfy the main function but are also critical in case of disruptions. It also encapsulates integration across functions that are not typically represented in practice, with multiple drawings used to reflect the individual subsystems, for example, the chilled water diagrams or electrical power distribution diagrams. In a typical ship engineering approach, these four sub-functions are independently addressed by engineers of the specific specialisation and the drawings and documentation that describes is in isolation.

The generation stage entails the functional network definition in DSM.



Type A	Gas Turbine 1	Gas Turbine 2	Gearbox 1	Gearbox 2	Diesel Generator 1	Diesel Generator 2	Diesel Generator 3	Diesel Generator 4	Transformer 1	Transformer 2	HV Switchboard 1	HV Switchboard 2	LV Switchboard 1	LV Switchboard 2	HV Switchboard Interconnector1	HV Switchboard Interconnector2	LV Switchboard Interconnector1	LV Switchboard Interconnector2	Emergency Generator	Emergency Switchboard	EDC 1	EDC 2	EDC 3	EDC 4	EDC 5	EDC 6	Propulsion Motor 1	Propulsion Motor 2	Converter Regular 1	Converter Regular 2	Steering Gear Hydraulic Power Pack 1	Steering Gear Hydraulic Power Pack 2	CW Plant Chiller 1	CW Plant Chiller 2	CW Plant Chiller 3	CW Plant Chiller 1-Manifold	CW Plant Chiller 2 -Manifold	CW Plant Chiller 3-Manifold	Essesential Consumer1 - Mast	Essesential Consumer2 -Sonar	LP Sea Water Pump 1	LP Sea Water Pump 2	LP Sea Water Manifold 1	LP Sea Water Manifold 2	GT LO Cooler 1	GT LO Cooler 2
Gas Turbine 1			1																																											
Gas Turbine 2				1																																										
Gearbox 1																																														
Gearbox 2																																														
Diesel Generator 1											1																																			
Diesel Generator 2												1																																		
Diesel Generator 3											1																																			
Diesel Generator 4												1																																		
Transformer 1													1																																	
Transformer 2														1																																
HV Switchboard 1									1						1														1																	
HV Switchboard 2										1						1														1																
LV Switchboard 1																	1				1		1		1						1															
LV Switchboard 2																		1				1		1		1						1														
HV Switchboard Inter1												1																																		
HV Switchboard Inter2											1																																			
LV Switchboard Inter1														1																																
LV Switchboard Inter2													1																																	
Emergency Generator																				1																										
Emergency Switchboard																															1	1														
EDC 1																																	1							1						
EDC 2																																		1					1							
EDC 3																																									1					
EDC 4																																										1				
EDC 5																															1				1											
EDC 6																																1														
Propulsion Motor 1																																														
Propulsion Motor 2																																														
Converter Regular 1																											1																			
Converter Regular 2																												1																		
Steering Gear Hydraulic Power Pack 1																																														
Steering Gear Hydraulic Power Pack 2																																														
CW Plant Chiller 1																																				1										
CW Plant Chiller 2																																					1									
CW Plant Chiller 3																																						1								
CW Plant Chiller 1 -Manifold																																					1		1							
CW Plant Chiller 2-Manifold																																				1		1		1						
CW Plant Chiller 3-Manifold																																					1									
Essesential Consumer1 - Mast																																														
Essesential Consumer2 -Sonar																																														
LP Sea Water Pump 1																																											1			
LP Sea Water Pump 2																																												1		
LP Sea Water Pump Manifold 1									1																		1		1															1	1	
LP Sea Water Pump Manifold 2										1																		1		1													1			1
GT LO Cooler 1	1																																													
GT LO Cooler 2		1																																												

Figure 15: DSM Type A

Moreover, in the generation stage, the Type A high-level of redundancy functional hierarchy, options, and Redundancy Threshold Criterion (RTC) diagram is defined as illustrated in Figure 16. The architectural options represent the sets of sources and sink components that were designed within the architecture to satisfy the same sub-function. The RTC was defined as the level of connectivity required amongst a set of the individual source and sink components in the architecture to satisfy the corresponding function.

The Redundancy Threshold Criterion (RTC) is defined as the level of connectivity required amongst source and sink components in the architecture to satisfy the corresponding function.

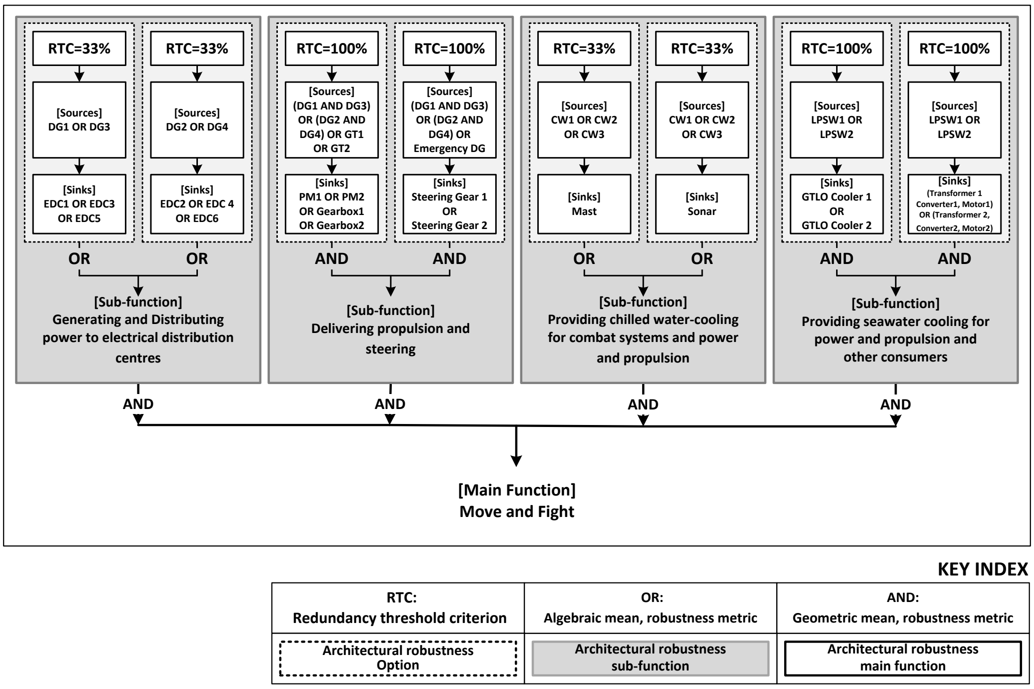


Figure 16: Functional hierarchy, architectural options and RTC for high redundancy system

The functional requirement for the power sub-function, the RTC_i is satisfied if 33% connectivity is available between the power source and sink. In contrast, for the propulsion sub-function, there are 4 options, however, each option requires 100% connectivity to be considered successful. Specifically, full connectivity amongst DG1 and DG 3 is required with PM1, a 100% connectivity of GT1 to Gearbox 1. The RTC relates to the level functional requirement (depends on the operational state) that the robustness of the architecture is assessed.

Figure 16 is based on technical information for Type A system. For example, two architectural options mean that they are two sets of sources and sink components that satisfy the same function. If full connectivity amongst source and sink is required to satisfy function, then RTC is 100%. If half connectivity amongst source and sink is sufficient to satisfy function, then RTC is defined as 50%.

8.2.2 Analysis stage

Redundancy and modularity are treated as the design variables that affect robustness, that is the response variable. Figure 17 is offering a snapshot of the RoMoGA methodology (Figure 6) illustrating that the disruptive loop is nested within the modularity loop, which is nested within the redundancy loop. That means for a single level of redundancy, iteration through the low-medium-high-level of modularity is suggested, and for each level of modularity iteration through the module and component disruption scenarios is proposed.

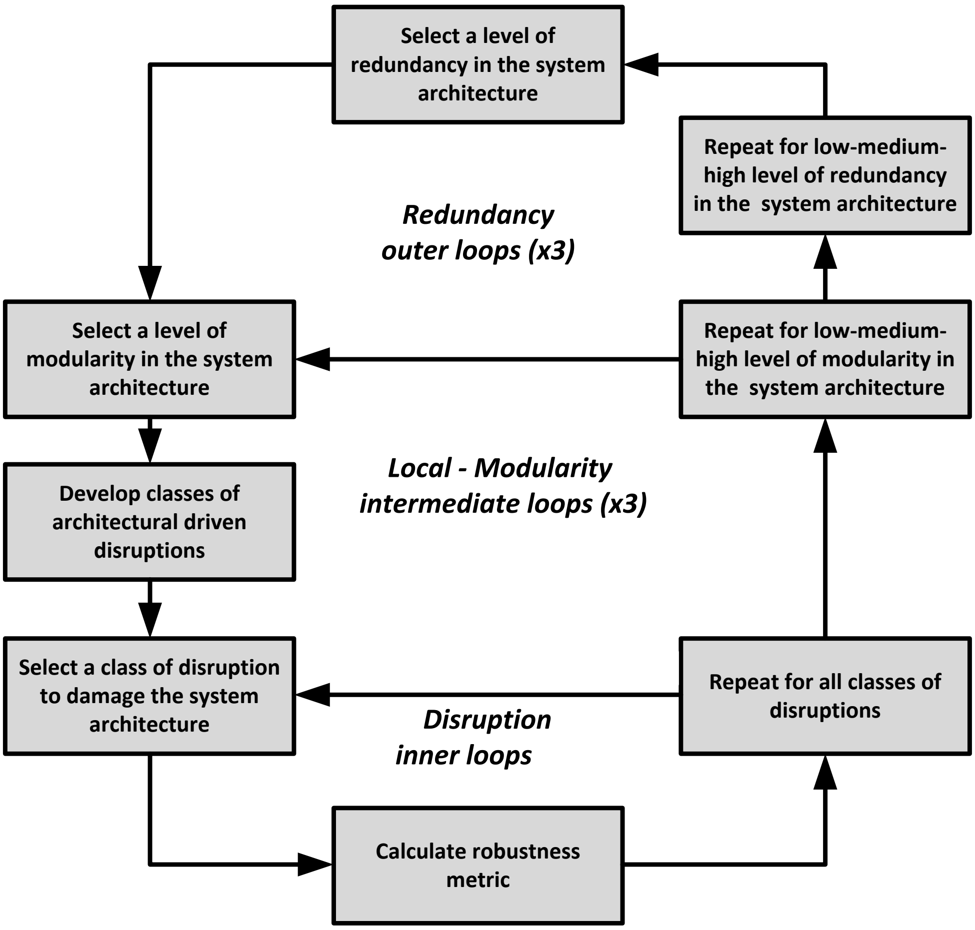


Figure 17: Analysis stage

Table 13 associates the iterations of the redundancy loop with the low-medium-high redundancy and provides a reference on the Figures that are presented in the following Sections. Redundancy is controlled through changing the functional hierarchical and architectural options diagram and the RTC of the architecture. For the analysis stage, additional alternative system architectures with different levels of redundancy were developed following discussions the SME in the redundancy loop. This was achieved by changes to the definitions of the functional hierarchy, architectural options and RTC diagram.

In this study system, architecture configuration for low and medium redundancy was derived based on the baseline high redundancy system architectures that were developed incorporation with the experts.

Table 13: Variations of the levels of redundancy in the system architecture

Design of Experiments (DOE) levels	Control factor	Level of redundancy in the system architecture	Iteration of the redundancy loops
3	High- level	Figure 14 and 16	First redundancy loop
2	Medium - level	Figure 21 and 22	Second redundancy loop
1	Low - level	Figure 25 and 26	Third redundancy loop

Modularity is treated as a design variable and its levels are manipulated using the stability method. The resolution parameter of the stability method was used to control the modularity as a design variable. Through the variation of the resolution parameter, the level of modularity in the system architecture was controlled as shown in Table 14 in each different iteration of the modularity loop.

Table 14: Variations of the level of modularity in the system architecture

DOE levels	Control factor	Level of modularity	Iterations of the modularity loops
1	Low - level	Stability resolution parameter: 0.2	First modularity loop
2	Medium - level	Stability resolution parameter: 0.5	Second modularity loop
3	High- level	Stability resolution parameter: 1	Third modularity loop

The following paragraphs the computational results are outlined.

8.2.2.1 1st Redundancy loop: High redundancy

The paragraphs below present the robustness results for a high-level of redundancy.

Combinatory disruption of components

The scenario of disruptions presented included all possible combinations of two components to be disrupted. Each row indicates a case of disruption. The second and third columns show the two components that were removed from the network, disrupting the network architecture. Robustness results calculated in MATLAB were collected in macro excel sheets, translating the component ID numbers into the names of the components, and the numeric robustness results were assessed using the system functional requirements to give a Yes or No answer. Because component-based disruptions were performed primarily to verify the reality of robustness calculation, Boolean logic was sufficient to capture results.

In addition, the expert confirms that the result generation as shown in Table 15 was useful as there is no automatic tool that can provide robustness calculation for all possible component disruption combinations used in practice. The experts were also more interested in robustness results after three and four-component disruptions, as the experts had more difficulty understanding these disruption scenarios. These were not calculated in this study, as the purpose of combined components was to verified results, but MATLAB's developed computational methodology tool is capable of performing this analysis.

The results are given in columns A to E showing if the sub-function remains operational (Y or N) following the disruption. The sub-functions are:

- A “Is the system generating and distributing power?”
- B “Is the system delivering propulsion and steering?”
- C “Is the system providing chilled water?”
- D “Is the system providing seawater?”
- E “Is the system able to move and operate?”, i.e. all sub-functions operational.

Table 15: Combinatory disruption of Components Type A high redundancy architecture

	If removed component 1	and removed component 2	A	B	C	D	E
1	Transformer 1	Transformer 2	N	Y	Y	Y	N
2	Transformer 1	HV Switchboard 2	N	Y	Y	Y	N
3	Transformer 1	LV Switchboard 2	N	Y	Y	Y	N
4	Transformer 1	Propulsion Motor 2	Y	Y	Y	N	N
5	Transformer 1	Converter Regular 2	Y	Y	Y	N	N
6	Transformer 1	LP seawater pump manifold 2	Y	Y	Y	N	N
7	Transformer 2	HV Switchboard 1	N	Y	Y	Y	N
8	Transformer 2	LV Switchboard 1	N	Y	Y	Y	N
9	Transformer 2	Propulsion Motor 1	Y	Y	Y	N	N
10	Transformer 2	Converter Regular 1	Y	Y	Y	N	N
11	Transformer 2	LP seawater pump manifold 1	Y	Y	Y	N	N
12	HV Switchboard 1	HV Switchboard 2	N	Y	Y	Y	N
13	HV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
14	HV Switchboard 2	LV Switchboard 1	N	Y	Y	Y	N
15	LV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
16	Propulsion Motor 1	Propulsion Motor 2	Y	Y	Y	N	N
17	Propulsion Motor 1	Converter Regular 2	Y	Y	Y	N	N
18	Propulsion Motor 1	LP seawater pump manifold 2	Y	Y	Y	N	N
19	Propulsion Motor 2	Converter Regular 1	Y	Y	Y	N	N
20	Propulsion Motor 2	LP seawater pump manifold 1	Y	Y	Y	N	N
21	Converter Regular 1	Converter Regular 2	Y	Y	Y	N	N
22	Converter Regular 1	LP seawater pump manifold 2	Y	Y	Y	N	N
23	Converter Regular 2	LP seawater pump manifold 1	Y	Y	Y	N	N
24	Steering Gear Hydraulic Power Pack 1	Steering Gear Hydraulic Power Pack 2	Y	N	Y	Y	N
25	CW Plant Chillier 1	CW Plant Chillier 2-Manifold	Y	Y	N	Y	N
26	CW Plant Chillier 1 -Manifold	CW Plant Chillier 2-Manifold	Y	Y	N	Y	N
27	CW Plant Chillier 1 -Manifold	Essential Consumer2 -Sonar	Y	Y	N	Y	N
28	CW Plant Chillier 2-Manifold	Essential Consumer1 - Mast	Y	Y	N	Y	N
29	Essential Consumer1 - Mast	Essential Consumer2 -Sonar	Y	Y	N	Y	N
30	LP seawater pump 1	LP seawater pump 2	Y	Y	Y	N	N
31	LP seawater pump 1	LP seawater pump manifold 2	Y	Y	Y	N	N
32	LP seawater pump 2	LP seawater pump manifold 1	Y	Y	Y	N	N
33	LP seawater pump manifold 1	LP seawater pump manifold 2	Y	Y	Y	N	N
34	LP seawater pump manifold 1	GT LO Cooler 2	Y	Y	Y	N	N
35	LP seawater pump manifold 2	GT LO Cooler 1	Y	Y	Y	N	N
36	GT LO Cooler 1	GT LO Cooler 2	Y	Y	Y	N	N
	Total		**9**	**1**	**5**	**21**	**36**

The outcome was verified by SMEs using their domain knowledge based on the system representations developed, technical documentation. The examination of the effects of disrupting all combinations of two components was used to inform and evaluate the methodology. The results were found to be viable by experts. Some results may require an additional explanation, for example, a loss of the mast and sonar (Combination 29), both sinks for the chilled water results in the failure of the chilled water sub-system and hence the loss of the operating function. Thus, the overall “move and operate” function fails.

In the 4th sub-function, LP seawater cooling of the propulsion equipment is grouped for shaft 1 and 2. The disruption to any motor, converter or transformer will result in the loss of the sink for that shaft. Thus, the disruption to any two of these components on either shaft will result in the loss of all propulsion function sink components. The cooling to the GT coolers is unaffected. However, the 4th sub-function, seawater cooling is defined as “the system providing seawater cooling for power and propulsion” and therefore the whole cooling function fails.

Modularity loop: Low – medium-high modularity

Table 16 presents the modular configuration for the different levels of modularity for the nominal system architecture (high-level of redundancy). The resolution parameter was changed for modularity iteration to enable the identification of different modular configurations. The first column indicates the identification number of the generated modules. The second column depicts the low-level modular configuration, whereas the third and four columns present the medium and high-level modular configurations.

Table 16: Modular configurations for high redundancy architecture Type A

First redundancy loop *high-level of redundancy in the system architecture*			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
1	Gas Turbine 2, Gearbox 2, GT LO Cooler 2	Gas Turbine 1, Gearbox 1, GT LO Cooler 1	Gas Turbine 2, Gearbox 2, GT LO Cooler 2
2	Gas Turbine 1, Gearbox 1, GT LO Cooler 1	Gas Turbine 2, Gearbox 2, GT LO Cooler 2	Diesel Generator 2, Diesel Generator 4, Transformer 2, HV Switchboard 2, HV Switchboard Inter1, HV Switchboard Inter2, Propulsion Motor 2, Converter Regular 2, LP Seawater Pump Manifold 2
3	Diesel Generator 1, Diesel Generator 3, HV Switchboard 1, HV Switchboard Inter1, HV Switchboard Inter2	Diesel Generator 1, Diesel Generator 3, Transformer 1, HV Switchboard 1	LV Switchboard 1, LV Switchboard Inter2, EDC 3, EDC 5, Steering Gear Hydraulic Power Pack 1,

First redundancy loop ***high-level of redundancy in the system architecture***			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
			CW Plant Chiller 3, LP Seawater Pump 1
4	Diesel Generator 2, Diesel Generator 4, Transformer 2, HV Switchboard 2	Diesel Generator 2, Diesel Generator 4, HV Switchboard 2, HV Switchboard Inter1, HV Switchboard Inter2	Gas Turbine 1, Gearbox 1, Diesel Generator 1, Diesel Generator 3, Transformer 1, HV Switchboard 1, Propulsion Motor 1, Converter Regular 1, LP Seawater Pump Manifold 1, GT LO Cooler 1
5	Emergency Generator, Emergency Switchboard	Emergency Generator, Emergency Switchboard	EDC 1, EDC 2, CW Plant Chiller 1, CW Plant Chiller 2, CW Plant Chiller 1 - Manifold, CW Plant Chiller 2-Manifold, CW Plant Chiller 3-Manifold, Essential Consumer1 - Mast, Essential Consumer2 -Sonar
6	EDC 4, LP Seawater Pump 2	EDC 1, EDC 2, CW Plant Chiller 1, CW Plant Chiller 2, CW Plant Chiller 1 - Manifold, CW Plant Chiller 2-Manifold, Essential Consumer1 - Mast, Essential Consumer2 -Sonar	LV Switchboard 2, LV Switchboard Inter1, Emergency Generator, Emergency Switchboard, EDC 4, EDC 6, Steering Gear Hydraulic Power Pack 2, LP Seawater Pump 2
7	LV Switchboard 1, LV Switchboard Inter2, EDC 5, Steering Gear Hydraulic Power Pack 1	LV Switchboard 1, LV Switchboard Inter2, EDC 5, Steering Gear Hydraulic Power Pack 1	
8	LV Switchboard 2, LV Switchboard Inter1, EDC 6, Steering Gear Hydraulic Power Pack 2	EDC 3, Propulsion Motor 1, Converter Regular 1, LP Seawater Pump 1, LP Seawater Pump Manifold 1	
9	Propulsion Motor 1, Converter Regular 1, LP Seawater Pump Manifold 1	Transformer 2, Propulsion Motor 2, Converter Regular 2, LP Seawater Pump Manifold 2	
10	Propulsion Motor 2, Converter Regular 2, LP Seawater Pump Manifold 2	LV Switchboard 2, LV Switchboard Inter1, EDC 6, Steering Gear Hydraulic Power Pack 2	
11	EDC 1, CW Plant Chiller 1, Essential Consumer2 -Sonar	CW Plant Chiller 3, CW Plant Chiller 3-Manifold	
12	CW Plant Chiller 3, CW Plant Chiller 3-Manifold	EDC 4, LP Seawater Pump 2	
13	EDC 2, CW Plant Chiller 2, CW Plant Chiller 1 - Manifold, CW Plant Chiller 2-Manifold, Essential Consumer1 - Mast		
14	EDC 3, LP Seawater Pump 1		
15	Transformer 1		
Robustness metric	0.757	0.73	0.467

First redundancy loop *high-level of redundancy in the system architecture*			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
Modularity metric	0.72	0.75	0.8
Classification of modules (Modules ID)			
central modules	[3]	[6]	[4]
peripheral modules	[5,6,9,10,12,14]	[5,11,12]	[1]
semi peripheral modules	[1,2,4,7,8,11,13,15]	[1,2,3,4,7,8,9,10]	[2,3,5,6]

Figures 18, 19 and 20 below illustrate the examples of the proposed modules generated by the RoMoGA analysis as presented in Table 16.

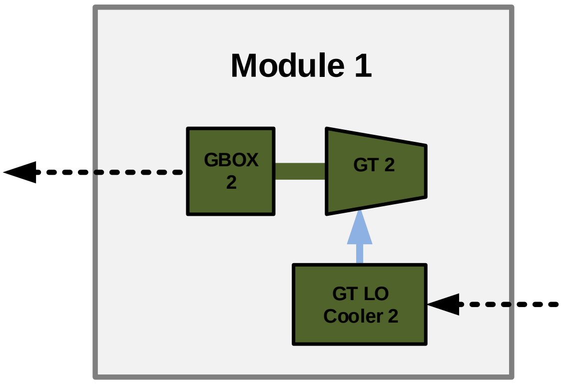


Figure 18: Module 1 Type A high redundancy and high modularity

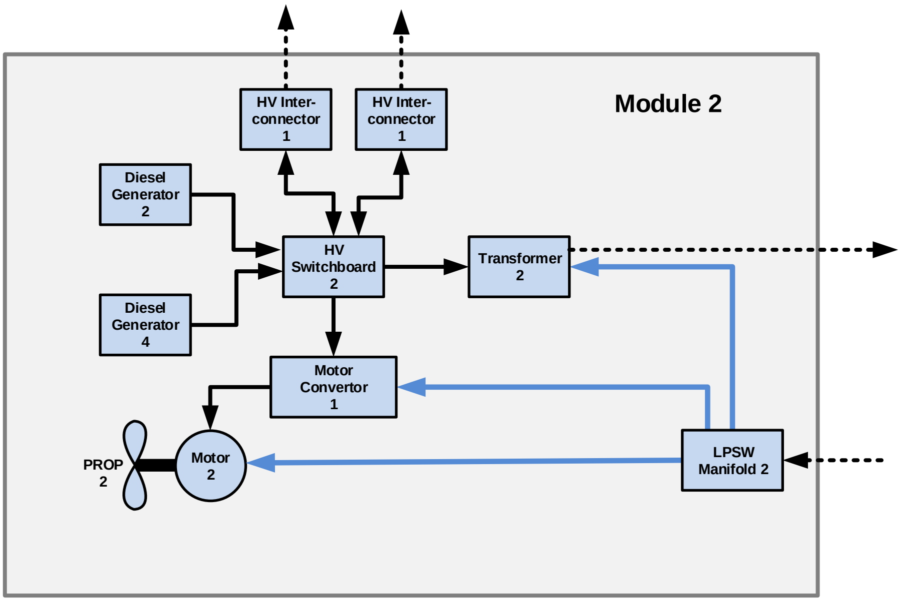


Figure 19: Module 2 Type A high redundancy and high modularity

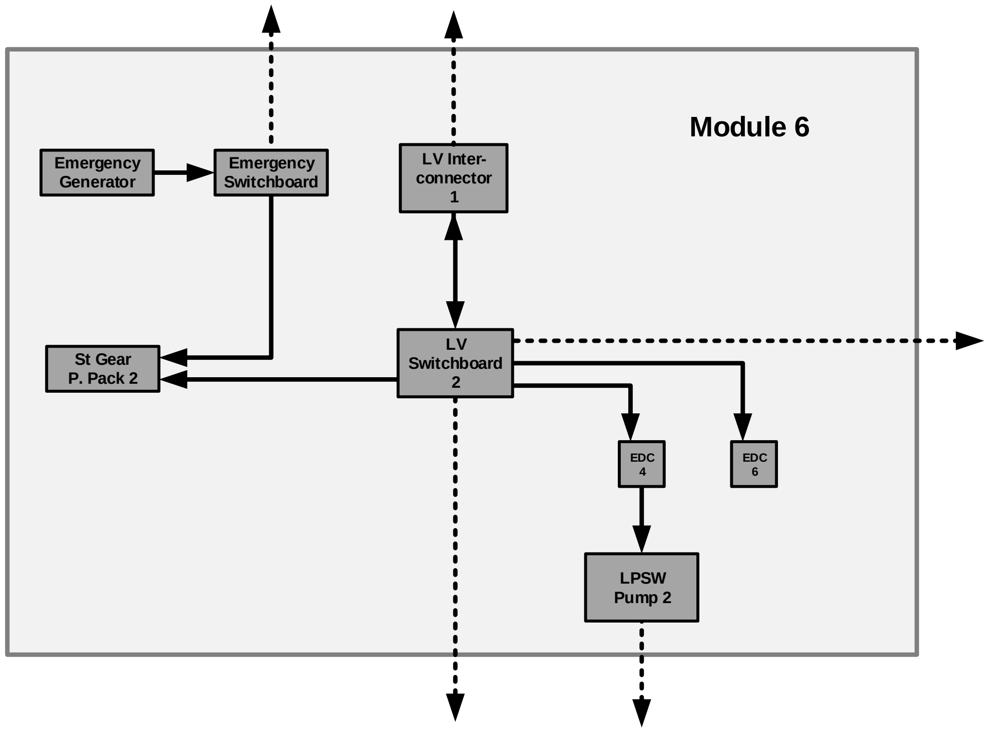


Figure 20: Module 6 Type A high redundancy and high modularity

The high-level of modularity generates a smaller number and largest module sizes, whereas the low-level of modularity generates a higher number of modules of smaller size. It can be observed that similar modules were found iterating through the modularity loop. Modules of nine to ten components were found to be robust. The third and fourth columns indicate that both high and medium modular configuration has only one non-robust module. In this instance, the highest-level modular configuration was considered the most appropriate and consequently used for reformulation in the evaluation stage.

8.2.2.2 2nd Redundancy loop: Medium redundancy

These paragraphs present the robustness results for a medium level of redundancy. The medium redundancy system architecture was devised through discussions with experts to demonstrate the implementation of the robust modular assessment methodology and does not reflect a real system architecture. The expert's suggested the removal of redundant electrical power generation source components and a chilled water pump (source for chilled water-cooling function).

The medium-level system architecture presented can satisfy the main function defined as to move and operate

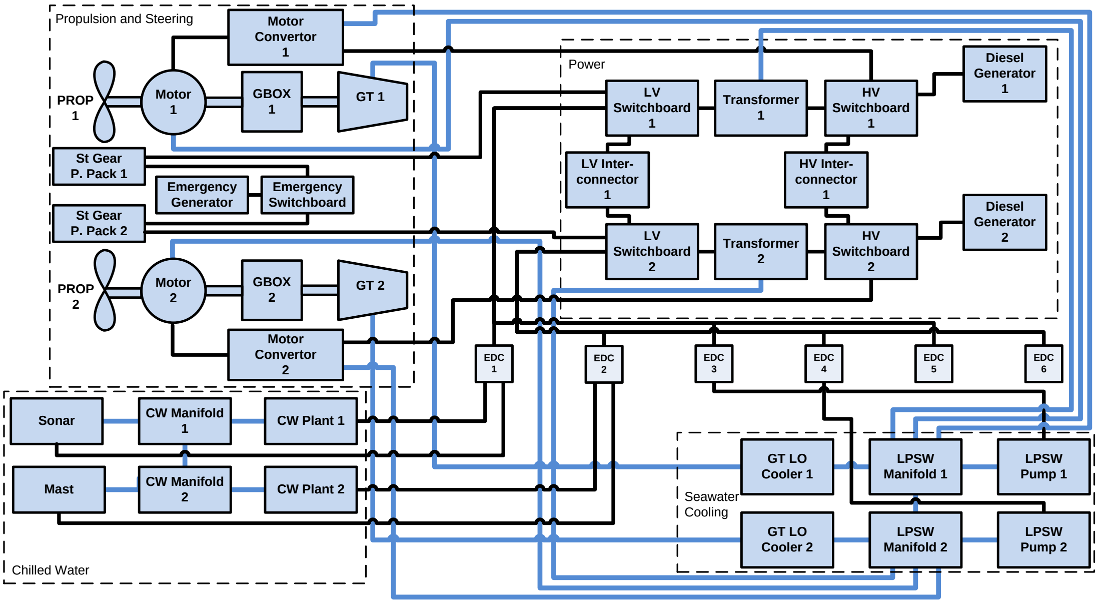


Figure 21: Type A medium redundancy schematic

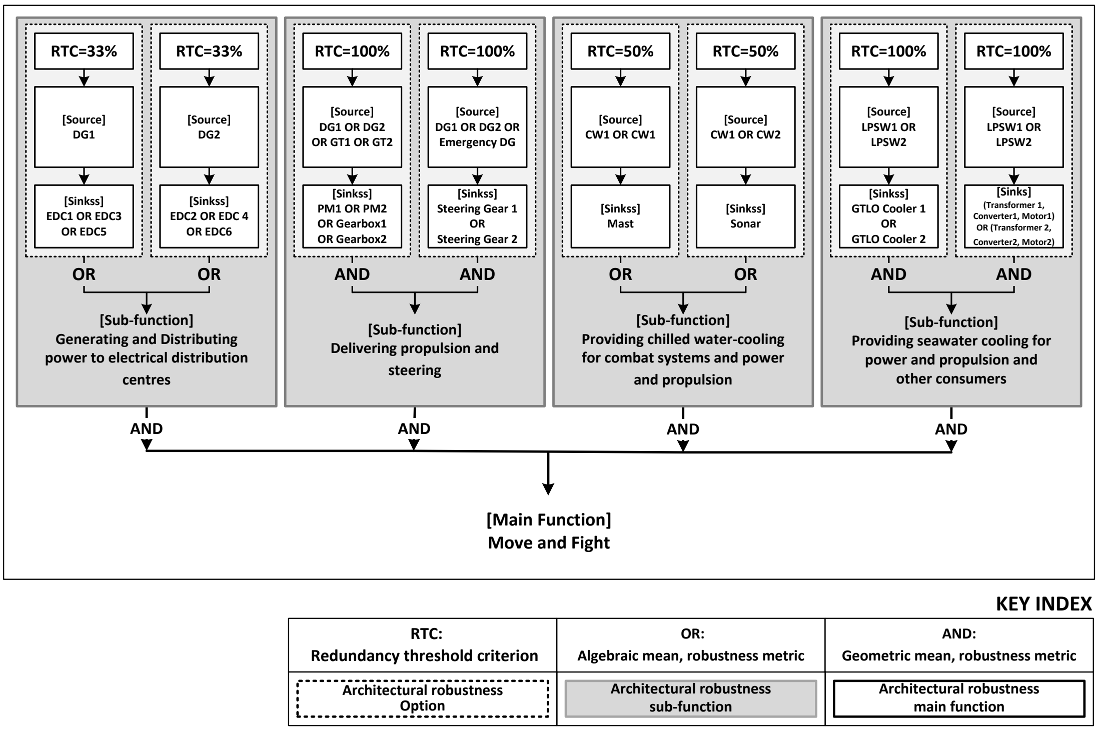


Figure 22: Functional hierarchy, architectural options and RTC for medium redundancy system

Combinatory disruption of components

The following combinatory disruptions of two components for the medium-level redundancy Type A system architectures were generated through the methodology. The results Table 17 Columns are as before:

- Column A “Is the system generating and distributing power?”
- Column B “Is the system delivering propulsion and steering?”
- Column C “Is the system providing chilled water?”
- Column D “Is the system providing seawater?”
- Column E “Is the system able to move and operate?”

Table 17: Combinatory disruptions of components Type A medium redundancy architecture

	If removed component 1	and removed component 2	A	B	C	D	E
1	Diesel Generator 1	Diesel Generator 2	N	Y	Y	Y	N
2	Diesel Generator 1	Transformer 2	N	Y	Y	Y	N
3	Diesel Generator 1	HV Switchboard 2	N	Y	Y	Y	N
4	Diesel Generator 1	LV Switchboard 2	N	Y	Y	Y	N
5	Diesel Generator 2	Transformer 1	N	Y	Y	Y	N
6	Diesel Generator 2	HV Switchboard 1	N	Y	Y	Y	N
7	Diesel Generator 2	LV Switchboard 1	N	Y	Y	Y	N
8	Transformer 1	Transformer 2	N	Y	Y	N	N
9	Transformer 1	HV Switchboard 2	N	Y	Y	Y	N
10	Transformer 1	LV Switchboard 2	N	Y	Y	Y	N
11	Transformer 1	Propulsion Motor 2	Y	Y	Y	N	N
12	Transformer 1	Converter Regular 2	Y	Y	Y	N	N
13	Transformer 1	LP Seawater Pump 2	Y	Y	Y	N	N
14	Transformer 1	LP Seawater Pump Manifold 2	Y	Y	Y	N	N
15	Transformer 2	HV Switchboard 1	N	Y	Y	Y	N
16	Transformer 2	LV Switchboard 1	N	Y	Y	Y	N
17	Transformer 2	Propulsion Motor 1	Y	Y	Y	N	N
18	Transformer 2	Converter Regular 1	Y	Y	Y	N	N
19	Transformer 2	LP Seawater Pump 1	Y	Y	Y	N	N
20	Transformer 2	LP Seawater Pump Manifold 1	Y	Y	Y	N	N
21	HV Switchboard 1	HV Switchboard 2	N	Y	Y	Y	N
22	HV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
23	HV Switchboard 2	LV Switchboard 1	N	Y	Y	Y	N
24	LV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
25	Propulsion Motor 1	Propulsion Motor 2	Y	Y	Y	N	N
26	Propulsion Motor 1	Converter Regular 2	Y	Y	Y	N	N
27	Propulsion Motor 1	LP Seawater Pump 2	Y	Y	Y	N	N
28	Propulsion Motor 1	LP Seawater Pump Manifold 2	Y	Y	Y	N	N
29	Propulsion Motor 2	Converter Regular 1	Y	Y	Y	N	N
30	Propulsion Motor 2	LP Seawater Pump 1	Y	Y	Y	N	N
31	Propulsion Motor 2	LP Seawater Pump Manifold 1	Y	Y	Y	N	N

	If removed component 1	and removed component 2	A	B	C	D	E
32	Converter Regular 1	Converter Regular 2	Y	Y	Y	N	N
33	Converter Regular 1	LP Seawater Pump 2	Y	Y	Y	N	N
34	Converter Regular 1	LP Seawater Pump Manifold 2	Y	Y	Y	N	N
35	Converter Regular 2	LP Seawater Pump 1	Y	Y	Y	N	N
36	Converter Regular 2	LP Seawater Pump Manifold 1	Y	Y	Y	N	N
37	Steering Gear Hydraulic Power Pack 1	Steering Gear Hydraulic Power Pack 2	Y	N	Y	Y	N
38	CW Plant 1	CW Plant Chillier 2	Y	Y	N	Y	N
39	CW Plant 1	CW Plant 2-Manifold	Y	Y	N	Y	N
40	CW Plant 2	CW Plant 1 -Manifold	Y	Y	N	Y	N
41	CW Plant 1 -Manifold	CW Plant 2-Manifold	Y	Y	N	Y	N
42	CW Plant 1 -Manifold	Essential Consumer2 -Sonar	Y	Y	N	Y	N
43	CW Plant 2-Manifold	Essential Consumer1 - Mast	Y	Y	N	Y	N
44	Essential Consumer1 - Mast	Essential Consumer2 -Sonar	Y	Y	N	Y	N
45	LP Seawater Pump 1	LP Seawater Pump 2	Y	Y	Y	N	N
46	LP Seawater Pump 1	LP Seawater Pump Manifold 2	Y	Y	Y	N	N
47	LP Seawater Pump 1	GT LO Cooler 2	Y	Y	Y	N	N
48	LP Seawater Pump 2	LP Seawater Pump Manifold 1	Y	Y	Y	N	N
49	LP Seawater Pump 2	GT LO Cooler 1	Y	Y	Y	N	N
50	LP Seawater Pump Manifold 1	LP Seawater Pump Manifold 2	Y	Y	Y	N	N
51	LP Seawater Pump Manifold 1	GT LO Cooler 2	Y	Y	Y	N	N
52	LP Seawater Pump Manifold 2	GT LO Cooler 1	Y	Y	Y	N	N
53	GT LO Cooler 1	GT LO Cooler 2	Y	Y	Y	N	N
	Total		**16**	**1**	**7**	**30**	**53**

There is a total of 53 combinatory disruptions that led to zero robustness for medium redundancy. It was observed that the medium redundancy configuration generated all the 36 failures identified in the high redundancy case with an additional 17 failures. For example, the combined disruption of generators DG1 and DG2 caused the loss of electrical power (sub-function A) and the total loss of the system to move and operate (function E) as these were the only two main electrical power sources in this medium redundancy configuration. Note the simplified representation of the sub-functions to provide propulsion (B), chilled water (C) and seawater (D) do not include the underlying need for electrical power and the Table shows these systems operational, however, the overall function (E) is calculated from A and B and C and D, capturing the interdependencies.

Modularity intermediate loop: Low – medium – high modularity

Table 18 presents the results of the modularity loops for the system architecture of medium level of redundancy.

Table 18: Modular configurations for Type A medium redundancy architecture

Second redundancy loop ***Medium level of redundancy in the system architecture***			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
1	Gas Turbine 2, Gearbox 2, GT LO Cooler 2	Gas Turbine 2, Gearbox 2, GT LO Cooler 2	Gas Turbine 1, Gearbox 1, GT LO Cooler 1
2	Gas Turbine 1, Gearbox 1, GT LO Cooler 1	Gas Turbine 1, Gearbox 1, GT LO Cooler 1	Diesel Generator 1, Transformer 1, HV Switchboard 1, HV Switchboard Inter1, Propulsion Motor 1, Converter Regular 1, LP Seawater Pump Manifold 1
3	Diesel Generator 1, Transformer 1, HV Switchboard 1	Diesel Generator 2, Transformer 2, HV Switchboard 2, HV Switchboard Inter1	LV Switchboard 1, LV Switchboard Inter1, Emergency Generator, Emergency Switchboard, EDC 3, EDC 5, Steering Gear Hydraulic Power Pack 1, LP Seawater Pump 1
4	Diesel Generator 2, HV Switchboard 2, HV Switchboard Inter1	Diesel Generator 1, Transformer 1, HV Switchboard 1	EDC 1, EDC 2, CW Plant Chiller 1, CW Plant Chiller 2, CW Plant Chiller 1 - Manifold, CW Plant Chiller 2-Manifold, Essential Consumer1 - Mast, Essential Consumer2 –Sonar
5	EDC 3, LP Seawater Pump 1	LV Switchboard 1, LV Switchboard Inter1, EDC 5, Steering Gear Hydraulic Power Pack 1	Gas Turbine 2, Gearbox 2, Diesel Generator 2, Transformer 2, HV Switchboard 2, Propulsion Motor 2, Converter Regular 2, LP Seawater Pump Manifold 2, GT LO Cooler 2
6	Transformer 2, LV Switchboard 2	Emergency Generator, Emergency Switchboard, EDC 6, Steering Gear Hydraulic Power Pack 2	LV Switchboard 2, EDC 4, EDC 6, Steering Gear Hydraulic Power Pack 2, LP Seawater Pump 2
7	LV Switchboard 1, EDC 5	EDC 1, EDC 2, CW Plant Chiller 1, CW Plant Chiller 2, CW Plant Chiller 1 - Manifold, CW Plant Chiller 2-Manifold, Essential Consumer1 - Mast, Essential Consumer2 -Sonar	
8	Propulsion Motor 2, Converter Regular 2, LP Seawater Pump Manifold 2	Propulsion Motor 1, Converter Regular 1, LP Seawater Pump Manifold 1	
9	Propulsion Motor 1, Converter Regular 1, LP Seawater Pump Manifold 1	Propulsion Motor 2, Converter Regular 2, LP Seawater Pump Manifold 2	

Second redundancy loop ***Medium level of redundancy in the system architecture***			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
10	LV Switchboard Inter1, Steering Gear Hydraulic Power Pack 1	EDC 3, LP Seawater Pump 1	
11	Emergency Generator, Emergency Switchboard, EDC 6, Steering Gear Hydraulic Power Pack 2	LV Switchboard 2, EDC 4, LP Seawater Pump 2	
12	EDC 1, CW Plant Chiller 1, Essential Consumer2 -Sonar		
13	EDC 2, CW Plant Chiller 2, CW Plant Chiller 1 - Manifold, CW Plant Chiller 2-Manifold, Essential Consumer1 - Mast		
14	EDC 4, LP Seawater Pump 2		
Robustness metric	0.745	0.702	0.544
Modularity metric	0.71	0.78	0.84
Classification of modules (Modules ID)			
central modules	[11]	[7]	[5]
peripheral modules	[5,6,7,8, 9,10,14]	[8,9,10]	[1]
semi peripheral modules	[1,2,3,4, 12,13]	[1,2,3,4,5,6,11]	[2,3,4,6]

Figures 23 and 24 below illustrate the examples of the proposed modules generated by the RoMoGA analysis as presented in Table 18.

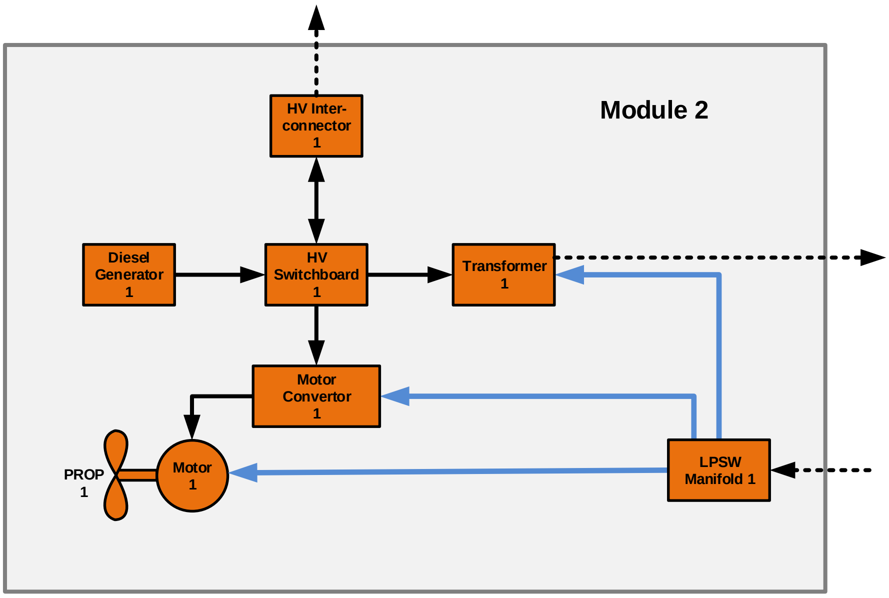


Figure 23: Module 2 Type A medium redundancy and high modularity

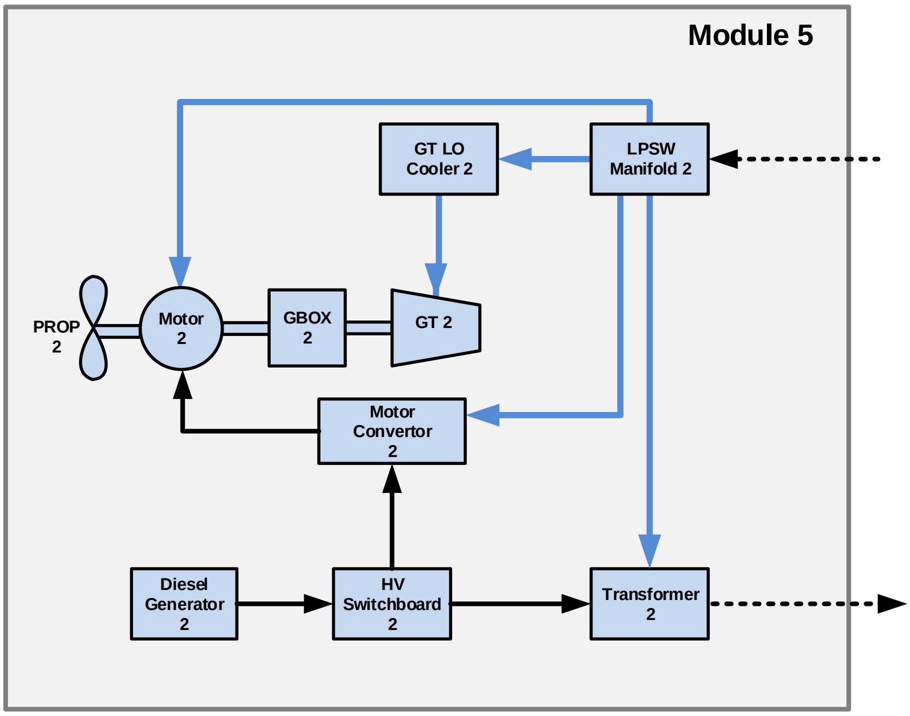


Figure 24: Module 5 Type A medium redundancy and high modularity

The results generated for the medium redundancy share similarities with the results generated for the high-level of redundancy. This suggests that iterating through the various levels of redundancy and modularity for the same system architecture can help architects identify common candidate robust modules. This could allow potential standardisation and commonality design initiatives to be implemented. Identifying standard and common robust modules can help to develop library robust modules that can be used at the beginning of the design

8.2.2.3 3rd Redundancy loop: Low redundancy

The low-level system redundancy architecture was developed in the third redundancy loop as shown in Figures 25 and 26. This was achieved by minimising the level of connectivity and number of the source components in the system architecture and was able to achieve the main functions defined above. However, in a naval engineering context, this architecture would not satisfy ship design requirements for redundancy or survivability and would not be realistic in the naval context. It was developed as an ideal architecture, for comparison and analysis purposes.

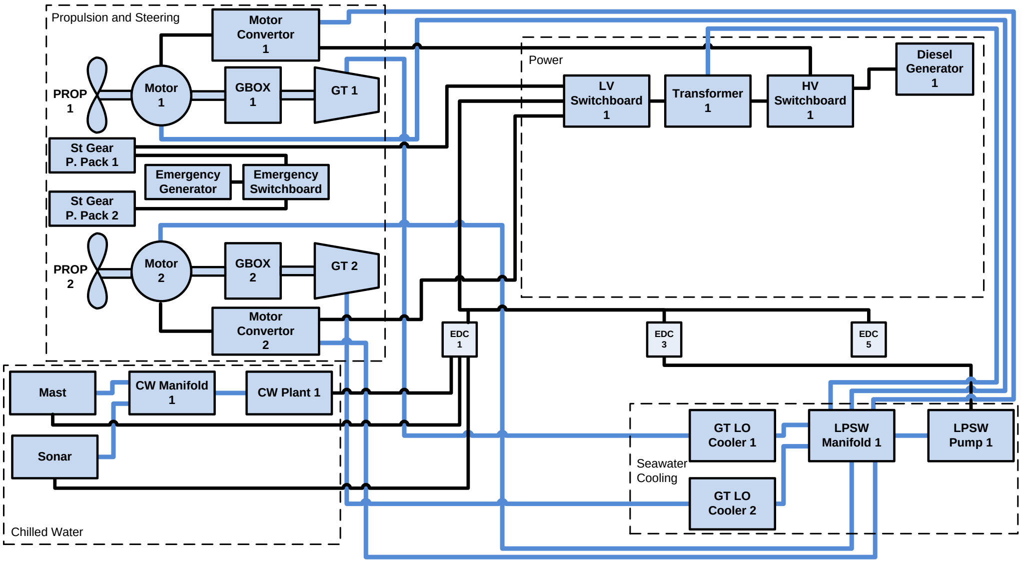


Figure 25: Type A low-level redundancy system schematic

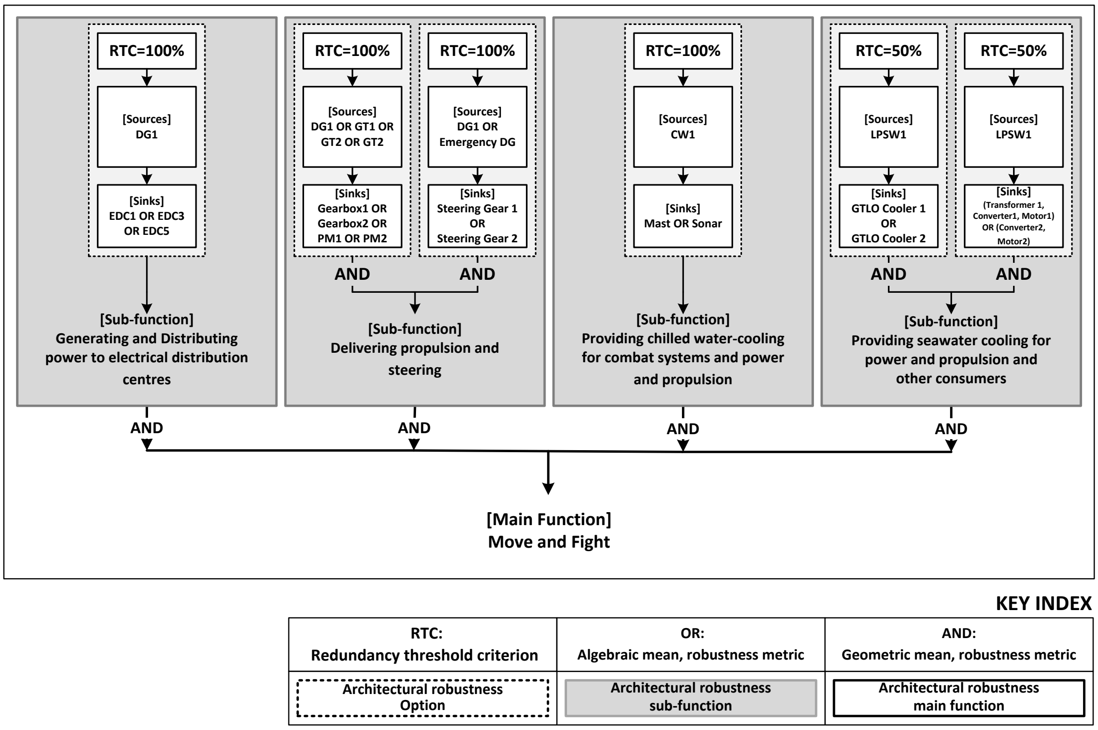


Figure 26: Functional hierarchy, architectural options and RTC for low redundancy system

Combinatory disruption of components

As there was no redundancy, the combinatory component level disruptions for two components gave results with a very high number of possible combinations with zero robustness (Table 19). This was the expected result for low redundancy system architecture and provides a point of comparison for this research.

Table 19: Combinatory disruptions of components Type A low redundancy architecture

If removed component 1	and removed component 2	A	B	C	D	E
Total		**161**	**18**	**98**	**52**	**264**

In the case of low redundancy in the system, each component becomes critical and hence the overall score of combinatory disruptions is 264.

Modularity intermediate loop: Low – medium-high modularity

Table 20 presents the modular configurations for the given low redundancy system architecture which has the highest possible degree of modularity; compared with the medium and high-level architectures.

Table 20: Modular configurations for Type A low redundancy architecture

Third redundancy loop *Low-level of redundancy in the system architecture*			
Module ID	**First modularity loop** *low-level of modularity*	**Second modularity loop** *medium level of modularity*	**Third modularity loop** *high-level of modularity*
1	Gas Turbine 2, Gearbox 2, GT LO Cooler 2	Gas Turbine 1, Gearbox 1, GT LO Cooler 1	Gas Turbine 2, Gearbox 2, GT LO Cooler 2
2	Gas Turbine 1, Gearbox 1, GT LO Cooler 1	Gas Turbine 2, Gearbox 2, Propulsion Motor 1	Transformer 1, Transformer 2, LV Switchboard 1, LV Switchboard Inter2, Emergency Generator
3	Diesel Generator 1, Transformer 1, HV Switchboard 1	Diesel Generator 1, Diesel Generator 2, Diesel Generator 3	Gas Turbine 1, Gearbox 1, LV Switchboard 2, HV Switchboard Inter1, HV Switchboard Inter2, LV Switchboard Inter1, EDC 5, EDC 6
4	LV Switchboard 1	Transformer 1, Transformer 2, LV Switchboard 1, LV Switchboard Inter2, Emergency Generator	HV Switchboard 1, Emergency Switchboard, EDC 1, EDC 2, EDC 3
5	Emergency Generator, Emergency Switchboard, Steering Gear Hydraulic Power Pack 2	HV Switchboard Inter1, LV Switchboard Inter1, EDC 5	Diesel Generator 1, Diesel Generator 2, Diesel Generator 3, Diesel Generator 4, HV Switchboard 2, EDC 4
6	EDC 3, LP Seawater Pump 1	LV Switchboard 2, HV Switchboard Inter2	
7	EDC 5, Steering Gear Hydraulic Power Pack 1	HV Switchboard 1, Emergency Switchboard, EDC 1, EDC 2, EDC 3	
8	Propulsion Motor 2, Converter Regular 2	Diesel Generator 4, HV Switchboard 2, EDC 4	
9	Propulsion Motor 1, Converter Regular 1		

<table>
<tr><th colspan="4">Third redundancy loop
<i>Low-level of redundancy in the system architecture</i></th></tr>
<tr><th>Module ID</th><th>First modularity loop <i>low-level of modularity</i></th><th>Second modularity loop <i>medium level of modularity</i></th><th>Third modularity loop <i>high-level of modularity</i></th></tr>
<tr><td>10</td><td>EDC 1, CW Plant Chiller 1</td><td></td><td></td></tr>
<tr><td>11</td><td>CW Plant Chiller 1 - Manifold, Essential Consumer1 - Mast, Essential Consumer2 -Sonar</td><td></td><td></td></tr>
<tr><td>12</td><td>EDC 5</td><td></td><td></td></tr>
<tr><td>Robustness metric</td><td>0.467</td><td>0.361</td><td>0.192</td></tr>
<tr><td>Modularity metric</td><td>0.71</td><td>0.85</td><td>0.9</td></tr>
<tr><th colspan="4">Classification of modules (Modules ID)</th></tr>
<tr><td>central modules</td><td>[1,2,3,5, 11]</td><td>[4]</td><td>[5]</td></tr>
<tr><td>periphery modules</td><td>[6,7,8,9,10]</td><td>[5,6]</td><td>[1]</td></tr>
<tr><td>semi periphery modules</td><td>[4,12]</td><td>[1,2,3,7,8]</td><td>[2,3,4]</td></tr>
</table>

The calculation of the degrees of modularity was - low redundancy 0.9; medium redundancy 0.84; high redundancy 0.8.

The low-level redundancy had the smallest potential to generate robust modularisation as shown in Table 20 whereas most of the modules generated were non-robust (shaded/red).

8.2.2.4 Accumulated robustness results

Table 21 gives the robustness results for a single module disruption for the different inputs of redundancy and modularity.

Table 21: Average robustness results for Type A system architecture options

RUNS	Design variables		Response variable
	Level of redundancy	**Level of modularity**	**Level of robustness**
Run 1	Low	Low	0.467
Run 2	Low	Medium	0.361
Run 3	Low	High	0.192
Run 4	Medium	Low	0.745
Run 5	Medium	Medium	0.702
Run 6	Medium	High	0.544
Run 7	High	Low	0.757
Run 8	High	Medium	0.730
Run 9	High	High	0.467

The Table shows that the level of robustness falls as the level of modularity increases. Systems with the highest level of redundancy have the highest levels of robustness with a broad similarity between systems with high and medium redundancy. In the low redundancy systems, a minimum equipment arrangement to meet the functional requirements, which would not form a practical, real-world arrangement, there is a significant reduction in robustness. The maximum robustness value in Run 1 is 0.467, about 40%, of the high redundancy system's robustness, Run 7. It was suggested that this could form a lower bound when assessing robustness responses. The high and medium redundancy systems have robustness values between 0.757 and 0.702 for low and medium modularity. This suggests that the calculated robustness values can be used as quantitative evaluation indicators to compare amongst different types of system architecture.

8.2.3 Evaluation stage

The evaluation stage of the descriptive implementation of RoMoGA methodology developed robust modular configurations for each architecture. Only the high and medium levels of redundancy were used. the low redundancy architecture was not used as it is not an appropriate solution to allow robust modularisation.

Moreover, the DOE analysis was performed based on the average robustness results presented in Table 21, to gain an overview of the effects of the different levels of modularity and redundancy to robustness

8.2.3.1 Robust modular reconfigurations

Table 22 presents the robust modular configuration devised following the guidelines given in the evaluation stage of descriptive RoMoGA methodology.

Table 22: Robust modular configuration for Type A medium and high redundancy architectures

MODULE ID	Medium level of redundancy in the architecture	High-level of redundancy in the architecture
1	Gas Turbine 1, Gearbox 1, GT LO Cooler 1	Gas Turbine 2, Gearbox 2, GT LO Cooler 2
2	Diesel Generator 1, Transformer 1, HV Switchboard 1, HV Switchboard Inter1, Propulsion Motor 1, Converter Regular 1, LP Seawater Pump Manifold 1	Diesel Generator 2, Diesel Generator 4, Transformer 2, HV Switchboard 2, HV Switchboard Inter1, HV Switchboard Inter2, Propulsion Motor 2, Converter Regular 2, LP Seawater Pump Manifold 2

MODULE ID	Medium level of redundancy in the architecture	High-level of redundancy in the architecture
3	LV Switchboard 1, LV Switchboard Inter1, Emergency Generator, Emergency Switchboard, EDC 3, EDC 5, Steering Gear Hydraulic Power Pack 1, LP Seawater Pump 1	LV Switchboard 1, LV Switchboard Inter2, EDC 3, EDC 5, Steering Gear Hydraulic Power Pack 1, CW Plant Chiller 3, CW Plant Chiller 3-Manifold, LP Seawater Pump 1
4	EDC 1, CW Plant Chiller 1, Essential Consumer1 - Mast, CW Plant Chiller 1 -Manifold	Gas Turbine 1, Gearbox 1, Diesel Generator 1, Diesel Generator 3, Transformer 1, HV Switchboard 1, Propulsion Motor 1, Converter Regular 1, LP Seawater Pump Manifold 1, GT LO Cooler 1
5	EDC 2, CW Plant Chiller 2, CW Plant Chiller 2-Manifold, Essential Consumer2 -Sonar	EDC 1, CW Plant Chiller 1, CW Plant Chiller 1 -Manifold, Essential Consumer1 – Mast
6	Gas Turbine 2, Gearbox 2, Diesel Generator 2, Transformer 2, HV Switchboard 2, Propulsion Motor 2, Converter Regular 2, LP Seawater Pump Manifold 2, GT LO Cooler 2	EDC 2, CW Plant Chiller 2, CW Plant Chiller 2-Manifold, Essential Consumer2 –Sonar
7	LV Switchboard 2, EDC 4, EDC 6, Steering Gear Hydraulic Power Pack 2, LP Seawater Pump 2	LV Switchboard 2, LV Switchboard Inter1, Emergency Generator, Emergency Switchboard, EDC 4, EDC 6, Steering Gear Hydraulic Power Pack 2, LP Seawater Pump 2
Modularity metric	0.78	0.76
Robustness metric	0.70	0.64

The robust modular configuration ensures that in case of a module disruption the system continues functioning. Figure 27 and 28 illustrate the robust module configuration, for medium and high redundancy systems, corresponding to Table 22. Seven robust modules were identified for the medium and high redundancy architectures. The different colours identify the modules.

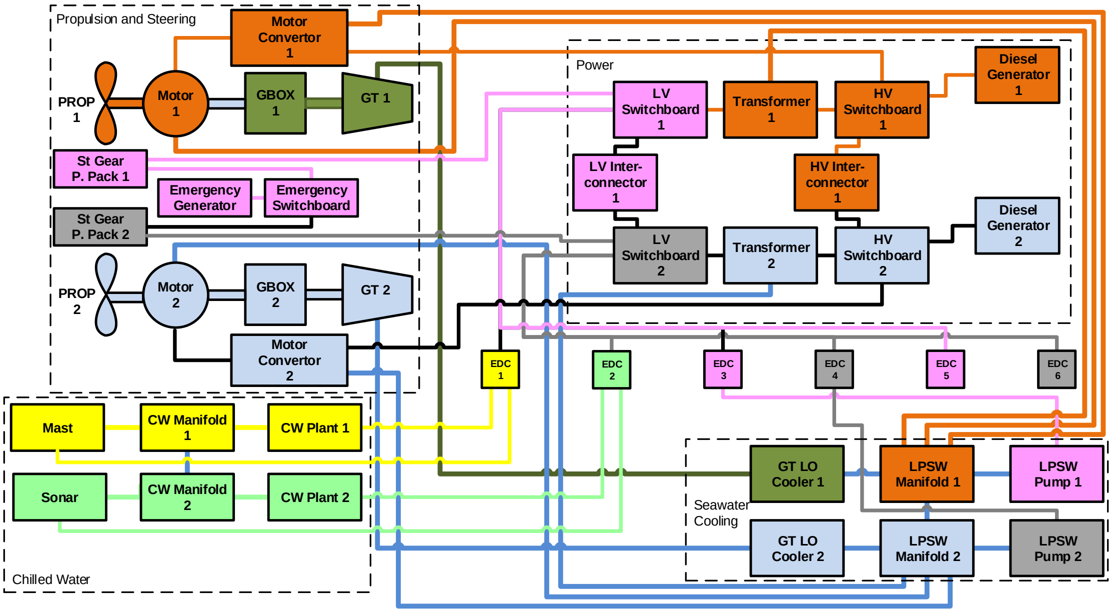


Figure 27: Resulting robust modular configuration for medium-level redundancy

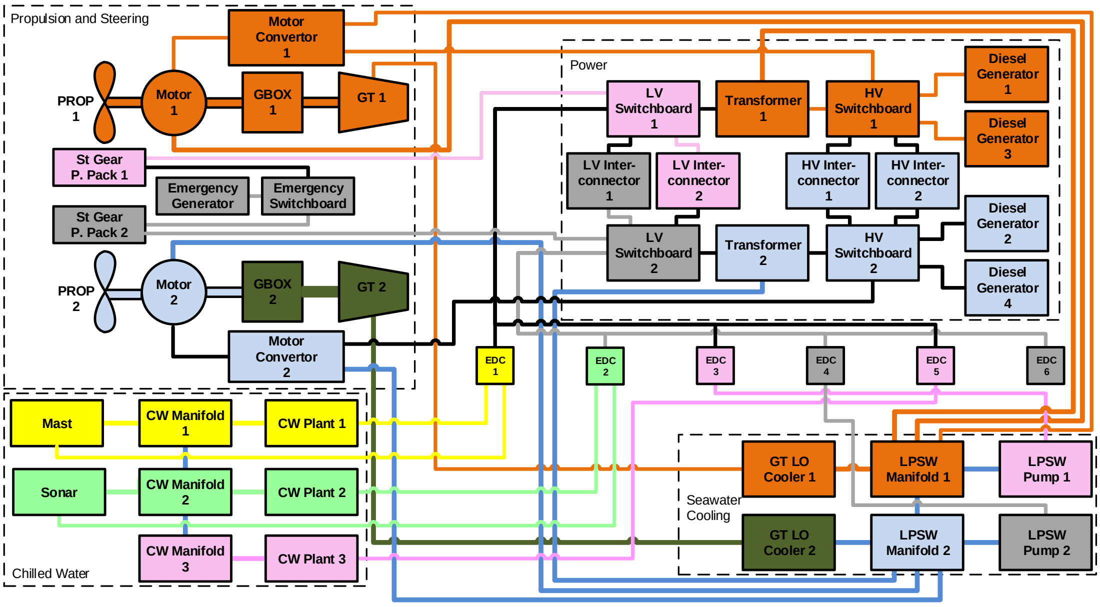


Figure 28: Resulting robust modular configuration for high redundancy system

The robust modular configuration for both the medium and high redundancy ensures a high-level (not maximum) of modularity in the system architecture by enabling the partition into modules without penalising the robustness. The robust modular configurations were discussed with the SMEs who advised that they considered the modules to be logical in the context of naval design.

Many modularity definitions in the literature (Ulrich and Tung, 1991) emphasis the one-to-one mapping between modules and functions. In the analysis stage of the proposed methodology modularity is determined through the structural network, and it's then evaluated the behaviour network, allow identification of the functional distinct modules. By identifying robust modules, that means that a module could be taken out of the architecture and the system will still keep functioning sufficiently, that implies that the robust modules it self-has a distinct functional character. The methodology does not produce functionally independent modules but generates robust modules with a functional distinct identity. For example, the orange module contains power, propulsion and low-pressure cooling system components. The dominant function is the propulsion, with additional elements showing electrical supplies for propulsion and the seawater cooling for propulsion.

8.2.3.2 DOE analysis: Redundancy and modularity effects on robustness

Figure 29 illustrates the results of the DOE analysis results in Table 21, showing average robustness for (a) redundancy (averaged across all modularity results) and (b) modularity (averaged across all redundancies).

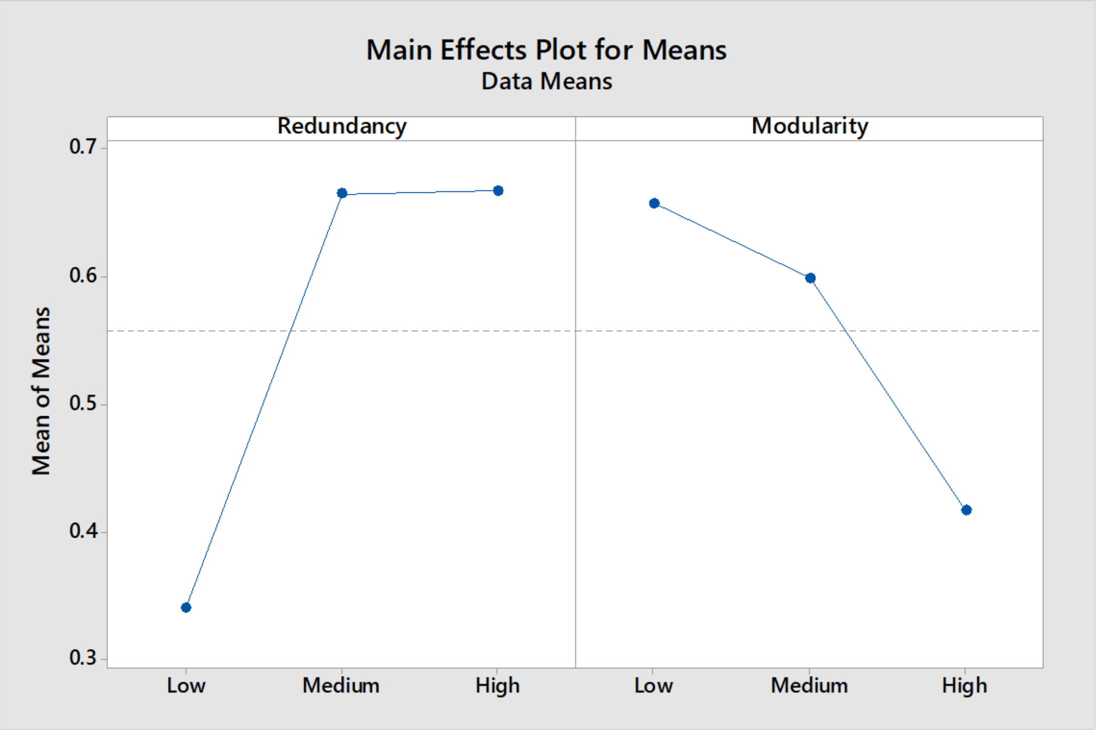


Figure 29: Main effects of redundancy and modularity on robustness Type A

Figure 29 indicates that an increase in the modularity level reduces robustness. This preliminary observation was expected, as disrupting a module of highest modularity would lead to the loss of a bigger size module which contains a higher number of components in the architecture. Medium redundancy compared to low redundancy, significantly improves robustness. However, changes to high redundancy show no improvement in robustness compared to the medium redundancy case. The findings of the application of the methodology to naval technical system Type A suggested that a medium redundancy and a medium level of modularity could be an acceptable trade-off to maintain a level of robustness of the system architecture.

8.3 Overview of the case studies: Type B and C

The methodology is applied to two more existing naval ship system architectures (Types B and C). Details of the application are provided in Appendix II, which describes the results of the stage of analysis. The following paragraphs provide an overview of the generation and evaluation stages for the application of the methodology to Type B and Type C.

8.3.1 Type B: Destroyer

Type B is an AAW ship (anti-aircraft warfare) with all-electric propulsion (Figure 30). Power is generated by three diesel generators and two gas turbines alternators, which supply the high voltage (HV) system for the two propulsion motors and, via transformers to the LV system power to the ship system. The third DG connects to a third HV switchboard, which is inserted into the interconnector so the third HV switchboard connects to both HV switchboards. Gas turbines (GT) are dependent on the seawater system for cooling (GT Intercooler and GT Lubricating Oil Module). The chilled water plants cool the transformers, the HV and LV filters associated with the propulsion motors, and the radar equipment in the fore and aft masts. The auxiliary seawater system cools propulsion converters and motors. Type B is an integral and complex architecture with many dependencies between the power and propulsion systems and cooling.

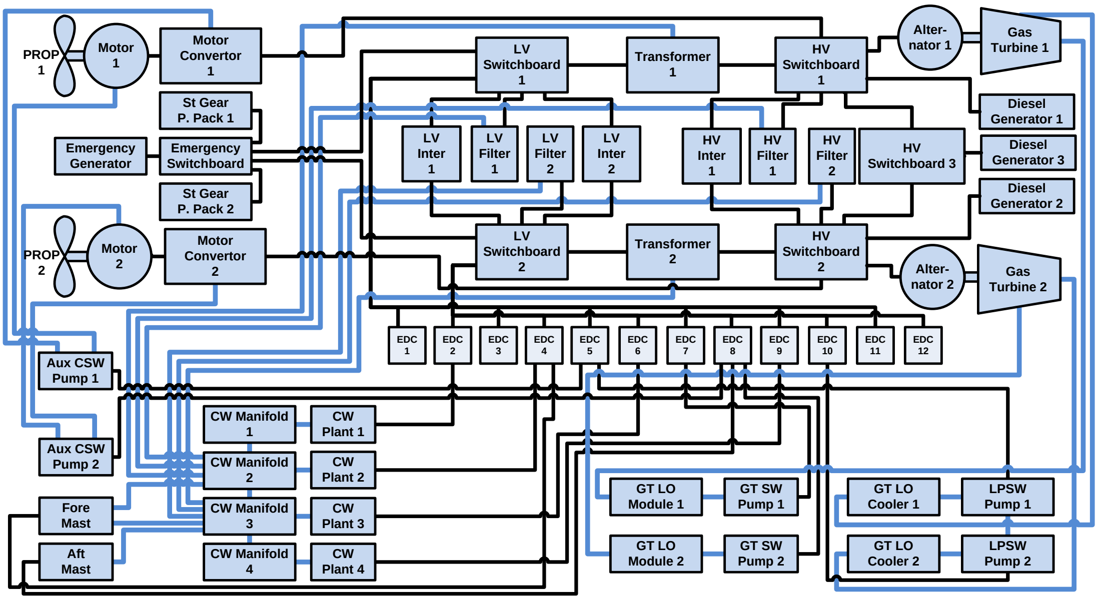
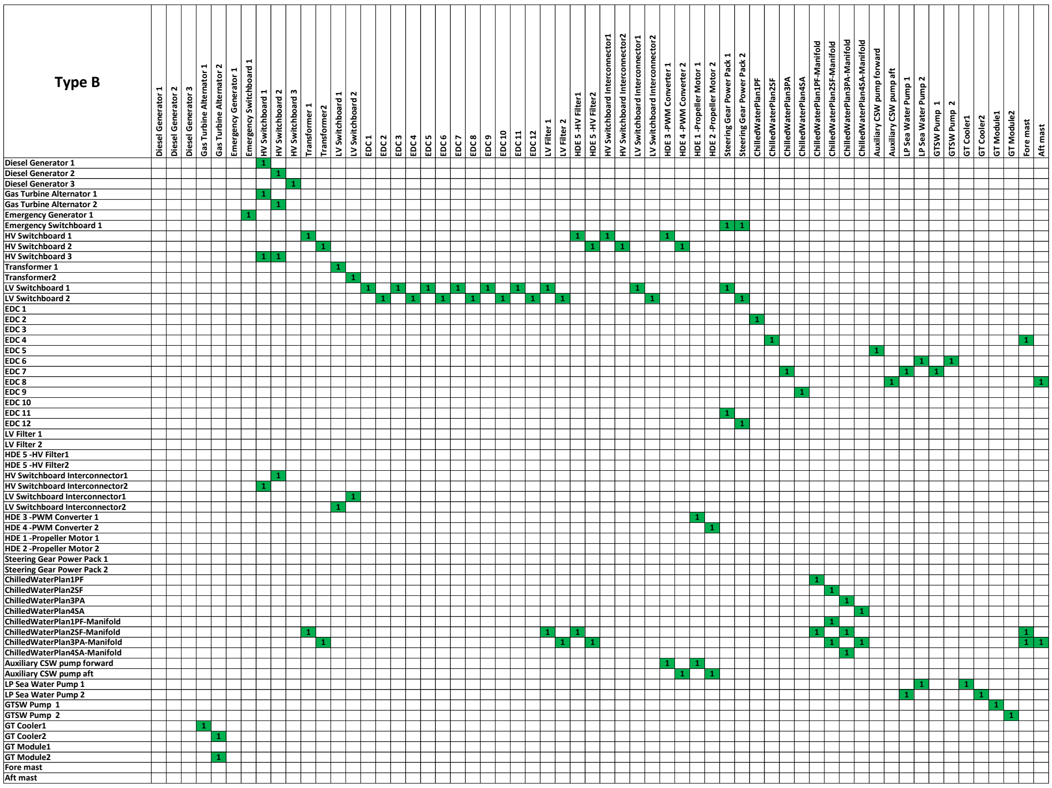


Figure 30: Type B nominal system architecture schematic (high redundancy)

8.3.2 Generation stage

The generation stage entails the functional network definition in DSM.



Type B	Diesel Generator 1	Diesel Generator 2	Diesel Generator 3	Gas Turbine Alternator 1	Gas Turbine Alternator 2	Emergency Generator 1	Emergency Switchboard 1	HV Switchboard 1	HV Switchboard 2	HV Switchboard 3	Transformer 1	Transformer2	LV Switchboard 1	LV Switchboard 2	EDC 1	EDC 2	EDC 3	EDC 4	EDC 5	EDC 6	EDC 7	EDC 8	EDC 9	EDC 10	EDC 11	EDC 12	LV Filter 1	LV Filter 2	HDE 5 -HV Filter1	HDE 5 -HV Filter2	HV Switchboard Interconnector1	HV Switchboard Interconnector2	LV Switchboard Interconnector1	LV Switchboard Interconnector2	HDE 3 -PWM Converter 1	HDE 4 -PWM Converter 2	HDE 1 -Propeller Motor 1	HDE 2 -Propeller Motor 2	Steering Gear Power Pack 1	Steering Gear Power Pack 2	ChilledWaterPlan1PF	ChilledWaterPlan2SF	ChilledWaterPlan3PA	ChilledWaterPlan4SA	ChilledWaterPlan1PF-Manifold	ChilledWaterPlan2SF-Manifold	ChilledWaterPlan3PA-Manifold	ChilledWaterPlan4SA-Manifold	Auxiliary CSW pump forward	Auxiliary CSW pump aft	LP Sea Water Pump 1	LP Sea Water Pump 2	GTSW Pump 1	GTSW Pump 2	GT Cooler1	GT Cooler2	GT Module1	GT Module2	Fore mast	Aft mast
Diesel Generator 1							1																																																					
Diesel Generator 2								1																																																				
Diesel Generator 3									1																																																			
Gas Turbine Alternator 1							1																																																					
Gas Turbine Alternator 2								1																																																				
Emergency Generator 1						1																																																						
Emergency Switchboard 1																																						1	1																					
HV Switchboard 1										1																		1		1				1																										
HV Switchboard 2											1																		1		1				1																									
HV Switchboard 3							1	1																																																				
Transformer 1												1																																																
Transformer2													1																																															
LV Switchboard 1														1		1		1		1		1		1		1						1						1																						
LV Switchboard 2															1		1		1		1		1		1		1						1						1																					
EDC 1																																																												
EDC 2																																								1																				
EDC 3																																																												
EDC 4																																									1																	1	
EDC 5																																																1												
EDC 6																																																			1		1							
EDC 7																																										1								1		1								
EDC 8																																																	1										1
EDC 9																																											1																	
EDC 10																																																												
EDC 11																																						1																						
EDC 12																																							1																					
LV Filter 1																																																												
LV Filter 2																																																												
HDE 5 -HV Filter1																																																												
HDE 5 -HV Filter2																																																												
HV Switchboard Interconnector1								1																																																				
HV Switchboard Interconnector2							1																																																					
LV Switchboard Interconnector1													1																																															
LV Switchboard Interconnector2												1																																																
HDE 3 -PWM Converter 1																																				1																							
HDE 4 -PWM Converter 2																																					1																						
HDE 1 -Propeller Motor 1																																																												
HDE 2 -Propeller Motor 2																																																												
Steering Gear Power Pack 1																																																												
Steering Gear Power Pack 2																																																												
ChilledWaterPlan1PF																																												1																
ChilledWaterPlan2SF																																													1															
ChilledWaterPlan3PA																																														1														
ChilledWaterPlan4SA																																															1													
ChilledWaterPlan1PF-Manifold																																													1															
ChilledWaterPlan2SF-Manifold										1																1		1																1		1												1	
ChilledWaterPlan3PA-Manifold											1																1		1																1		1											1	1
ChilledWaterPlan4SA-Manifold																																														1														
Auxiliary CSW pump forward																																			1		1																							
Auxiliary CSW pump aft																																				1		1																						
LP Sea Water Pump 1																																																			1			1						
LP Sea Water Pump 2																																																		1					1					
GTSW Pump 1																																																								1				
GTSW Pump 2																																																									1			
GT Cooler1				1																																																								
GT Cooler2					1																																																							
GT Module1																																																												
GT Module2					1																																																							
Fore mast																																																												
Aft mast																																																												

Figure 31: DSM Type B

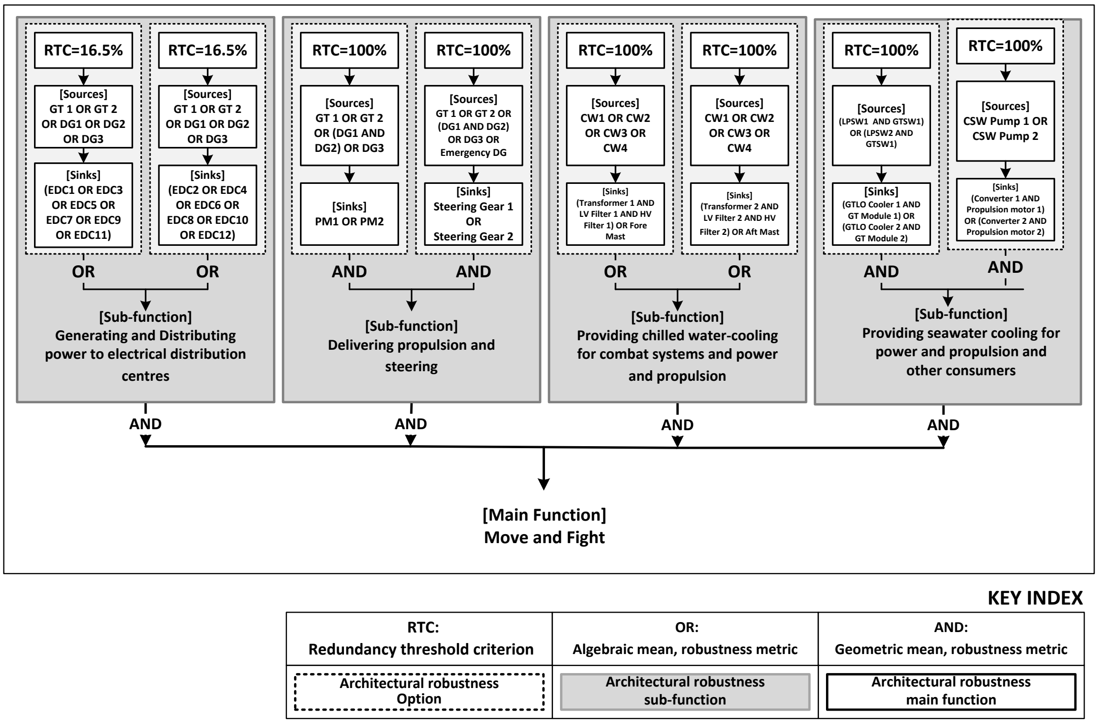


Figure 32: Functional hierarchy, architectural options and RTC for high redundancy Type B

The above Figure 32 presents the functional hierarchy, architectural options, and redundancy threshold criterion for the Type B high-level redundancy system architecture. It is noted that for the seawater cooling sub-function is fulfilled when there is cooling to the (propulsion motor 1 OR propulsion motor 2) AND (cooling GT1 OR GT2) i.e. cooling is always required for propulsion. This is noted as it reflects on the calculation of robustness in Type B.

8.3.3 Evaluation stage

The evaluation part of the methodology is presented below, which includes robust modular reconfigurations and a DOE diagram.

8.3.3.1 Robust modular reconfigurations

Given that Type B is a highly redundant system architecture the iterations through the redundancy loops of the proposed robust modular assessment methodology led to the formulation of robust modular configuration for all the levels of redundancy (high, medium and low, See Appendix II).

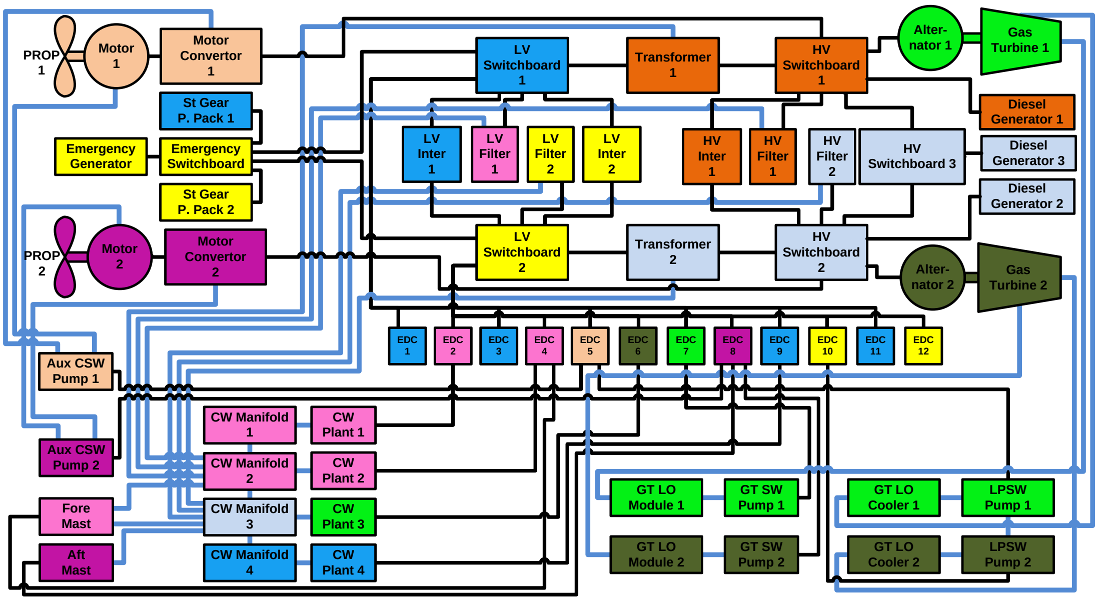


Figure 33: Resulting robust modular configuration for high redundancy system Type B

The robust module for high-level redundancy (Figure 33) generates four robust power generation modules, although the design includes five individual power source components (GT 1 & 2, DG 1-3). The light blue robust module in Figure 33 contains the two DGs inside (Diesel Generator 2 & 3). In comparison, the light blue robust module in Figure 34 for medium redundancy is the same but the redundancy (second diesel generator in blue modules) is removed.

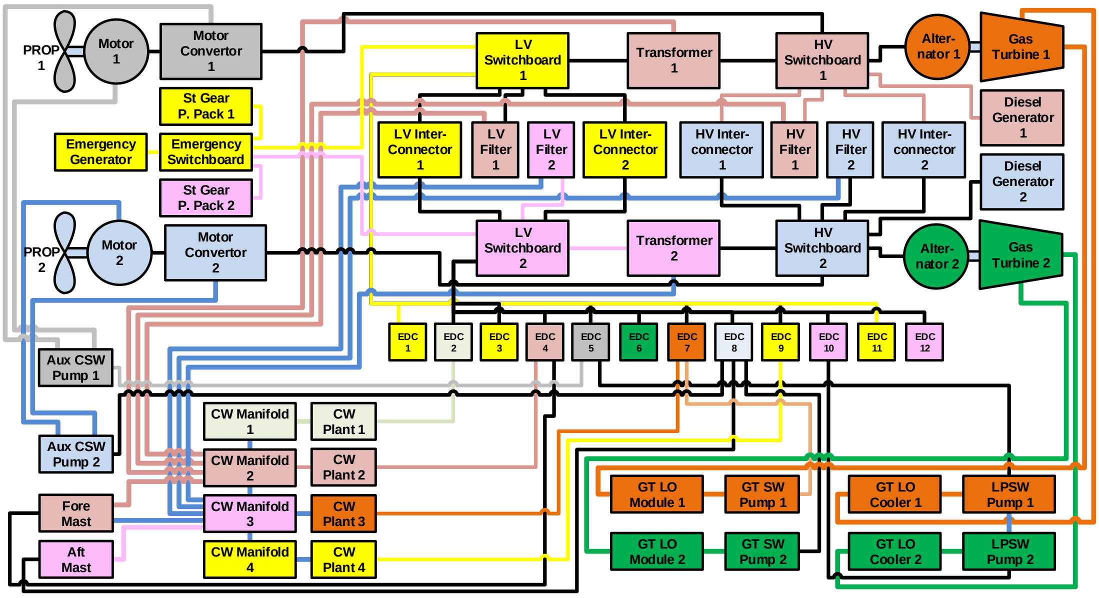


Figure 34: Resulting robust modular configuration for medium-level redundancy system Type B

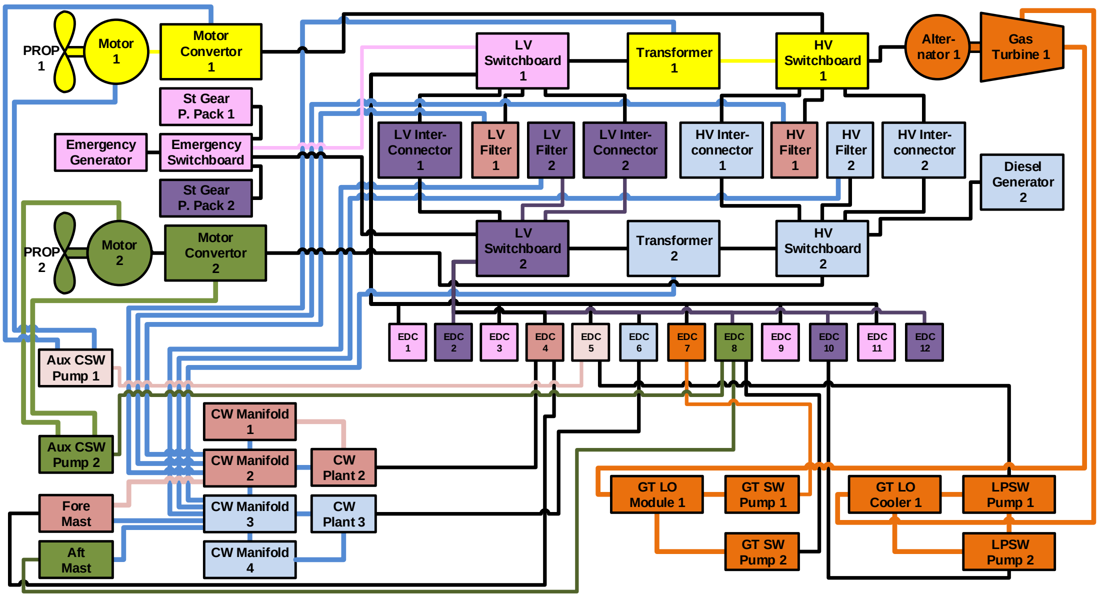


Figure 35: Resulting robust modular configuration for low-level redundancy Type B

They are two robust low redundancy power generation modules based on two power generation source components in Figure 35. In contrast to the low redundancy Type A configuration, the low redundancy Type B system architecture can reconfigure a robust modular configuration because there is sufficient redundancy. The reason is that the derivations of the medium and low system architectures were based on the high redundancy Type B system, which is more redundant than the high redundancy Type A.

8.3.3.2 DOE Analysis: redundancy and modularity effects on robustness

Table 23 presents the robustness results for a single module disruption for the different levels of the redundancy and modularity.

Table 23: Average robustness results for Type B system architecture options

RUNS	Design variables		Response variable
	Level of redundancy	Level of modularity	Level of robustness
Run 1	Low	Low	0.823
Run 2	Low	Medium	0.730
Run 3	Low	High	0.659
Run 4	Medium	Low	0.805
Run 5	Medium	Medium	0.721
Run 6	Medium	High	0.680
Run 7	High	Low	0.819
Run 8	High	Medium	0.790
Run 9	High	High	0.711

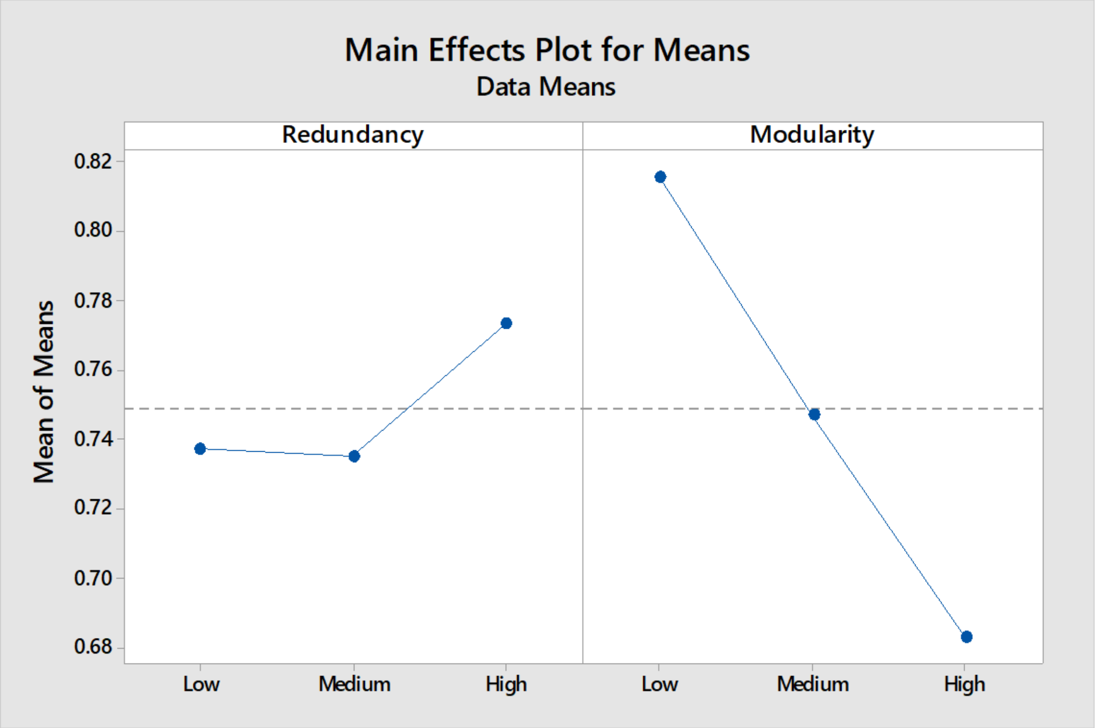


Figure 36: Main effects of redundancy and modularity on robustness

It is observed in Figure 36 that for Type B increasing the level of redundancy from medium to high benefits robustness more than the increase from low to medium. This is different from the DOE analysis results generated for the Type A architecture. The Type B architecture relies on a higher number of redundant components than Type A architecture. This observation is further discussed in the cross-case study comparisons at the end of this Chapter.

8.3.4 Type C: Modern frigate

Type C is a modern anti-submarine warship ship (Figure 37). Power is generated by four diesel generators, which supply two MV switchboards for the two propulsion motors mounted on the propulsion shafts, and via transformers the LV switchboards to power to the whole ship systems. Propulsion can also be provided by a single gas turbine through a cross-connect gearbox to both shafts. LP seawater cooling is provided for the gas turbine intercooler, lubrication cooler, the two propulsion motor converters, two steering gear hydraulic packs, gearbox, and radar. The four chilled water plants are supplied directly from the MV switchboards and provide cooling for the sonar and computer rooms' HVAC. In the action state the chilled water system is split into four quadrants, reducing redundancy but limiting the spread of damage. Type C is considered a medium complex architecture, as it has fewer dependencies to a simpler CW cooling system architecture. The power and propulsion systems are depended to the LP seawater system.

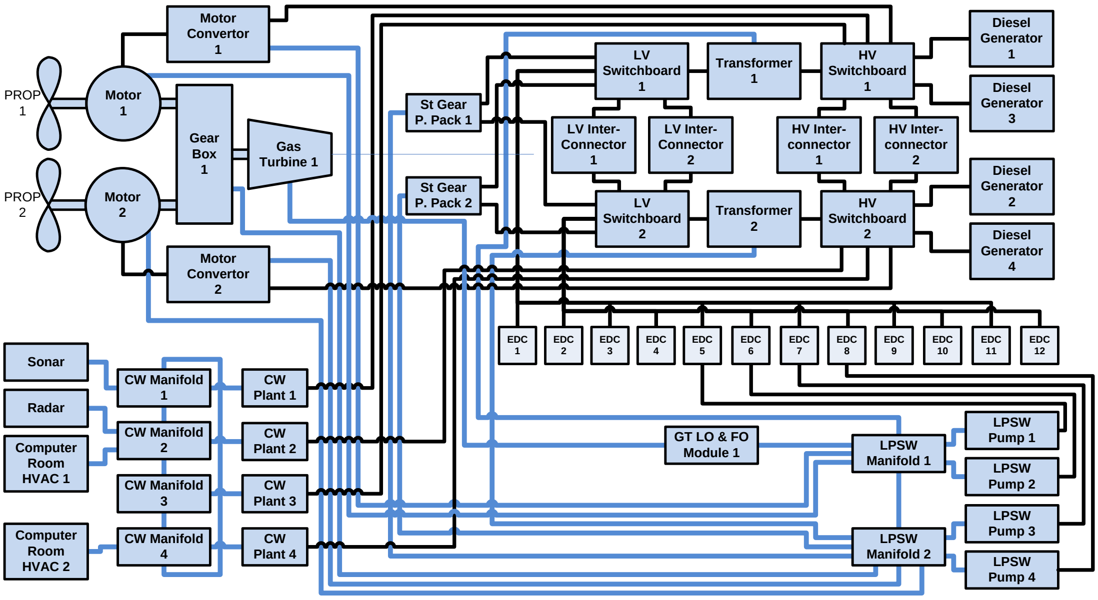
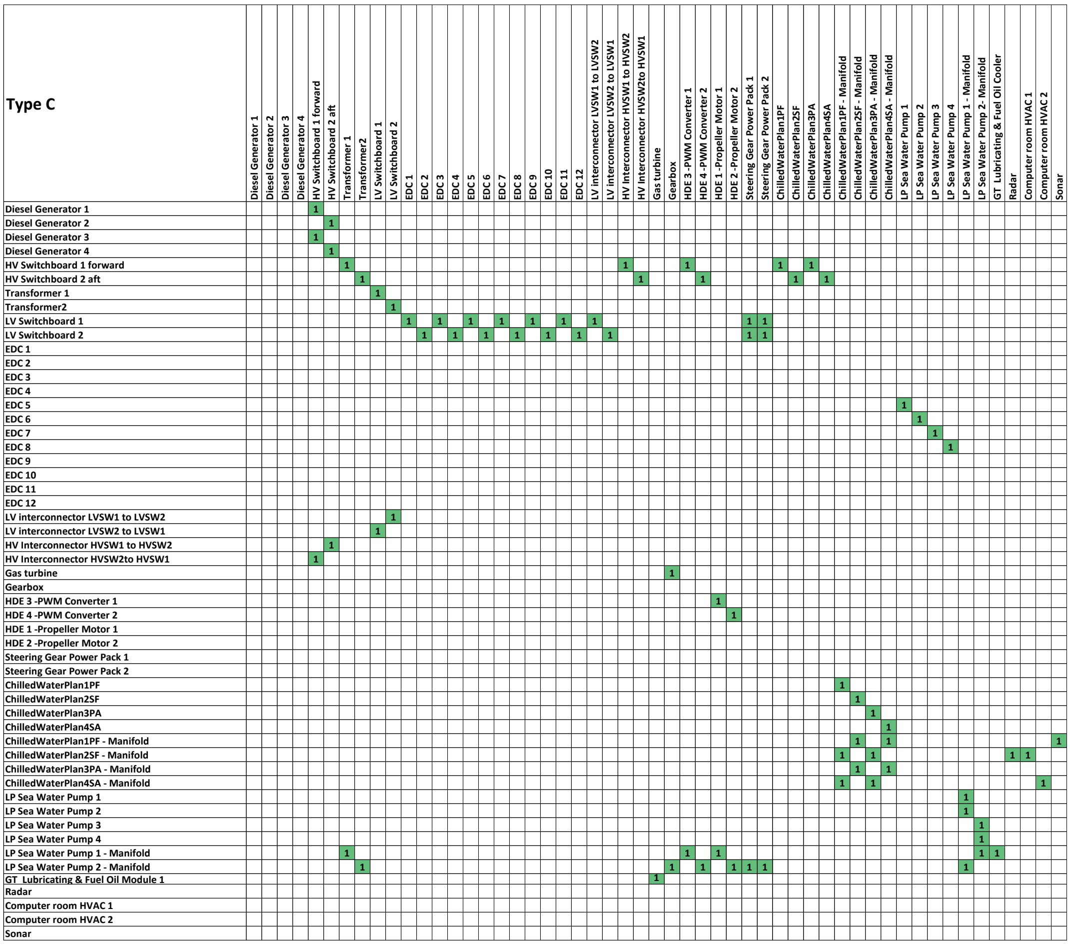


Figure 37: Type C high redundancy schematic

8.3.5 Generation stage

The generation stage entails the functional network definition in DSM.



Type C	Diesel Generator 1	Diesel Generator 2	Diesel Generator 3	Diesel Generator 4	HV Switchboard 1 forward	HV Switchboard 2 aft	Transformer 1	Transformer2	LV Switchboard 1	LV Switchboard 2	EDC 1	EDC 2	EDC 3	EDC 4	EDC 5	EDC 6	EDC 7	EDC 8	EDC 9	EDC 10	EDC 11	EDC 12	LV interconnector LVSW1 to LVSW2	LV interconnector LVSW2 to LVSW1	HV Interconnector HVSW1 to HVSW2	HV Interconnector HVSW2to HVSW1	Gas turbine	Gearbox	HDE 3 -PWM Converter 1	HDE 4 -PWM Converter 2	HDE 1 -Propeller Motor 1	HDE 2 -Propeller Motor 2	Steering Gear Power Pack 1	Steering Gear Power Pack 2	ChilledWaterPlan1PF	ChilledWaterPlan2SF	ChilledWaterPlan3PA	ChilledWaterPlan4SA	ChilledWaterPlan1PF - Manifold	ChilledWaterPlan2SF - Manifold	ChilledWaterPlan3PA - Manifold	ChilledWaterPlan4SA - Manifold	LP Sea Water Pump 1	LP Sea Water Pump 2	LP Sea Water Pump 3	LP Sea Water Pump 4	LP Sea Water Pump 1 - Manifold	LP Sea Water Pump 2- Manifold	GT Lubricating & Fuel Oil Cooler	Radar	Computer room HVAC 1	Computer room HVAC 2	Sonar
Diesel Generator 1					1																																																	
Diesel Generator 2						1																																																
Diesel Generator 3					1																																																	
Diesel Generator 4						1																																																
HV Switchboard 1 forward							1																		1				1						1		1																
HV Switchboard 2 aft								1																		1				1						1		1															
Transformer 1									1																																													
Transformer2										1																																												
LV Switchboard 1											1		1		1		1		1		1		1										1	1																			
LV Switchboard 2												1		1		1		1		1		1		1									1	1																			
EDC 1																																																					
EDC 2																																																					
EDC 3																																																					
EDC 4																																																					
EDC 5																																											1										
EDC 6																																												1									
EDC 7																																													1								
EDC 8																																														1							
EDC 9																																																					
EDC 10																																																					
EDC 11																																																					
EDC 12																																																					
LV interconnector LVSW1 to LVSW2										1																																												
LV interconnector LVSW2 to LVSW1									1																																													
HV Interconnector HVSW1 to HVSW2						1																																																
HV Interconnector HVSW2to HVSW1					1																																																	
Gas turbine																												1																									
Gearbox																																																					
HDE 3 -PWM Converter 1																															1																						
HDE 4 -PWM Converter 2																																1																					
HDE 1 -Propeller Motor 1																																																					
HDE 2 -Propeller Motor 2																																																					
Steering Gear Power Pack 1																																																					
Steering Gear Power Pack 2																																																					
ChilledWaterPlan1PF																																							1														
ChilledWaterPlan2SF																																								1													
ChilledWaterPlan3PA																																									1												
ChilledWaterPlan4SA																																										1											
ChilledWaterPlan1PF - Manifold																																								1		1											1
ChilledWaterPlan2SF - Manifold																																							1		1									1	1		
ChilledWaterPlan3PA - Manifold																																								1		1											
ChilledWaterPlan4SA - Manifold																																							1		1											1	
LP Sea Water Pump 1																																															1						
LP Sea Water Pump 2																																															1						
LP Sea Water Pump 3																																																1					
LP Sea Water Pump 4																																																1					
LP Sea Water Pump 1 - Manifold							1																						1		1																	1	1				
LP Sea Water Pump 2 - Manifold								1																				1		1		1	1	1													1						
GT Lubricating & Fuel Oil Module 1																											1																										
Radar																																																					
Computer room HVAC 1																																																					
Computer room HVAC 2																																																						
Sonar																																																						

Figure 38: DSM Type C

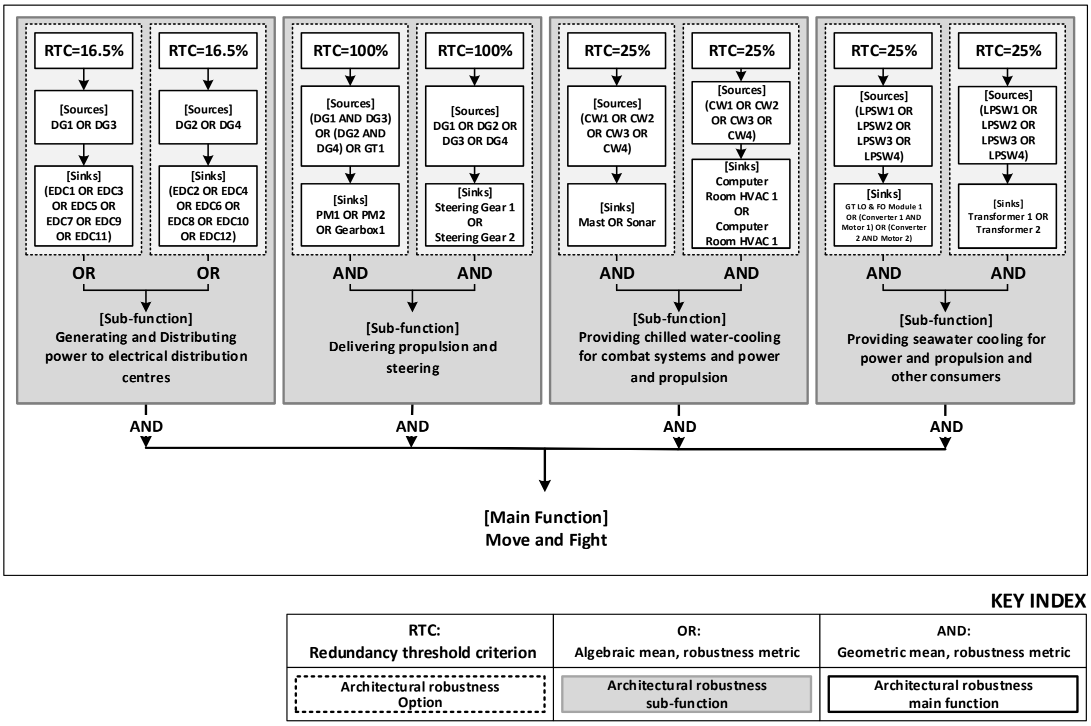


Figure 39: Functional hierarchy, architectural options and RTC for high redundancy Type C

In Figure 39, at the level of the sub-function of providing seawater cooling for power and propulsion and other consumers the architectural option is designed with a pair of "AND". The seawater cooling supports both power and propulsions. They are three seawater supply to propulsion options motor 1 OR motor 2 OR GT LO & FO Module 1 that feeds the GT that drives the gearbox. Also, there is a seawater cooling supply to transfomer1 and transformer 2 that support power.

8.3.6 Evaluation stage

Following the two stages of the evaluation robust modular reconfigurations and DOE results are presented.

8.3.6.1 Robust modular reconfigurations

The application of the robust modular assessment methodology generated the following results for Type C. The robust configuration is shown in Figure 40, proposed for the high redundancy type C system, provides new insights, highlighting the need to separate seawater and chilled water pumps, which are key components of the auxiliary systems, into different modules to ensure the entire system remains functional post a module disruption.

In the case of a non-robust modular configuration, the loss of a module which combines CW pumps or LPSW pumps could result in a total loss of the main function of moving and fighting.

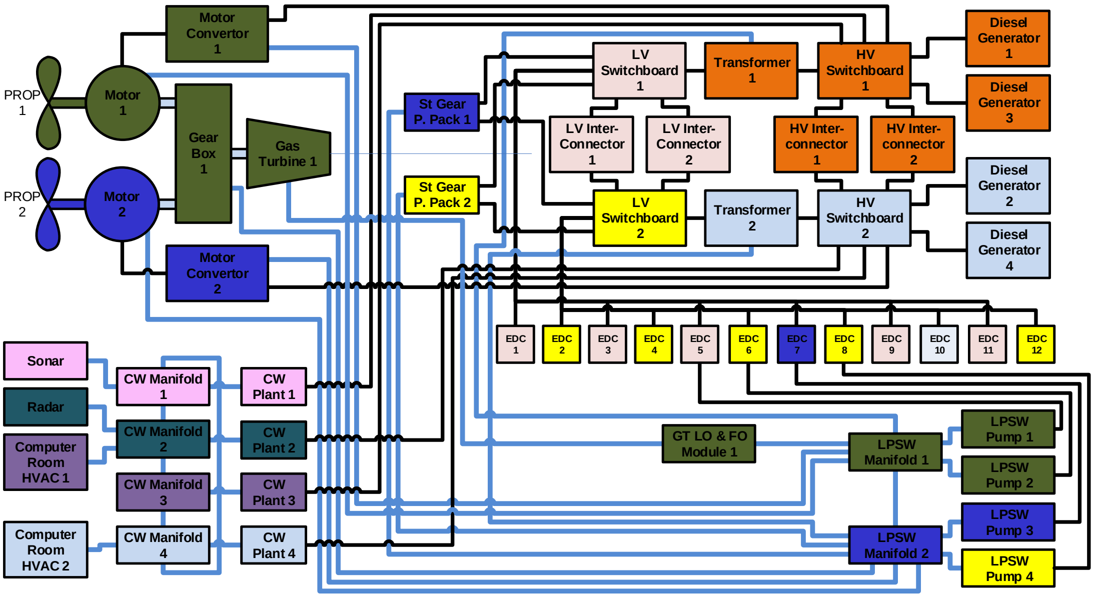


Figure 40: Resulting robust modular configuration for high-level redundancy Type C

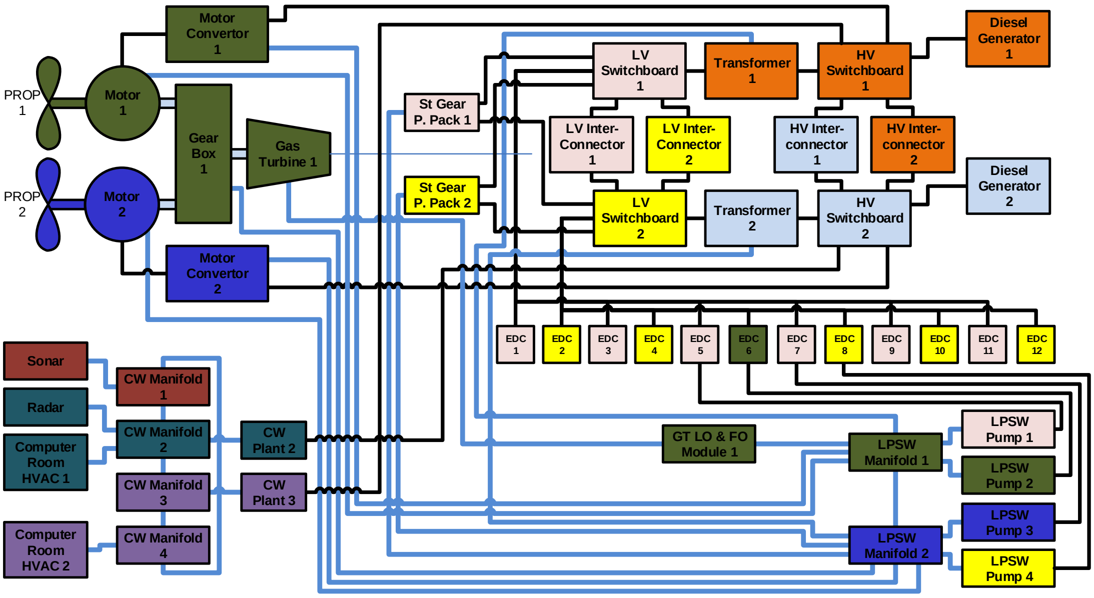


Figure 41: Resulting robust modular configuration for medium-level redundancy Type C

The robust modular configuration generated for Type C has similarities with the configuration of Type A. Type C high redundancy system architecture (Figure 40) has triple redundancy offering alternative options for propulsion, however, given that the way the robust modular configuration is generated considers a single module disruption (and not, for example, two modules disruptions) this lead on a robust module (green colour in Figures 40) that includes together the gas turbine, gear box1, motor converter1, motor 1, GT LO & FO module 1, LPSW Pump 1, LPSW Pump 2, LPSW Manifold 1. In case that the design should consider combinatory two modules disruptions, the green module will be further divided into two smaller modules.

8.3.6.2 DOE Analysis: redundancy and modularity effects on robustness

Table 24 presents the robustness results for a single module disruption for the different levels of the redundancy and modularity.

Table 24: Average robustness results for Type C system architecture options

RUNS	Design variables		Response variable
	Level of redundancy	Level of modularity	Level of robustness
Run 1	Low	Low	0.371
Run 2	Low	Medium	0.446
Run 3	Low	High	0.273
Run 4	Medium	Low	0.818
Run 5	Medium	Medium	0.691
Run 6	Medium	High	0.361
Run 7	High	Low	0.818
Run 8	High	Medium	0.683
Run 9	High	High	0.559

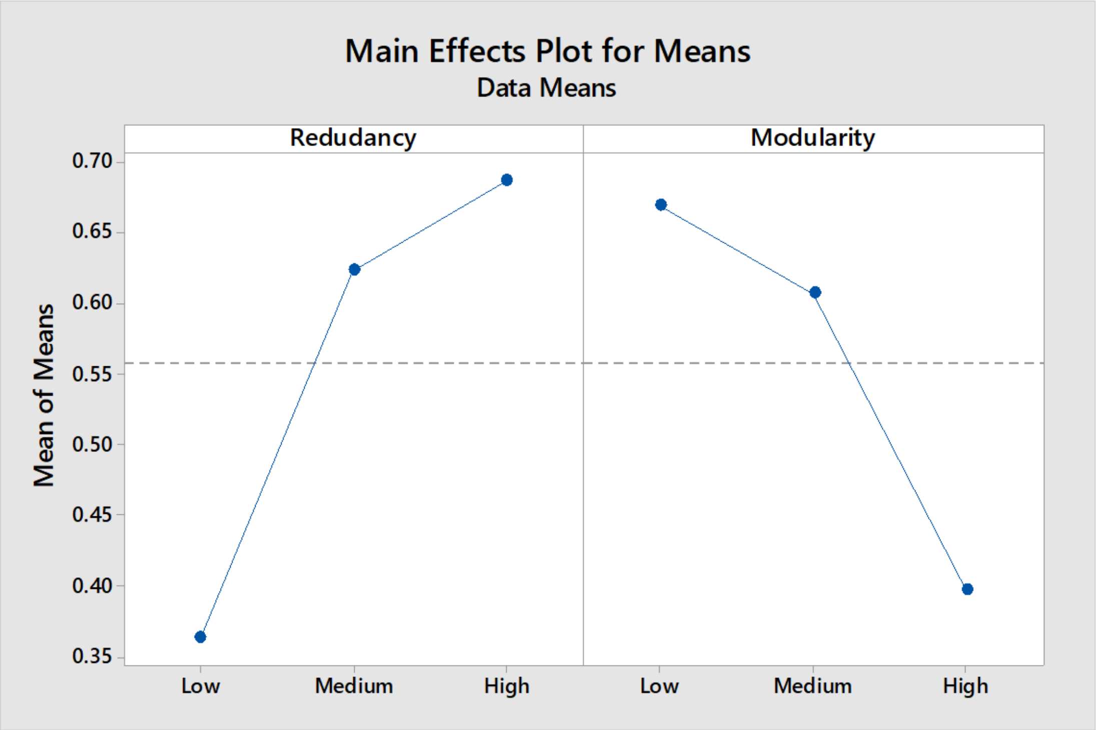


Figure 42: Main effects of redundancy and modularity on robustness Type C

Figure 42 shows that there is a significant improvement when increasing from low to medium redundancy and that it also improves when increasing from medium to high. Results for Type C share similarities with Type A and are different from Type B. This was expected as Type A and Type C are similar type ships (frigates) and therefore have similar performance and capability requirements that have driven their robustness and redundancy.

8.4 Cross case studies comparisons

This section demonstrates how the descriptive implementation of RoMoGA methodology can be used to compare different options of system architecture designs.

This Section presents comparative results for three naval distributed system architectures (Type A, B and C). Type A is an older design, with simpler system architecture, Type B is a modern design with high complexity and redundancy, and Type C is a modern evolution of Type A, however, simpler design than Type B. Thus, was expected based on expert's input and the technical documentation that:

- Type A had the lowest level of robustness and the highest level of modularity.
- Type B has the highest level of robustness and the lowest level of modularity.
- Type C had medium-level robustness and modularity.

The decomposition approach adopted for the three system architectures and the level of granularity is the same, making it possible to compare these systems. This is noted because different levels of decomposition and granularity in system architectures directly affect the architectural analysis and modularity results (Chiriac et al., 2011a, 2011b). The following sections present comparisons under the combined component and module disruptions for the three systems.

8.4.1 Robustness after component disruptions

Figure 43 presents the robustness results calculated for the three systems under disrupting an increasing number (1-10) of components. Results of robustness are presented for existing technical systems (high redundancy) which the expectation of robustness has been established as discussed above.

The calculated robustness decreases as more components are disrupted which is logically expected. The comparative results amongst the three systems show that Type A has the worst robustness results, which is expected as it was the oldest and simplest design. The outcome of the methodology is consistent with prior established knowledge of the robustness of the three systems.

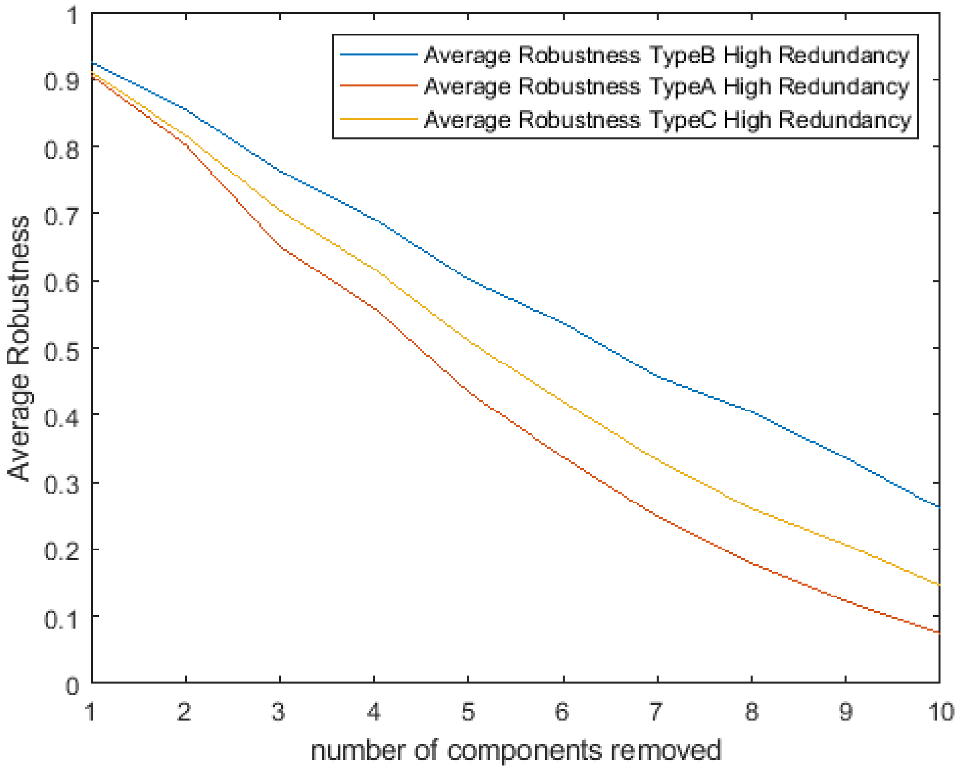
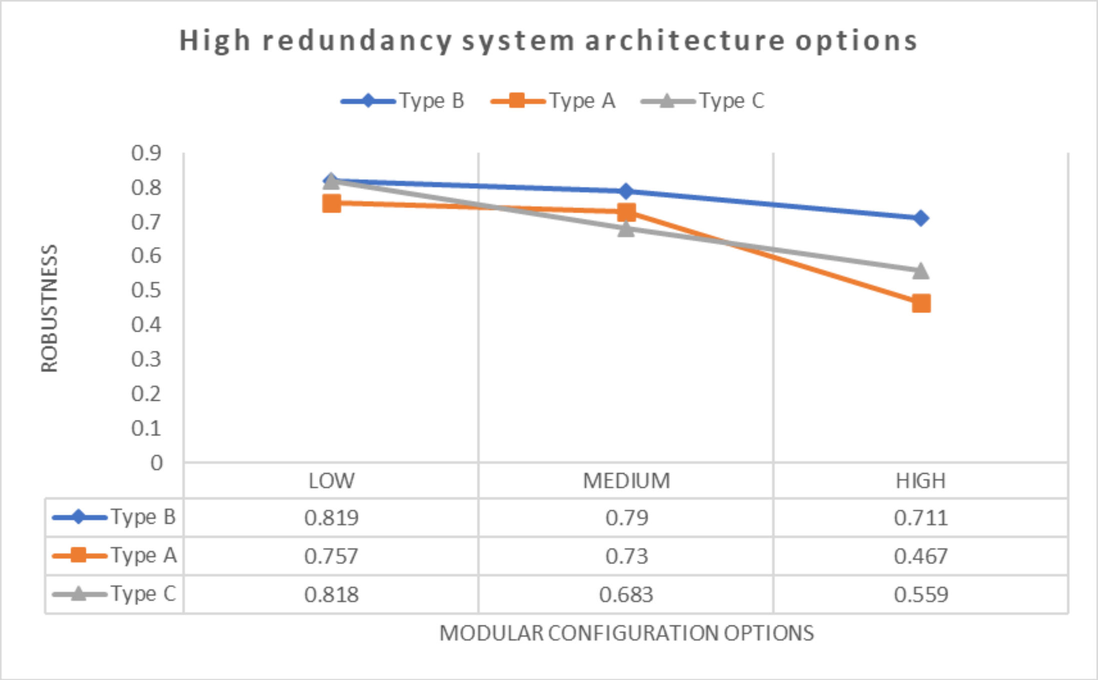


Figure 43: Average robustness under disruption of components (1-10) for Type A, B and C

The above Figure 43 shows the average robustness calculated for all possible combinations of a given number of components. Type B high redundancy is an improved design with electric propulsion which has additional redundancy (3rd HV Switchboard and a 3rd Diesel Generator). The results show that Type C has better average robustness than Type A which can be explained as Type C is more advance design. In terms of the level of redundancy Type B has the highest redundancy, Type C has medium redundancy and Type A has the lowest redundancy. Average robustness under component level disruptions shows consistent improvement as the level of redundancy increases, as there more alternative components, and paths available to achieve the required functionality. This result provided a reference point for comparisons with module-level disruptions.

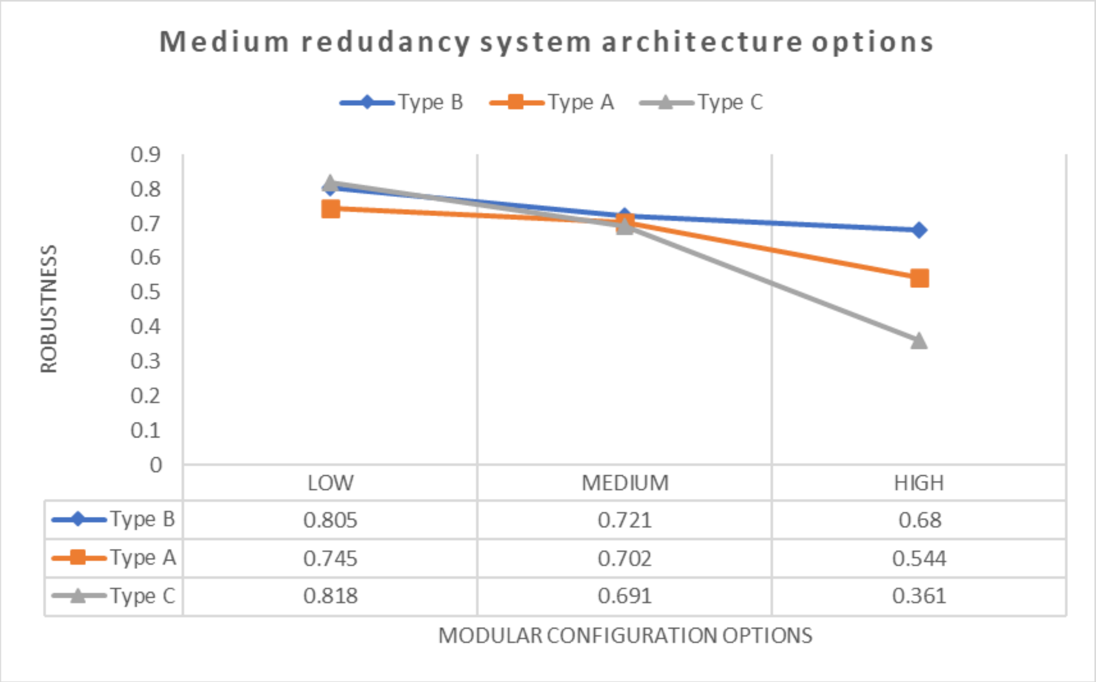
8.4.2 Robustness after module disruptions

Figure 44-46 shows the robustness of Type A, B and C high-medium-low redundancy system architectures under different size module disruptions. Robustness depends on the design of the system architecture (Type A, B, C), the level of redundancy and the option of modular configuration. The analysis shows the effects on the robustness of variations in these parameters. In Figures 44-46, the colour of the blue line indicates type B, the colour of the grey line indicates type C, and the colour of the orange line indicates the technical system type A. The x-axis shows modular configuration options which are low-medium-high. The robustness results presented in Figures 44-46 below are based on robustness calculations for type A (Section.8.2.2) and type B and type C (Appendix II) redundancy and modularity loops.



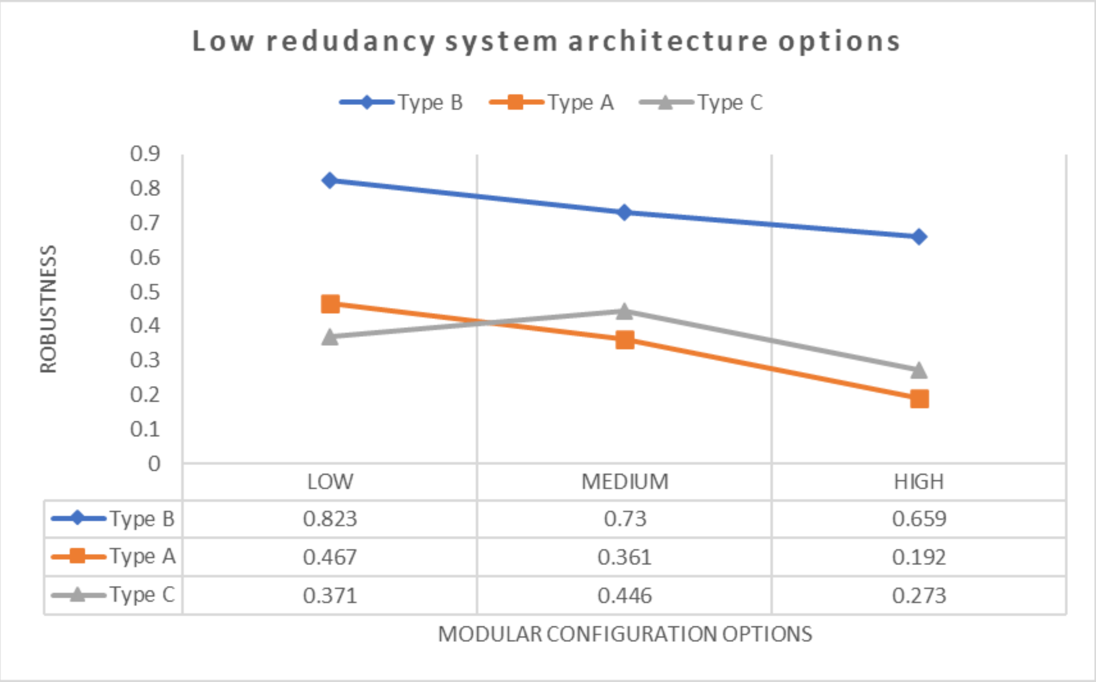
	LOW	MEDIUM	HIGH
Type B	0.819	0.79	0.711
Type A	0.757	0.73	0.467
Type C	0.818	0.683	0.559

Figure 44: Robustness for high redundancy system architecture options



	LOW	MEDIUM	HIGH
Type B	0.805	0.721	0.68
Type A	0.745	0.702	0.544
Type C	0.818	0.691	0.361

Figure 45: Robustness for medium redundancy system architecture options



	LOW	MEDIUM	HIGH
Type B	0.823	0.73	0.659
Type A	0.467	0.361	0.192
Type C	0.371	0.446	0.273

Figure 46: Robustness for low redundancy system architecture options

Figure 44 shows the high redundancy of system architectures. Unlike the average robustness results in Figure 43, Type B and C are equally robust at low modularity Figure 44 & 45. As expected, increasing modularity reduces the robustness of the system, a similar trend as shown in the average robustness previously discussed. Type C medium modularity, however, falls below Type B and A, which is not based on the average robustness results as expected. This differs from the trend of the average robustness post component disruption as the robustness is calculated under disruption for selected components based on the particular modular configuration. Increasing modularity level in Type C leads to a sharp decrease in robustness level, whereas for Type B the decrease in robustness is more stable with an inherently redundant architecture and a higher number of key components.

Type B architecture includes additional hub component (3rd HV Switchboard), enabling a more equal distribution along with the modular configuration (i.e. enables separation of hub component in different modules). However, the redundant component in the Type C architecture cannot be equally distributed in the modules, thus reducing robustness. It is also noted that even under module disruptions (low modularity) Type B has more redundancy than Type C, they have similar robustness. This contrasts with component level disruption, which shows that Type B has always improved robustness in comparison to Type C.

This observation shows the potential influence of modular configuration on robustness, as a module could be disrupted. This also highlights the need for robust modular reconfiguration, and the insight gained can alert architects and decision-makers to be aware of modularity sensitivity on robustness. In the medium redundancy system architectures (Figure 45) shows that after disruption of the medium-level module, the robustness results for the three different systems are similar. This suggests trade-offs between high and medium redundancy system architectures are acceptable.

However, compared to medium and low modularity, the variation of robustness at high modularity is significantly large. In contrast, there is no variation for medium modularity and redundancy, and robustness results are almost the same. This is unlike the average robustness results calculated for component level disruptions. The low-level redundancy system architectures in Figure 46, the robustness results are broadly similar as expected at the high-level of modularity.

The key finding is robustness behaviour varies with different module disruption sizes and increasing redundancy does not consistently improve robustness. This reinforces the need to examine the level of modularity and redundancy effects to robustness to select the right system architecture for different system architecture options.

8.5 Chapter summary

In this Chapter proposed methodology is applied in three existing naval system architectures. The input data for the case studies were collected through technical drawings, specification, manuals, and meetings with experts. The application provided evidence of realism as the robustness results as the combinatory components were found realistic as per the actual results established before the application. The key finding is that robustness behaviour under module disruption is different than the average robustness under combined component disruptions. It was observed that for a medium level of redundancy and modularity similar robustness was calculated for the three types of system designs (Type A, B and C). The results reinforce the need to assess variations in the level of these attributes amongst different system architecture options. The Chapter answers the research objective 6: apply the methodology in technical system architectures. The results of case studies for Type A and B are utilised in the Industrial Evaluation in Chapter 11. Next Chapter 9 presents the explorative application of RoMoGA methodology using the network tool.

Chapter 9: Explorative and prescriptive applications – experiments and redesign

The aim of Chapter 9 is to achieve objective 6 research, which is to apply the methodology in theoretical systems. The RoMoGA explorative application presented in this Chapter involves the use of a novel network tool developed in Chapter 7. The generated system architecture options as recommended in Chapter 5 are assessed in terms of robustness and modularity. As discussed in Chapter 7, preliminary exploration is recommended to allow the user to develop the boundaries for the exploration of the theoretical system architecture that shares topological similarities with the nominal system architecture. The identification of the topological dominant characteristics of the nominal system architecture Type A and B are reported in Appendix III.

The explorative application of RoMoGA is referred to as experiments in the Chapter, as the methodology is used to discover the effects of the topological features on modularity and robustness when a particular parameter of the network tool is manipulated. The findings of the evaluation of the explorative application of RoMoGA guided the redesign of the nominal type A system architecture. The redesign of Type A in Section 9.3 demonstrates a prescriptive application of RoMoGA. The redesign of the Type A system architecture is reviewed by experts who have concluded that the new solution is rational and feasible.

9.1 Explorative application: preliminary experiments

The network tool experiments used Monte Carlo technique to examine the influence of the parameters on modularity and robustness under both random and targeted attacks. A total of 2,500 system architecture option instances were generated for each experiment. While experimenting with one parameter, the other parameters of the generator were randomly varied in the range of the limits sets for each parameter shown in Table 23. The limits of the ranges of the parameters were decided based on discussions with SMEs involved in the technical case study. The limits of the ranges were reasonably consistent with similar engineering systems, with the reference point being the technical systems of the case studies. The same approach was adopted for all the experiments that are following presented.

9.1.1 Main structure pattern experiments

The parameter under investigation (type of pattern of the main structure) is highlighted in bold in Table 23, which presents the experimental set-up. For the following experimental set-up, five experiments were performed for each of the main structure patterns under examination (S1 = bus modular, S2 = path, S3 = hierarchical, S4 = integral, S5 = cycle). This means for each experiment the main structure was constant, whereas the other parameters were randomly selected for each instance within the predefined range and options of Table 25.

Table 25: Experiments: main architectural pattern

Parameters		**Experimental setup**
1	**Number of components (n)**	**40-80**
2	**Type of pattern of the main structure**	**Options**
	S1	*Bus modular*
	S2	*Path*
	S3	*Hierarchical*
	S4	*Integral*
	S5	*Cyclical*
3	**Number of hubs (k)**	**1-10**
4	**Density of hubs (m_x)**	**5-20**
5	**Type of pattern of hubs (Ty)**	**Options**
	Ty1	*Hierarchical*
	Ty2	*Bus modular*
	Ty3	*Integral*
	Ty4	*Path*
6	**Probability of hub's nodes are sources (P_{so})**	**1/3**
7	**Probability of hub's nodes are sinks (P_{sk})**	**1/3**
8	**Redundancy threshold criterion (RTC)**	**0.5**
9	**Level of connectivity amongst hubs (pal)**	**0.5**

For each experiment, the robustness under random and targeted attacks and the modularity of the hybrid pattern were evaluated. Instances of the hybrid patterns populated by the generator for each of the main structure patterns are shown in Figures 47–50.

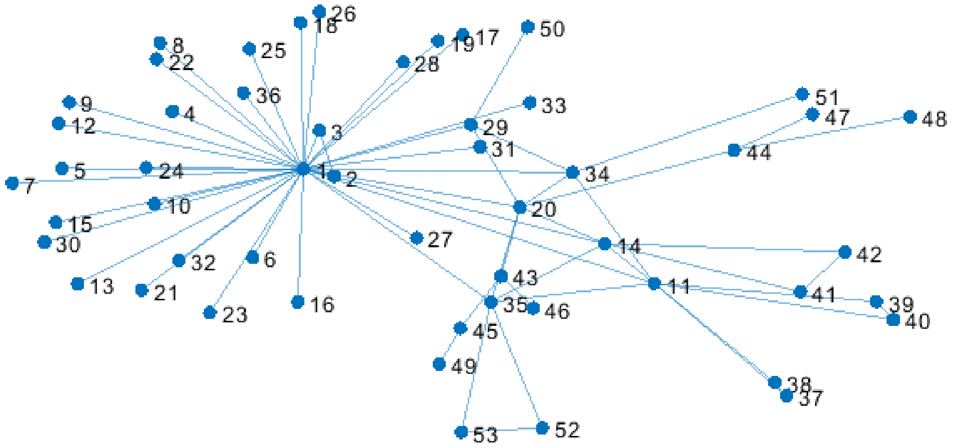


Figure 47: Random instance of a hybrid bus modular main structure network (S1)

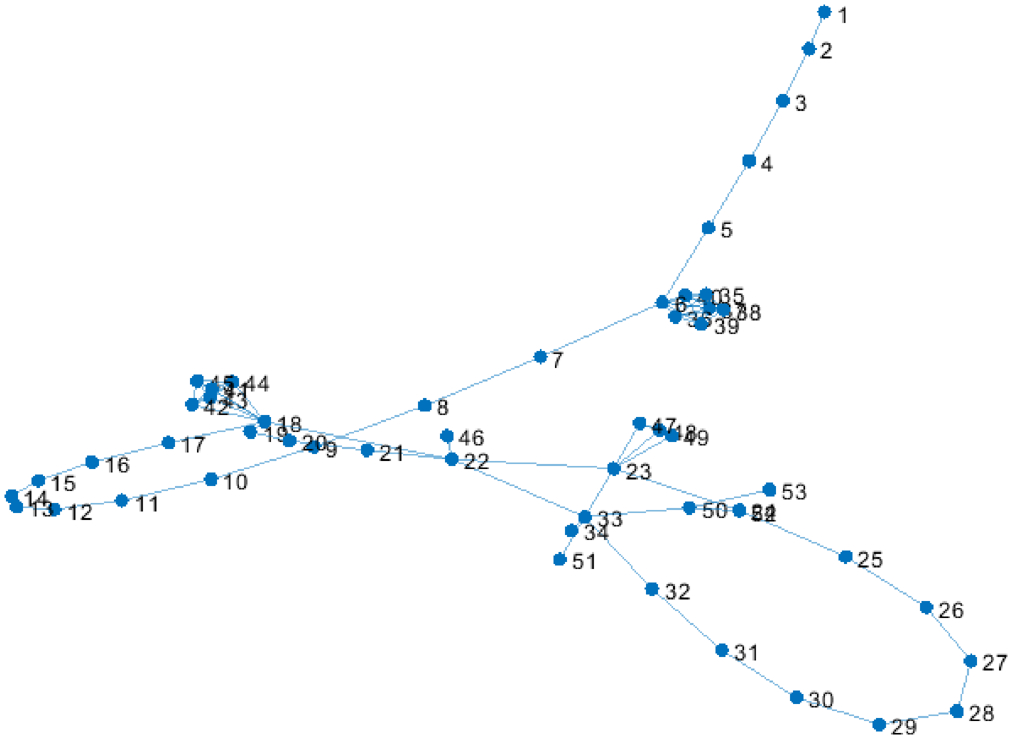


Figure 48: Random instance of a hybrid path main structure network (S2)

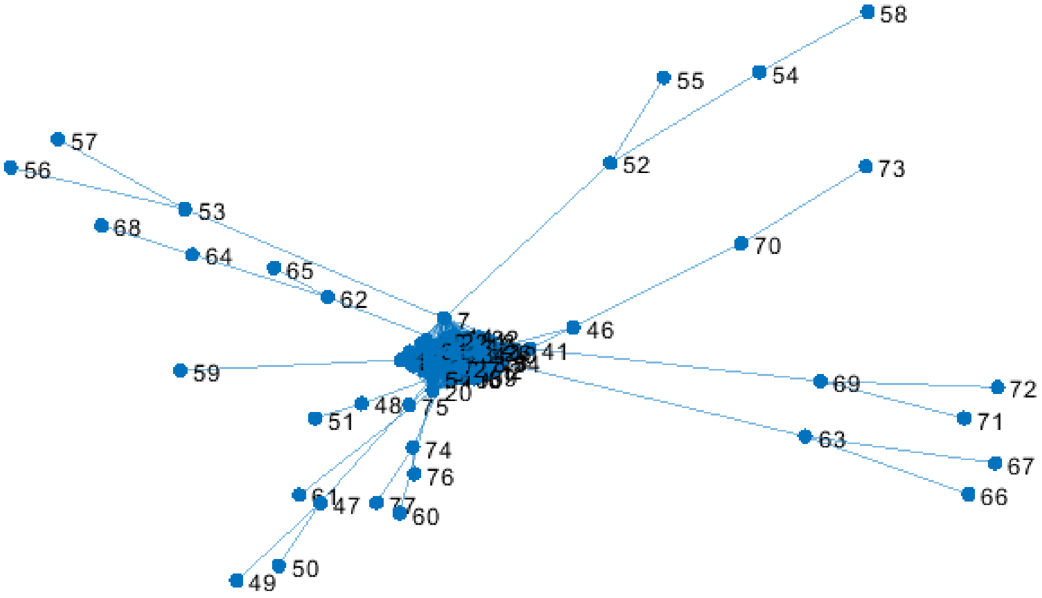


Figure 49: Random instance of a hybrid integral main structure network (S4)

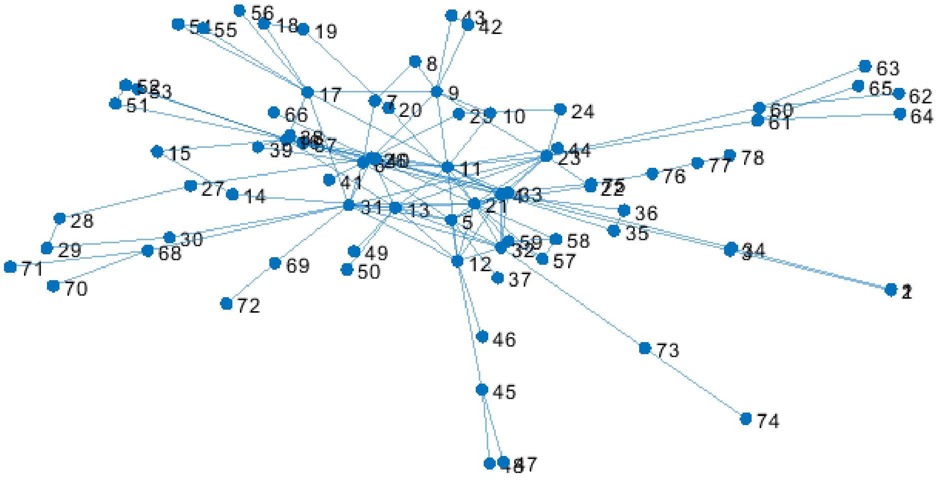


Figure 50: Random instance of a hybrid cyclical main structure network (S5)

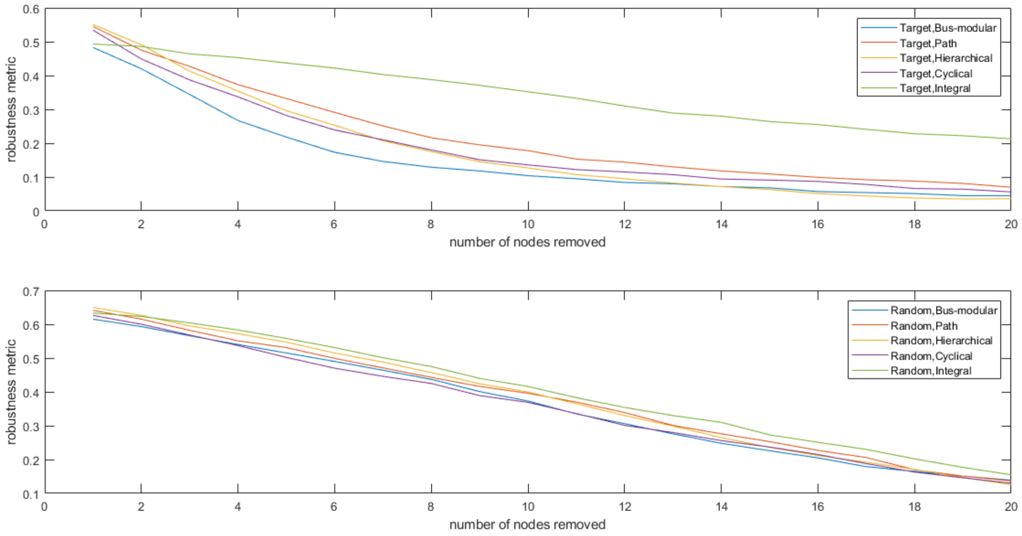


Figure 51: Robustness of the various main structure' pattern

Figure 51 shows a reduction in robustness for an increasing number of random and targeted disruptions for all hybrid patterns. The hybrid 'integral' simulated network with a high number of interconnections, exhibited the highest robustness under both targeted and random attacks. However, this high level of connectivity is not an efficient feasible arrangement – as Chan et al. (2014) explained: "ideally, a fully connected network is the most robust; however, it is not feasible to design such real-world networks due to constraints in physical space, budget, etc." The hybrid bus modular network had the worst performance under targeted disruptions. Disruption of the most central component of the bus modular pattern influences the whole pattern, leading to increased vulnerability, although it has acceptable levels of modularity. The 'cyclical' pattern has a more straightforward, low connection, low-cost arrangement, with potentially the most desirable robustness under targeted disruption.

The robustness of the architecture patterns to random attacks is broadly similar, showing an almost linear decline as the number of nodes removed increased. However, with targeted attacks, all the patterns showed a rapid nonlinear reduction in robustness, with variations between the patterns examined. This implies that the choice of pattern is important, particularly in environments where target disruptions can occur.

The results (Figure 52) show the modularity assessment, whereas the path (PM), hierarchical (HE), and cyclical (CY) hybrid patterns provided the best modularity results, between 0.52 and 0.54, whereas the two technical patterns exhibited a comparable degree of modularity, between 0.63 and 0.48. The integral (IN) pattern is highly interconnected and could not be readily divided into modules, giving a Q value close to zero. This experiment with the generator provided an overview of the effects of the different main structure patterns on robustness and modularity.

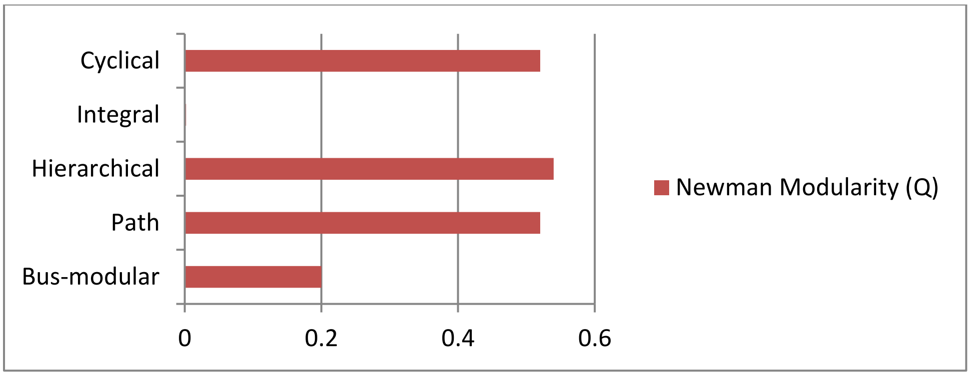


Figure 52: Modularity for the various hybrid main theoretical patterns

In the next stage of the study, it was decided further experiments should focus on the cyclical main structure pattern.

9.1.2 Selecting the cyclical main structure pattern

The cyclical pattern shares similarities with respect to its configuration with the Type A and B technical systems presented in Chapter 8. As outlined in Appendix III, the two technical systems have a cyclical network configuration, showing a loop-style pattern with the main hubs identified, and additional connectivity among the hubs and other components. The existence of closed loops (cycles) was a key observation in the technical systems. The robust behaviour of the cyclical pattern was comparable to that of the technical systems. The robustness of a single node under targeted and random disruption for the cyclical pattern was like the two technical systems. The level of modularity of the cyclical pattern was slightly better than that of the technical systems. This means that the cyclical pattern has the potential to be readily portioned into modules. Although the cyclical pattern is not widely mentioned as a theoretical pattern in the engineering design literature, specific references were found that support the drive for its deeper exploration. For example, Chalfant et al. (2017) discussed a ship propulsion mechanical drive configuration, explaining that there could be two styles, a cyclical and an integral (mesh): “[it] may require closed-loop paths for operability as in lube oil and chilled water systems, or may allow many redundant paths through a complex mesh-restorable network as in an electrical distribution system”. Whitney (2003) argued that patterns (motifs) that generate functions are cyclical (closed loops) in the context of technological networks such as mechanical assemblies and electronic circuit. Moreover, a study by Agarwal et al. (2014), which aimed to identify an appropriate system of system architecture to satisfy multiple objectives, including robustness and modularity, found that a circular graph was the best pattern. These reasons motivate further experimentation with the cyclical pattern.

9.1.3 Experiments: varying the number of nodes

The experimental set of Table 26 focuses on the variation of the number of nodes in the simulated network.

Table 26: Experiments: number of nodes

No.	Parameters	Experimental setup		
1	Number of nodes in the network (n)	n=40-50	n=50-60	n=70-80

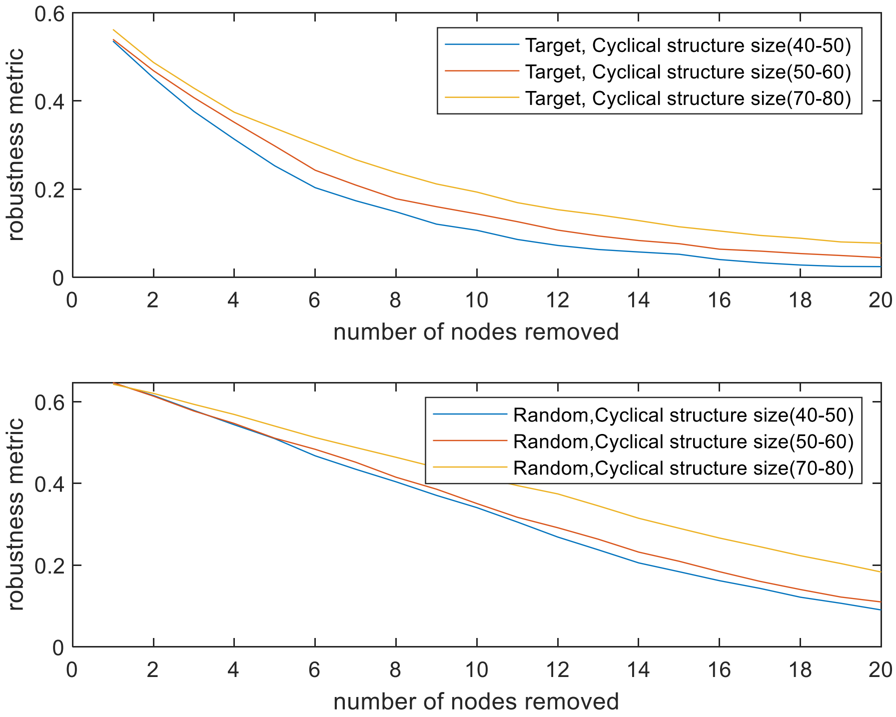


Figure 53: Robustness of hybrid cyclical patterns - different sizes of the main structure

Figure 53 shows a reasonable behaviour with the robustness response, falling as the number of disruptions increased. There was broadly similar behaviour for all sizes of the cyclical pattern for both random and targeted disruptions, with a small reduction in robustness as the size of the main structure was reduced. It can, therefore, be deduced that scaling a cyclical pattern will not crucially influence robustness.

With respect to modularity, Table 27 results indicate that the modularity increases slightly as the size of the cyclical pattern increases.

Table 27: Modularity of hybrid cyclical patterns - different sizes of the main structure

Hybrid cyclical pattern	**Modularity**
N=40:50	0.48
N=50:60	0.51
N=70:80	0.55

The results show a slight increase in the degree of modularity with an increase in pattern size. From the previous results with a broader size range (from 40 to 80), the degree of modularity was 0.52. This suggests that the scaling of the main cyclic structures does not significantly affect the degree of modularity of this specific experimental setup. In summary, the size of the main structure pattern does not significantly affect either its robustness or its modularity.

9.2 Explorative application: redundancy experiments

The following five experimental set-up presents an explorative application of RoMoGA, which involves different system architecture options of varying redundancy to be generated, which are assessed in terms of modularity and robustness under different types of disruption.

9.2.1 Experiments: varying the number of hubs

The experiments presented in Table 28 is dedicated to the variation of the number of hubs.

Table 28: Experiments: number of hubs

No.	**Parameters**	**Experimental setup**		
3	Number of hubs	K=1-5	K=6-14	K=15-20

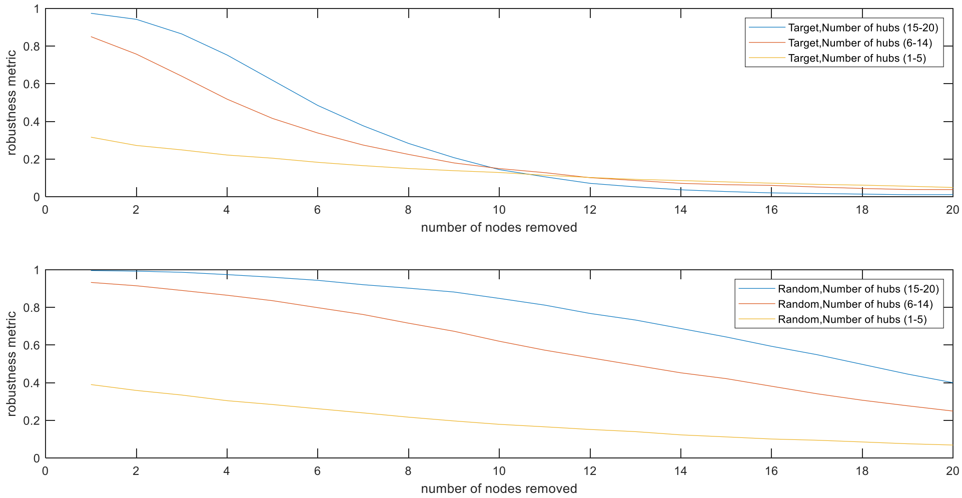


Figure 54: Robustness of hybrid cyclical patterns - different number of hubs

It can be observed that increasing the number of hubs initially protects the network from targeted disruptions but then lead to an increased vulnerability once the scale of the attack passes a certain threshold (in Figure 54 this threshold is 10 nodes) whereas for random attacks no such threshold is apparent. Large scale targeted attacks will tend to remove complete hubs. In the simulated network, the increased frequency of hubs also increases the frequency of removal of interconnections between hubs in target attacks which might explain the phenomenon observed here as the removal of these “arteries” proves catastrophic. But in random attacks, the chances of disrupting an artery are diminished and the added communicability provided by a plethora of hubs proves beneficial of robustness. Such contradictory behaviour highlights the need for balance in the design.

In technical systems, hubs may be designed with spatial separation from each other because the is the requirement to avoid a single disruption affecting more than one hub simultaneously. In the case of an extended disruption of the system that can damage more than one hub, the robustness decreases significantly. The findings suggest that the number of hubs should be carefully considered, given the expected operational environment of the system: many hubs could be detrimental for robustness, and so the spatial separation of hubs in terms of the physical design is fundamental. However, addressing only the spatial location of hubs does not resolve the problem, as an intelligent attacker may be aware of the location of hubs and intentionally disrupt them simultaneously. Technical systems architectures have specific structures where the increased number of hubs suggests increased redundancy and improved robustness. In network science, the number of hubs is considered to be a point of vulnerability in scale-free networks (Albert et al., 2000). However, in the simulated networks with a limited number of hubs, the vulnerabilities introduced by hubs are more considerable only if an extensive targeted attack occurs in the system, rather than an attack of a smaller magnitude. Sosa et al. (2011) argued that “the presence of hubs has a significant effect on system quality (measured by the number of defects in the system)” and concluded that there is an “optimal presence of hubs which minimize the number of defects in the system”. Their work is different from the experiments presented herein, as the focus of their study is on quality. However, our results provide further evidence that there is an optimal number of hubs in the design of technical systems. The redundant hubs are the ones that are readily available to take over the functionality in the case of a targeted attack on a hub, as they are designed to avoid such targeted attack effects. However, determining the correct number of hubs in a technical system is not a straightforward task and requires a careful investigation and consideration of the expected operational environment in which a system is designed to survive.

Table 29: Modularity of hybrid cyclical patterns - different number of hubs

Hybrid cyclical pattern	**Modularity**
Number of hubs=1:4	0.57
Number of hubs=5:7	0.50
Number of hubs=8:10	0.50

Keeping the number of hubs in the simulated cyclical network small gives a higher level of modularity than when there are a larger number of hubs. Since there is a constant level of interconnections between hubs in the generator, their number will increase as the number of hubs increases. The higher interconnectivity reduces the ability to divide the network pattern into modules, and hence reduces the modularity. This result is also complemented by Section 9.2.5 findings showing lower connectivity between hub increases modularity. This implies that increasing the number of hubs while limiting connectivity between the hub, can be positive for modularity.

The experimental findings indicate that a smaller number of hubs positively affect modularity in the development of coherent modules; however, robustness falls under targeted attacks. The important finding is that robustness has a cut-off point, meaning that for disruptions that are extensive in magnitude, an increase in the number of hubs in the architecture provokes a sharper decline in robustness.

These experimental findings were used to inform the redesign of Types A (Section 9.3). The original designs Type A and B show that the level robustness of the two naval designs collapses with more than four or five central components removed. As was previously discussed a more linear degradation of robustness is preferable. The naval technical systems have in total of four main hubs (2x LV and 2x HV switchboards). The robustness results for the simulated networks shown in Figure 54 suggest the possibility of improved robustness by increasing the number of hubs above four given that disruptions that remove more than ten central nodes simultaneously are not expected to occur during operational life. The worst-case scenario in naval design examples only considers single attack events and not multi-attacks. In this case, increasing the number of hubs can positively affect robustness under the target and random disruptions. This finding can provide insights to guide the selection of new design options. Section 9.3 implements these experimental results in the redesign of the Type A technical system. Another example, a future warship power system design may have six to eight discrete power generators modules “DG Source feeding HV hubs” configured, separated, and controlled to deliver specific functions on-board.

9.2.2 Experiments: varying the density of hubs

The following experimental set (Table 30) is focused on the variation of the density of hubs.

Table 30: Experiments: density of hubs

No.	Parameters	Experimental setup		
4	Density of Hubs	mx=5-10	mx=10-15	mx=15-20

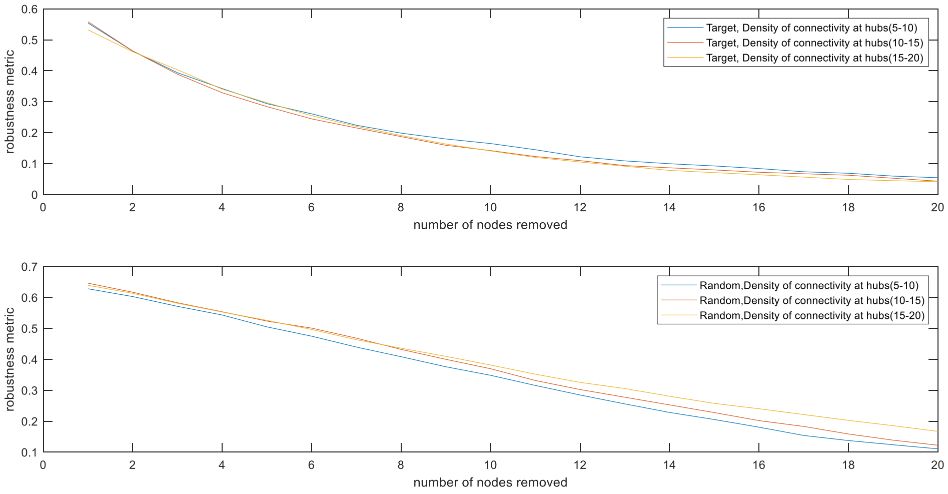


Figure 55: Robustness of hybrid cyclical patterns - density of hubs

The density (i.e. the number of source/sink interconnections, for example, the number of consumers/sinks connected to a hub) of the hubs does not significantly influence the robustness of the patterns under targeted or random disruptions. There is a marginal indication that robustness with higher density hubs is slightly reduced under targeted attacks and slightly increased following random attacks.

Table 31: Modularity of hybrid cyclical patterns - density of hubs

Hybrid cyclical pattern	Modularity
Density of hubs=5:10	0.56
Density of hubs=10:15	0.51
Density of hubs=15:20	0.52

The results of experiments on the density at the hub level suggest that the modularity falls slightly to a limiting value as the density of the hub increases (Table 31). This may be explained by the increased level of interconnections between hubs as the hub size increases. It is worth noting that this may seem as surprised as it could be expected that increasing density at the hub will create more high-density modules.

By increasing the nodes around the hub, however, this also increases the possibilities these nodes are connected to other nodes in another hub, making partitioning into modules more difficult. The way the generator is developed involves a constant level of interconnections between hubs.

The density of hubs does not drastically influence either the robustness or the modularity of the system architecture. The experimental findings indicate that the density of hubs is not a significant factor in influencing the robustness of the architecture. The density of hub relates to the number of sources and sink nodes connected to the hub. This finding could be useful when designing the system architecture because the additional or fewer sink and source components could be added onto the hubs. In practice, the number of source and sink at hub level are considered key elements for designing redundancy in the architecture. However, the experimental findings suggest that a reduction in the number of source and sink could be an aspect of trade-off in a future redesign. This is implemented in the Type A redesign (Section 9.3) that the number of sources is reduced at the hub level and an additional hub is designed in the architecture

9.2.3 Experiments: varying the hub patterns

Table 32 shows an experimental set on the variation of the patterns of the hubs.

Table 32: Experiments: hub patterns

No.	Parameter	Experimental setup			
5	Type of Pattern of Hubs	hierarchical	bus-modular	integral	path

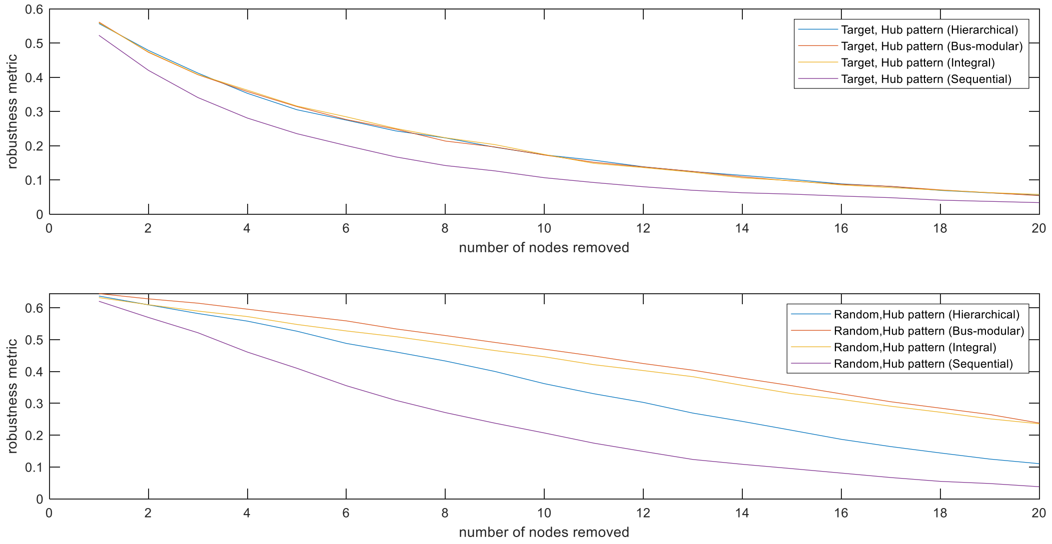


Figure 56: Robustness of hybrid cyclical patterns - different hub patterns styles

The robustness of architectures subjected to targeted disruptions was found to be insensitive to the type of hub pattern style (apart from a path pattern). Path patterns at hub level showed the worst robustness under both random and targeted disruptions, as any loss of node reduced it.

In general, robustness improved as the level of interconnectivity increased; however, the integral hub pattern with the highest level of interconnection did not show the best robustness. The simulated networks of this experiments show that a bus modular hub level pattern has acceptable performance (Figure 56) while at the main structure level a bus modular has the worst performance (Figure 51). Bus modular style hub level patterns are also identified in the two naval designs examples. The power and propulsion expert gave the following example: “In ships’ electrical systems there are separate generators and switchboards. Power to ships’ equipment is taken from the EDCs, which are located throughout the ship. Each EDC has a changeover switch connecting either the normal supply from one switchboard or an alternative supply from the other switchboard. To reduce vulnerability, the switchboard/EDC cables run on opposite sides of the ship. The interconnections between EDCs and equipment are typically a single path, which tends to increase system vulnerability. For some of the critical ship systems, additional changeover switches are located close to the equipment to provide additional normal and alternative supplies from adjacent EDCs”. This arrangement suggests two ‘bus modular’ patterns, with ‘star’ interconnections at a hub level.

The choice of hub pattern has little effect on robustness in the case of targeted disruptions; however, for random disruptions may be more beneficial to design bus modular or integral hub patterns because of increasing reliability and availability of engineering systems.

Table 33: Modularity of hybrid cyclical patterns - Different hub patterns styles

Hybrid cyclical pattern	**Modularity**
hierarchical hubs	0.66
bus modular hubs	0.59
Integral hubs	0.29
Path hubs	0.65

The baseline hybrid cyclical patterns have a modularity of 0.52 (Figure 52) that is based on random allocations of hub patterns (bus modular, path, hierarchical, cycle, and integral). The results in Table 33 show a significant reduction in modularity when it is defined as integral hub patterns (the hub pattern is not randomly selected). This means that the integral pattern at hub level can influence the overall degree of modularity. All the other hub configurations show higher modularity when compared with the baseline, indicating also that the hub pattern is an influential factor in controlling the degree of modularity.

This is a noteworthy finding in line with the argument that the level of granularity affects modularity (Chiriac et al., 2011b). It is useful for architects to recognise that although they may have modularised the system at a low granularity level (high size system elements) if the system creates integral hubs at a high granularity level(small size system elements), this reduces the architecture's overall modularity. This is also discussed in the semi-structured where experts suggested that the network tool could be used to manage the evolution of the system architecture from the initial stage that it is represented at a high level of granularity (more abstractly) to detail design, that more information is available and it is described at a low granularity level (more detail and individual information about subsystems).

9.2.4 Experiments: varying the redundancy threshold criterion

The Experiments of Table 34 concentrated on the variation of the redundancy threshold criterion that relates to the level of redundancy in the system architecture.

Table 34: Experiments: redundancy threshold criterion

No.	**Parameter**	**Experimental setup**		
8	Redundancy threshold criterion	RTC=0.25	RTC=0.5	RTC=1

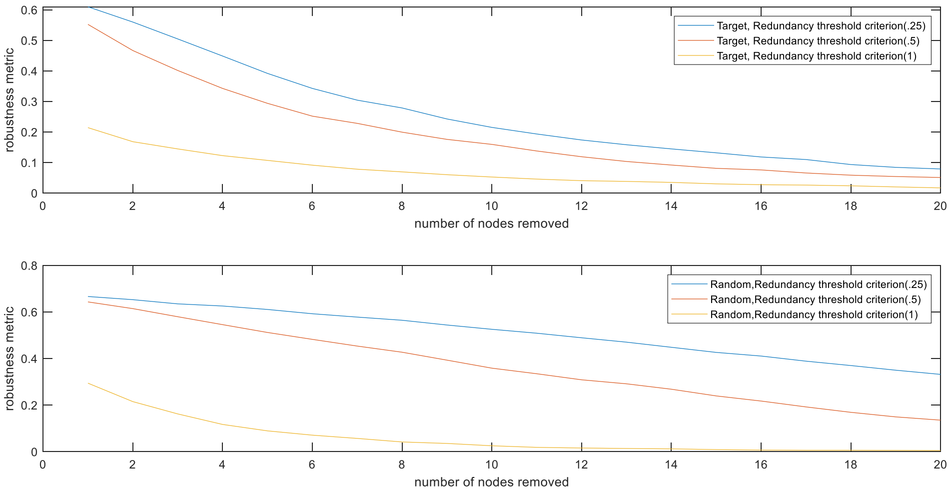


Figure 57: Robustness of hybrid cyclical pattern - redundancy threshold criterion

The threshold redundancy criterion represents the level of redundancy of sources: 0.25 = four times the redundancy, 0.5 = double the redundancy and 1.0 = no redundancy. The results in Figure 47 show that double the redundancy of sources in the architecture improved overall robustness under random and targeted attacks. However, with four times the redundancy, there was only marginal improvement in the robustness. The experiment findings suggest a decreasing benefit in robustness as the level of redundancy is increased. In the naval design example, this finding suggests that an optimum level of redundancy exists, given the associated additional costs, weight, space requirements, and maintenance. The proposed generator can provide an early indication of the relative merits of levels of redundancy at the initial stage of design. This may suggest areas that the initial design efforts might examine new or novel approaches without compromising robustness. For example, this finding guided the redesign of Type A (Section 9.3) that included three instead of four alternative independent redundancy paths between the source and the sink to satisfy power functionality.

9.2.5 Experiments: varying level of connectivity amongst hubs

Table 35 presents the Experiments of varying the level of connectivity between hubs.

Table 35: Experiments: level of connectivity amongst hubs

No.	Parameters	Experimental setup		
9	Level of connectivity amongst hubs	pal=0.25	pal=0.5	pal=1

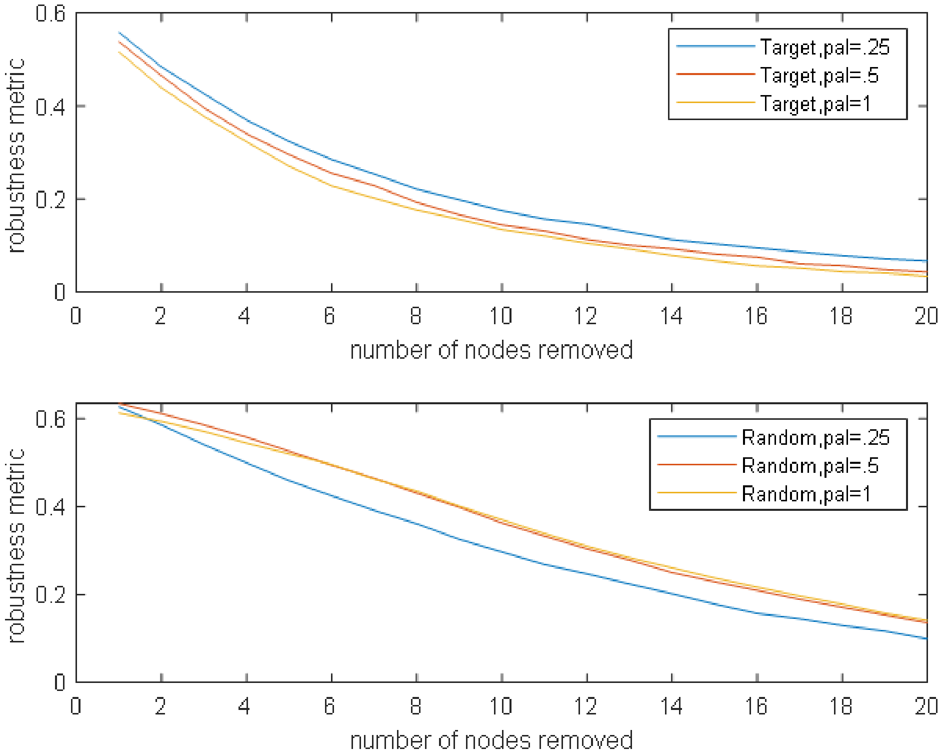


Figure 58: Robustness of hybrid cyclical patterns - different connectivity amongst hubs

The level of connectivity among hubs was controlled in the proposed generator using the ‘pal’ parameter. The pal parameter indicates the level of hub interconnectivity: pal = 0.5 means that 50% of hubs are connected; pal = 1 means that all hubs are interconnected. The results in Figure 49 show that for high connectivity among hubs, robustness is better for random disruptions than targeted disruptions. In targeted disruptions with high levels of interconnectivity, the loss of central nodes reduces the robustness. In random disruptions, the high levels of hub interconnection provide alternative pathways and a higher robustness score.

Simulated patterns with less connectivity among hubs were found to be more robust against targeted attacks than patterns with more connectivity. In contrast, under random attacks patterns with high connectivity perform better. This finding suggests that if there are only a few hubs then low connectivity between them can reduce the damage from targeted attacks. This applies to any relatively small size network whose degrees do not follow a power-law distribution.

Table 36: Modularity of hybrid cyclical patterns – different connectivity amongst hubs

Hybrid cyclical pattern	Modularity
Pal =0. 25	0.55
Pal=0.5	0.53
Pal=1	0.49

With respect to modularity, a lower level of connectivity among hubs improves modularity, whereas high connectivity among hubs reduces the degree of modularity in the simulated networks. The level of connectivity among hubs was found to be an influential parameter, based on the experiments performed. The level of connectivity among hubs has different effects on robustness and modularity. These experimental findings guided the development of the redesign of Type A (Section 9.3): while it is decided to add a hub (HV Switchboard 3) to the improved design, it was decided that this additional hub would not directly feed the other hubs.

9.3 Prescriptive application: Type A redesign

To demonstrate the prescriptive RoMoGA a redesign of the Type A technical system presented in Chapter 8.3 is developed (Figure 59 & 60).

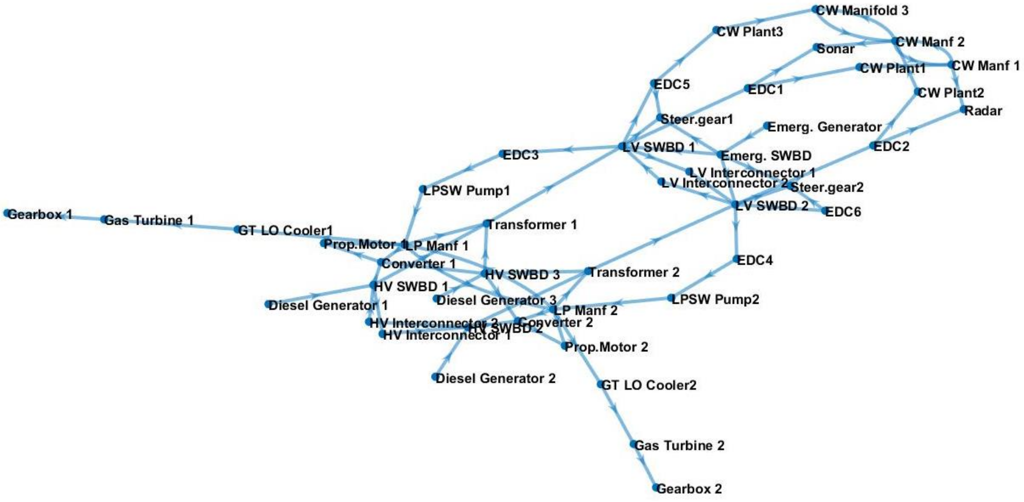


Figure 59: Technical network Type A redesign

The redesign of Type A was driven on the experimental results achieved in Section 9.3 using the network tool. The experimental findings of Section 9.2.1 indicated that the addition of one HV Switchboard would improve the robustness of the architecture, whereas the findings of Section 9.2.2 imply that the reduction in the number of sources (density of hub) may be an acceptable trade-off. Sections 9.2.4 suggested the proposal of three alternative independent paths between source and sink component for the power function. The experimental findings of Section 9.2.5 suggested that the additional hub (HV Switchboard 3) should not be connected directly to the other hubs (HV Switchboards 1 and 2).

Following the experimental findings, the Type A redesign proposal includes one additional HV Switchboard and reduced the total number of DGs from four to three. The redesign of Type A is shown in the following Figure 60.

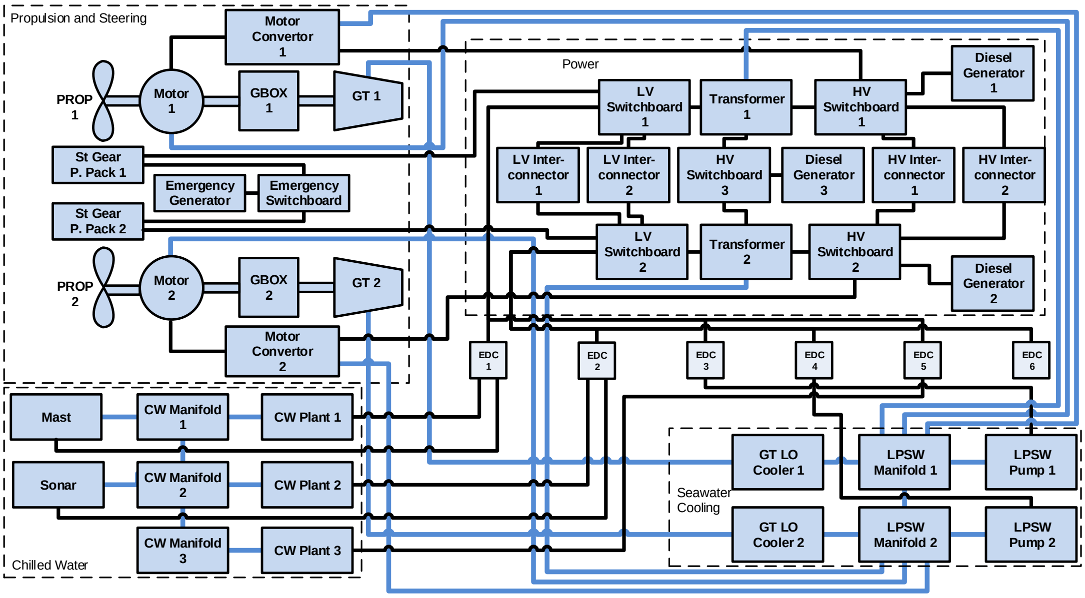


Figure 60: Type A redesign schematic

The redesign of Type A was carried out without changing the total number of system components (one HV SWBD was added and one DG removed) to allow a comparison of networks of the same size. For the Type A system, the cost of a diesel generator was higher than the cost of an HV switchboard that means the Type A redesign is a cost-improved solution. Moreover, the Type A redesign involves hub patterns of single source-DG supply power to HV Switchboard-hub. This is advantageous as electrical synchronisation problems can be avoided by having only one DG supply for each HV Switchboard (not two DG supplies to one HV Switchboard). The simpler source-hub pattern would improve integration time and cost. The redesign of Type A was discussed with experts who verified that the Type A redesign is a rational and feasible solution.

The results (Figure 61) show a graceful degradation of robustness compared to the original Type A under targeted central component attacks and improved robustness under random attacks.

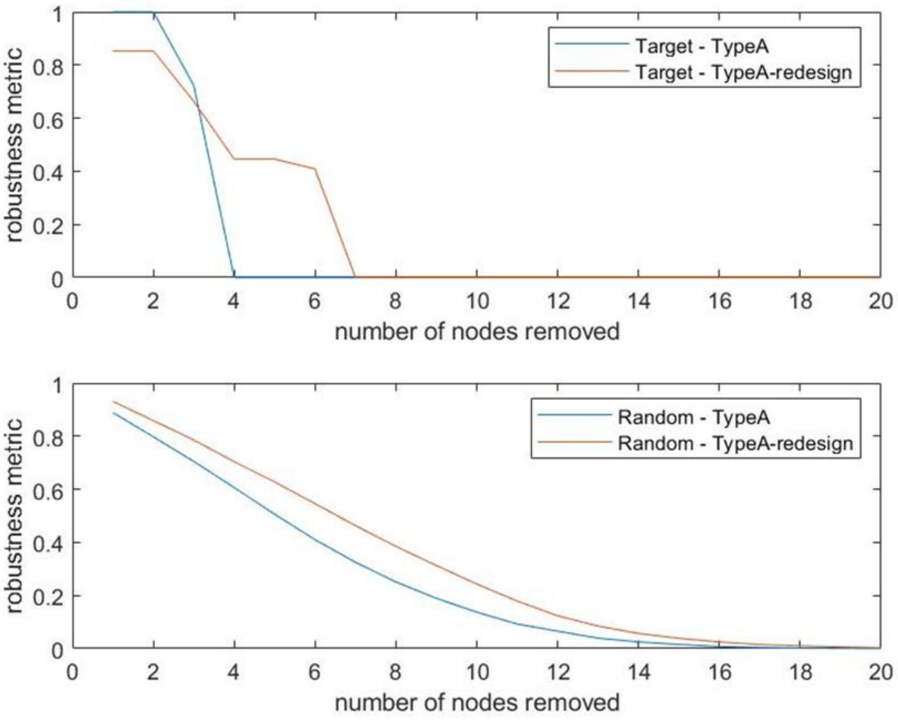


Figure 61: Robustness for Type A Redesign technical system architectures

Moreover, the modularity results (Table 37) show that the redesign Type A has a lower level of modularity due to the increase in the number of connections.

Table 37: Modularity for Type A and Redesign technical system

Technical systems	Modularity
Type A	0.63
Type A-Redesign	0.56

Crawley et al. (2015) recommended to identify architectural decisions early and carefully review them because making the wrong choices will decisively influence the realisation of the system, where no amount of detail design or feature optimisation can solve the key problems. Typically, in the naval context, the number of switchboards or diesel generators are key architectural decisions that are usually taken at the initial design stage and are influenced by previous designs, expert opinions, and other business factors. Such decisions have a significant impact on development time, system costs and operating behaviour. In particular, the robustness of the systems is typically assessed at a later stage in the design process when more detailed information is available. The redesign of Type A demonstrates the value of the new network tool for expanding explorative and analytical approaches during the initial design phase, which can help in architectural decision-making. The Prescriptive RoMoGA is demonstrated in this redesign example, but no further detailed robust modular configuration and trade-off analysis similar to Chapter 8 have been carried out.

9.4 Chapter summary

Chapter 9 presents the application of the explorative RoMoGA by presenting the experiments performed using a network tool. The experiments focus on a cyclical pattern, as identified as a dominant feature of the technical systems studied in Chapter 8 of the Case Studies. The application of the explorative RoMoGA aid in developing new design improvement approaches that have not been previously proposed by the expert engineers following the RoMoGA descriptive application presented in Chapter 8. Experimentation with the different generator parameters in the simulated networks helps to gather findings that inform the redesign of the technical system Type A system presented in Section 9.3. The redesign of Type A was discussed with the experts and verified as a rational and feasible solution. The next Chapter 10 discuss the usefulness, appropriateness, and applicability of RoMoGA methodology with experts, while the Chapter 11 outlines the industrial design practices performed, concluding the evaluation part of the study reported in this thesis.

Chapter 10: Industrial evaluation - interviews

This chapter discusses the semi-structured interviews undertaken to evaluate the usefulness, appropriateness and applicability of the RoMoGA in an industrial context. The evaluation process involved the researcher conducting semi-structured interviews with six experts from different departments and domains within BAE Systems. The RoMoGA methodology was discussed with stakeholders of the development project from different backgrounds, such as production, electrical engineering, forward design management, and supply chain and system engineering. The analysis and discussion of the data collected are presented. Research findings from the interviews provide evaluation evidence for the proposed RoMoGA methodology.

10.1 Establishing evaluation criteria

In order to formulate the right questions for the semi-structured interviews, evaluation criteria needed to be established. Regarding the design evaluation process, Duffy and Donnell (1999) advise that "evaluation, according to some criteria, measures the relation between a result, concept, method, tool, etc. against a datum of some kind such as a requirements specification, known practice, or performance targets". Three evaluation criteria were decided in this study: usefulness, appropriateness and applicability, as discussed below.

The usefulness of the proposed methodology was the main evaluation criterion that was considered during the semi-structured interviews. Useful as per the (Oxford English Dictionary, 2019) is "capable of being put to good use; suitable for use; advantageous, profitable, beneficial" where usefulness is defined as "the state or condition of being useful or serviceable; utility, serviceableness" (Oxford English Dictionary, 2019). Robinson (2008) defined utility as "usefulness as an aid to decision-making within to specified context". Pidd (2010) suggested building models "with some intended use(s) in mind" and argues that "careful consideration on how a model may be used is an important part of any modelling project". Therefore, semi-structured interviews questions aimed to discuss the usefulness of the proposed methodology in the context of the design and development of complex engineering systems.

For the evaluation of the proposed methodology, two additional criteria were used which were appropriateness and applicability. In this respect, there is a need to assess how far the methodology developed is appropriate for its purpose and how effectively it can be implemented in practice. The company designs and develops complex engineering systems and uses a variety of different computational software and methods. Also, there are different domains to consider, such as the electrical aspect or that of the naval architect or expert engineers, as well as outfitting, auxiliaries, and instrumentation. That means the proposed methodology had been evaluated with respect to its fitness for purpose in the context of such a complex multi-disciplinary engineering system. Appropriateness as per the Oxford English Dictionary (2019) is "the state of being appropriated or devoted to some special purpose; special destination". Thus, the evaluation criterion was to assess the extent to which the methodology developed was able to attend its purpose. Additionally, the assessment criterion of applicability was defined in order to evaluate how effectively the proposed methodology could be applied and implemented within the company's complex development process. This was a more specific criterion than simply asking experts to advise how they thought the methodology could be implemented in practice. The applicability criterion relates to the feasibility criterion discussed by Robinson (2008). Feasibility as a critical assessment criterion which answers the following questions: "Can the model be developed and used within the time available? Are the necessary skills, data, hardware and software available?" (Robinson, 2008).

The semi-structured interviews were developed to assess these three evaluation criteria of usefulness, appropriateness and applicability. The researcher divided the four questions into two sections. Questions 1 and 2 related to current practice and questions 3 and 4 related to the proposed methodology's usefulness, appropriateness and applicability. More specifically, question 3 addressed the proposed methodology's usefulness and appropriateness and question 4 inquired about how it could be implemented within the current design process and explored its applicability to designing complex engineering systems. The semi-structured interviews were generic and were the same for each participant in order to allow the comparability. The researcher tailored specific questions to the interviewees' particular knowledge and experience. For example, the researcher did not ask the production engineering manager questions relating to the initial system design stages, since he was not involved in that stage of the design. At the start of the interview, the researcher had an introductory discussion with the interviewees where she presented the development of the methodology and provided an overview of the research. Then, the researcher posed the five specific questions to each interviewee which were documented in a preformat. The researcher engaged in a participant interview style which, as per Saunders et al. (2015), allowed her to manage the flow of the interview and to discuss the predefined questions. The data collected from the semi-structured interviews are presented as follows.

10.2 Industrial context and selection of interviewees

DeJonckheere and Vaughn (2019) argue that “good interviewees are those who are available, willing to be interviewed and have lived experiences and knowledge about the topic of interest”. All the interviewees had in-depth knowledge and experience of the design and development of complex engineering systems in the sector of naval ships. The researcher interviewed senior and principal system architects, a production engineering manager, an engineering manager responsible for the development of future designs, an electrical engineering manager, and a senior electrical engineer. Thus, the six experts in BAE Systems were the most appropriate to be interviewed when evaluating the proposed methodology. It is worth emphasizing that individuals who were selected to be interviewed were not involved in the explorative focus group, in the application of the methodology in the case studies, or the industrial design practice. The reason for this was to allow the research evaluation method of semi-structured interviews to draw in opinions from different experts within the company and so as not to lead to conflict between the data collection and this evaluation process. This consideration was aligned with the recommendations for the triangulation approach followed in the research study as discussed in Section 2.7.

Table 38: List of interviewees

	Interviewees position	Years in the company
1	Production Engineering Manager	12
2	Senior System Architect	5
3	Senior Consultant Engineer - Systems Safety & Software, Engineering	10
4	Engineering Manager Development of Future Designs	38
5	Senior Electrical Engineer	5
6	Senior Supply Chain Manager	20

Table 38 summarizes the positions of the interviewees. The research intended to collect data from participants with different backgrounds and expertise and to gain diverse feedback through their views. A brief overview of the interviews is provided, while the findings of the evaluation are discussed in Section 10.3.

Interview 1 with the Production Engineering Manager

The production engineering manager had been responsible previously for identifying spatial modules in ship distributed engineering. However, this approach for developing modularity took place in the latter stages of the design process when the main system architecture was already fixed. The production team faced modularity challenges due to the system having a high degree of integrality with many interconnections.

Interview 2 with the Senior System Architect

The Senior System Architect provided their insights on the current lack of systematic and analytical tools to assess the system architecture during the initial system design stages of the design process when the high-level architectures were defined. The modularity and robustness of the architectures were not assessed from a system engineering viewpoint, for example, at the level of the block diagrams. It was envisaged that the system architects would use the proposed methodology, and, therefore, the evaluation process needed to gain the senior system architect's reflections on how the methodology could be utilised and incorporated into the system engineering software for the current project.

Interview 3 with the Senior Consultant Engineer - Systems Safety & Software, Engineering

The Senior Consultant Engineer is a member of INCOSE with extensive experience in systems engineering in various fields of application, such as naval vessels, submarines, and the nuclear industry. He is involved in the reliability studies and the system engineering modelling, posing knowledge of the Sparx EA software. This allows discussing the potential applicability of the proposed methodology during the system engineering activities. Appendix IV provides further details on the ability of system engineering modelling to automatically generate the input required for RoMoGA.

The correct timing of RoMoGA's application was discussed during interview 3 that was suggested to be an architectural definition stage after the requirement definition. It was further discussed that this point in product development may vary for different companies and industries, depending on external factors. It was suggested that experiments using the novel network tool with a known system architecture at a high abstraction level could be carried out by different possible structural types at a lower abstraction level, which would allow the examination of larger networks that human minds cannot understand. In this way, the potential evolution of the known system architecture through detail design and design change could be predicted.

Interview 4 with the Engineering Manager Development of Future Designs

The Engineering Manager for the Development of Future Designs advised that the company used a variety of tools that focused on customer preferences and cost trade-offs. He explained that, traditionally, the block diagram schematics were drawn during the early stages of the design. It is noteworthy that he was involved in early conceptual decisions, before the system architects. The redundancy and robustness of the system is a critical driver for designing naval systems, and decisions on the level of redundancy are based mainly on regulations that require either a minimum level of redundancy or relate to the customer's specific requirements. He advised that the customer influences greatly the approach adopted for redundancy.

Interview 5 with the Senior Electrical Engineer

The researcher interviewed the Senior Electrical Engineer because the ship's electrical and propulsion distribution subsystems were of critical importance. However, in many cases, these are electrical analyses conducted in isolation from the other subsystems, even though they are interconnected and supported by other subsystems, such as the chilled and seawater systems. Their input was to advise on the usefulness of the proposed methodology, which was driven mainly by the architecture's connectivity. This was an alternative approach compared to the traditional electrical simulation software and other approaches typically used in this domain.

Interview 6 with the Senior Supply Chain Manager

The senior supply chain manager is responsible for communicating with the various key suppliers of major equipment for the system. Her role includes establishing the contractual requirements, and she is supported by the engineering department. The supply chain team expects to receive information from the engineering departments of the company to be able to successfully assign contracts and detail the relevant clauses and requirements. Not having sufficient information about the system architecture will lead to unfavourable contracts, which will impact the construction of the system, its cost and the time required for its development.

10.3 Evaluation findings

The objective of the analysis of the findings is to evaluate the proposed RoMoGA methodology based on the three pre-defined evaluation criteria. The three evaluation criteria set out the main thematic categories in which the findings were classified. In addition, the discussion on redundancy and modularity demonstrates the need to address existing industrial challenges. Section 10.3.1 discusses findings from the interviews regarding redundancy and robustness, while Section 10.3.2 presents arguments relating to modularity. Sections 10.3.3. to 10.3.5 present findings from the interviews with respect to the usefulness, appropriateness, and applicability of the RoMoGA methodology.

10.3.1 Need

The following two Sections provide opinions of the experts on the current practice related to designing redundancy, robustness and modularity that highlight the industrial challenges reinforcing the need for methodologies applicable during the initial stages of the design. The Senior Consultant Engineer (Systems Safety & Software, Engineering) elaborated during interview 3 regarding the current design practice: “Industry-wide I would say that all three (redundancy, modularity and robustness) are treated separately, with relatively poor communication across the boundary”. He explained that the reasons are driven by contractual aspects, i.e. who is responsible for each element and also by project management techniques that focus on decomposing to measurable tasks and ignore the holistic management of complex systems and also to the engineering education and experience not keeping pace with increasing system complexity. He stated that in practice “requirements are normally poorly defined by systems engineering; the delivery teams develop their design and their level of modularity is driven from manufacturing and cost improvement demands; system safety is only later considered and assessment of the design happens post-development (instead of at the start) and are only really interested in reliability, redundancy, diversity and separation/segregation to ensure that any identified risks to personnel are reduced to acceptable. There is no desire or drive to improve the modular design and build”.

Robustness relates “ilities” such as reliability and survivability are assessed only after sufficient information are available. He explained that “Reliability teams have a similar focus to System Safety but concentrating on success instead of failure (likelihood of successful mission delivery). The Survivability teams seem to come in very late to assess the robustness of the design against damage scenarios”. The need for a different approach that integrates the analysis of these three aspects is highlighted “all three aspects are insufficiently linked early enough in the design process to adequately develop a coherent and complete requirement set to drive design development”. Despite the lack of methodological approach, it is suggested that in current approaches the expertise of the engineering teams who have been through this development process may fill this gap, however, this is only partially successful.

10.3.1.1 Redundancy and robustness

The effects of redundancy were also discussed during the interview, as the RoMoGA methodology provides a design approach to analyse options for redundancy in system architectures during the early stages of the design. During interview 1 with the Production Engineering Manager, he stated that “a design with increased redundancy naturally has increased production work-scope to connect the physically separated equipment with pipes and cables. The increase in work is proportional to the number of redundant equipment and the separation distance. This increases the number of parts, but also the congestion of the space which makes completing the work harder”. Redundancy has a direct correlation with the overall cost of the system and the time of integration. This relationship between redundancy and cost establishes redundancy as a key aspect of design that requires careful study and analysis. That is why the focus of the study includes parameterising redundancy, allowing it to be analysed in the RoMoGA methodology.

Interview 2 with the Senior System Architect revealed that the “system engineering department does not currently model redundancy using SysML in the Sparx EA toolset on the project”. This finding indicates that the RoMoGA methodology can contribute by filing an existing gap in practice, namely, the lack of modelling and analysing redundancy in system engineering models.

Interview 4 with the Engineering Manager Development of Future Designs clarify that “rredundancy is inherent in the development of system block diagrams, the first stage in internal ship system design. The need for redundancy is driven by both component failure potential and susceptibility to system damage”. The regulator (classification society) defines the minimum level of redundancy. Furthermore, for operational reasons, the customer can need a level above the minimum (e.g. military). The findings agree that there are various reasons for redundancy and that it is necessary to investigate the correct level of redundancy.

The Senior Supply Chain Manager explained in interview 6 that, “at present within the contract, the company requires the supplier to identify if they have any obsolescence within their equipment. This is managed through a vendor’s deliverable. Once the vendor’s technical documentation is submitted it is shared with key stakeholders. However, there is no knowledge if the redundant components are to be located in the same or different modules”. This finding suggests that the outputs of the descriptive implementation of the RoMoGA methodology, which is a robust modular configuration, may also support the supply chain team in entering into contracts and setting delivery dates for redundant components belonging to modules that need to be manufactured earlier during the development process or at different locations.

10.3.1.2 Modularity

The effects of modularity were also discussed during the interviews, as the RoMoGA methodology supports the investigation of modularity in the initial stages of the design. Interview 1 with the Production Engineering Manager highlighted that modularity needed to be decided in the initial stages of the design: “introduction of modularity after the concept/system design stages enables the grouping of outfit items in a way that they can be pre-assembled off the ship in a safer and more efficient environment. However, there is limited opportunity to co-locate items which are interconnected by this stage. Hence, the number of interfaces between modules is likely to be higher, and the number of outfit items installed on modules lower than if integrating the modular approach in early design". These findings underscore the need to consider modularity while deciding on the system architecture at the initial stage of the design. Otherwise, it is challenging to implement modularity and realise its benefits. In interview 4 with the Engineering Manager Development of Future Designs, it was stated that modularity is a key driver for producibility. The potential for modularity benefits is established in the early stages of system integration and general arrangement integration and the actual benefits are established only during later design development. Modularity may be required for various reasons for ship design:

1. Operational. The ability to swap complete modules in and out quickly for either role changing or repair by replacement
2. Build cost. If a module can be built, set to work and tested away from the ship and installed when it is complete then there can be substantial cost savings.

He highlighted that the disadvantage of modularity is that may require more space and volume. Therefore, there is a cost-benefit trade-off in choosing the right degree of modularity. The aim of concept studies during the early stages is to explore these trade-offs and decide where the optimum balance is for specific system architecture. These comments indicate that finding the right level of modularity is seen as the appropriate strategy for achieving competitive advantages during the development and life cycle of the system. The Engineering Manager Development of Future Designs advised that RoMoGA methodology answers this need by encouraging the architect to find a trade-off by considering the various potential modular configuration of system architectures.

10.3.2 Usefulness

In this section, the arguments identified during the interviews with respect to the usefulness of the proposed methodology are outlined.

The RoMoGA methodology was suggested by the Senior System Architect that may assist in managing the complexity by allowing a high-level characterisation of the system architecture, through its key topological features (hubs, source, sinks) and offering a robust modular configuration. The Senior System Architect said during interview 2 that "the design complexity is increasing with the progress of technology. Previous methods of design are no longer appropriate for addressing the new challenge". He highlighted that there are many industrial examples of deficient design approaches during the initial stages leading to inappropriate designs. He advised that methodologies that enhance how the design approach can handle complexity are considered useful and are appropriate during the initial stages of the design of complex projects.

The ability of RoMoGA to generate alternative candidate system architecture options and investigate theoretically the role of key features of the architecture (such as hubs, source and sink) was discussed with the Senior Consultant Engineer (Systems Safety & Software, Engineering). He highlighted during interview 3 that “change is the expensive project killer and understanding the impact of any change is critical. The value of the proposed methodology that could be gained from the method is the high level of complexity, where the system is starting to become opaque, especially around interfaces and interactions between systems". The potential value of the methodology is in the assessment of design changes and option trade studies. The initial concept is driven by the contract functional and performance requirements, and based on the engineer’s experience; however, the optimisation of that concept and the increasing disconnect between systems makes it difficult to manage complexity. Attempts are made to manage complexity by decomposition that requires holistic analysis techniques to assess the integrated system and bridge the systemic failures in engineering delivery methods. The Senior Consultant Engineer suggested that the predicted benefits are expected to be realised throughout the project life cycle, as part of the ongoing change impact assessment process. He stated, “the proposed methodology could be useful throughout the lifecycle, but I suspect the potential is in change impact assessment and option trade studies”. This input offered a different viewpoint, as the researcher did not envisage the use of this proposed methodology in relation to change management, as the key focus is to contribute to the initial system architecture design studies. This suggested that comparing system architecture before the change, and architecture after the change, and comparing robustness and modularity, could help evaluate the evolution of system architecture design.

RoMoGA methodology offers preliminary quantitative comparison measures for redundancy, modularity, and robustness to support the evaluation of alternative system architecture design options. During interview 4, the Engineering Manager Development of Future Designs emphasised that “during the concept stage, ship systems are designed against conflicting demands for a set of key factors: performance, redundancy, survivability, product-ability, availability, reliability and maintainability and affordability. However, at the earliest stages when the system is defined at a block and line level, showing how equipment components are connected, the general arrangement (GA) in which the system must fit, is still evolving. As the system designs and GA are brought together and their interactions defined, the key drivers and potential maximum benefit levels achievable for all the above characteristics are set. The proposed approach is appropriate for use at this stage, as the data it requires is available and it considers the interaction between different factors". He concluded that the methodology could be "very useful in providing initial comparison measures for alternative designs, thus providing two key benefits: supporting a cost-benefit trade-off process and reducing the potential for downstream degradation of the factor target benefit levels". The positive feedback from the experts on the initial conceptual studies offers evaluation evidence of the usefulness for the proposed methodology.

RoMoGA was suggested to support better business planning through an early formulation of a robust modular system architecture that can guide the contracts with main sub-suppliers and subcontractors, and through early decisions on the division of work, grouping of design teams and delivery dates for components that are to be located in the same modules. The Senior Supply Chain Manager stated during Interview 6 that knowledge of component allocation to modules and specifically redundant components will allow the business to plan better. She stated that "modules were not taken into consideration for the supply chain element prior to contract award on the current project. This is an area that requires improvement in the next projects, as components located onto modules require to be available earlier than the contract delivery dates. In the current project, contracts were placed based on the lead-times required to design and manufacture the equipment in line with the required in yard dates (RIY)". She pointed out that the required delivery dates to the yard (integrator premises) of supplier equipment are shifted earlier in the schedule when the equipment is added to the modules. This highlights the need to know the early stages of the project of the components that are to be added in a particular module. From a production point of view, the gains from a design that carefully considered the trade-offs between modularity and redundancy are significant, as they have an impact on the time and cost of construction and integration. During Interview 1 the Production Engineering Manager suggested that the "RoMoGA methodology could enable the identification of modules with fewer module-interconnections, which would provide the following benefits to production: fewer pipes and cables to be manufactured, installed and tested; possibly reduced number of penetrations to link equipment together between spaces; less congestion in the design, making the outfit easier to install; increased connections on the modules may increase the amount of off-ship testing that can be completed".

10.3.3 Appropriateness

The three key requirements of interest to the interviewees were the design of redundancy (adding additional and alternative components), robustness (also related to resilience and survival) and modularity (in relation to product-ability due to parallel and early testing). The ability of the methodology to consider these three system attributes together was found to be appropriate for the design and development of complex engineering systems, as described in the interviews. However, the different interviewees had differing views on these three requirements, since they dealt with each requirement individually, for example, robustness and resilience were assessed by a survivability team, whereas modularity was considered primarily by the detail design and production team, and redundancy was designed by concept designers at an early stage.

Senior Consultant Engineer (Systems Safety & Software, Engineering) argued in Interview 3 that the methodology is appropriate as “part of the interactive system architecture implementation post initial requirement derivation. The methodology could then also provide part of the interactive process throughout the lifecycle, assessing change”. The Senior System Architect explained during Interview 2 that the “Sparx EA software is simply a tool for better understanding the design through modelling. The proposed methodology certainly supplements this, through providing clarity in an area previously not considered”. As it was discussed with the experts, the RoMoGA methodology can be incorporated with the Sparx EA software (used in the system engineering design practice as discussed in Appendix IV) to provide analytical capabilities in the system engineering department.

During Interview 5, the Senior Electrical Engineer stated that the “creation of high-level options would be useful for rapid prototyping of conceptual designs, both for the electrical power system and wider whole-ship functionality. It is difficult to achieve the required levels of fidelity to build a suitably detailed model that crosses domains”. The example discussed was that while a chilled water pump is included within the electrical power system modelling, this considers the problem only from an electrical perspective (chilled water pump as a consumer demanding power). However, this type of modelling does not consider how the chilled water pump interacts with another component within the whole system. If all such components from other subsystems were included, the complete model would become computationally very difficult to run. In contrast, the proposed network tool that is incorporated in the explorative implementation of RoMoGA methodology, enables the generation of various system architecture options computationally, rapidly, allowing a wider exploration of the design space. He continues that “in several detailed electrical modelling packages, such as Simpower Systems, equipment that does not belong to the electrical system is not included in the analysis. For example, the cooling fans for the diesel generator that are directly required for the DG to function are not included. The proposed methodology allows a more holistic analysis and permits the expansion of the analysis beyond the electrical system". Therefore, components and systems that lie on the boundaries of the electrical system (demanding of/or supporting the electrical system) that are not modelled within the electrical modelling studies can be modelled within the proposed methodology, and the behaviour of the electrical system in the broader system architecture could be characterised.

The Senior Electrical Engineer also stated that "in current practice, the electrical modelling approaches primarily address system performance assessment rather than early conceptual, exploration and analysis purposes". The chosen design's performance is informed, but the decision-making process to reach that agreed system design was not an analytical, quantitative approach that was well documented. His comments highlighted that the current analytical and computational tools used in the electrical modelling department post the initial stage of the design and are focused on assessing the performance of the selected system.

He continues that "the proposed methodology would allow testing of different system architecture designs. The methodology allows for novel and innovating architectures to be considered that may naturally feel counterintuitive and, by using the methodology, a set of configuration options could be selected which can feed a more detail analysis. Through the methodology, a wider design space could be considered, and the selected potential/preferred solution could be analysed in greater detail". This finding reinforces the appropriateness of the RoMoGA methodology during the initial phase of the design and is useful in generating a wide range of system architecture options at a high-level of abstraction. Moreover, he stated that "with regards to the appropriateness concerning electrical performance; the methodology has some obvious limitations. For example, a limitation of the methodology is the inability to include capacity in cables and more characteristics of individual equipment. However, in the context of early decision-making activities, the methodology is sufficient to generate and assess high-level solutions, without expert electrical knowledge and can help managers and customers to consider possible designs more thoughtfully". His comments also point out the limitation of the proposed RoMoGA methodology to be used as a specific disciplines tool, as it cannot capture specific technical details. For example, RoMoGA methodology does not capture power capacity, cable resistance, therefore, is not appropriate for detailed electrical power modelling simulation. Also, it does not capture spatial and geometrical location or individual components volumes, therefore it cannot substitute the design vulnerability modelling which will be further discussed in Chapter 11.

In short, RoMoGA is not appropriate for detailed technical studies because it is not capable of capturing specific details and technical information. However, it is appropriate to capture the high-level representation of complex systems involving interrelated subsystems and to provide a modelling and analytical method for generating and evaluating different system architecture options.

10.3.4 Applicability

The applicability of the proposed methodology is discussed in relation to the availability of the input data and the current industrial design practices. The Engineering Manager Development of Future Designs stated that "the data requirements for the proposed methodology seem to be compatible with the data available at the early system concept design stage. Key requirements of early-stage design tools are the availability of high-level design data; ability to input and change data quickly and easily, thus allowing multiple alternatives to be analyzed quickly, and provision of readily understood results that can be used to provide meaningful comparisons between alternatives". These findings verify that the input data required by the RoMoGA methodology can be available during the initial design phases.

The Senior Consultant Engineer (Systems Safety & Software, Engineering) suggested that the "methodology requires the development of efficient and effective model-based systems engineering techniques and tools for the dataset to be generated for analysis as the system architecture is created, optimised and changed". Model-based system engineering (MBSE) design activities are now performed in the BAE Systems and many complex engineering companies that can aid "in the understanding of how the underlying data could be mined and analyzed should drive how MBSE is implemented to support these possibilities". The RoMoGA methodology is to be used as a starting point during the concept design and can be combined with the current tools employed in the system engineering departments. The RoMoGA methodology was found to be compatible with the data available at the initial design system stage. The Senior System Architect indicated in Interview 2 that "SysML and the Sparx EA toolset could be used in the same way as the toolset used in the proposed methodology: block definition diagrams can be used to create a visual representation of the interfaces between systems. The straightforwardness of the process is dependent on the skillset of the implementer; however, developing block definition diagrams in EA is relatively simple". The required Sparx EA software to support the data collection is already in use in the BAE Systems, however, RoMoGA methodology cannot be used to realise the value on its own, but must be part of a broader shift to system engineering models and resources to produce the data needed.

The Senior Electrical Engineer indicated that “the methodology could be combined with a simple electrical study. This will allow several electrical configurations to be studied in a steady-state fashion. Different states of operation could become the inputs of the methodology, including the percentage that each state of operation will occur during the system lifecycle. In this way, an overall assessment of the design of system architecture, given a typical operational profile of performance (depending on the expected operational environment and life of the system), will be able to be derived. Each conceptual design will have multiple possible operational states. Computationally, being able to rapidly assess a number of given designs would allow for rapid exploration of the feasible search space associated with a system design”. These comments were noteworthy, as the Senior Electrical Engineer points out that the different operational reconfigurations of a system architecture could be studied through the proposed methodology, and robustness characterisation of the system could be performed considering its operational through life scenario. This finding also informed the recommendation of future work that is outlined in Section 12.4.3.

In summary, the findings from the interviews indicated that the data needed as input for the proposed methodology were available during the initial phase of the design. In addition, the establishment of Model-Based System Engineering (MBSE) approaches in complex engineering firms reinforces the applicability of the methodology. The existence of specific positions for system architects and the incorporation of system engineering software modelling tools enables the development of diagrammatic representations of the systems required as inputs to the methodology and its implementation through system engineering software.

10.3.5 Consolidated overview

A consolidated overview of key findings, as discussed above, on the usefulness, appropriateness and applicability of the proposed methodology, is provided in Table 39 below, which includes key quotes from the interviews.

Table 39: Consolidated overview of key interview quotes

Participant standpoint Interviews	**Usefulness**	**Appropriateness**	**Applicability**
Interview 1 Production	“The methodology can provide the ship designers with a **library of functional modules** suitable for different ship sizes and uses, it can help **to accelerate the initial system design stage** and result in improved spatial arrangements”	“I think the potential use of a library of the functional modules would enable design with **modules and fewer interconnections**”	“The library should be used as **a starting point during concept design**, which is then augmented as the design matures to include other Bill of Material items onto the early defined modules”
Interview 2 System Architect	“A methodology which improves how the design approach may **address complexity** would certainly be **useful and appropriate** to be used during both conceptual stages and throughout the design of complex projects”.	“Sparx EA software is simply a tool for better understanding the design through modelling. The proposed methodology **certainly supplements** this by providing **clarity** in an **area previously not considered**”	“SysML and the Sparx EA toolset could **be used in the same way** as the toolset used in the proposed methodology”
Interview 3 Systems Safety & Software, Engineering	“The value that could be gained from the method is **the next level of complexity** where the system is starting to become opaque especially around interfaces and interactions between systems”	“Part of the **interactive system architecture implementation** post initials requirement derivation ... could then also provide part of the **interactive process throughout the lifecycle assessing change**”	“methodology cannot be implemented on its own to realise the value but **would need to be part of a wider move to systems engineer modelling methods** and tools that generate the data required”
Interview 4 Conceptual Development	“It could, therefore, be very useful in providing **initial comparison measures** for alternative designs”	“The proposed approach is **appropriate** for use at this stage, as the **data it requires is available** and it considers the **interaction between the factors** redundancy, survivability, and product-ability”.	“The data requirements for the proposed process seem to be **compatible** with the **data available** at the early ship and system concept design stage”
Interview 5 Electrical	“The creation of **high-level options** would be useful for rapid prototyping of conceptual designs”	“**Computationally**, being able to **rapidly** assess a number of given designs would allow for **rapid exploration** of the feasible search space associated with a system design”	“The methodology **could be combined** with a simple electrical study. This will allow **a number** of electrical **configurations** to be studied, in a steady-state fashion”
Interview 6 Supply Chain	“Will allow the **business to plan** better”	“enable **components located in modules** to be delivered in integrator installations **earlier**”	“inform **contract placement** in the early stages of the project”

10.4 Chapter summary

This chapter presents the work conducted to evaluate the usefulness, appropriateness, and applicability of the RoMoGA methodology, in an industrial context. Semi-structured interviews were carried out with six key stakeholders from different departments, with different backgrounds and expertise, who are involved in complex engineering development projects. The findings of the semi-structured interviews support the evaluation of the study by providing evidence that the RoMoGA methodology is useful, appropriate, and applicable in supporting the architects in designing robust modular system architectures. This indicates that it is appropriate to address the relevant design challenges and is potentially applicable in the system engineering design practise. The next Chapter 11 includes additional evaluation performed through industrial design practice.

Chapter 11: Industrial evaluation - design practice

This Chapter reports the industrial design practice elaborated with BAE Systems. Duffy and Donnell (1999) defined industrial studies “actual design practice is studied and analysed through a variety of techniques” and suggested them as tools to perform validation and evaluation in design research. The evaluation criteria that are considered as a datum for the industrial design practices are the same as Chapter 10: usefulness, appropriateness, and applicability. The industrial design practice consists of actual design practice currently employ in the BAE Systems: ship vulnerability design that is elaborated using SURVIVE software. The Chapter is structured as following: SURVIVE industrial design practice is presented, that provides an introduction on ship vulnerability; the framework and outcomes of the evaluation. Chapter 11 concludes with a discussion on findings gained through the industrial design practice.

11.1 Ship vulnerability background

The industrial design practice is motivated by the vulnerability design aspect required for the survivability of naval designs. Survivability as per the NATO (2002) definition is the “capability of a system to continue to carry out its designed missions (s) in a combat threat environment” (Schofield, 2018). A combat threat environment entails extended magnitude disruptions by external attackers. This combat threat operational environment is not a typical environment that nonmilitary related systems are designed to survive. Specifically, survivability consists of three aspects: susceptibility, vulnerability and recoverability. A system with higher robustness is expected to have fewer design vulnerabilities. Vulnerability entails the probability of the loss of capability because of disruption. Robustness as defined in this research as the ability of the instantiated system architecture to support sufficient functional continuity after a disruption and is a precondition of survivability.

Float, move and fight are the main functional requirements of naval systems. Capability is defined as “the combination of equipment, trained personnel and support that gives the armed forces the capacity to achieve the task they are given” (Ministry of Defense, 2015). Capability is the fundamental statement describing the need of the customer, which relates to being capable of achieving a task. Capability encompasses the need for components and connections of the architecture of the system (a ship is a system of systems) to remain sufficiently connected to support a level of functional requirements post disruption. In this industrial design practice, vulnerability and robustness are accepted as two inverses in meaning attributes of the system. A system highly robust is less vulnerable. These are examined and compared through two approaches: the proposed RoMoGA methodology and the SURVIVE software. Within BAE Systems the SURVIVE software is used as the MoD approved tool to evaluate the vulnerability of the naval ship designs. SURVIVE modelling and assessment toolset software is developed to help naval architects to design ships that can survive a dynamically evolving environments (Qinetiq, 2020). SURVIVE can support important design decisions by comparing the vulnerability of a vessel against the lethality of a variety of threats (Qinetiq, 2020).

Vulnerability reduction strategy defined by (Ministry of Defense, 2015a) mandates the adoption of features in the design to prevent a catastrophic loss. In addition, the design should minimise single points of failures and if they exist, they must include additional protection. Another recommendation relates to the concentration of critical components and separation of alternative (redundant) sources. The concentration of components shares analogies with the concept of modularity, which concerns the grouping of components into coherent (highly interconnected) modules. The design philosophy for separation or redundancy of the design of key systems to reduce vulnerabilities in design is key aspects in the naval engineering systems.

The necessity to carefully examining the connectivity amongst subsystems and the influence in ship vulnerabilities is identified in the naval ships related literature. Schofield (2018) explained that “with increasing warship complexity the interplay between seemingly unrelated systems becomes important”. van Oers (2011) identified that there is a gap on assessing the system architectures and the configuration of the connections during the initial system design stage and proposed an automated method to generate alternative routing in ship systems, given the positions of the components was known, that was suggested that could be a part of a vulnerability assessment process. Jansen et al. (2018) explained that they are different measures to reduce the vulnerability in designing naval ship such as “damage containment by zoning, redundancy and separation of systems, and protection with blast-resistance materials”. In essence, the distributed systems architecture impacts the overall ship robustness and vulnerability. In the next section, the evaluation framework formulated in order to evaluate the RoMoGA methodology in this industrial design practice is explained.

11.2 Industrial design practice analysis framework

The industrial design practice entails the use of SURVIVE software that is the main means of assessing the vulnerabilities present in naval ship design. SURVIVE software is the UK Ministry of Defence approved tool to perform a ship design vulnerability study on naval ships. An unclassified version of the SURVIVE software was provided by BAE Systems for the study reported in the thesis. A naval architect helped with developing the SURVIVE models. A significant amount of time and effort is required to learn to use the software, as there is a learning curve process to become more familiar with its use. The hull models developed in SURVIVE were the same for the two technical system case studies (Type A and B) and were built based on information provided by BAE Systems. The hull models were the same in order to allow the comparison of findings to be focused on the distributed systems, neglecting the influence of the hull.

The evaluation approach was elaborated through a comparative analysis. The same inputs, that are the system architectures (Type A and B) presented in this thesis case studies were used as inputs in both the proposed methodology and the SURVIVE software modelling. The evaluation approach relates to using the outcomes of the RoMoGA methodology in a practical design situation in an industrial context. The evaluation approach used a single reference design (ship hull and general arrangement). Additionally, the definition of function considered by the proposed methodology and SURVIVE software had to be examined. SURVIVE software suggests that achievement of function post disruption is successful when “a route can be traced from the top to the bottom without passing through damaged equipment the function survives” (Schofield, 2018). The definition of a function that SURVIVE software uses is similar with the definition that the study embraces as discussed in Chapter 6.2.2.1 which states that a function can be satisfied if a set of source and sink components of the flow network maintain sufficient connectivity after a disruption. Moreover, the differences amongst the proposed RoMoGA methodology and the SURVIVE software; had to be taken into consideration, to allow valid comparisons. Thus, a number of assumptions had to be accepted. The following paragraphs discussed the assumptions; scenarios of disruption and the evaluation approach adopted to perform the industrial design practice using SURVIVE software.

11.2.1 Assumptions

The main assumption that was adopted during the SURVIVE industrial design practice was that all case studies system architectures were encapsulated in the same ship hull, which for simplification purposes were set to be a sufficiently large to enclose all the system architectures. This is an assumption as in reality; Type A and B systems are enclosed in different ship hulls and have a different general arrangement. The reason for this assumption was to neglect the effects that the ship hull has on the design vulnerabilities aspects, to allow the comparisons with the robustness generated by the RoMoGA methodology. The software includes features that enhance the structure of the ship with specific naval features, such as reinforcing critical ship bulkheads or including shock mountings in systems. In the industrial design practice, such features as blast resistance structure and shock mounting are applied equally to all the system architectures under investigation. The focus of the industrial design practice is the distributed systems architecture of the power generation, propulsion equipment and supporting systems that are critical for the reduction of the vulnerability in the ship design. The resulted in robust modular system architectures presented in the Case Studies Chapter 8 were used as input to update the original designs and were assessed through the SURVIVE software and compared against previous reference designs provided by BAE Systems.

A second assumption used was that the probabilistic aspects of the disruptions were normalised by applying the same disruption scenarios for all the various case studies. The disruptions generated by SURVIVE were documented, and the same disruptions were used as input in the different instances (original and updated) of designs. Moreover, to evaluate the robust modular configurations, the same general arrangement drawings of the reference design were used to all instance of designs examined. The reference design is a generic warship design containing no proprietary or legacy platform information, allowing it use for educational and publishable research. The reference design general arrangement is presented in following Figure 62 and the ship hull is displayed in Figure 63.

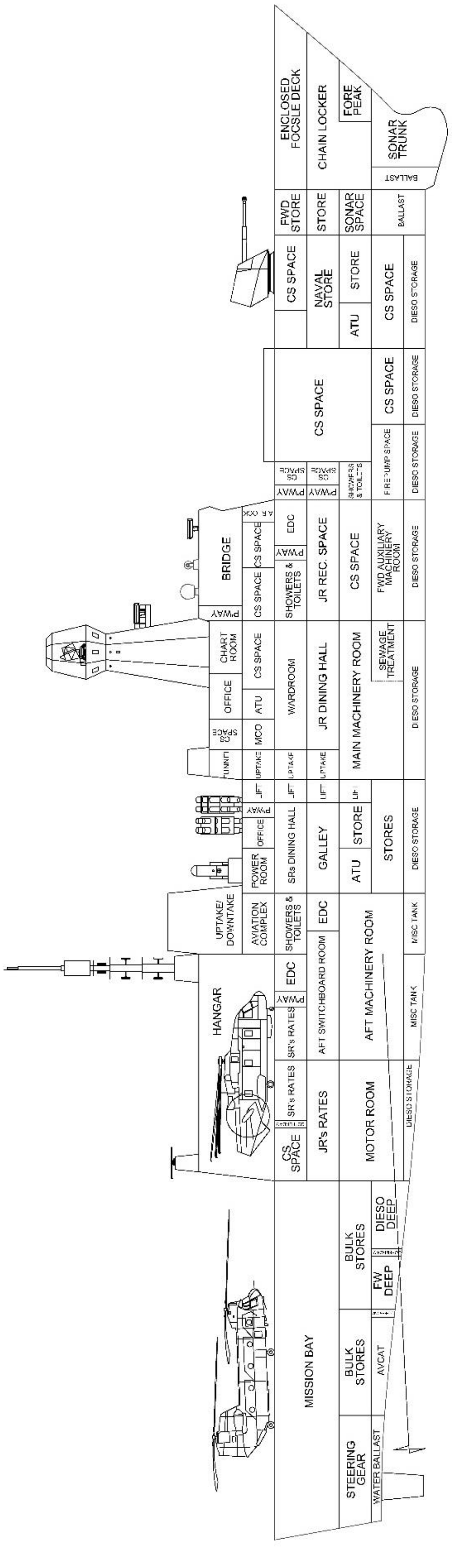


Figure 62: Section view general arrangement of the reference design

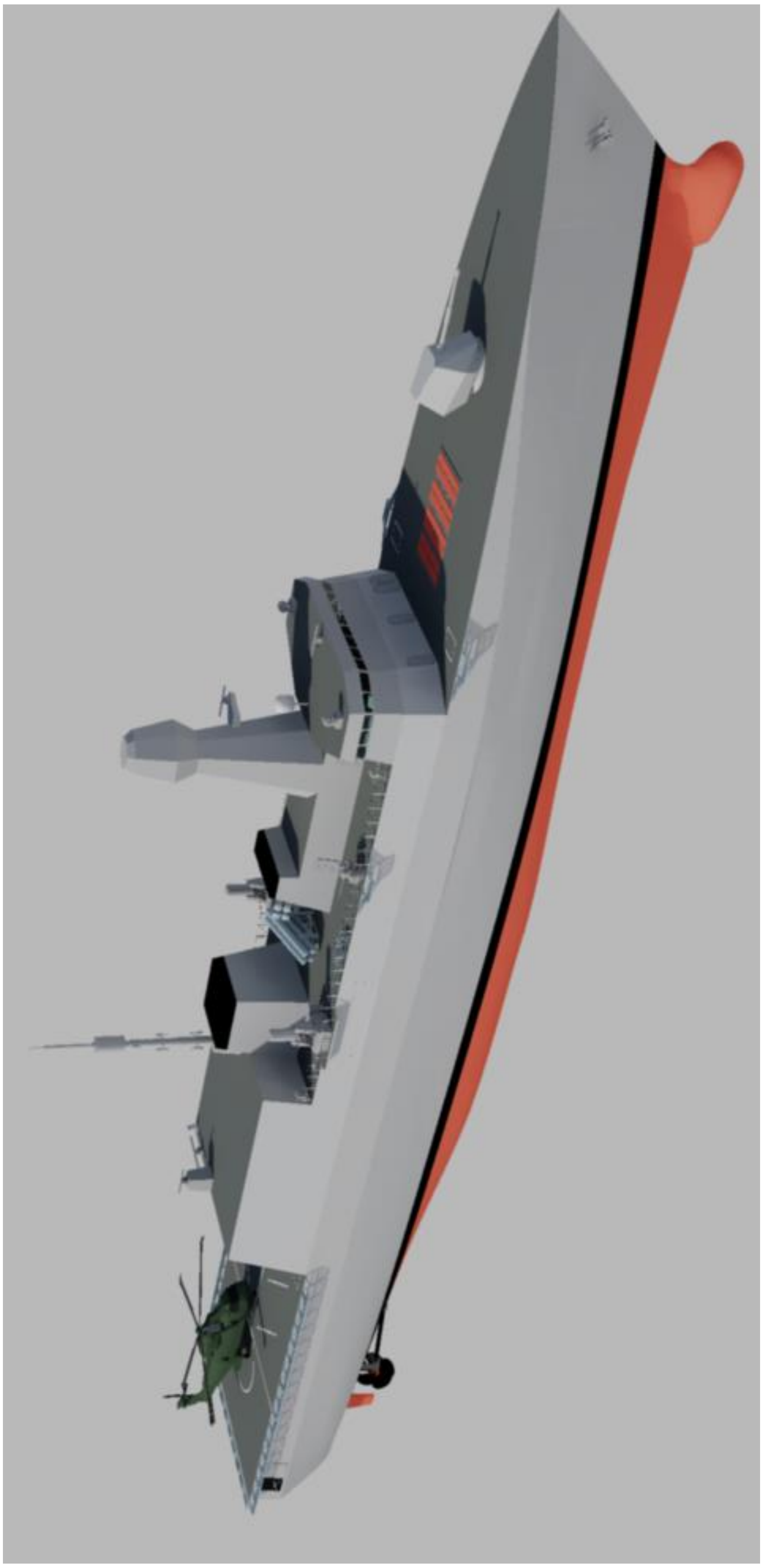
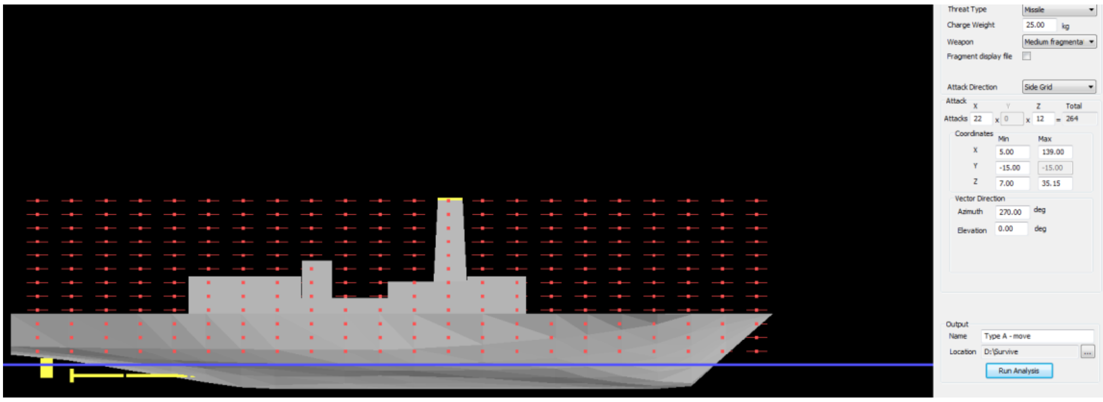


Figure 63: Ship hull of the reference design

These assumptions were accepted to standardise the designs (in respect to ship hull; ship general arrangement and disruption scenarios), enhancing the soundness of the comparisons performed in the industrial design practice.

11.2.2 Scenario of disruption

All the instances of designs examined were simulated against the same generic disruption. This was a generic anti-ship missile, penetrating the ship side and exploding on the centre line of the ship. This disruption was applied as a standard transverse grid 1 meter by 1 meter for the full length and height of the ship as shown in Figure 64.


Threat Type: Missile
Charge Weight: 25.00 kg
Weapon: Medium fragmenta
Fragment display file ☐
Attack Direction: Side Grid

Attack

	X	Y	Z	Total
Attacks	22 x	0 x	12 =	264

Coordinates

	Min	Max
X	5.00	139.00
Y	-15.00	-15.00
Z	7.00	35.15

Vector Direction
Azimuth: 270.00 deg
Elevation: 0.00 deg

Output
Name: Type A - move
Location: D:\Survive

Figure 64: Standard threat type 25 kg missile, medium fragmentation (Snapshot from SURVIVE)

The magnitude of the disruption that was simulated in SURVIVE was set up to be approximate of the loss of a single module (meaning that an approximate number of components from 4-7 were disrupted in a single disruption). The magnitude of the disruption is calibrated to be below that of an overmatching threat that would cause catastrophic damage to the ship as a whole. In this way, a more detail evaluation and comparison analysis could be performed. The following paragraph discusses the evaluation approaches employed.

11.2.3 Evaluation approach

The evaluation stage was intending to ascertain reality in the robust modular configuration outcomes proposed by the RoMoGA methodology for the technical systems Type A and B examined in the Case Studies Chapter 8. A reference design was used to define ship hull and general arrangement and the Type A, and B were modelled and simulated for a specific disruption scenario in SURVIVE software. An updated version of the reference design informed by the robust modular configuration proposed by RoMoGA methodology as presented in the results of the Case Studies (Chapter 8), was subsequently modelled and simulated in SURVIVE software. This updated version of the design was developed with the help of the Naval Architect and Research and Technology Engineering Manager.

SURVIVE results generated for the original reference designs were compared against SURVIVE results generated for the updated version of the reference design that was developed guided by the robust modular configuration. Each instance of design under disruption was assessed for its effect on the same functional requirements as discussed in Chapter 8 (move and fight). For each disruption, the SURVIVE calculates the damage to the equipment and systems and finds the corresponding loss of functionality. Table 40 consolidates the evaluation approach adopted.

Table 40: Evaluation approach

Comparison of SURVIVE software simulation results amongst against the same disruption	
Basic design **(Case studies Type A and B)**	Reference basic design a typical arrangement which was not optimised
Updated robust modular design **(Case studies Type A and B)**	Updates on the basic design made based on the robust modular configuration outcome presented in the Case Studies (Chapter 8)

The results gathered from the modelling and disruption simulation performed in SURVIVE software for the Type A and Type B system architectures are following presented.

11.2.4 RoMoGA-design changes

The design changes made by the naval architect were based on the reference design, which defines the location of the equipment in the hull of the vessel. The location of the equipment was described in respect to the longitudinal, transverse and vertical dimensions based on the centre of gravity of each component (LCG, TCG, VCG) and the identification number of the compartment and the area of the vessel. Based on this information, the naval architect modelled the reference design in the SURVIVE software and therefore the SURVIVE reference model was created.

RoMoGA's results proposed grouping components into modules at the network level, after which the naval architect made design changes to the reference design to convert the network grouping of components into a physical design solution. Design changes were made to allow the components shown in the network to be grouped within the module, into physical modules in the SURVIVE model. The design changes were the result of the relocation of the components to physically co-locate the ship components into a compartment in the same area of the ship, which could thus become an independent sub-assembly during production. This would allow the module to be independently produced and tested by a sub-supplier and could be integrated with other modules during the construction of the ship. The relocation of equipment to different locations of the ship led to the creation of an updated design.

First, the naval architect determines whether the proposed components to be grouped by RoMoGA were already been placed in the same physical area in the original design. In this case, a pre-existing potential module in the reference design has been identified which would allow the architect to further concentrate the components in a single compartment. The architect then identifies the non-adjacent components and moves the components to a different location in the SURVIVE software model to enable network-level grouping as per RoMoGA. It is noted that a network-level module could be translated into a variety of physical implementations. In this way, the architect attempted to move the components to different physical locations, and the technically reasonable design changes recorded were the physical implementation of the network module. The design changes were therefore made by the naval architect who, following the suggestions of the RoMoGA colour network diagrams, attempted to redesign the position of the equipment in the hull of the vessel to enable the robust module physical creation.

If the component could not be moved to form a network module due to different spatial constraints, a discussion was held on how the robust colour module generated by RoMoGA could be achieved by incorporating a different technical solution. In this way, the newly proposed design changes to the SURVIVE software model have been implemented and a design vulnerability assessment simulation has been carried out. In the following paragraphs, the designs of Type A and Type B are reported.

11.3 Type A and B updated designs

The updated instances of designs developed are termed robust modular designs in this section. These updated instances of design relate to specific changes made from the reference design guided by the robust modular configuration of Case Studies Chapter 8. Both the reference (basic) design and the updated (robust modular) designs were simulated for each design comparison in the SURVIVE software. The following photo Figure 65 shows how the robust modular schematics were used to inform the updated design and aid the approach to group the equipment in the SURVIVE software. It is noted that the colour schematic provided information to the naval architect who had the responsibility to implement spatial the schematic arrangement in the ship hull. The modelling and implementation of the robust modular system architectures were performed by the naval architect and the research and technology engineering manager. Expert knowledge was helped to model and analysis the systems in SURVIVE software. The below photo Figure 65 was taken during the industrial design practice, illustrate how the outcomes of the RoMoGA methodology were used in practice to support the architect and perform the domain-specific vulnerability analysis using SURVIVE software.

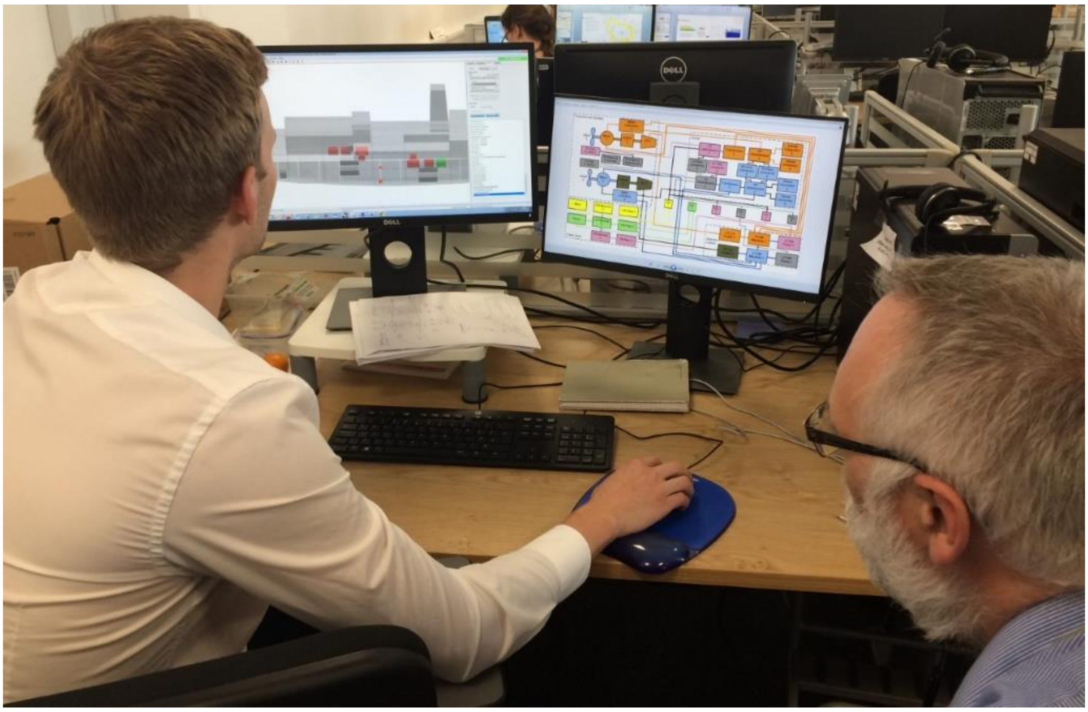


Figure 65: Naval Architect and Research & Technology Manager utilising RoMoGA outcomes

Three design comparisons were elaborated:

1) Type A high redundancy: basic versus robust modular updated design
2) Type A high versus medium redundancy
3) Type B high redundancy: basic versus robust modular updated design

The modelling in SURVIVE software of the components and subsystems is a time-consuming process, and due to the time limitations of the study the Type C system was not achieved to be modelled and analysed. As discussed before, the same scenario of disruptions was simulated in the basic design and the updated robust modular designs, and specific instances of disruptions were compared which are following presented.

11.3.1 Type A high redundancy comparisons basic versus updated design

Figure 66 ship encapsulates the Type A schematic previous presented in Figure 14 (Section 8.2.1).

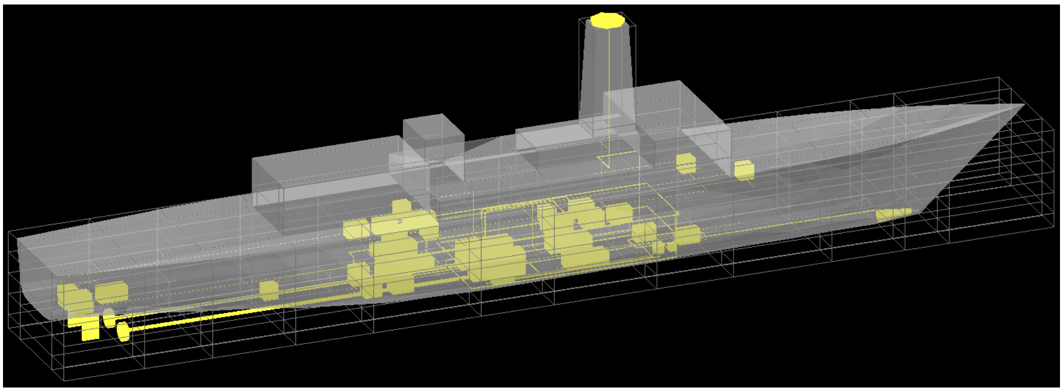


Figure 66: Type A - High redundancy basic design

The outcomes of the descriptive RoMoGA methodology Chapter 8 were used to devise an updated version of the basic design. The following list of changes was decided were incorporated, as guided by the results of the RoMoGA methodology to develop the update robust modular version of the design. The design changes incorporated to devise the updated robust modular design instance were the following:

1. Thruster forward (replacing the SG Pack 2)
2. Aft switchboard room (High voltage 1) moved aft 1 compartment
3. Forward switchboard room (High voltage 2) moved forward one compartment.
4. Low voltage switchboard moved further away from the main switchboard room.
5. Electrical distribution centres (EDC 1) moved further forward.

Figure 67 encapsulates the Type A robust modular configuration of Figure 28 (Section 8.2.3).

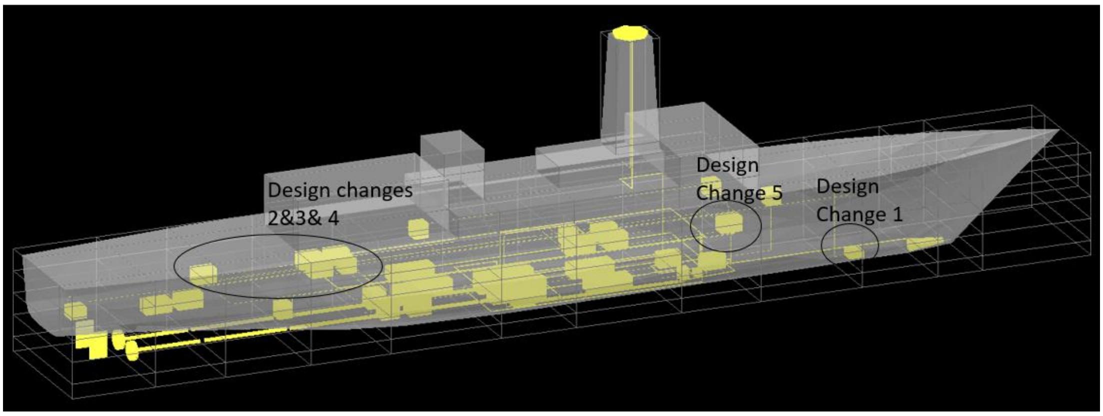


Figure 67: Type A- High redundancy updated robust modular design version

The same simulation scenario (Standard threat type 25 kg missile, medium fragmentation) is performed at the basic design (Figure 66) and the updated robust modular design (Figure 67).

For the Disruption ID 1, the list of equipment damaged is reported and the sub-function affected is shown in Table 41.

Table 41: Type A - Results for high redundancy basic design (Disruption ID: 1)

Type A High – Basic design ***Function: Move and Fight – Disruption ID: 1***	
Equipment Damaged	***Sub-Function Disrupted***
1. Emergency switchboard 2. Emergency generator 3. Steering gear pack 1 4. Steering gear pack 2 5. Starboard rudder 6. Steering gear pack 1 – LV 1 cabling 7. Steering gear pack 2 – LV 2 cabling 8. Emergency switchboard – steering gear pack 1 cabling 9. Emergency switchboard – steering gear pack 2 cabling	• Propulsion and Steering

Figure 68 illustrated the equipment damaged. The ID of the red boxes corresponds to the ID of the equipment damaged in Table 41.

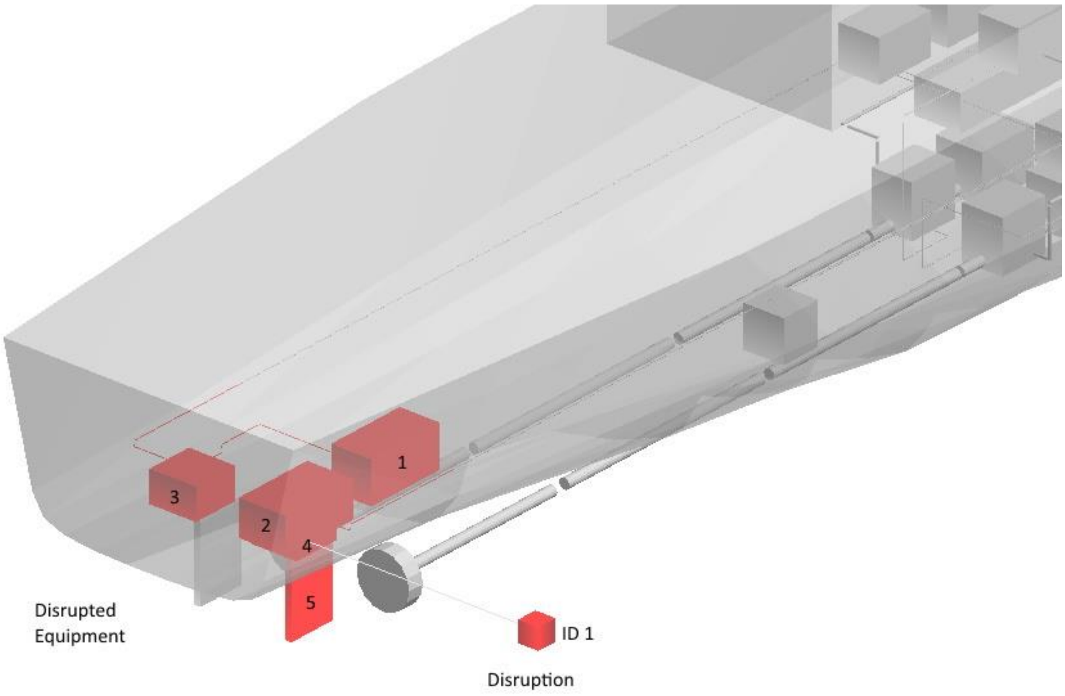


Figure 68: Type A - Basic design results (Disruption ID: 1)

Figure 68 shows that a single disruption could lead on the loss of both steering gears which essentially is a total loss of the function of the system under examination (steering sub-function equals that the main function move and the fight are not able to be achieved). This is a critical hit, and this design area requires attention with respect to its robustness.

The following Table 42 displays the results for a high redundancy robust modular design.

Table 42: Type A - Results for high redundancy robust modular design (Disruption ID: 1)

Type A High – Robust modular design ***Function: Move and Fight – Disruption ID: 1***	
Equipment Damaged	***Sub-Function Disrupted***
1. Steering gear pack 1 2. Starboard rudder 3. Steering gear pack 1 – LV 1 cabling 4. Emergency switchboard – steering gear pack 1 cabling	• Propulsion and Steering ***Non-critical*** • Propulsion and Steering system maintain adequate functionality using Thruster (SG Pack 2) forward.

Figure 69 illustrated the equipment damaged. The ID of the red boxes corresponds to the ID of the equipment damaged in Table 42.

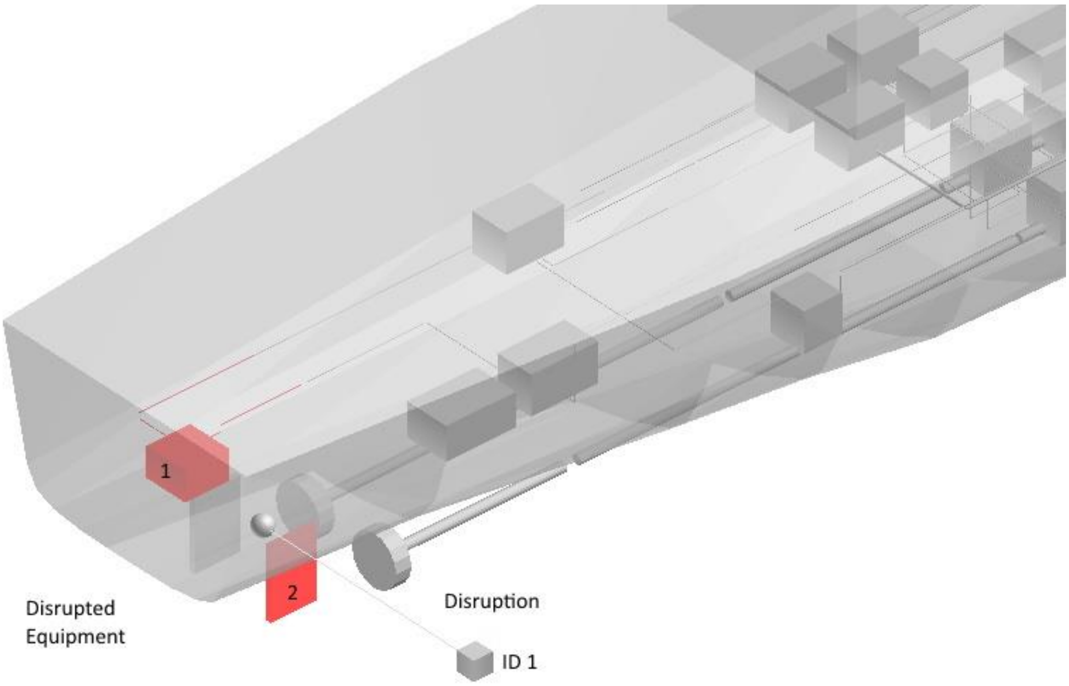


Figure 69: Type A - Updated robust modular design results (Disruption ID: 1)

Figure 69 shows the updated robust modular design, which was conceived through the discussions with Naval Architect and Research and Technology Engineering Manager based on the outcomes of RoMoGA methodology that identified that grouping the two steering gears in the same module could lead on a total loss of the system. In practice, there is a requirement that required two steering gear power packs and two rudders for reliability reasons. Also, in practice the physical area that these two steering gear power packs and rudders occupied is minimised by their concentration, reducing the probability of a critical hit. For the following updated robust modular design, an independent directional thruster was located as far forward as possible on the ship to maintain the manoeuvring capability in the event of a critical hit on the steering gear compartment. In the past the design proposal of a directional thruster located forward of the ship was discussed, however, domain-specific experts had raised concerns because of the technical limitation of such a proposal (a naval ship requires to be highly silence and a forward thruster may increase noise). Thus, the domain-specific experts (acoustic) prevail, the system architecture recommendation that a forward independent directional thrust was a better solution, and that was worthy to be more in-depth investigated. The use of the RoMoGA methodology could have provided objective and analytical results to the debate between the domain experts and the system architects, reinforcing the need for a more innovative technological solution. The proposed RoMoGA methodology is envisaged to be used in such type of early debates by offering traceable and objective agreements to the debates that are predominantly influenced by domain experts. By providing to the architect analytical capabilities to investigate alternative solutions in a systematic methodology factual reasoning can be formed during the initial stage. The methodology offers evidence from an architectural viewpoint on the decision-making that individually domain-specific knowledge lacks to capture. A comparison of the two designs under a different type of Disruption ID 13 in the same area is also shown. Table 43 displays the results of a different Disruption ID 13 for a high redundancy basic design.

Table 43: Type A - Results for high redundancy basic design (Disruption ID: 13)

Type A High – Basic design ***Function: Move and Fight – Disruption ID: 13***	
Equipment Damaged	***Sub-Function Disrupted***
1. Emergency switchboard 2. Emergency generator 3. Steering gear pack 1 4. Steering gear pack 2 5. Starboard rudder (minor damaged) 6. Steering gear pack 1 – LV 1 cabling 7. Steering gear pack 2 – LV 2 cabling 8. Emergency switchboard – steering gear pack 1 cabling 9. Emergency switchboard – steering gear pack 2 cabling	• Propulsion and Steering

Figure 70 illustrated the equipment damaged. The ID of the red boxes corresponds to the ID of the damaged equipment and the ID of the light blue boxes corresponds to the minor damaged equipment described in Table 43.

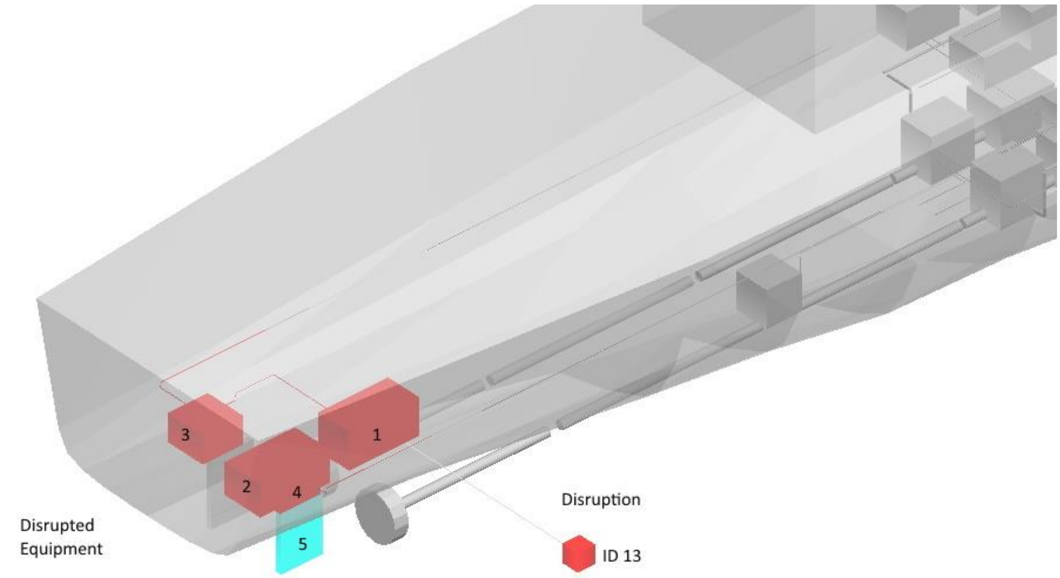


Figure 70: Type A - Basic design results (Disruption ID: 13)

Table 44: Type A - Results for high redundancy robust modular design (Disruption ID: 13)

Type A High – Robust modular design ***Function: Move and Fight – Disruption ID: 13***	
Equipment Damaged	***Sub-Function Disrupted***
1. Steering gear pack 1 2. Starboard rudder (minor damaged) 3. Steering gear pack 1 – LV 1 cabling 4. Emergency switchboard – steering gear pack 1 cabling	• Propulsion and Steering **Comments** • Non-critical • Propulsion and Steering system maintain adequate functionality through use of Thruster (SG Pack 2) forward.

Figure 71 illustrated the equipment damaged. The ID of the red boxes corresponds to the ID of the damaged equipment and the ID of the light blue boxes corresponds to the minor damaged equipment described in Table 44.

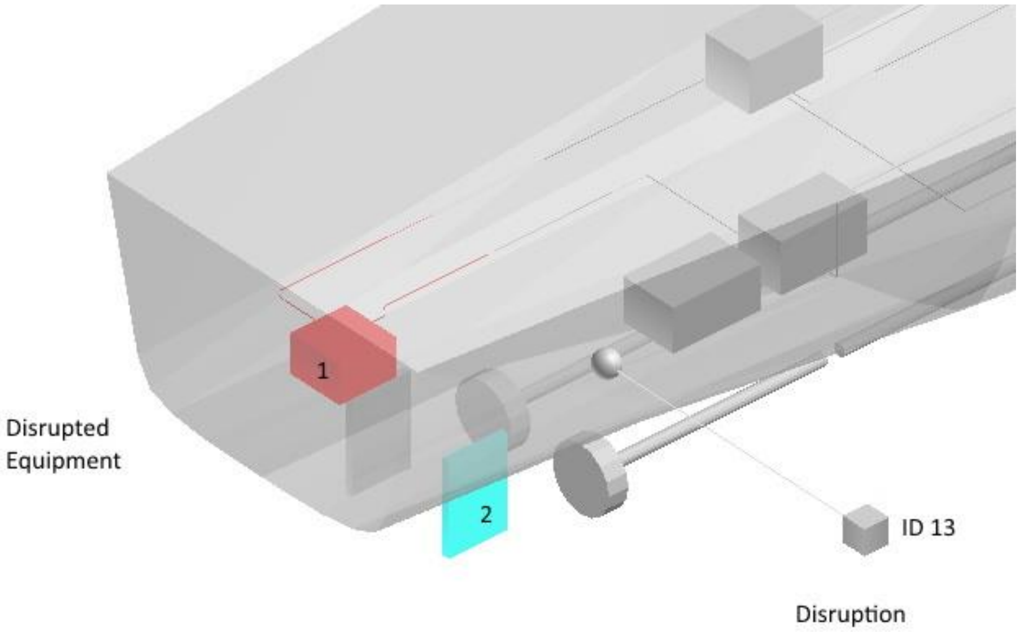


Figure 71: Type A - Updated robust modular design (Disruption ID: 13)

11.3.2 Type A high redundancy versus medium updated designs

The methodology recommends the investigation of various levels of redundancy system architectures through the redundancy loop. The second comparison was based on contrasting the Type A high redundancy comparison versus medium redundancy robust modular designs. According, and based on the robust modular configurations evaluation outcomes previously presented, a medium redundancy system was also modelled and comparisons with the high redundancy model were made. The components removed for the high redundancy model to formulate the medium redundancy system basic design were the following:

1. HV & LV Interconnectors
2. DG 3
3. DG4
4. CWP 3

Figure 72 ship encapsulates the Type A robust modular configuration of Figure 27 (Section 8.2.3).

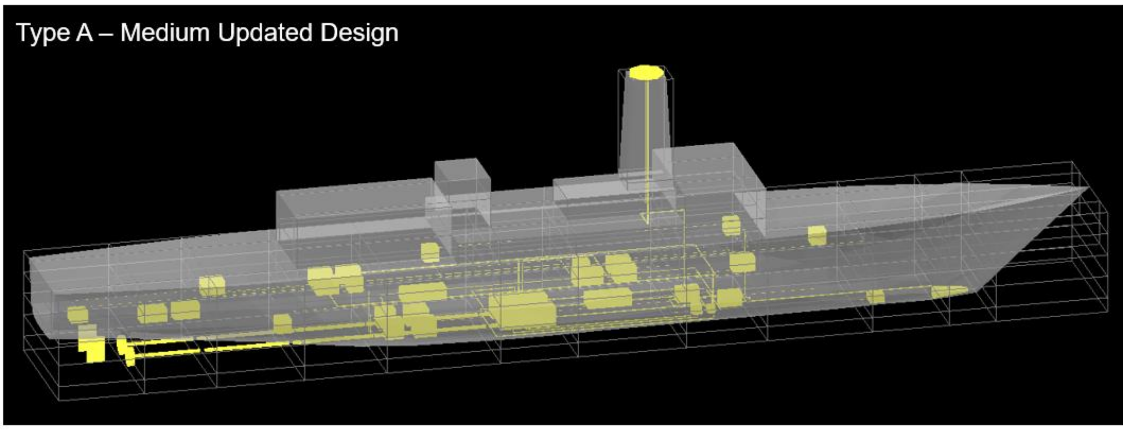


Figure 72: Type A - Medium redundancy updated robust modular design

The Table 45 summarises the damaged equipment for high redundancy in the updated design.

Table 45: Type A - Results for high redundancy robust modular design (Disruption ID: 133)

Type A High – Robust modular design ***Function: Move and Fight – Disruption ID: 133***	
Equipment Damaged	***Sub-Function Disrupted***
1. DG 4 2. HV switchboard 2 3. Transformer 2 4. Converter 2 5. EDC 3 6. LPSW 1 – converter 1 piping 7. CWP 3 – CWP 2 piping 8. LV interconnectors 9. HV interconnectors 10. Converter 2 – motor 2 cabling 11. HV2 – DG2/4 cabling 12. LV 1 – EDC 1,3 cabling 13. LV 2 – EDC 6 cabling 14. EDC 3 – LPSW 1 cabling 15. EDC 5 – CWP 3 cabling	• Power • Propulsion and Steering • Chilled Water • Seawater Cooling

Figure 73 illustrated the equipment damaged. The red boxes ID correspond to the equipment damaged ID in Table 45.

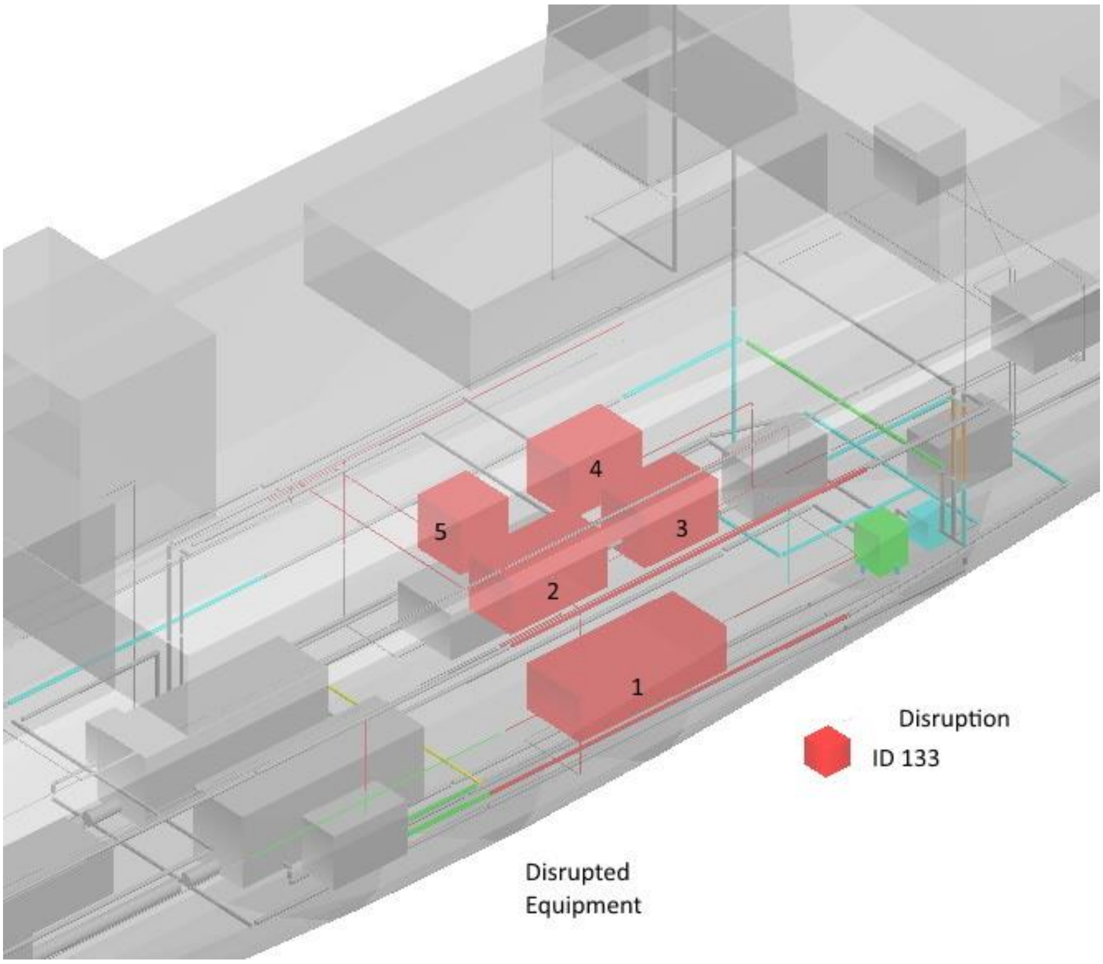


Figure 73: Type A - Robust modular design high redundancy (Disruption ID: 133)

Then the same Disruption ID 133 is simulated in the medium level of redundancy Type A robust modular design as shown in Table 46.

Table 46: Type A - Results for medium redundancy robust modular design (Disruption ID: 133)

Type A Medium – Robust modular design ***Function: Move and Fight – Disruption ID: 133***	
Equipment Damaged	***Sub-Function Disrupted***
1. HV switchboard 2 2. Transformer 2 3. Converter 2 4. EDC 3 5. LPSW 1 – converter 1 piping 6. HV interconnectors 7. Converter 2 – motor 2 cabling 8. HV2 – DG2 cabling 9. LV 1 – EDC 1,3 cabling 10. LV 2 – EDC 6 cabling 11. EDC 3 – LPSW 1 cabling	• Power • Propulsion and Steering • Chilled Water • Seawater Cooling

Figure 74 illustrated the equipment damaged. The red boxes ID correspond to the equipment damaged ID in Table 46.

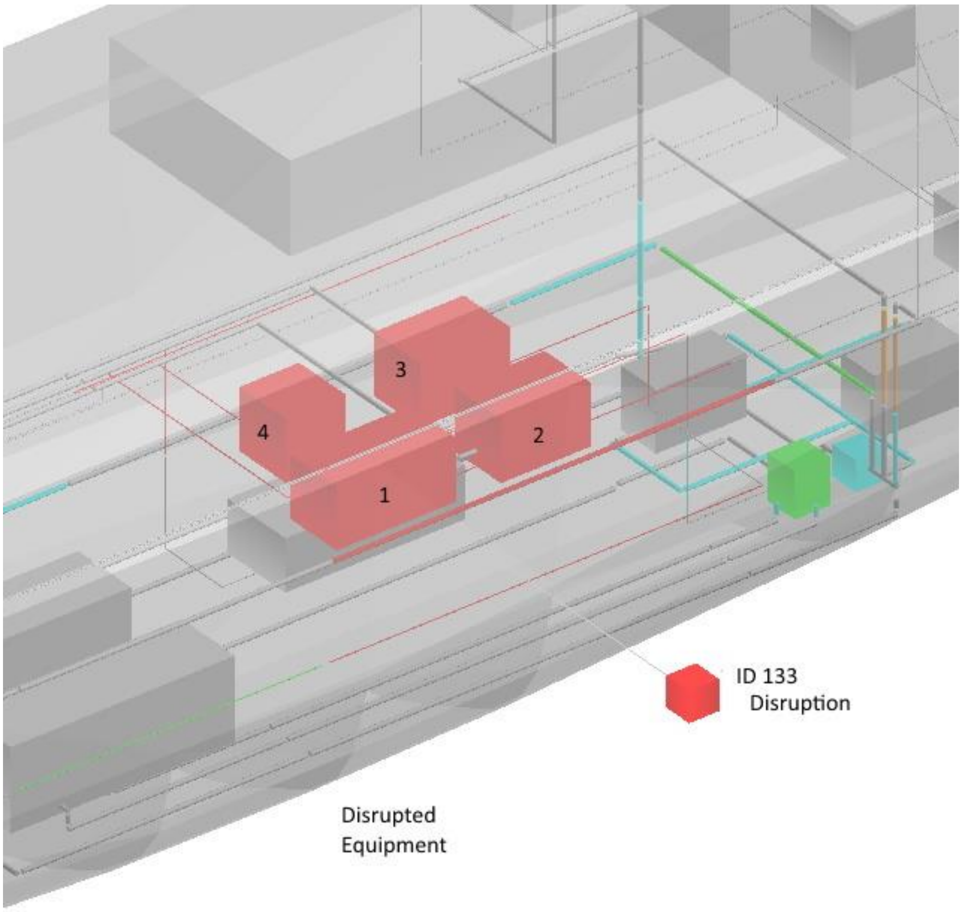


Figure 74: Type A - Robust modular design medium redundancy (Disruption ID 133)

The main findings of the second comparisons simulation indicated that all critical disruptions appear to be in the same areas of the ship comparing medium and high redundancy. Additionally, the medium redundancy model experiences one additional critical hit that was caused because of dependency between high- level and low-level interconnector on the starboard side. In all the other cases that the critical disruptions occur, the damage experienced is similar for both models (medium and high redundancy).

The comparison examples showed indicated that the same consequences from Disruption ID 133 occur for both systems of medium and high redundancy. The high redundancy system loss four more equipment and connections (DG 4, LV interconnectors, ECD3-LPSW 1 Cabling, EDC 4-CWP 3 Cabling) compare to medium redundancy architecture (because there were fewer items in the area affected). This example indicates that redundancy benefits are a non-trivial task that requires analytical and systematic examination.

The findings of the industrial design practice are consistent with the findings of the evaluation part of the Case Studies Chapter 8, whereas the DOE results showed that increasing the level of redundancy from medium to high did not significantly improve robustness under module' disruption. The findings of the industrial design practice offer evidence of realism on the outcomes of the proposed RoMoGA methodology.

11.3.3 Type B high redundancy comparisons basic versus updated design

The Type B high redundancy design presented in Figure 75 which encapsulates the Type B schematic previous presented in Figure 30 (Section 8.3.1).

Figure 75: Type B - High redundancy basic design

The following Table 47 captures the results of SURVIVE software calculated on the ship. For the Disruption ID 85, the list of equipment damaged is reported.

Table 47: Type B - Results for high redundancy basic design (Disruption ID: 85)

Type B High – Basic design ***Function: Move and Fight – Disruption ID: 85***	
Equipment Damaged	***Sub-Function Disrupted***
1. HV SWBD 1 2. HV filter 1 3. Motor 1 4. Convert 1 5. EDC 9 6. HV SWBD 1 – HV SWBD 3 cabling 7. HV SWBD interconnectors 8. LV SWBD interconnectors 9. LV 1 – EDC 1,3,5,7,9 cabling 10. Steering gear pack 1 – LV 1 cabling 11. Emergency SWBD – LV 1 cabling 12. Steering gear pack 2 – LV 2 cabling 13. Emergency SWBD – LV 2 cabling 14. EDC 9 – CWP 4 cabling	None

Figure 76 illustrated the equipment damaged. The red boxes ID correspond to the equipment damaged ID in Table 47.

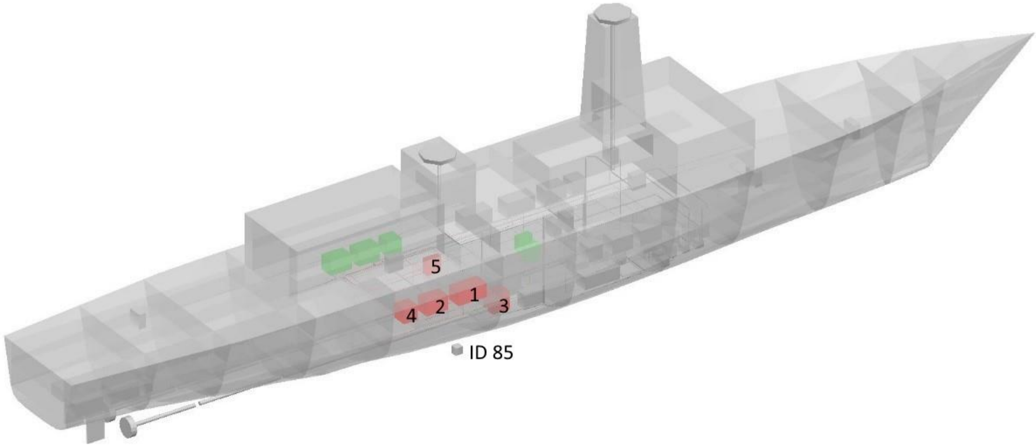


Figure 76: Type B - Basic design results (Disruption ID: 85)

Disruption ID 85 is shown to affect the sub-functions: power, propulsion and steering, chilled water, however, the disruption is not critical, because there is high-level redundancy in the design (including a third switchboard).

The following robust modular design was developed based on four design changes made to the basic design based on the RoMoGA outcome:

1. Aft HV SWBD room moved 1 compartment aft to increase separation from Fwd. HV SWBD Room

2. Aft LV SWBD room moved 1 compartment aft to increase separation from Fwd. HV SWBD Room
3. HV SWBD 3 moved to position 1 deck above DG 3.
4. SG Pack 2 moved forward acting as a Thruster.

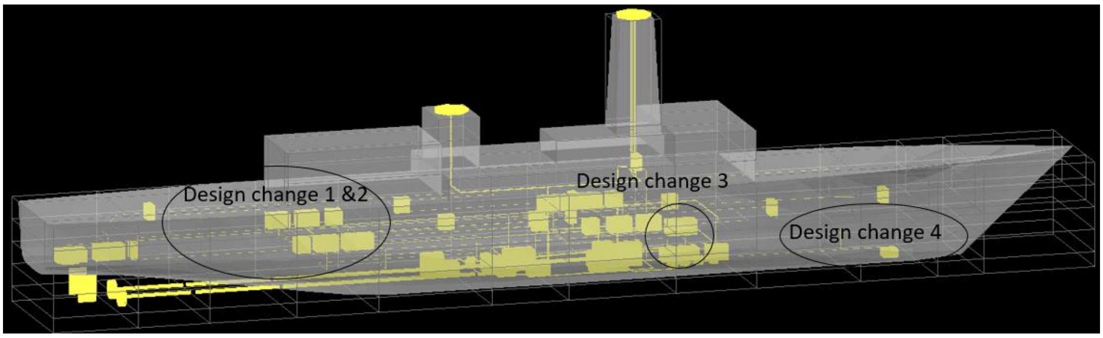


Figure 77: Type B - High redundancy updated robust modular design version

Table 48 summarises the equipment damaged by the Disruption ID 85.

Table 48: Type B - Results for high redundancy robust modular design (Disruption ID: 85)

Type B High – Updated robust modular design ***Function: Move and Fight – Disruption ID: 85***	
Equipment Damaged	***Sub-Function Disrupted***
1. Propulsion motor 1 2. EDC 9 3. Emergency SWBD – LV 1 cabling 4. Emergency SWBD – LV 2 cabling 5. EDC 9 – CWP 4 cabling	None

Figure 78 illustrated the equipment damaged. The red boxes ID correspond to the equipment damaged ID in Table 48.

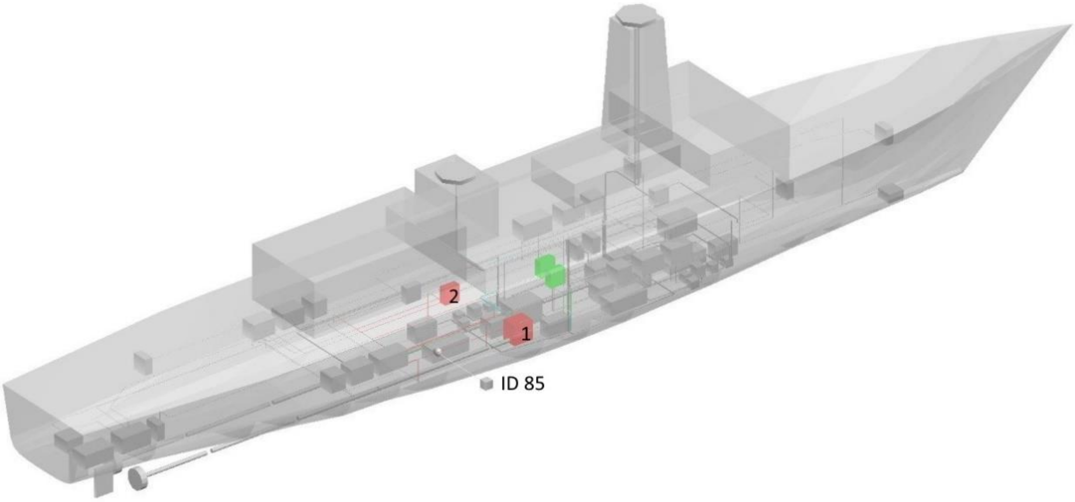


Figure 78: Type B – Updated robust modular design results (Disruption ID: 85)

The Type B was inherently a highly robust design given that had high redundancy in hubs. The update robust modular version of Type B shows an improved result, given that the number of equipment disrupted for the same Disruption ID 85 are significantly less than the number of equipment disrupted for the basic design under Disruption ID 85.

11.3.4 Insights gained using RoMoGA and SURVIVE

The Naval Architecture who performed the modelling and simulation in the SURVIVE software provided his view on SURVIVE software experience that was combined with the use of the robust modular methodology outcomes.

Time labour

“I found Survive requires a large amount of time and patience to be able to build a model. Cable runs and piping between equipment took up the largest amount of my time”

Accuracy

“Given Survive is the UK Mister of Defence’s accepted software for survivability analysis I believe that it does paint an accurate picture of what you’d expect in real-life scenarios”

Ease of use

“Survive is easy to use in terms of creating a ship type and setting up decks, superstructure, watertight bulkheads. Survive runs into difficulty as your model becomes larger with several 100 different pieces of equipment and can often fall over or freeze.

Easy of understanding the output/friendly interfaces

“Survive takes some time to understand the outputs through its user interface. There is a Help guide available, although this is slightly vague in the results section. Without the schematics and hierarchy diagrams, I believe I would have spent a far greater amount of time working on developing the models and the analysis stage. For modelling, I would’ve had to work solely off of ship general arrangement drawings, the schematics combined with the general arrangement drawings gave me a much better understanding of what we were trying to achieve, and this allowed me to create the models in much less time.”

11.4 Discussion of findings

The industrial design practice presented in this Chapter used the outcomes of the RoMoGA methodology in Case Study A and B as presented in Chapter 8, as inputs to perform updates on a basic design. The basic and updated robust modular system architecture designs are then both modelled and simulated in SURVIVE software. The industrial design practice used a generic design as a reference point and based on the outcomes of the RoMoGA methodology a Naval Architect formulates an updated version of the design. The original design and the robust modular design are compared, and reflections on the industrial design practice are given. In respect to the usefulness and applicability of the proposed methodology, it was found that systematic nature of the methodology of establishing the inputs, analysing different levels of modularity and redundancy, and the robust modular configuration outcome had been useful to the Naval Architect to inform the SURVIVE analysis. The Naval Architect that was involved in the industrial design practise (Chapter 11), had acknowledged that outcomes established through the application of the methodology in the Case Studies Chapter 8 had helped him to be able to model and simulate the system in SURVIVE. The outcomes involve the schematics and the functional hierarchical diagrams that are established in the generation stage and the colour robust modular configuration outcomes that are generated in the evaluation stage. The application of the proposed methodology in the systems, prior to the modelling and simulation of the systems in SURVIVE Software offer a foundation of knowledge for Naval Architect to better perform the naval design vulnerability analysis. It is noted here, that a limitation of the SURVIVE industrial design practice is that the interpretation of the robust modular configuration outcomes of Chapter 8 was not debated by a group of experts for example in a workshop and was a significant time-limitation in the process of the interpretation the robust modular network into the updated ship design.

In respect of the appropriateness, it was observed that the proposed methodology was most appropriate in the initial stage before technical constraints are fixed for example spatial. The reason that this limitation with regards to the appropriateness of the methodology was acknowledge was that the assumption is that the ship hull was fixed thus the geometrical and spatial viewpoint did not allow a broader investigation of options that could accommodate the robust modular configuration. That means that the outcome of the methodology could be most useful when the technical constraints are not yet become unchangeable (or too expensive if they change). It was concluded that the greatest benefit of the proposed RoMoGA methodology is when using in initial system design stages. For example, in the simulation examples presented the ship hull was fixed, and the geometric already decided limits that possibilities to allocate the components in the way the robust modular configuration may propose. In addition, the types of components in the schematics were pre-decided. An alternative possible novel technological solution could be encouraged to be employed guided by the recommended robust modular configuration. The SURVIVE software has spatial knowledge of the components that can identify for example long cables and pipes, which increase the vulnerability of the design because they offer a large spatial area that disruption may occur. This aspect of the design is not considered in the proposed methodology. Additionally, the interpretation of the robust modular configuration results in any domain-specific software, the SURVIVE software, depends on the engineer implementing them. Moreover, the robust modular configuration may conflict with technical constraints, for example, electromagnetic or acoustic constraints. Thus, an additional refinement for the robust modular configuration which considers the technical constraints given the specific system context is required.

The industrial design practice carried out as part of the evaluation of this study with the proposed RoMoGA methodology led to the identification of some disadvantages of the proposed methodology. The disadvantages of the proposed methodology are that cannot capture technical constraints, for example, the geometrical and spatial dimension in respect to design for ship vulnerability. It was found that the application of the RoMoGA methodology should precede the use of detailed design practices, and the results of RoMoGA can provide valuable knowledge to designers that can be used as inputs for a more detailed analysis.

11.5 Chapter summary

Chapter 11 outlines the industrial design practice involving the development of updated designs based on the RoMoGA results reported in Chapter 8 and a comparison between the original and the RoMoGA updated design using SURVIVE-ship design vulnerabilities software. The findings suggest that the RoMoGA methodology can help the architect to develop improved system architecture designs during the initial design process and prior to detailed and domain-specific design analysis. Chapter 11 concludes the evaluation part reported in the thesis which includes also: industrial case studies (Chapter 8), explorative experiments (Chapter 9), semi-structured interviews (Chapter 10). The following Chapters 12 offers a reflective discussion on the findings of the evaluation part of the thesis.

Chapter 12: Discussion and reflections

This chapter presents reflections on the previous stages of the study. The research findings from the four evaluation methods are discussed. Then the key limitations of the RoMoGA methodology are summarised. Finally, recommendations for future work are set out.

12.1 Reflections on the research findings

The three industrial challenges that were identified through the explorative focus group discussions (Chapter 3) and investigated through the literature review (Chapter 4) are now used in this section to structure the discussion of the findings of four evaluative research methods. These are as follows: the importance of system architecture design, finding the right level of modularity; balancing modularity and robustness. The following discussion aims to synthesis and compares findings from the four evaluation methods, position the findings in the wider literature and interpreted and reflect on the evaluation part of the study.

12.1.1 Importance of system architecture design

The RoMoGA methodology was developed to address the research aim to develop a methodology to support the design of robust modular system architectures. The requirements for the development of the methodology established in Chapter 4.7 were: 1) able to generate alternative system architectures; 2) combine analysis and evaluation based of redundancy, modularity and robustness requirements; 3) provide computational and quantitative evaluation metrics and an analysis stage; 4) consist of high-level modelling and to require the input of simple data. The following section discusses the findings of the evaluation of the research methods in relation to the four requirements for the proposed methodology. In addition, the methodology is discussed as a decision support tool.

12.1.1.1 Generation of alternative system architectures

The selection of the right system architecture is one of the key decisions which determine whether a system is successful. In the ideal design process, alternative system architecture options should be generated, analysed and comparatively evaluated prior to the final selection at the initial system design stages of design. The findings of the focus group discussion (Chapter 3) and the semi-structured interviews (Chapter 10) indicate that, in the industrial context, there is a phenomenon of design fixation to create new designs based on previous examples, driven by expert opinions that prevent a broad investigation of different system architecture options. In a semi-structured interview with a senior electrical engineer, it was pointed out that typically an exhaustive investigation was not carried out at an initial system design stage, as most of the new designs are evolutions of previous, existing designs. SME stated that "in the context of early decision-making activities, the methodology is sufficient to generate and evaluate high-level solutions without expert electrical knowledge and can help managers and customers to consider possible designs more thoughtfully." However, it was noted that in relation to domain-specific (i.e. electrical) performance, the RoMoGA methodology has some obvious limitations, such as the inability to include capacity in cables and individual characteristics of the equipment. This limitation in methodology concerns the approach of network science, which assumes equality between nodes and high-level abstraction, which does not capture the specific details of the systems.

The explorative implementation is suggested in the RoMoGA methodology following the development of a descriptive system architecture assessment. The reason for this is to establish the dominant topological characteristics and behaviours of the technical system under investigation that require improvement. It is therefore expected that the exploration of the generator will be broader than the descriptive approach based on subject-matter experts and previous design-based approaches. Exploration using a network tool, however, depends on the user, and if a limited number of experiments are developed it may not be as broad as, for example, using an automated generic algorithm generation approach, whereas a large number of random concepts are populated and then optimised. Since the network tool takes into account the structural patterns and features shown to dominate the nominal system architecture, it is assumed that there is a higher likelihood that the new generated system architecture design options are relevant and capable of fulfilling the desired functionality.

System decomposition is an effective approach to study the system. The performance of a system in terms of properties will vary significantly depending on how the system is decomposed into smaller components (Sinha et al., 2019) and the level of granularity has been found to affect the degree of modularity in the architecture (Chiriac et al., 2011b). The network tool is proposed to provide architects with the capacity to explore and analyse different designs at different levels of decomposition. During a semi-structured interview (Chapter 10) with the Senior Consultant Systems Safety & Software Engineer, it was suggested that the use of the RoMoGA methodology could be to explore large-scale simulated networks in more detail at a low level of granularity (detail and smaller subsystems). This can help to compare the properties in the system architecture in different levels of decomposition (high and low). The generator could be used to simulate networks with the same topological characteristics as a given technical system under examination at a high abstraction level, and then a variation of the parameters at a lower decomposition level to analyse structures at lower levels of the network (e.g. structural patterns at the hub, source and sink levels) could be performed. This is advantageous because architects cannot understand a large-scale complex system in the lower level of granularity (detail and smaller subsystems) because of the inherent limitation of the human mind. In the case studies reported in Chapter 8, experienced engineers were able to design and understand the technical systems of the 55-60 components and their inter-relationships due to their in-depth knowledge of the system. However, increasing the scale of the system of interest from 55 to 60 components (higher level of decomposition) to 100 to 120 components (lower level of decomposition) will lead to a lack of understanding by experts of the same systems. A suggestion was that the network tool could be used to populate a simulated network that shares similar characteristics with the given technical system, at a high abstraction level of 55-60 components, and to carry out an explorative investigation by expanding the simulated networks at a low abstraction level and populating a larger network of 100-120 components. Such network tool implementation could motivate the investigation of larger networks with identifiable topological characteristics, and the analysis focuses on properties, for example, modularity and robust behaviours at a lower granularity level.

The findings of the study suggest that the proposed RoMoGA methodology, consisting of a network tool, can help in the inception, analysis and assessment of new system architecture options furthermore could help in the analysis and evaluation of new designs at different levels of decomposition.

Effects of the main structural pattern

In the literature, it is suggested that engineering systems reassemble to some degree one of these architectural patterns: integral, bus-modular or linear-modular (Min et al., 2016). The findings of this study suggest that robustness and modularity are inherent properties influenced by the main structural pattern of the system architecture. Hölttä-Otto et al (2012) evaluated various modularity metrics, based on theoretical patterns, given that their level of modularity could be anticipated. The proposed RoMoGA methodology offers the potential to assess the main structure pattern of a system architecture in the explorative implementation of RoMoGA. During the descriptive RoMoGA, it is not possible to significantly change the main structural pattern of system architecture because an existing defined technical system, with a fixed structural pattern, is considered to be the input. The explorative RoMoGA enables the main structural pattern investigation due to the second parameter of the network tool that allows for variation between bus-modular, hierarchical, path, cyclical, and integral options. The main structural pattern options developed in the network tool relate to the preliminary work of the main structural pattern (Paparistodimou et al., 2017). The main structural pattern was found to have a specific level of properties, such as modularity or complexity. Paparistodimou et al (2017) indicated that the different patterns have different characteristics and that it is appropriate for the architect to select the right pattern from the design of the system architecture in line with the design objectives. The design objectives that this work focuses on are robustness and modularity. The network tool was developed to enable the architect to explore a set of different main structural patterns. The experiments presented in Chapter 9 show that the integral main structural patterns are highly robust and less modular. Bus modular patterns had the worst performance under targeted disruptions (of the most central component) indicating that they were not robust patterns, although they had good modularity levels. Such observations reveal the trade-offs between robustness and modularity on the basis of a specific pattern. This implies that the type of pattern affects both modularity and robustness in different ways. It was found that the hybrid cyclic pattern had an acceptable level of robustness and modularity. This means that an appropriate main structural pattern becomes the enabler of a robust modular system whereas the selection of an inappropriate main structural pattern (e.g. bus modular pattern) prevents a good level of robustness and modularity from coexisting. In particular, the experimental findings presented in Chapter 9 show that the mesh cyclic pattern had relatively medium levels of modularity and robustness. The findings of the study reported in this thesis complement the existing literature on system architecture patterns. The results of the experiments share similarities with Hölttä-Otto et al. (2012) who used a pattern to study modularity metrics. The integral pattern was established as having the lowest modularity level, while the modular bus pattern had a medium modularity level. In addition, the present study identified the hierarchical pattern as having the highest modularity level, while the cyclic and path patterns also had the second-highest modularity level.

The network tool makes it possible to discuss the inherent ability of a system architecture option to be robust and modular preliminary design studies. The architect could experiment with different patterns in order to identify system architecture options that provide the desired level of modularity or robustness as per the design objectives. Experiments are artificial, meaning that they do not represent realistic and feasible patterns of system architecture, and however, offer a tool to explore and generate various potential patterns of system architecture options.

12.1.1.2 Combine analysis and evaluation based on redundancy, modularity and robustness

De Weck et al. (2012) suggested that "investigating ilities in sets may be more meaningful than the study of single ilities in isolation". In agreement with the De Weck et al. (2012) recommendation, this study developed the RoMoGA methodology, which includes an analysis phase, consisting of a nested analysis of redundancy, modularity, and disruption, which enables a combined analysis of the different properties. A level of redundancy is implemented for given system architecture, modularity is investigated, and robustness is calculated. This analysis is repeated for different system architectures descriptively, explorative, and prescriptively. RoMoGA methodology makes it possible to vary system architecture, redundancy, modularity, and disruption, to encourage the architect to identify the correct system architecture option and to make appropriate trade-offs between redundancy, modularity and robustness.

The redundancy loop indicates the investigation of the architecture of the system at different levels of redundancy. The case studies presented in Chapter 8 accepted that the existing system architectures, developed from the data collected from BAE Systems, corresponded to a high-level of redundancy because the systems under consideration were naval systems that were inherently highly redundant. Thus, for medium and low-level redundancy iterations, the system architectures for each case study have been formulated with expert advice. However, the medium and low-level redundancy system architectures for Types A, B and C were not realistic or based on existing architectures and were primarily formulated to demonstrate the proposed methodology. It has been noted that the redundancy loop motivates architects to consider alternative options for the implementation of redundancy in the systems. In the case studies carried out by the experts, the redundancy variation in the technical system was found to be focused on varying the additional capacity of the system. The network tool offers the capability to explore alternative ways that redundancy could be implemented in the system architecture.

For a system architecture with a given level of redundancy, a modular analysis was performed. The modularity loop employed a stability method as a flexible and tuneable approach in helping to identify robust modules. The usefulness of the flexible modular identification approach was that it could generate a variety of candidate modular configuration corresponding to different levels of modularity.

Next, the disruption loop was performed, which included different disruption strategies: component and modular, random, and targeted. The disruption loop allows the calculation of a robustness evaluation metric that provides an overview of the potential robustness of the system. In particular, the disruptions of the target module establish a direct link between modularity and robustness, to investigate the consequences of the disruption of the module. Specifically, this resulted in a key outcome of the case studies that modularity maximisation was not an appropriate approach to the identification of modules (Chapter 8). If only a maximum modular configuration is considered, there is a possibility that non-robust modules will be formulated. A robust modular configuration could, however, be identified by iterating possible modular configurations, assessing the robustness, and then undertaking a final evaluation stage. Modules identified as non-robust must be revised during the evaluation stage and a robust module must be selected from another level (medium or low) of modular configurations. This means that an appropriate level of modularity and robustness can be achieved if careful consideration is given throughout the design studies.

The decision on the level of modularity and redundancy is suggested to be dependent on the architecture of the system under review and to the specific requirements for redundancy, modularity, and robustness. For each different system architecture, a study needs to be carried out to determine the correct level of the set of properties. The RoMoGA methodology is suggested to be applied in particular cases of system architectures; the assessment stage depends on the specific design requirements established for redundancy, modularity, and robustness. Regarding the methodology, the SME submitted the following observations: "It must be a good thing to be able to understand the trade-offs between redundancy, robustness and modularity! I think this helps to make sense of conflicting requirements. It's always easier to develop a design when you have some starting points, even if they're not perfect ". In conclusion, guided by the existing literature that recommended the study of the set of properties together and not isolated, this study proposed the RoMoGA methodology that considers three attributes of the system architecture: redundancy, modularity, and robustness. The proposed RoMoGA methodology was evaluated in the study and the findings suggest that it can aid in the design of robust modular system architectures.

12.1.1.3 Computational and quantitative evaluation metrics and an analysis stage

One noteworthy aspect of the research was the computational analytical stage and the quantification of the methodological outputs for modularity and robustness. The computational phase of the analysis of the RoMoGA methodology is implemented in MATLAB. It is suggested that the user of the methodology iterates through the proposed loops guided either by DOE or until the results are saturated. The computational analysis stage provides an automated and time-efficient approach. But the user oversees the process as the

RoMoGA requires inputs to be inserted, the number of repetitions to be decided and the results to be recorded. The integration of the computational and quantitative evaluation metrics offers an objective approach to the methodology, but the user, who is an expert, can intervene at the stage of the computational analysis as he or she controls inputs and iterations. RoMoGA is a combined manual and computational approach that helps the architect rationalise computational and quantitative results in the technical context. The hybrid manual and automated methodological approach adopted for the development of RoMoGA is consistent with other research that integrates "computational capability of clustering algorithms with the flexibility of manual approaches" (Sanaei et al., 2016).

The quantitative results generated are essential evaluation indicators that support comparison, and evaluation and must be interpreted by the experts. This was a critical finding of the research. The RoMoGA methodology functions as a comparison methodology for the systematic assessment of alternative options in the system architecture. RoMoGA aims to encourage the architect to consider different system architecture options, enabling a filtering exercise at the early stages of the design. The computed evaluation results corresponding to the level of modularity or robustness cannot, on its own, offer any clear conclusion or the final selection to the architect. The proposed methodology does not generate a single optimum system architecture option. Industrial design practise (Chapter 11) has demonstrated how the use of RoMoGA results in practice to inform another established design tool (SURVIVE software) can help the architect to consider different redesign and updated system architecture options.

12.1.1.4 High-level modelling and simple data input

A key requirement for the proposed methodology was to be able to capture system architecture at a high-level as a whole and to required simple data available at an early stage of design. Parker (2014) stated that there is a need for a mathematical framework that represents a wide range of design fields and provides a foundation for analysis and understanding of design before commitment. This mathematical framework needs to operate with only low fidelity inputs because this is what is available in early-stage design. The research described in this thesis adopts network science close to (Parker, 2014). The evaluation part of the study shows how advantageous are the low-level of detail and simple information required by the RoMoGA methodology, which offers a preliminary step before current design modelling techniques can be used.

The semi-structured interview with the Engineering Manager for the Development of Forward Designs indicated that typically analytical methodologies require detailed information that is not available in the early stages, and thus they used a number of calculated assumptions. The interviewer explained that RoMoGA methodology, by contrast, requires information at a low-level of detail that is available during initial system designs. Moreover, the discussion with this engineering manager also highlighted the need for this systematic analytical process to be used in the early design stages so that decisions can be traced and reasoning for the selection of the architectures in order to help with the discussions with stakeholders.

The industrial design practice indicates that using specific domain software (SURVIVE) design vulnerability does not allow trade-offs to be investigated early on. The software requires high accuracy in modelling, and the results are directly related to the detail of the modelling; re-doing the modelling for a high number of alternative possible variations of designs is not a practical option. The RoMoGA methodology allows for early analysis, based on a low-detail, schematic representation, allowing trade-offs to be studied. The naval architect involved in the industrial design study of Chapter 11 stated that "Without the schematics [outcomes of RoMoGA] and hierarchy diagrams [inputs of RoMoGA], I believe I would've spent a far greater amount of time working on developing the models and the analysis stage. For modelling, I would've had to work solely off of ship general arrangement drawings, the schematics combined with the general arrangement drawings gave me a much better understanding of what we were trying to achieve, and this allowed me to create the new models in much less time."

In addition, during the data collection process of the case study (Chapter 8), it was found that no single individual drawing was available to capture the high-level system architecture. The researcher had to create an interconnected view of the system on the basis of specific domain diagrams representing the subsystems through secondary data. This finding from the case study indicated the need for a high-level representation of the system architecture during the initial system design stage. The development of a schematic representation of interconnected systems, which is a high-level representation used by the RoMoGA methodology, was shown to be vital in order to better understand the system as a whole. Typically, the subsystems were represented on independent drawings and there was no representation at a higher level of abstraction in the company's documentation, at the time of the data collection (Chapter 8).

The evaluation findings indicated that the use of high-level modelling techniques to capture the complexity of the technical systems was necessary. The Senior Consultant Systems Safety & Software Engineer stated in the semi-structured interview that "the move to MBSE (Model-Based System Engineering) has begun and the understanding of how the underlying data could be mined and analysed should drive how MBSE is implemented to support these possibilities. In conclusion, the methodology cannot be implemented on its own to realise the value but would need to be part of a wider move to systems engineering modelling methods and tools that generate the data required". The process of applying the RoMoGA methodology highlighted that the quality and correctness of the required inputs are essential for the outcomes to be meaningful. The formulation of system engineering structural diagrams block definition diagrams (BBD) and internal block diagrams (IBD) modes that can be transformed in directed multi-flow graphs facilitate the formulation of the required input for the RoMoGA methodology.

During the semi-structured interviews, it was identified that the company's system engineering software, which is the main tool used by the system architects, offers a design structure matrix (DSM) representation called relationship matrix that is the main input for the methodology. However, this is not currently being used by the system engineering team. The identification of the software potential to help determine the input required by RoMoGA was a key finding to reinforce the practical applicability of the methodology. This is because as the manual development of the three DSMs for the case studies were a time-consuming process that highlighted the absence of a high-level schematic representation of the systems. The evaluation findings indicated that the proposed RoMoGA methodology could be used in combination with commercial system engineering software.

Moreover, the process of applying the RoMoGA methodology in the case studies to existing systems, in an industrial context and with the participation of experts, provided new insights relating to the inputs required. The process of acquiring the inputs that represent the system architecture as a flow network and defining the source and sink components corresponding to sub-functions was found to be critical to the methodology. This generation stage of the methodology does not match the typical naval engineering design process. This approach was found to improve understanding of the system specifically concerning the input required to define the source and sink components by sub-functions. The construction of the hierarchical functional diagram was also found to be a valuable part of the methodology. The case study application of the RoMoGA methodology found that even if the methodology does not require high fidelity of information, it does require a good knowledge of the system from the architects. This relates to the ability of the user/architect to be able to map sub-functions to source and sink components and to define the required level of redundancy. It is recommended that the required input of RoMoGA to be collected during the system engineering activities and that the systematic thinking on the input can help the architect to better model and understand the systems.

12.1.1.5 Decision support tool

In the following two paragraphs, consideration is given to the ability of the proposed methodology to be used as a decision support tool, beyond the focus on selecting the right system architecture option in terms of modularity and robustness. By establishing the results of the methodology, the architect would have selected the desired system architecture, which could serve as a basis for informing the decision on the choice of the correct technological solution or spatial configuration for the system.

Supporting the selection of technological solutions

Modern engineering system architectures rely heavily on key components and technological solutions, as they hold a high-level of critical functionality of the entire system. Crawley et al. (2015) explained that "one of the pivotal roles of the architect is deciding whether and how to infuse new technology into an architecture". System architects are responsible for selecting the right components and technological solutions that can deliver the desired functionality and that can be integrated successfully into the systems during development stages, ensuring that they do not introduce additional complexity or make the system less robust or modular. It is suggested that the proposed RoMoGA could be used to guide the selection of the appropriate subsystems or technological solutions that could contribute to robust modular system architecture.

The industrial design practise consists of using the results of case studies to inform an improved design, that SURVIVE software has assessed with respect to design vulnerability, provided new insights into its usefulness. Reflections on how RoMoGA results could be used as evidence to support decision-making and to provide system architect and integrators with advice on the choice of new technological solutions proposed by suppliers are provided. In the industrial context, typically architects and experts will consider the technical solutions currently known or available in the market and on the basis of possible combinations of technical solutions; they will be able to envisage feasible system architectures with varying degrees of redundancy. RoMoGA methodology reverses this process, providing system architects and integrators with more knowledge to select and decide on a technical solution for developing the desired system architecture, rather than designing system architecture to accommodate established technical solutions. The RoMoGA results could be used by system architects and integrators to encourage suppliers to improve technology or to innovate based on the chosen system architecture. An example of this is the finding of the SURVIVE industrial design practice, that separation of the two steering gears is proposed for the robust modular configuration.

This is a rational engineering result; however, the example is given to discuss the evidence-based system architecture approach that an initial system design stage could motive an alternative decision-making process. In practice, this suggests that, instead of just steering gear at the stern of the ship, a ship could be designed with a thruster located towards the forward part of the ship. Multiple steering gears (with rudders) and thrusters are two different technical solutions that could deliver similar functionality in steering and manoeuvring the ship. However, in the naval context, it is not appropriate to introduce a thruster due to constraints relating to noise. By applying the RoMoGA methodology, it becomes apparent that this solution is consistently recommended (separating the two steering gears into different modules that relate with two LV switchboards); thus the integrator that designs and builds the system had legitimate evidence to further encourage suppliers to improve thruster technology and also had evidence to argue to stakeholders that this is an appropriate solution. This example shows that knowledge of the fundamental system architecture in initial stages could inform decision-making, related to the informed selection of components and technological solutions. Crawley et al. (2015) stated the suppliers may influence the architecture for their good if they are involved early in the architectural process, and that “once a supplier is slated to provide a component, that component becomes a hard boundary”. RoMoGA can, therefore, help to combine architectural decisions with supply chain and technological solutions decision-making at the early stages.

Bayrak et al. (2018) argued that, in some cases, modular design takes place at the system embodiment stage rather than at the design stage, which limits the possibilities for identifying innovative modules. The RoMoGA methodology can recognise potential robust modules before physical implementation is fixed and can, therefore, be subject to technological development. This is a benefit, as it can guide the direction of new technologies that should be chosen and incorporated into the design of the system architecture. In this way, modularity can improve technological innovation as claimed by (Baldwin and Clark, 2000).

Modern architecture is also becoming more complex due to new technologies in the main subsystems. In this way, significant power is ceded to suppliers, allowing them to have significant control over the design of the system architecture. System architectures are thus primarily designed to accommodate these novel solutions. A better understanding of the system architecture can also help to make informed choices regarding subcomponents and technical solutions suitable for system architecture by designers and integrators.

One of the findings from applying the RoMoGA in industrial design practice was that the interpretation of the robust modules could allow different solutions to be decided in practice. For example, the finding that the two steering gears should be separated into two distinct modules motivated an alternative engineering approach to how the same function could be achieved. This means that, instead of redundancy being associated with duplication (adding an additional identical component), a functional substitution type of redundancy was considered, in order to realise the proposed robust modular configuration.

Supporting the selection of spatial configuration

Crawley et al. (2015) explained that spatial relationships explained “where things are: the location or placement of the objects of form. These relationships imply only location and placement, not the ability to transmit anything between the objects”. RoMoGA methodology does not explicitly consider the spatial dimension of the systems, as it adopts a network point of view. The spatial relationships between the components of system architecture are not examined. The proposed RoMoGA methodology captures at the structural network connectivity formal relationships that are “instruments of functional interaction, so they directly support the emergence of function and performance” (Crawley et al., 2015). Even so, RoMoGA is not directly concerned with the spatial relationship between the components; its results could be used to inform the spatial solution of the system.

Typically, the spatial relationships of the systems are not yet fixed at the initial system design stage when RoMoGA is proposed to be used. This gives the architect the freedom to consider alternative system architecture options that relate to connectivity before the spatial solution is constrained. This means that RoMoGA methodology should be implemented before any spatial constraints are established. Then, the findings of the RoMoGA methodology can be used to inform an appropriate spatial solution.

This was found applicable; in particular, in the naval design process of the spiral design. (Page, 2012) explained that “the typical naval design process follows the design spiral ... The design progresses along in a rather linear fashion, assessing each discipline in sequence, e.g., payload, then hull geometry, then space and arrangements, etc.”. Page (2012) clarified that “naval architects and other engineers make educated guesses as to the final hull dimensions, weights, electrical demands, etc., and then refine the initial estimates as better information becomes available from customers, vendors, or other engineers”. In line with the above-mentioned ability to select subsystems and technological solutions guided by the RoMoGA results, an understanding of the system architecture at the connectivity relationships level can better inform the selection of the spatial dimensions of a system, in the case of a naval ship.

In the case studies that were performed on naval engineering systems, which are enclosed within a predefined spatial geometry (ship hull), the SME commented regarding the robust modular configuration results: “The module suggestions are interesting and provide a start point for consideration. There is sound logic in their formulation but transforming into a ship, of course, requires a bit more consideration, generally of constraints. For example, a module that has a diesel generator and propulsion motor. It may not be possible to provide enough longitudinal separation for the diesel generators given the constraints on motor location within the ship, perhaps leading to the need to split each of these modules into two. This does not devalue the results in any way as they give a good starting point for the design which has already been shown to display the characteristics required”

Spatial constraints were not considered, and it was expected that revisions of the proposed robust modular configuration would be required to make it technically feasible. The establishment of a robust modular configuration early in the design process, before spatial and geometrical constraints are fixed, can provide a compass for architects to direct the design towards the desired requirements and to consider spatial configurations that could accommodate the proposed robust modular configuration. As the SME explained in the semi-structured interviews: “at the earliest stages when the system is defined at a block and line-level showing how equipment components are connected, the General Arrangement (GA) in which the system must fit, is still evolving. As the system designs and GA are brought together, and their interactions defined the key drivers and potential maximum benefit levels achievable for all the above characteristics are set.” This shows that, although the spatial aspect of the system influences the relationship between modularity and robustness, investigation at a network level could counteract potential spatial constraints.

The industrial design practice demonstrated the implementation of the robust modular configuration generated by the RoMoGA methodology in the SURVIVE software. A robust modular configuration suggested a grouping of components; however, interpreting this in a spatial configuration requires additional constraints and considerations. The approach in the industrial design practice was to co-locate in the same compartment (or an adjacent one, where possible) the components that were proposed to be grouped into robust modules and to simulate a disruption that will cause these grouped module components to be disrupted simultaneously. The compartmentalisation of components and the dimensions of the ship also affected the vulnerability results generated by the software. The industrial design practice that implemented the robust modular configuration into a specific spatial hull found that it is more difficult to implement the results of the RoMoGA methodology when the spatial aspects of design are fixed. The RoMoGA methodology is more advantageous when used in the initial stages of the design when most of the technical aspects of design are not yet decided and recommendations can be offered on the directions of the design. For example, if the ship hull was not fixed, the compartmentalisation of components into the robust modules that the RoMoGA methodology suggested could be achieved more effectively. In terms of naval architecture and marine engineering, this finding also indicates that the distributed systems that are enclosed within the ship's hull require consideration before the hull of the ship is decided, contrary to the typical spiral design process. In conclusion, the results of the evaluation methods have shown that the results of the RoMoGA methodology can inform the spatial and physical solution of the system.

12.1.2 Finding the right level of modularity

Finding the right level of modularity was recognised in the literature and the explorative focus group discussions as a central theme of the study. The findings of the evaluation part of the study provide further insights on the subject, and these are summarised in three sub-themes: first, regulating the level of modularity; second, flexible and interactive methods for identifying modular configurations; and finally, the structural classification of modules.

12.1.2.1 Optimum level of modularity

RoMoGA methodology recommends a modular reconfiguration based on robust modules, thus rejecting modules that are found by maximising modularity and that are found not to be robust (i.e. their disruption causes the total loss of the system), and includes instead comparably robust modules that were found at medium or low-levels of modularity.

The findings from the case studies were discussed with the SME, who emphasised that a "high-level of modularity is not necessarily better than a medium-level of modularity when all considerations of buildability, installation, setting to work etc. are considered. The benefits of modularity will need to be assessed for each possible configuration". The research findings from the case studies indicate that care should be taken when considering increasing modularity, because an approach that focuses solely on maximising modularity and grouping components strictly based on the maximum number of interactions with modules and the minimum number of interconnections amongst modules may not be the most appropriate.

During the semi-structured interviews with the Engineering Manager for the Development of Future Designs, these tensions regarding the desirability of modularity and the need to regulate the level of modularity in a way that will bring in practice benefits were also highlighted. The SME stated that "modularity is a key driver for producibility and is greatly influenced by system component location. As such it is likely to conflict with survivability requirements.

The potential for modularity benefits is determined at the early stage system and general arrangement integration. The actual benefits are only determined during later design development". He continued, "the downside is that modules may require more space in the ship. There is, therefore, a cost-benefit trade-off in deciding the correct level for both! The purpose of the methodology is to explore these trade-offs and decide where the optimum balance is for a particular project". Therefore, it is established that the right level of modularity for a system should be examined through a study of trade-offs, rather than through a single task approach, aiming to maximise modularity. In addition, the experiments with the network tool show that there are patterns that perhaps have a medium (not the maximum) level of modularity (mesh cyclical) and that also have a medium to a high-level of robustness under targeted and random attacks. An architect might decide to proceed with a theoretical system architecture pattern that has a medium level of modularity, but a high-level of robustness. For example, the redesign of Type A presented in Section 9.3, considered by experts to be an improved system architecture design due to improved robustness, has a slight decrease in the level of modularity. The industrial design practice also provided evidence that a maximisation approach to implementing modularity is challenging because of the various constraints that apply in practical design. In the SURVIVE study, for example, dividing the system architecture into distinct modules within a ship hull was difficult because of the geometrical constraints, while separating two modules could increase the length of a critical interconnection between the modules, which could endanger the robustness of the system. Consequently, minimising the number of connections between modules, but at the same time increasing their length, was not an all-around advantageous approach.

The methodology proposed in this study seeks to find a modular configuration solution, by accepting that an absolute modularisation of the system may not be the desirable approach. This study findings agree to the findings of the research of Sinha and Suh (2017) that performed Pareto-optimisation of complex system architecture between structural complexity and modularity, and found that "modularity maximizing decompositions tend to have a smaller number of modules with a higher degree of asymmetry in module-level complexity distribution". This is considered negative as "a few numbers of highly complex modules can act as bottlenecks and results in system development delay" (Sinha and Suh, 2017).

This research study suggests that the approach to modularity maximisation may have negative effects on robustness, and the development of few in number and complex modules (in size and connectivity) may, in the event of a disruption, jeopardise the entire system's functional continuity.

Apart from the two points of view discussed above: structural complexity and robustness, other aspects are considered when selecting a modular configuration. An important reason for the pursuit of modularity, for example, is to facilitate parallel development and reduce the assembly time of the system. Modularity is a positive attribute of the system because it can support the development and construction of the system. As discussed above, maximising modularity can lead to fewer modules of higher complexity. This may be positive for assembly when the manufacturing facilities are capable of handling large-scale modules. There are examples from the industry that large size modules (due to weight or space) have in practice introduced additional problems during production, where it was difficult to move in the assembly line due, for example, to the limitation of the existing lifting equipment.

Overall, the findings of the evaluation also agree with Walsh et al. (2018) that “a system’s modularity is situation-dependent rather than universal” and that continuous assessment of the modularity is required.

12.1.2.2 Flexible and interactive modularity methods

The proposed methodology provides the techniques to perform this continuous assessment of different system architectures, to help the architect to identify the right level of modularity that corresponds to the given system architecture. The most desirable level of modularity is subject to other aspects and trade-offs that architects are expected to take into consideration. The findings of the case studies show that interactively identifying various possible modular configurations at different levels of modularity helps the user to find good modules that are potentially feasible and poor modules that are not robust and that contain redundant components. The use of a flexible approach for identifying modules provides the architect with a whole set of possible modules to evaluate. In the industrial design practice, the set of modular configurations for various levels of modularity were used to address problems arising also from other technical constraints (mainly geometrical). The information gathered from examining different levels of modularity can supply the architect with alternative potential modules that could be more feasible for a given design. The idea proposed by Eppinger et al. (1994) of maximising interactions within modules while simultaneously minimising interactions between modules has set a trend for modular design methods developed in this field which have adopted this approach (e.g. Whitfield et al., 2002; Yu et al., 2007). However, the findings of the present study suggest that alternative approaches are required. Or, as Sanaei et al. (2016) put it, “Novel approaches are needed to enable interactive definition of modular architectures ... also capable of coping with large complex systems using algorithms to suggest or guide more effective modularity schemes.”. This also aligns with Bayrak et al. (2018), who adopts a multi-objective optimization approach for modular design and “generate modular solutions quantifying trade-offs with respect to varying degrees of modularity”.

12.1.2.3 Structural classification of modules

The classification included in RoMoGA embraces the same idea of developing a modular configuration by selecting appropriate modules founds at different levels of modularity. The classification of modules is a construct to investigate the architectural role of modules in the system architecture. For the case studies and the industrial design practice, technical system architectures that relate to naval engineering system analogies were used to devise a classification. The concept of diameter was adapted from network theory to suggest the classification of modules.

In network theory, the diameter is used as an analogy with the concept of damage extent used in studies of survivability. This analogy suggests an approximation between damage extent and diameter that allows for correlation between the network and the instantiated system architecture. In the context of survivability, damage extent is the physical length between components that are vulnerable to impairment by disruption. Based on the results of this study’s technical case study, inferences on the association between the classes of modules were made. Peripheral modules were deduced to have an association with the damage extent of survivability. Central modules were deduced to have an association with system functionality. As for semi-peripheral modules, their association with functionality or survivability was less pronounced. A high diameter implies a higher length for the nodes in the network, which was assumed to relate to an increased separation between components in the system and a higher extent of damage, increasing the possibility of impairment following a disruption. Based on this analogy, it was suggested that the most peripheral module, which has the maximum diameter and therefore the maximum extent of damage, was consequently the type of module most possible likely to be affected by the disruption.

Accepting that peripheral modules have a high possibility of being disrupted, this finding suggests that a higher level of modularity could be preferable for specific periphery modules, given that a concentration of components into highly coherent peripheral modules is preferable. The central modules in the technical system case studies were observed to possess a high degree of system functionality. This implies that the largest central modules could endanger the robustness of the architecture at a high-level of modularity: given that these are central, their disruption may lead to the loss of a high degree of the functionality of the system. Semi-peripheral modules are suggested to have a reduced association with survivability (peripheral) and with system functionality (central) and, therefore, provide opportunities for further manipulation and updated development. For example, in the case study Type A the non-robust module that needed to be updated in order to reformulate a robust modular configuration was a semi-peripheral module. It is expected that different analogies would be appropriate for different system architectures which would generate different results; thus, it is not suggested that the findings be generalised. This discussion also suggests that different rules and recommendations be developed to formulate new robust modular configurations. If architects find analogies and associations between classes of modules with other design requirements, then new principles could be developed for selecting modules. For example, another principle could be that the robust semi-peripheral module in the maximum modular configuration is most preferable. The purpose of classifying of modules was to offer an additional way of expanding the search for the desired robust modular configurations by helping architects to reflect on the architectural role and behaviour of different classes of modules.

12.1.3 Balancing modularity and robustness

The findings of the four different research methods with regards to the relationship between redundancy, modularity and robustness will now be discussed. Here, three sub-themes were identified: effects of redundancy on robustness and modularity, effects of the type of redundancy and type of disruptions.

12.1.3.1 Effects of level of redundancy on robustness and modularity

In this section, the effects of redundancy are discussed with respect to robustness and modularity. The outcomes of the RoMoGA methodology in the Type A case study reported in Section 8.2.3 were discussed with the SMEs, who advised that they considered the modules to be logical in the context of naval design, given that typically at least a minimum level of double redundancy exists across the system (in specific subsystems the level of redundancy could be either triple or quadruple). In relation to redundancy, the SME commented: “The observation that increasing redundancy for a similar level of modularity does not necessarily increase robustness is to be expected, as some of the larger modules start to contain redundant systems, for example, two generators in a single module. The results for all modules, therefore, seem logical.”

The findings of the case studies indicate that redundancy at a component level requires investigation to ensure that it contributes to improving robustness. That means that the implementation of redundancy is also a situation-dependent principle. In response to disruption, the existence of redundant components does not ensure the robustness of the architecture as a whole. It was, however, acknowledged that such redundancy was included for reasons of reliability and availability, rather than to survive an extensive disruption which, for example, damages all components in a module. Medium redundancy significantly improves robustness, compared to low redundancy. However, high redundancy does not continue to improve the robustness of the architecture, compared to medium redundancy, when it is subject to a module disruption at the same extent in the case of Type A and Type C case studies.

This observation depends on the type of system architecture under examination, the type of disruption against which the robustness is calculated, and the type of redundancy involved. For example, the results from Type B show that increasing redundancy from medium to high did improve the robustness of the overall architecture. This observation relating to the level of redundancy is noteworthy because increasing redundancy has cost implications; it increases the spatial and power requirements; it increases weight, and it increases susceptibility to accidents and errors when additional components and connections are relied on to satisfy its function. In general, the optimal approach from an engineering perspective is to achieve robustness with increased modularity and decreased redundancy. The semi-structured interviews with the production manager highlighted their views on redundancy. The SME stated that "a design with increased redundancy naturally has increased production work-scope to connect the physically separated equipment with pipes and cables. The increase in work is proportional to the number of redundant equipment and the separation distance. This increases the number of parts, but also the congestion of the space which makes completing the work harder".

With respect to the naval distributed system architecture, increasing redundancy from low to medium significantly increased its robustness. However, high redundancy that relates with additional source components and connections was not found to considerably improve the robustness of the system architecture when a module was disrupted. Increasing the level of modularity tends to lead to an increase in the size of modules. If a disruption were to happen to a large-sized module, a higher number of components (which includes the adjacent, redundant component) would be simultaneously lost, meaning that the redundant component would not add to the robustness of the system architecture. The results of the case studies suggest that redundancy acts as an enabler in the relationship between modularity and robustness and is the key to the design of robust modular systems. The positive effect of redundancy on robustness is widely known in engineering systems. However, these findings also indicate that the capacity of redundancy to have a beneficial effect on the relationship between modularity and robustness depends on the specific system architecture under consideration.

The positive effects of redundancy on robustness, independent of modularity, were also evident in the experiments with the network tool where a redundancy threshold criterion was varied in the theoretical system architecture patterns. The results show that increasing redundancy (RTC) was positive for robustness under both targeted and random component disruptions. In the industrial design practice that was elaborated using the SURVIVE software, which also considers geometry and technical details of the system without giving attention to specific instances (as was reported in Chapter 10), the overall results calculated by SURVIVE indicate that, for a higher level of redundancy, there was a lower probability of vulnerability for the ship. These findings indicate that attention is needed to decide the level of redundancy. It is evident that redundancy improves robustness; however, an examination of redundancy in combination with modularity is required in systems that are intended to be designed as robust and modular.

Regarding the effects of redundancy on modularity and robustness, it was observed that redundancy mostly contributed positively when the level of modularity was medium rather than high. This is a logical finding because high modularity means that many redundant components will be assembled in large modules. The case studies results show that the highest modularity is calculated for low redundancy architecture. The results showed that the modular configurations for low redundancy systems had the highest possible degree of modularity, compared with medium and high-level architectures in technical system A. This finding confirms that increasing redundancy also reduces the potential to modularise the system architecture. The finding of the case study agrees with the literature which suggests that “many implementations of redundancy reduce the modularity of the system because the mapping between functions and components becomes more complex” (Chen and Crilly, 2014).

The semi-structured interviews also indicate that the production engineer believed that a highly redundant system could not be modularised: “there is limited opportunity to co-locate items which are interconnected by this stage. Hence, the number of interfaces between modules is likely to be higher, and the number of outfit items installed on modules lower than integrating the modular approach in early design”. The experiments with the network tool, manipulating the redundancy threshold criterion and calculating the level of modularity, also indicate that a higher level of redundancy has a negative effect on the level of modularity. The effects of redundancy on modularity were also documented during the industrial design practice that included modelling the schematics to SURVIVE and taking into consideration the geometry of the system’s elements and interconnections. The industrial design practice indicated more clearly that additional connectivity between components and additional redundant components made it difficult to separate the system into coherent modules. The findings of the four research methods provided both qualitative and quantitative evidence to support the negative relationship between modularity and redundancy. The findings suggest that a medium level of redundancy and modularity could be an acceptable trade-off to satisfy an appropriate level of robustness required by the system architecture. This is suggested considering the desired trade-offs between the preferred level of modularity, the degree of redundancy and the situation-dependent nature of the design problem.

Discussion of the effects of redundancy and modularity was found in the literature. Although the study undertaken by Siemaszko and Pittet (2011) differs from this one, as concerns reliability, their results offer a basis for comparison with the findings of this study. Siemaszko and Pittet (2011) concluded that the “best reliability figures are obtained with the use of one redundant module only. It seems that the use of more modules adds more possible failures to the system than it actually saves”. In keeping with the work reported in this thesis in respect, they find that redundancy does not seem to have a constant positive effect on robustness. Similarly, Fricke & Schulz (2005) provide an example of poorly implemented redundancy in an “Ariane 501 control system, where the redundant control system was based on similar systems [HW and SW] and therefore both units failed at the same time for the same reason”. Although the research reported in this thesis does not concern reliability (component failure issues) and focuses on the robustness of the system architecture, the potential redundancy gains to justify its implementation in design need to be considered across its various dimensions: cost, production, reliability, availability and robustness.

The finding that design for redundancy is a design approach which needs careful consideration is in line with the principles of good architectural development. Crawley et al. (2015) argued that good system architecture means that achieving the desired function at a minimal structure is synonymous with delivering value (minimum cost-benefit) and referred to the axiom of lean manufacturing that more components attract costs.

12.1.3.2 Effects of the type of redundancy on robustness and modularity

The proposed RoMoGA methodology includes a redundancy loop that suggests experimentation with redundancy levels to investigate the best possible redundancy options. In a descriptive RoMoGA, the redundancy loop experiments with the level of redundancy by varying the number of source components and the level of connectivity between the source and the sink. In Case Studies Chapter 8, the experts provided alternative design options of the system architectures with varying levels of redundancy (Type A, B, C). The researcher observed that the experts changed the redundancy in all alternative system architecture options by varying the number of source components. However, a key difference between the descriptive (technical systems-Chapter 8 Case Studies) and the explorative (simulated networks-Chapter 9 Experiments) implementation of the RoMoGA methodology is that the explorative approach allowed a broader, maybe counterintuitive exploration of the parameters that decide redundancy than the experts may have performed. This can contribute to the design of system architectures with different levels of redundancy. The main parameter that was found to influence robustness is the number of hubs. In the experiments presented in Chapter 9, it was found that increasing the number of hubs below the cut-off point (referring to the size of the expected disruptions) is beneficial for the robustness of simulated networks which share similarities with the technical systems (referred to in Chapter 8). An increase in the number of hubs in technical systems (from 4 to 5-10 hubs) was found to have a positive influence for the target and random robustness, given that major disruptions were not expected. Again, this finding of the experiments is in agreement with the cross-case studies in Chapter 8, and the finding that the high-level redundancy type B system architecture, which has five main switchboards (three HV switchboards and two LV switchboards), is, therefore, the technical system with the highest number of hubs, this had the best robustness among the three technical systems examined. It is noted that the number of hubs in the descriptive RoMoGA concerning the technical system did not vary in the redundancy loops. This was because of the redundancy variation recommended by experts focuses on changing the number of redundant source components (DG or GT power sources).

A common definition of redundancy is that "the concept of redundancy concerns the provision of additional capacity in the system so that system performance is maintained despite partial system failure" (Chen and Crilly, 2014). However, the experiments reported in Chapter 9, with the parameters of a network tool that allow for variations in the number of hubs, indicate that other parameters that may not be associated with additional capacity could be used to design redundancy in systems. The results show that an effective way to design for redundancy is to increase the number of hubs in technical systems. This is an alternative to the redundancy options of increasing the number of sources and the level of connectivity between the source and the sink components. The findings of the case studies (Type B) and the experiments agree that the system architecture with an increased number of hubs (5-10) is more robust. These findings relate to technical systems with a size of 40-80 components based on the experimental set-up range (Chapter 9) and a total of 58 components of the Type B system. Another finding was that connectivity between hubs manifested inverse behaviour as a result of the random and targeted disruption. This is also critical because it means that consideration should be given to how to design connectivity between the main hubs when designing a system that expects to have targeted disruptions. Another noteworthy finding was that the density of the hubs (number of connections, feeding or supplying the hub) was not as important as the number of hubs.

The findings of the experiments and the case studies suggest that different approaches at the topological level should be considered when the level and options of redundancy vary. The experimental findings in relation to hubs are important in the design of a robust system. This approach is different from introducing redundancy with the addition of supply or capacity in the systems. It is suggested that the user experiments with the level of redundancy, the varying number and connectivity between hubs, the number of sources, and the level of connectivity between source and sink components in the prescriptive implementation of the RoMoGA methodology. The findings of the study suggest that there is a need for a careful investigation into how redundancy is implemented in the design of robust systems. As shown in Chapter 8 Case Studies results of Type A, there are specific situations, e.g. additional redundancy (in source components) may not improve the robustness of the architecture. In contrast to Type A, Type B has additional redundancy at the hub level (main switchboard), which improves robustness. This may be because additional source components tend to be grouped in the same module, while additional hub components tend to form different modules, which may result in the overall modularity level slightly decreasing. However, as the goal is to balance modularity and robustness, it is suggested that designing the system with smaller robust modules can be acceptable. Therefore, in order for hub level redundancy to be beneficial for robustness, it is necessary to make sure that alternative hubs are located in different modules. The results suggest that a robust modular system architecture can be achieved by modularising in such a way as to contain redundant hubs in various modules as more balanced and logical modules can be generated. For example, the Type B high redundancy and low-level modular configuration, has generated modules that each hub (HV SWBD) is contained in different modules (Appendix II). In contrast, the Type C high redundancy and low-level modular configuration, the redundant components are not able to be separated in different modules (Module 1 contains source components DG1&3 and Module 2 contains DG2&4) (Appendix II). Architecture with separate redundant source components will lead to an architecture that is more closely resembling an integrated architecture, that is, one "made up purely of the lowest level of components without having intermediate assemblies" (Yu et al., 2007). In short, it is suggested that hub redundancy could make it acceptable to create a medium-level modular configuration, while extensive redundant source component capacity would make it difficult to modularise architecture. The medium-level modular architecture is not a "fully modular architecture is one with clear clusters of elements, and where the relationships between the elements within an assembly are hidden to the elements outside the assembly" (Yu et al., 2007).

The study suggests modular configuration requires consideration of redundant component allocations. Furthermore, the study findings suggest that the concept of redundancy should be extended beyond the design of additional capacity in the system since the provision of additional hubs is a successful approach to balancing robustness and modularity in the architecture.

12.1.3.3 Effects of the type of disruptions on robustness

Results of robustness calculated against component and module level disruptions and random or targeted disruption strategies indicate that the type of disruption is critical to the estimated robustness of the architecture. The robustness results calculated through the case studies and experiments show dependency on the type of disruption. This implies that the environment in which the system is operated needs to be defined both clearly and early in the design stage, as this will guide the robustness design requirement. In the case of an uncertain future design environment, further investigation, including uncertainty analysis, could be considered using a probabilistic approach. This study only suggests deterministic disruptive strategies, but it could be adapted in the future to include probabilistic disruptions as well.

The operational environment of the naval distributed technical systems that were examined in the case studies, suggests that they expected significant size single disruptions. This translates that the technical systems can lose at the same time a number of components that are adjacently located, therefore, can lose a module. This type of disruption is different than reliability failures, that for example, a single component fails, or a number of components fail due to the meantime to failure (MTTF) or the mean time between failures (MTBF). The simultaneous disruption of grouped component shares analogies with the disruption of a module (a group of highly connected components). In this research, the premises are that disruptions at module and component levels have a different effect on the robustness of the system. Moreover, careful consideration of the implementation of redundancy is required with technical systems that are expected to experience module-level disruption. The results of Type A showed that the robustness of the architecture under module disruption was significantly improved when medium redundancy was designed into the architecture, compared with non-redundancy (low redundancy) architecture. In contrast, in the case of high-redundancy architecture (Type A), the robustness of the architecture under module disruption was not significantly improved, compared with medium redundancy. However, the results of Type B which had a higher level of redundancy in hub components showed that an increase in redundancy from medium to high does benefit the robustness of the architecture under module disruption. By contrast, robustness under component level disruption did benefit consistently from the increase in the redundancy. This reinforces the need to consider and decide on modularity, redundancy, and robustness together during the design of the systems.

The robustness of the simulated networks was calculated based on random and targeted disruptions during the experiments in the network tool Chapter 9. In some cases, the effects of the simulated network parameters were found to be different, depending on the disruption strategy. For example, the level of connectivity between hubs is beneficial for robustness in random attacks, but not in target attacks. Therefore, the type of disruption strategy influences the estimated robustness. For the architect to decide on the effects of an architectural parameter on robustness, consideration must be given to the future operational environments, and potential disruptions surrounding it.

In conclusion, the type of disruptions component or module, random or target critically informs the calculation of robustness. The establishment of an overview of the robustness of the system architecture involves considering the different types of disruption, given the future operating scenarios and the environment in which the system will operate. The design of a modular system enables a modular disruption scenario for a variety of reasons, which also make it essential to consider the robustness of these modular systems.

12.2 RoMoGA position in the literature

This section of the literature review was performed after the development of the RoMoGA methodology. In light of the literature review findings, this section specifically compares scholarly works which are recognised as sharing the most commonalities with the suggested methodology reported in this thesis. The studies are primarily from the engineering design literature, but also include some examples from wider literature. To conclude the literature review, a reference comparison Table 49 was developed, and three criteria were developed for comparison:

1. Attributes: Does the methodology examine redundancy, robustness and modularity?
2. Initial system design stage: Does the methodology address the generation, analysis and evaluation stages of design?
3. Method: Does the methodology include qualitative or quantitative techniques?

Table 49: Literature comparison Table

	References	**Attributes/Properties**			**Initial system design stage**			**Method**	
		Redundancy	**Modularity**	**Robustness**	**Generation**	**Analysis**	**Evaluation**	**Quatitative**	**Qualitative**
1	(Zakarian et al., 2007)		√	√		√	√	√	
2	(Mehrpouyan et al., 2015)		√	√	√	√	√	√	
3	(Sanaei et al., 2016)		√			√	√	√	√
4	(Hvam et al., 2017)		√				√		√
5	(Bayrak et al., 2018)		√		√	√	√	√	
6	(Walsh et al., 2019)		√	√		√	√	√	
7	(de Vos and Stapersma, 2018)			√	√	√	√	√	

The study summarised the seven research works which were discovered to be most adjacent to the study's solution. Chang's original piece of research (1996) was titled: 'Conceptual robustness in distributed concurrent engineering and modular design'. Despite the apparent title, this work approaches robustness from a quality engineering point of view, rather than the robust system design approach that concerns this research. Chang (1996) explains that his method was "to obtain modular design by considering the devices to be fitted as conceptual noise and designing modular parts to be conceptually robust against such noise". Therefore, the conceptual robustness mentioned in Chang's work is distinct from the robustness in a system design point of view that this research scope on. This is the reason that was not included in the following Table 47.

Existing literature was included in the comparative Table 47 which was recognised to share the following view on modularity: namely, that not seeking for maximum ideal modularity (Bayrak et al., 2018; Hvam et al., 2017; Sanaei et al., 2016). Quantitative based methods (Bayrak et al., 2018; Sanaei et al., 2016) were identified, which make use of multi-objective techniques to discover modular configuration taking into consideration additional constraints leading to non-maximal modularity configuration. Bayrak et al.'s (2018) method include generation, analysis, and evaluation stages, but the generation is function-based (not architectural). Sanaei et al.'s (2016) method is focused on analysis and evaluation and combines both algorithm and expert’s inputs. Hvam et al. (2017) proposed a conceptual tool to aid in understanding partial modularisation in engineer-to-order companies, accepting that maximising modularity is not the best strategy in engineer-to-order companies. The modularity application matrix is a qualitative framework that can support an evaluation phase. However, these studies (Bayrak et al., 2018; Hvam et al., 2017; Sanaei et al., 2016) do not include the properties of redundancy and robustness in their work.

Approaches focusing on robustness and which consider modularity were also identified in the literature (Chang, 1996; Mehrpouyan et al., 2014; Walsh et al., 2019; Zakarian et al., 2007). Mehrpouyan et al. (2014) and Walsh et al. (2018) employ network science methods to examine the robustness of engineering systems, and also considered the trade-offs between modularity and robustness. Mehrpouyan et al. (2014) proposed a methodology that includes design generation which is based on maximising robustness and applies the methodology in three designs that have different kinds of modularity. The proposed methodology does not include a modularisation stage in the design, but the authors reflect on the relationship between robustness and modularity based on the findings of the application of their approach to the case studies. Walsh et al. (2018) analysed and evaluate three engineering systems using network-based metrics for robustness and modularity. Mehrpouyan et al. (2014) and Walsh et al. (2018) are the most recent works found in the literature to identify the tension between modularity and robustness. However, they do not propose a methodology to support the architect in formulating robust modular system architecture. Zakarian et al. (2007) proposed a design methodology focusing on system robustness, which also implicitly addresses modularity, as the methodology proposes using clustering algorithms to group components, and then experimenting with interconnections between modules to find out which ones are most suitable for designing a robust system. However, this method does not consider redundancy, does not use a quantitative metric to assess the robustness and does not employ a flexible clustering approach. De Vos and Stapersma (2018) proposed an approach that includes generation, analysis, and evaluation and which examines the robustness of systems. Their work does not consider modularity and redundancy and is focused on distributed ship engineering systems. However, the robustness metric proposed by de Vos and Stapersma (2018) shares the similarity with the robustness metric developed in this study as they are both network-based and both use the notation of source and sink connectivity to achieve functionality. In addition, none of these works examines redundancy and its effects on modularity and robustness. In general, no method was found to consider modularity and robustness of system architecture together.

Explorative RoMoGA methodology includes a novel network tool at the generation stage that is suggested as a tool for exploring potential system architecture options. The network tool populates simulated networks which share similarities with engineering systems. Subsequently, the network tool is positioned to other recent works in network science engineering design literature. Comparing the simulated networks that the generator produces against the range of technical networks exhibited in the recent work of Walsh et al. (2019) it is demonstrated that the generator populates networks that share many of the attributes of established technical systems. It is observed that the technical networks reported in Walsh et al. (2019) exemplified instances of the artificial models proposed in this study. Focusing on the component-based networks reported in the Walsh et al. (2019) it is observed that the bicycle drivetrain has a cyclical hub with hubs linked to paths, and the automobile drive train has a path as a backbone with the bus-modular hub. The aircraft network has a hybrid structure with a path backbone and hubs which are generally integral but there is also a level of interconnection between hubs (around 25% here).

In the design research, literature scholars have simulated networks and compare with the original examples to identify areas of improvement. Piccolo et al. (2018) developed two null models: one that conserves degree distribution and a second which preserves the number of edges but redistributes them randomly between nodes. In contrast, the network tool proposed in this study focuses on a different factor, namely the simulation attempts to preserve structural patterns (e.g. cyclical pattern) and features (e.g. hubs) for small size networks. In this study, it is chosen to develop a structural based generator that is in agreement with the recommendation by Tangmunarunkit et al. (2002) structural generators "are better choices for small-scale simulation studies". In this research study, hubs are identified in the network not simply because they have a degree but because their degree is high relative to their neighbours. In the case of networks which do not have the power-law degree distribution seen in the preferential attachment, this relative measure is suggested to be more appropriate to better characterise the hub in the local areas of the network.

12.3 Limitations of RoMoGA methodology

The limitations identified with respect to the proposed methodology are discussed in the following.

12.3.1 Expertise of user

The proposed methodology is intended to be used during the initial system design stages when a low-level of information is available. One limitation that could be claimed is that the user of the methodology is preferably an expert or someone with a good level of knowledge of the technical system. This is needed to define the inputs well and appropriately interpret the outputs. The definition of inputs, specifically the definition of source and sink components corresponding to different sub-functions, is a critical part of the methodology. A wrong definition of input will provide the wrong outputs. The combinatory disruptions scenario that has been included in the methodology also functions to verify the inputs and test that the results generated are not irrational. Given that technical systems have several technical constraints, regulatory requirements and additional lifecycle requirements, the user must be able to appreciate and comprehensively evaluate the computational generated solutions. These are the reasons why the proposed methodology includes MATLAB automation at the analysis stage and a predominantly manual evaluation stage, which is essentially created to allow expert input and understanding to be included in the methodology.

12.3.2 Spatial viewpoint

The methodology does not explicitly include a spatial or physical layout, and essentially considers a system to be a network. It is worth noting here that some thought was given to including more physical or spatial details in the network representation, for example by weighting the edges. However, the decision was made to avoid creating a network as an engineering construct, and the approach was instead to treat the engineering system as a simple network and based on that representation develop the methodology. The reason for this is that the methodology aims to aid at an initial stage of decision making before spatial or physical constraints are considered. This approach provides the freedom to think before constraints are considered; however, this could be considered a limitation, as maybe some generated results might not realistically be spatial possible.

12.3.3 Dynamic investigation

The methodology addresses the robustness of the system, immediately after the disruption, and therefore the limitation could be that it does not address robustness from a dynamic point of view. The robustness issue examined in this study relates to the immediate event following the disruption and does not include dynamic robustness aspects related to the spread and propagation of error. This is appropriate for the application, since immediate damage can have catastrophic results, but can be referred to as a limitation because it limits the applicability of the analysis.

Dynamically also the modularity effects on robustness are not considered, entailing how the clustering of components affects the propagation of the disruption. Thus, a limitation of the research is that it does not investigate how modularity supports the system's robustness dynamically. The independence of modules means that they can fail without propagating the effects to the whole system, thus reducing the possibility of a spread of disruptions. Another limitation is that other factors that influence robustness such as the style of the pattern and the complexity of the architecture were not investigated dynamically.

12.3.4 System boundaries

The system boundaries influence the completeness, and rationality of results. For example, in the process of identification of modules, it was noted that the fact that there were missing consumers components in the level of sinks, lead to some not technically feasible of allocation of components into modules. For a complex engineering system to holistically assess the architecture it requires to set broad system boundaries to be able to capture the whole otherwise, the proposed methodology will only assess partially the system. However, setting wider system boundaries will require additional work and more inputs which may make the methodology more difficult to be used. Caution on the system boundaries is suggested, to ensure that a sufficiently complete representation of the system is used as input in the methodology.

12.4 Recommendations for future work

The previous section of Chapter 12 presents the key limitations of RoMoGA. Following paragraphs outline recommendations for future work.

12.4.1 Complexity metrics

The complexity metrics existing in the engineering literature could also be incorporated in the proposed methodology, to additionally evaluate the level of structural complexity in relation to redundancy, modularity, and robustness. Complexity relates to redundancy, modularity and robustness; however, in this research, it was not directly considered and quantified. In the focus group discussion, the need to quantify complexity was mentioned, and a discussion regarding complexity was raised at different points of the research. Complexity can be a perceptual attribute or an objective. There are in the literature complexity metrics are proposed (Sinha, 2014), and future work will include more consideration on the complexity and its effects.

12.4.2 Additional disruption scenarios

In additional future recommendations on the evolution of the methodology, more disruption scenarios will be included, which will widen the robustness assessment. Additional combinatory scenarios could be combined to disrupt inter-module connections with other components or combined to disrupt robust modules with individual connection or combination. Such disruptions could further contribute to assessing robustness at the early stages of design. A probabilistic approach to disruption could also be included in future research to address more uncertain future scenarios of operation; however, it was avoided herein to keep the disruption generic and principally deterministic.

12.4.3 Weighted robustness metric for reconfiguration

The distributed engineering system architectures are typically reconfigurable that means a different style of configuration will have different robustness under a different type of disruption. A possible future work is to automate the calculation of the robustness under different system architecture configuration styles and calculated a weighted version of the robustness metric given the importance (critically or frequency) of a configuration state to appear during the operational lifecycle of the system. In the research, the robustness was calculated in the case studies only under normal operation (cruising state configuration). For example, for the naval engineering systems studied there are two other configuration states (survive and recovery) that the system architecture will be differently connected (components and connections) which will lead to having different modularity and robustness. Future work could update the methodology to be able to study various configuration states of the systems and developed accumulated metrics that condense the various configuration states into a weighted robustness and modularity metric. The weighted version of the robustness metric could be calculated to correspond to different operational performance requirement. The case studies presented relate with the operational condition of the systems at cruising which the ship spends most of its life being in peace operations. However, the ship or other systems through their life are expected to operate in different performance limits. Thus, the robustness metric could be calculated for each respective operational performance requirement a weighted total could be calculated given the importance or the percentage that the system is expected to be required to operate in such conditions.

12.4.4 Evaluation in different technical contexts

Due to the research being applied and evaluated within the context of naval distributed engineering systems, further research into different technical contexts, systems and companies would benefit and contribute to allowing a broader generalisation of findings and the validation of the proposed methodology. Because of the time limitation of a PhD study and the lack of access to different technical systems and companies, this thesis does not presently work on different systems and contexts. Even though the methodology aims to address generic engineering systems and not specifically naval engineering systems, more investigation into different types of systems has not been managed to be included in the thesis. Therefore, the general nature of the proposed methodology remains to be further validated. Future research will aim to include applications for evaluation and validation in different technical system contexts.

Chapter 13: Conclusion

The defined research aim of the study was to develop a methodology to support the design of robust modular system architectures. The aim was achieved by developing the RoMoGA methodology that is composed of three stages: generation, analysis, and evaluation.

The overview of RoMoGA is provided in Chapter 5, while the descriptive implementation of RoMoGA is detailed in Chapter 6 and the explorative implementation of the RoMoGA is outlined in Chapter 7. RoMoGA enables generation, and robustness assessment of system architecture options, of varying redundancy and modularity based on the input nominal system architecture. Figure 79 illustrates the proposed RoMoGA methodology.

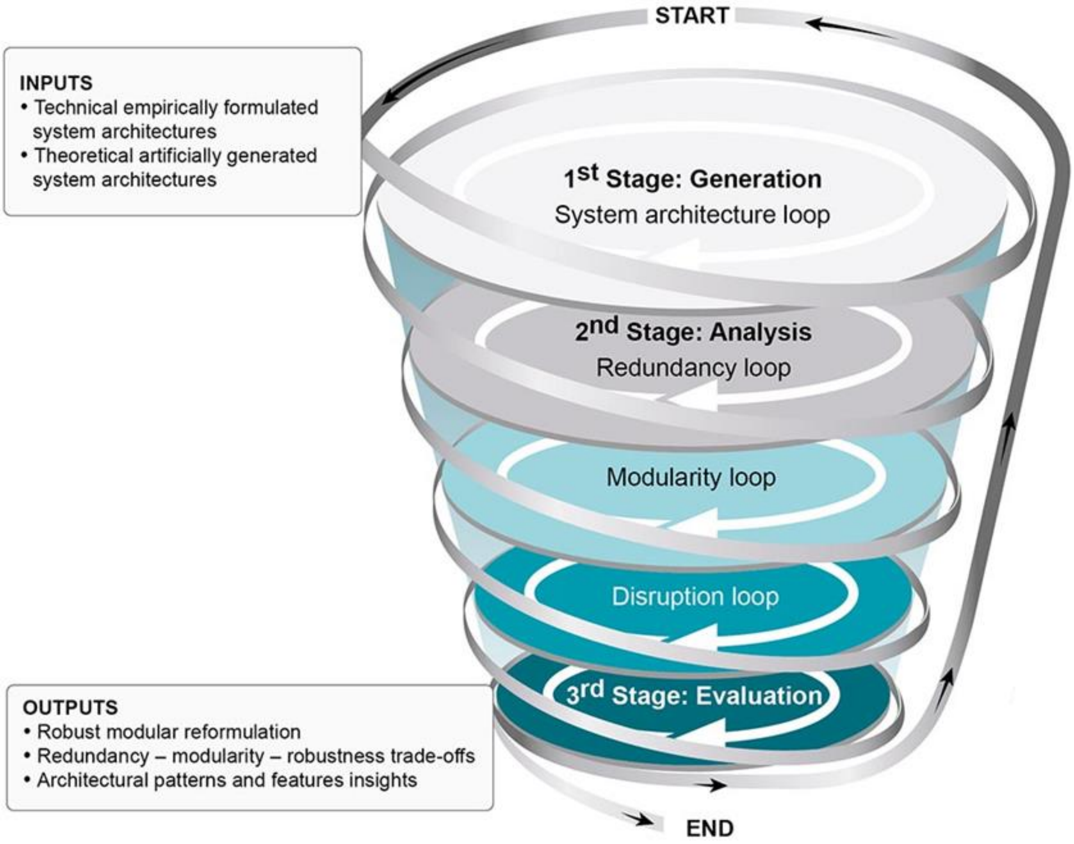


Figure 79: RoMoGA methodology illustration

13.1 Novel contribution to the knowledge

The main knowledge contribution of the study is the new, validated method of system architecting introduced in Chapters 5, 6 and 7. The method contributes to knowledge through the creation of a new robustness metric, a new network generation tool, and a new methodological approach that facilitates the integrated study of two desirable attributes of system architecture: robustness and modularity. Earlier engineering field research has either concentrated on modularity or robustness, without taking into account redundancy, specific characteristics and challenges of engineering systems, failing to establish a systematic approach to support the design of robust modular system architecture in the field of engineering design.

The study enhances knowledge through the convergence of these two research streams, by systematically combined the literature streams of modularity and robustness in the engineering design research area. This contributes by moving away from the isolation study of system architecture attributes, focusing on the development of a method that is specific for the engineering systems, systematic and quantitative, and that respects the complexity and uniqueness of the engineering systems.

The method developed is new, as it is the first method to combine quantitative generation, analysis and quantification of system robustness and modularity in engineering systems. The method generic in nature and allows it to be applied in different engineering disciplines, such as aerospace, nuclear, electronics or software engineering. The method is applicable to the different domains of distributed engineering systems, i.e. systems dictated by their configuration of source and sink components, structured in such a way as to provide a specific set of functions.

The research study focuses on function, behaviour and structure and, accordingly, robustness, redundancy and modularity by proposing that the function be satisfied if the source and sink remain sufficiently connected after disruption. Previous applications of network science-based methods in engineering design did not incorporate the concept of source and sink in the quantification of robustness in system architectures.

In summary, the novel contribution to knowledge have been achieved in the study reported in this thesis by:

- Introducing a quantitative robustness assessment of system architectures at the initial design stage by developing a **new metric** that can be calculated based on the sufficiency of connectivity between the source and the sink components and a redundancy threshold criterion. The novel metric is tailored for the engineering systems, moving away from robustness metrics discussed in the extended network science literature, that do not consider connectivity between specific nodes and redundancy within the network. The novel robustness metric can be calculated without the need for detailed information of components which are typically not available at very early design stages. The metric is computationally efficient, allowing a large number of system architecture options to be quickly assessed in terms of their potential robustness, which contributes to enlarging the generation and analysis of alternative system architecture options during initial design stages. The research study novel contribution to the knowledge is the introduction of the first robustness metric that is based on the source and sink concept in the engineering design literature.

- Introducing a new generation system architecture method by developing a **new network tool** that generates novel system architecture options that can be assessed concerning robustness and modularity. This is the first network science structural pattern generation approach proposed in engineering literature, moving the field away from simple pattern identification, offering the ability to mix patterns to create novel solutions. The tool incorporates structural patterns and differentiates nodes into sources, sinks and hubs. The generated networks have engineering relevant characteristics and can be subsequently simultaneously analysis concerning robustness and modularity. The research study novel contribution to the knowledge is the introduction of network science structural-based method to generate of system architecture options during the early stages of design.
- Extending the modular design literature, by adopting a **multi-resolution method** to modularity assessment, moving away from modularity-based maximisation methods that are traditional in modular design literature. A non-maximisation-based approach to modularity may be particularly useful for industrial contexts dealing with engineering to order systems, that are unique and small in scale, and are not mass-produced, extending modularity implementation beyond the typical industries discussed in the literature. The research study novel contribution to the knowledge is the introduction of a multi-resolution modular approach to the engineering design literature.

13.2 Research objectives achievement

The research aim reported in Section 1.3 was achieved by fulfilling the research objectives outlined in Section 1.4. Table 50 summarises the research methods used to achieve the objectives and where the fulfilment of research objectives is presented in the thesis. Research objectives achievement is discussed in the following paragraphs.

Objective 1

Define an approach to the representation, generation, and analyses of system architectures.

Objective 1 was answered based on the literature review on DSM and network-based modelling and analysis presented in Section 4.2.5 and Network-basis generation methods in Section 4.2.6. The findings in the literature justify the definition of the network science as the approach to represent, generate and analysis system architectures. The use of network science is suggested in the design science literature (Chen et al., 2018). There is a growing body of literature implementing network and graph theory in product and system architectures (Baldwin et al., 2014; Braha and Bar-Yam, 2007, 2004a; Parraguez et al., 2015; Piccolo et al., 2018; Sarkar et al., 2014; Sosa et al., 2007).

Network representation also relates to the popular in engineering design literature DSM (Browning, 2015) and has a strong mathematical basis (Estrada and Knight, 2015) that offers established knowledge which is transferable from mathematics to engineering systems. Crawley et al. (2004) defined system architecture as “an abstract description of the entities of a system and the relationships between those entities”, though in most instances system architecture can be abstractly depicted as a DSM: that is, equivalent to a network representation. By representing complex systems as networks and using the adjacency matrix, metrics can be calculated that quantify distinct attributes of the network topology.

A network science approach offers a strong mathematical background to help on the development of a computational analysis stage of the methodology. That became apparent in this study, as the network science field offers a wide range of existing knowledge, tools and computational codes that have been critical in developing the proposed methodology. An additional reason for selecting network modelling was that it allows a high-level presentation of system architectures, thus appropriate during the initial stages for design when there is a lack of detailed technical information.

Network science has been used to develop the analysis stage of the RoMoGA methodology in a novel way. Previous studies in the engineering design field (Braha and Bar-Yam, 2004a; Piccolo et al., 2018) exploited popular statistical-based networks science metrics such as degree distribution, average path length or clustering coefficient. This study has not followed this route and has designed bespoke methods for the particular class of engineering-related networks of interest. This study has used the mathematical framework of network science to develop an engineering-related robustness metric that captures the direct and indirect connectivity between source and sink components. Moreover, the study has exploited several conceptual ideas from network science (such as diameter, eccentricity, radius) to generate engineering analogies (central component and module) and to devise disruption strategies.

Besides, network science provided the basis for the development of a network tool that generates networks with identifiable engineering characteristics, such as hubs and patterns that can be assessed in terms of robustness and modularity. Section 4.2.6 discussed the methods of network generation found in the engineering system and the wider area of network science research literature. The findings in the literature have motivated a network approach to support the generation of options for system architecture in the field of engineering systems. The network generation tool proposed in this study is tailored to engineering systems, incorporating a structural and hub-based approach on the basis that the hub is a component with a high degree of connectivity relative to its neighbours. In conclusion objective 1 is achieved by employing network science as the study's approach for the representation, generation and analyses of system architectures.

Table 50: Overview of research objectives

Research Objectives		**Research methods**	**Thesis Chapter**
O1	Define an approach to the representation, generation and analyses of system architectures.	➢ Literature review.	➢ Section 4.2.5 DSM and network-based modelling and analysis. ➢ Section 4.2.6 Network-based generation methods
O2	Develop a methodology to combine the assessment of modularity and robustness in system architecture.	➢ Conceptual, mathematical, and computational modelling.	➢ Section 5.2 Analysis stage. ➢ Section 6.3 Analysis stage. ➢ Section 7.3 Analysis stage.
O3	Identify a method for quantifying the level of modularity in the system architecture.	➢ Literature review.	➢ Section 6.3.1 Modularity loop. ➢ Section 7.3.1 Modularity assessment
O4	Identify a method for quantifying the level of robustness in the system architecture.	➢ Conceptual, mathematical, and computational modelling.	➢ Section 6.3.2 Disruption loop. ➢ Section 6.3.3 Robustness evaluation metric. ➢ Section 7.3.2 Robustness assessment
O5	Develop a method that generates alternative system architectures.	➢ Conceptual, mathematical, and computational modelling.	➢ Section 7.2 Generation stage
O6	Apply the methodology in technical and theoretical system architectures.	➢ Case studies. ➢ Experiments.	➢ Chapter 8 Industrial application - case studies. ➢ Chapter 9 Explorative application – experiments.
O7	Evaluate the methodology in an industrial context.	➢ Interviews. ➢ Industrial design practise.	➢ Chapter 10 Industrial evaluation – interviews ➢ Chapter 11 Industrial evaluation - design practices.

Objective 2

Develop a methodology to combine the assessment of modularity and robustness in system architecture.

The research objective 2 is achieved by the proposed analysis stage of the RoMoGA methodology summarised in Section 5.2 which outlines the combined assessment of the modularity and robustness of the system architecture options generated. The analysis stage of the RoMoGA methodology is discussed in detail in Section 6.3 which outlines the descriptive implementation and Section 7.3 which describes the explorative implementation.

The descriptive RoMoGA analysis stage, as discussed in Section 6.3, consists of modularity and robustness assessment. For a given level of redundancy in the system architecture, a modularity loop is designed to assess the potential level of modularity by finding different modular configurations. Then a loop of disruptions occurs, which uses the different modular configuration found, as input. Modules corresponding to the different configurations are disrupted and the robustness of the architecture is determined for a given redundancy. The analysis stage consisting of these nested loops enable the combined analysis of modularity and robustness. The results of the analysis phases feed into the evaluation phase, which involves the reformulation of an appropriate robust modular configuration and a trade-off examination of the effects of modularity and redundancy on robustness.

Explorative RoMoGA uses a network tool to populate system architecture options and incorporates modularity and robustness metrics and methods at the analysis stage to assess these options. The classification of the nodes as source, sink and hub and the directionality at the hub level, which is included in the network tool, makes it possible to assess the robustness of the network, enabling a combination of robustness and modularity analysis (Section 7.3). Variation of generator parameters — structural patterns and topological features (e.g. hubs)— leads to a change in the level of redundancy and modularity of the system architecture options. For example, the variation in the number of hubs or the redundancy threshold criterion is a way to vary the redundancy. An approach to controlling the degree of modularity of the system architecture is also to select a structural pattern that influences modularity (e.g. integral modularity reduction or modular bus increase). The use of a network tool that enables robustness analysis and the incorporation of modularity and robustness metrics and methods in the Explorative RoMoGA enables the combined analysis of redundancy and modularity effects on robustness.

Objective 3

Identify a method for quantifying the level of modularity in the system architecture.

Objective 3 of research is accomplished through the proposed modularity assessment which is included in the RoMoGA methodology analysis stage outlined in Section 5.2.1. Section 6.3.1 discusses the modularity assessment proposed for the descriptive implementation and Section 7.3.1 explains the explorative implementation. The modularity assessment stages include the method and a metric. In the descriptive and explorative implementation of RoMoGA the same normalised modularity metric (Magee et al., 2010) is used that was identified in the literature because it allows comparisons between the levels of modularity amongst system architectures of different sizes. The normalised modularity metric is incorporated in Section 6.3.1 & 7.3.1.

The modularity method chosen for the descriptive implementation of RoMoGA in Section 6.3.1 is the stability method by Delvenne et al. (2008) that is computationally implemented by Martelot and Hankin (2013). This community detection method that includes a resolution parameter allows the generation of potential modules corresponding to different modularity levels. This was not an approach previously used in the field of engineering design and was selected as appropriate in this study to be implemented in the descriptive RoMoGA.

In addition, the research objective 3 is fulfilled in Section 7.3.1 of the explorative RoMoGA, which uses the modularity method-Louvain community detection method (Blondel et al., 2008) which allows for the characterisation of the modularity of the simulated network. In the explorative RoMoGA, there is no need to identify in detail the potential modular configuration, as the assessment of modularity is at a high abstract level. The Louvain Community Detection Method was therefore found to be appropriate, as was previously established in engineering design literature. (Parraguez et al., 2019; Sinha et al., 2019).

Objective 4

Identify a method for quantifying the level of robustness in the system architecture.

The research objective 4 is accomplished in the robustness assessment of the proposed RoMoGA methodology (Section 5.2.2) by the development a novel robustness metric (Section 6.3.3) and different types of disruptions (Sections 6.3.2 & 7.3.2). A novel robustness metric was developed in Section 6.3.3 which was a network science and engineering system literature tailored for distributed engineering systems. The robustness metric is capable to assess distributed engineering systems i.e. dictated by their configuration of source and sink components, structured in a way to deliver a particular functionality. The robustness metric was formulated based on the mathematical expression of the exponential matrix that captures direct and indirect connectivity (any path) amongst components and classifies nodes to sources and sinks. The robustness metric asks the question of whether a function is satisfactory after the disruption. The function is defined to be satisfactory if there is sufficient connectivity amongst predefined source and sink nodes. Therefore, the robustness metric assesses after a disruption whether sink components remain with a sufficient level of direct or indirect connectivity with corresponding source components.

For the descriptive and explorative implementation of RoMoGA, different types of disruptions have been developed because the descriptive implementation concerns established technical systems, while the explorative implementation involves simulated networks. For this reason, the descriptive implementation proposed deterministic approaches that could provide results that could be verifiable by the user of the expert. The proposed types of disruptions to the descriptive RoMoGA are combined components and module-based disruptions that are defined to apply to technical systems (Section 6.3.2). Explorative RoMoGA defines a target and random disruptions for simulated networks (Section 7.3.2) that are analogous to other research works discussed in the literature (Braha and Bar-Yam, 2007; Piccolo et al., 2018). The novelty of the disruption approach is that it uses the network science concept of eccentricity to establish the central module and nodes (i.e. the lowest eccentricity nodes or modules) to the device to target priority disruptions.

Objective 5

Develop a method that generates alternatives system architectures.

The RoMoGA Methodology (Chapter 5) is designed to be generative, meaning that the user can populate a variety of different system architecture options. In the descriptive implementation of chapter 6.2.2 of RoMoGA, the generation of alternative system architecture options takes place at the redundancy loop. The generation of alternative system architecture options in the descriptive RoMoGA maybe not as explorative, as is bound by the characteristics of the input nominal architecture and user' previous experiences. This study develops a novel network tool in Section 7. 2 to answer Objective 5.

A network tool has been developed as part of the RoMoGA methodology to broaden the exploration and analysis of possible system architecture design. The network tool offers explorative capabilities that allow the user to change the patterns and parameters of the hubs in order to investigate their effects on robustness and modularity. The main parameters of the generator enable the selection of a theoretical pattern at the main structure and at the hub level, based on a set of options (modular bus, hierarchical, path, cyclical, integral). Another important parameter of the generator is its ability to parameterise the hubs; therefore, the number, density, pattern, and connectivity between the hubs could be changed. Another parameter is that components fed or demanding from a hub could be identified as sources or sinks; therefore, the system architecture options generated can be assessed in terms of their robustness. In this way, the user of the methodology can resourcefully use the network tool to develop system architecture options based on nominal system architecture. The network tool may populate and evaluate a network simulation that may counteract an expert's intuition or previous knowledge.

Objective 6

Apply the methodology in technical and theoretical system architectures.

The developed RoMoGA methodology has been applied to industrially and exploratively. The Chapter 6 descriptive RoMoGA was applied to the industrial case studies (three naval distributed engineering systems) in Chapter 8 and the application and results were assessed by SMEs at BAE Systems. In the first place, the robustness results of the component combined disruptions were used as a means of verifying the realism of the results and ensuring that the calculations were correct. This was possible because the combination of the two components disrupting the systems was well understood by the expert and the established technical documentation was available to provide a basis for a comparison between the calculated robustness of the methodology and the actual results. In addition, Type A, B and C robust modular configuration results were presented to experts for assessment and were also compared with existing solutions. The experts had verified that the results were rational, and that could be particularly useful if they were available at the initial stage of the design before other constraints were fixed. The RoMoGA outcomes were considered appropriate when considering potential options for system architecture during discussions with stakeholders at an initial stage of design.

The explorative RoMoGA described in Chapter 7 was applied through experiments in Chapter 9. The experiments focus on the cyclical main structure pattern, as it was found that the technical systems (nominal system architectures) presented in the case studies (Chapter 8) had a cyclical topology. The experimental setup is designed to vary the number of hubs, the number of sources or sinks (the density of hubs), the connectivity between hubs, the type of pattern at the level of the hub, and the redundancy threshold criterion (connectivity between source and sink). Specifically, the experimental results of Chapter 9 highlighted the importance of carefully selecting the right number of hubs in the design of a system architecture given the expected operating environment during the design process.

It was found that an increase in the number of hubs could be beneficial for small-scale networks (40-80 nodes) that do not exhibit power-law distribution, given that large simultaneous disruption is not an expected operating scenario. In addition, the findings of the experiments indicate that increasing connectivity between the hub is to be avoided when target disruption is expected and that increasing redundant connectivity between the source and the sink does not provide a linear benefit to robustness. Based on the findings of the explorative application (Chapter 9), a redesign of the technical system A was developed in Section 9.4, which was verified by the experts as a rational and feasible solution.

Objective 7

Evaluate the methodology in an industrial context.

The evaluation of the methodology in the industrial context was carried out through semi-structured interviews (Chapter 10) and industrial design practice (Chapter 11).

The industrial evaluation was carried out through semi-structured interviews (Chapter 10) with experts from different disciplines (production, electrical engineering, forward design engineering, system safety, system engineering and supply chain) of BAE Systems. The interview questions were designed to assess the usefulness, appropriateness, and applicability of the proposed methodology. The findings of the interviews indicated that the methodology was appropriate to act as a foundation before a detailed analysis was carried out, as the network representation was able to capture the system more holistically than the current tools. The disadvantage of the methodology was that it was not able to capture technical details, such as the calculation of electrical power or fluid flow, and therefore it was not appropriate to provide detailed information on the performance of the system. In addition, the interviews indicated that the system engineering department of the company has design processes in place to model systems in the SysML language, which enables the development of structural diagrams that can generate automatic matrix representation of systems. This finding reinforces the applicability of the proposed methodology to current engineering design practices.

In addition, the industrial design practice (Chapter 11) provides the industrial evaluation of the RoMoGA methodology. It has been developed to evaluate how the results of the RoMoGA methodology can be used in the industrial context using domain-specific design software. The results of the methodology were used by a Naval Architect and Research and Technology Engineering Manager to develop updated designs of the original type A and B naval distributed systems. The results of case studies (Chapter 8) informed the technical updates that were simulated in vulnerability software (SURVIVE) in Chapter 11. The proposed methodology was able to identify robust modules prior to the use of detailed design software that is time-consuming to be used, required detailed information and is expensive to purchase by companies. Moreover, industrial design practice found that the fact that the methodology could be used at an early stage was critical because it could influence decision-making and provide objective arguments to support discussions at an architectural level, counteracting the opinions of domain experts who may not be able to view the system as a whole, at a high-level of abstraction.

13.3 Practical and theoretical implications

Potential practical implications are that the robust modular configuration results of the descriptive implementation of RoMoGA could be used to develop a library of robust modules for a given class of technical systems. The results of the case studies in Chapter 8 and Appendix II of the three technical systems which share the same functions led to the identification of robust modules which share similarities between different levels of redundancy and modularity. The application of RoMoGA could allow the user to find a set of robust module choices that can be assessed in terms of cost, obsolescence, involvement of the supplier, potential for technological improvement, time and ease of manufacture and assembly. In this way, these robust module options can then be classified and informed about the development of a library that could be reused in different system class evolutions.

The practical implication of the explorative implementation of the RoMoGA methodology is that it could help practitioners to decide on novel designs that could counteract traditional thinking, allowing for more open-minded exploration of design. The generator can help to demonstrate new system-architectural options which are not yet available for engineering solutions. In this way, potential beneficial system architecture designs are identified and, potential appropriate functional solutions that satisfy the option of system architecture are suggested. Moreover, the practical implications of the network tool are that it can populate a large number of simulated networks that can be used as inputs for optimisation or other analytical approaches.

In practice, the results of the methodology may function as objective evidence to allow visibility and traceability in the decision-making process during the initial stages of the design process. This is important considering the contractual and legal responsibilities of complex engineering companies and projects. Such responsibilities may be linked to their funding, requiring early decision-making to be transparent and objectively justifiable to a wider audience.

The theoretical implications of the proposed methodology are to encourage the development of quantitative approaches which view key attributes of system architectures in combination. Modern engineering systems are required to exhibit a variety of key attributes that have an impact on one another, have trade-offs, and some are more important than others. Finding metrics for quantifying these attributes and using metrics together in systematic methodologies that can help design more stable system architectures. RoMoGA contributes to the literature on systematic methodologies by examining the multi-properties of systems, avoiding analyses in isolation of attributes.

This study contributes to the literature that discusses the trade-off between modularity and robustness. Another theoretical implication of the study is that increasing redundancy in modular architecture does not linearly increase robustness and that systematic methods are required to study the desired level and type of redundancy in modular system architectures.

Lastly, the theoretical implications of the network tool are the deliberate development of complex system architecture options and the knowledge of the topology that has created complexity.

13.4 Consolidated overview of the research

The study accepted the philosophical standpoint of pragmatism meaning that subjectivity may be included in the research and that the researcher used research methods pragmatically, in each case considering their capacity to help achieve the research aim and objectives. Therefore, this study has used mixed methods, both qualitative and quantitative. The mixed-method approach was considered appropriate in this study since it allows different methods to be used to collect data. Firstly, the identification of research scope for the study was achieved by combining a literature review with explorative focus group discussion.

To develop the RoMoGA methodology, the researcher combined the literature review with conceptual, mathematical, and computational modelling. The three aspects of modelling are interrelated. Mathematical and computational modelling requires prior conceptual modelling. The researcher performed the three forms of modelling in parallel and iteratively.

The evaluation part of the study used four research methods: case studies; experiments; semi-structured interviews; and industrial design practices. The case studies provide a foundation of established technical systems with known behaviours to which the proposed methodology can be applied. The case studies (Chapter 8) were critical research methods to provide evidence of how realistic and viable the proposed methodology was.

Experiments (Chapter 9) employing the proposed network tool were performed to demonstrate the explorative RoMoGA methodology. The main part of the experiments is focused on the hybrid cyclical patterns that are simulated networks, which share similarities with the technical systems that were studied in the case studies Chapter 8.

A prescriptive implementation of the RoMoGA methodology was also demonstrated in Chapter 9, that included a redesign of Type A based on the findings of the experiments. The Type A redesign was verified by the experts as a rational and feasible solution.

Additionally, semi-structured interviews and industrial design practice were employed for evaluation purposes within BAE Systems. In summary, the various research methods employed in the study offer pluralism of evidence, opinions, and viewpoints, aligned with the pragmatism philosophical assumptions accepted. The following Table 51 is formulated to offer a consolidated overview of the research study.

13.5 Chapter summary

This chapter concludes the study reported in this thesis by presenting the fulfilment of the research aim and objectives, practical and theoretical and implications and a consolidated overview, concluding the research study.

	Strengths	Limitations	Recommendations
Research Contribution			
RoMoGA methodology	➢ Combined multi-attributes ➢ Semi-automatic: computational efficient and allow manual input ➢ Descriptive, explorative and prescriptive ➢ Systematic and repeatable ➢ Design and business support tool	➢ Expert's knowledge is required ➢ Lack of spatial viewpoint ➢ Lack of dynamic robustness viewpoints ➢ System boundaries affect the results	➢ Complexity metrics ➢ Additional disruption scenario ➢ Weighted robustness metric for reconfiguration ➢ Evaluation in different technical contexts
Research Approach			
Pragmatism	➢ Less restrictive, more pluralism	➢ May be contradictory	➢ Engineering design domain to adopt a unified research philosophical standpoint
Mixed methods	➢ Use of quantitative and qualitative methods	➢ Time-consuming activity	➢ Follow a specific type of mixed methods as per literature
Research Methods			
Exploration focus group discussion	➢ Interactive and flexible discussion ➢ Appropriate for exploration ➢ Efficient on gaining opinions in short time	➢ A limited number of participants, and a single focus group discussion ➢ The discussion was diverted in specific company-related issues	➢ Follow-ups focus group discussions ➢ More experienced facilitator
Literature review	➢ Broad and various literature streams	➢ Lack of a systematic approach	➢ Systematic literature review
Conceptual, mathematical and computational modelling	➢ Abstract, time and computationally efficient modelling	➢ High-level of representation of reality, lack of details and technical specifics	➢ Improve the computational modelling into a user-friendly software tool
Case studies	➢ In-depth knowledge of the technical systems and their expected behaviour	➢ A single company, industrial context and technical system (naval ship systems)	➢ Investigate different technical systems design
Experiments	➢ Control over the characteristics of patterns and parameters of hubs	➢ Could generate artificial and non-realistic system architecture options	➢ Tailor the experiments into the context of the technical systems
Semi-structured interviews	➢ Freedom and flexibility to gain input in a semi-structured approach	➢ Time limitation of the participants ➢ Personal bias and subjectivity of opinions	➢ Perform multi-companies and different industrial context, and technical systems
Industrial design practices	➢ A practical instance of design providing insights into the reality of the ways that the proposed methodology could be implemented in the industry	➢ Depends on the knowledge of engineer involved ➢ Time-consuming process ➢ Accepted assumptions	➢ Perform industrial design practices with additional technical specific software and cross-check the results

Table 51: Consolidated research overview

References

Adams, K.M., 2015. Non-functional Requirements in Systems Analysis and Design, Topics in Safety, Risk, Reliability and QualityDesign Methodologies, in: Design Methodologies. pp. 15–43. https://doi.org/10.1007/978-3-319-18344-2

Agarwal, S., Pape, L.E., Dagli, C.H., 2014. A hybrid genetic algorithm and particle swarm optimization with Type-2 Fuzzy sets for generating systems of systems architectures, in: Procedia Computer Science. pp. 57–64. https://doi.org/10.1016/j.procs.2014.09.037

Agogué, M., Poirel, N., Pineau, A., Houdé, O., Cassotti, M., 2014. The impact of age and training on creativity: A design-theory approach to study fixation effects. Think. Ski. Creat. 11, 33–41. https://doi.org/10.1016/j.tsc.2013.10.002

Albert, R., Barabási, A.L., 2002. Statistical mechanics of complex networks. Rev. Mod. Phys. 74, 47–97. https://doi.org/10.1103/RevModPhys.74.47

Albert, R., Jeong, H., A, 2000. -L. Barabási, Error and attack tolerance of complex networks. Nature 406, 378–382.

Alexander, C., 1964. Notes on the Synthesis of Form. Harvard University Press.

AlGeddawy, T., ElMaraghy, H., 2013. Optimum granularity level of modular product design architecture. CIRP Ann. - Manuf. Technol. 62, 151–154. https://doi.org/10.1016/j.cirp.2013.03.118

Allen, K.., Carlson-Skalak, S.C., 1998. Defining product architecture during conceptual design, in: In 1998 ASME Design Engineering Technical Conference, Atlanta, GA.

Andersson, P., 1997. On robust design in the conceptual design phase: A qualitative approach. J. Eng. Des. 8, 75–89. https://doi.org/10.1080/09544829708907953

Andersson, P., 1996. A Semi-Analytic Approach to Robust Design in the Conceptual Design Phase. Res. Eng. Des. 8, 229–239.

Anvarifar, F., Zecenbergen, C., Thissen, W., 2015. Comparing flexibility and robustness for flood defence development planning under uncertainty, in: 36th IAHR World Congres. pp. 246–262. https://doi.org/10.1002/sys.20098

Arabian-Hoseynabadi, H., Oraee, H., Tavner, P.J., 2010. Failure Modes and Effects Analysis (FMEA) for wind turbines. Int. J. Electr. Power Energy Syst. 32, 817–824. https://doi.org/10.1016/j.ijepes.2010.01.019

Arenas, A., Fernández, A., Gómez, S., 2008. Analysis of the structure of complex networks at different resolution levels. New J. Phys. 10, 053039. https://doi.org/10.1088/1367-2630/10/5/053039

Arianos, S., Bompard, E., Carbone, A., Xue, F., 2009. Power grid vulnerability: A complex network approach. Chaos 19. https://doi.org/10.1063/1.3077229

Baldwin, C., Maccormack, A., Rusnak, J., 2014. Hidden structure: Using network methods to map system architecture. Res. Policy 43, 1381–1397. https://doi.org/10.1016/j.respol.2014.05.004

Baldwin, C.Y., Clark, K.B., 2000. Design Rules: Volume 1. The Power of Modularity. The MIT Press, Cambridge, Massachusetts, London, England.

Baldwin, C.Y., Clark, K.B., 1997. A Guide to Modularity. Harv. Bus. Rev. 84–93.

Bar-Yam, Y., 2003. When systems engineering fails - Toward complex systems engineering. Proc. IEEE Int. Conf. Syst. Man Cybern. 2, 2021–2028.

Bayrak, A.E., Collopy, A.X., Papalambros, P.Y., Epureanu, B.I., 2018. Multiobjective optimization of modular design concepts for a collection of interacting systems. Struct. Multidiscip. Optim. 57, 83–94. https://doi.org/10.1007/s00158-017-1872-4

Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp. 2008, P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008

Bompard, E., Wu, D., Xue, F., 2011. Structural vulnerability of power systems: A topological approach. Electr. Power Syst. Res. 81, 1334–1340. https://doi.org/10.1016/j.epsr.2011.01.021

Bonvoisin, J., Halstenberg, F., Buchert, T., Stark, R., 2016. A systematic literature review on modular product design. J. Eng. Des. 4828, 1–27. https://doi.org/10.1080/09544828.2016.1166482

Borjesson, F., Hölttä-Otto, K., 2014. A module generation algorithm for product architecture based on component interactions and strategic drivers. Res. Eng. Des. 25, 31–51. https://doi.org/10.1007/s00163-013-0164-2

Braha, D., 2016. The Complexity of Design Networks: Structure and Dynamics. pp. 129–150. https://doi.org/10.1007/978-3-319-33781-4

Braha, D., Bar-Yam, Y., 2007. The Statistical Mechanics of Complex Product Development: Empirical and Analytical Results. Manage. Sci. 53, 1127–1145. https://doi.org/10.1287/mnsc.1060.0617

Braha, D., Bar-Yam, Y., 2004a. Topology of large-scale engineering problem-solving networks. Phys. Rev. E 69, 016113. https://doi.org/10.1103/PhysRevE.69.016113

Braha, D., Bar-Yam, Y., 2004b. Information Flow Structure in Large-Scale Product Development Organizational Networks. J. Inf. Technol. 19, 244–253. https://doi.org/10.1057/palgrave.jit.2000030

Brefort, D., Shields, C., Habben Jansen, A., Duchateau, E., Pawling, R., Droste, K., Jasper, T., Sypniewski, M., Goodrum, C., Parsons, M.A., Kara, M.Y., Roth, M., Singer, D.J., Andrews, D., Hopman, H., Brown, A., Kana, A.A., 2018. An architectural framework for distributed naval ship systems. Ocean Eng. 147, 375–385. https://doi.org/10.1016/j.oceaneng.2017.10.028

Browning, T., 2015. Design Structure Matrix Extensions and Innovations : A Survey and New Opportunities. IEEE Int. Eng. Manag. Conf. 63, 27–52. https://doi.org/10.1109/TEM.2015.2491283

Buede, D., 2000. The Engineering Design of Systems: Models and Methods. John Wiley & Sons.

Bulleit, W.M., 2017. Pragmatism and Engineering. pp. 13–22. https://doi.org/10.1007/978-3-319-45193-0_2

Cardoso, C., Badke-Schaub, P., 2011. The influence of different pictorial representations during idea generation. J. Creat. Behav. 45, 130–146. https://doi.org/10.1002/j.2162-6057.2011.tb01092.x

Carlson, J.M., Doyle, J., 2002. Complexity and robustness. Proc. Natl. Acad. Sci. U. S. A. 99, 2538–2545. https://doi.org/10.1073/pnas.012582499

Carlson, J.M., Doyle, J., 1999. Highly optimized tolerance: A mechanism for power laws in designed systems. Phys. Rev. E - Stat. Physics, Plasmas, Fluids, Relat. Interdiscip. Top. 60, 1412–1427. https://doi.org/10.1103/PhysRevE.60.1412

Çetinkaya, E.K., Broyles, D., Dandekar, A., Srinivasan, S., Sterbenz, J.P.G., 2011. Modelling communication network challenges for Future Internet resilience, survivability, and disruption tolerance: a simulation-based approach. Telecommun. Syst. 52, 751–766. https://doi.org/10.1007/s11235-011-9575-4

Chakrabarti, D., Faloutsos, C., 2006. Graph Mining: Laws, Generators, and Algorithms. ACM Comput. Surv. 38, 2. https://doi.org/10.1145/1132952.1132954

Chalfant, J., Chryssostomidis, C., 2017. Application of Templates to Early-Stage Ship Design 111–117.

Chalfant, J.S., Chryssostomidis, C., 2011. Analysis of various all-electric-ship electrical distribution system topologies, in: 2011 IEEE Electric Ship Technologies Symposium, ESTS 2011. pp. 72–77. https://doi.org/10.1109/ESTS.2011.5770844

Chan, H., Akoglu, L., Tong, H., 2014. Make It or Break It: Manipulating Robustness in Large Networks. Proc. 2014 {SIAM} Int. Conf. Data Mining, Philadelphia, Pennsylvania, USA, April 24-26, 2014 325–333. https://doi.org/10.1137/1.9781611973440.37

Chang, T.S., 1996. Conceptual robustness in distributed concurrent engineering and design-in-modularity. University of Michigan.

Chao, M.T., Fu, J.C., Koutras, M. V., 1995. Survey of Reliability Studies of Consecutive-k-out-of-n:F & Related Systems. IEEE Trans. Reliab. 44, 120–127. https://doi.org/10.1109/24.376531

Chen, C.-C., Crilly, N., 2016. From Modularity To Emergence.

Chen, C.C., Crilly, N., 2014. Modularity, redundancy and degeneracy: Cross-domain perspectives on key design principles, in: 8th Annual IEEE International Systems Conference, SysCon 2014 - Proceedings. IEEE, pp. 546–553. https://doi.org/10.1109/SysCon.2014.6819309

Chen, L., Kou, Y., Li, Z., Xu, A., Wu, C., 2018. Empirical research on complex networks modeling of combat SoS based on data from real war-game, Part I: Statistical characteristics. Phys. A Stat. Mech. its Appl. 490, 754–773. https://doi.org/10.1016/j.physa.2017.08.102

Chen, W., Heydari, B., Maier, A.M., Panchal, J.H., 2018. Network-based Modeling and Analysis in Design. Des. Sci. https://doi.org/10.1017/dsj.2018.8

Chiriac, N., Hölttä-Otto, K., Lysy, D., Suh, E.S., 2011a. Three approaches to complex system decomposition, in: Invest on Visualization - Proceedings of the 13th International DSM Conference. pp. 3–17.

Chiriac, N., Hölttä-Otto, K., Lysy, D., Suk Suh, E., 2011b. Level of Modularity and Different Levels of System Granularity. J. Mech. Des. 133, 101007. https://doi.org/10.1115/1.4005069

Cooper, H., 1989. Integrating research: A guide for literature reviews.

Crawley, E., Cameron, B., Selva, D., 2015. System architecture: Strategy and product development for complex systems.

Crawley, E., Week, O.L. De, Eppinger, S.D., Magee, C., Moses, J., Seering, W., Schindall, J., Wallace, D., Whitney, D., 2004. The influence of architecture in engineering systems. Pap. Present. MIT Eng. Syst. Symp. 29.

Crilly, N., 2019. Creativity and fixation in the real world: A literature review of case study research. Des. Stud. 64, 154–168. https://doi.org/10.1016/j.destud.2019.07.002

Dahmus, J.B., Gonzalez-Zugasti, J.P., Otto, K.N., 2001. Modular product architecture. Des. Stud. 22, 409–424. https://doi.org/10.1016/S0142-694X(01)00004-7

Danese, P., Filippini, R., 2013. Direct and mediated effects of product modularity on development time and product performance. IEEE Trans. Eng. Manag. 60, 260–271. https://doi.org/10.1109/TEM.2012.2208268

Daniilidis, C., Ensslin, V., Eben, K.G.M., Lindemann, U., 2011. a Classification Framework for Product Modularization Methods. ICED11 18th Int. Conf. Eng. Des. 400.

de Vos, P., Stapersma, D., 2018a. Automatic topology generation for early design of on-board energy distribution systems. Ocean Eng. 170, 55–73. https://doi.org/10.1016/j.oceaneng.2018.09.023

de Vos, P., Stapersma, D., 2018b. Automatic topology generation for early design of on-board energy distribution systems. Ocean Eng. 170, 55–73. https://doi.org/10.1016/J.OCEANENG.2018.09.023

de Weck, O.L., 2015. 16.842 Fundamentals of Systems Engineering.

De Weck, O.L., Ross, A.M., Rhodes, D.H., 2012. Investigating Relationships and Semantic Sets amongst System Lifecycle Properties (Ilities). Third Int. Eng. Syst. Symp. CESUN 2012, Delft Univ. Technol. 18-20 June 2012 18–20.

DeJonckheere, M., Vaughn, L.M., 2019. Semistructured interviewing in primary care research: A balance of relationship and rigour. Fam. Med. Community Heal. 7, e000057. https://doi.org/10.1136/fmch-2018-000057

Delvenne, J.-C., Yaliraki, S.N., Barahona, M., 2008. Stability of graph communities across time scales. Proc. Natl. Acad. Sci. U. S. A. 107, 12755–60. https://doi.org/10.1073/pnas.0903215107

Department of Defense, 2011. FACT SHEET: Resilience of Space Capabilities.

Doar, M.B., 1996. Better model for generating test networks, in: Conference Record / IEEE Global Telecommunications Conference. pp. 86–93. https://doi.org/10.1109/glocom.1996.586131

Dong, A., Sarkar, S., Moullec, M.-L., Jankovic, M., 2016. Eigenvector Rotation as an Estimation of Architectural Change, in: Volume 7: 28th International Conference on Design Theory and Methodology. ASME, p. V007T06A014. https://doi.org/10.1115/DETC2016-59114

Dong, Q., 2002. Predicting and managing system interactions at early phase of the product development process.

Du, X.P., Chen, W., 2000. Towards a better understanding of modeling feasibility robustness in engineering design. J. Mech. Des. 122, 385–394. https://doi.org/10.1115/1.1290247

Duffy, A.H.B., Donnell, F.J.O., 1999. A Design Research Approach. Crit. Enthusiasm - Contrib. to Des. Sci. Work. Res. Methods AI Des. 33–40. https://doi.org/82-91917-08-6

Easterby-Smith, M., Thorpe, R., Jackson, P., 2012. Management Research, 4th eds.

Edmonds, J., Karp, R.M., 1972. Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems, Jc~urnal of the Association for Computing Machinery.

Efatmaneshnik, M., Ryan, M. j., 2015. On optimal modularity for system construction. Complexity. https://doi.org/10.1002/cplx.21646

Eisenhardt, K.M., 1989. Building theories from case study research. Acad. Manag. Rev. 14, 532–550. https://doi.org/10.2307/258557

Elias, G., Jain, R., 2017. Assessing systems architecture: an exploratory framework. J. Bus. Inf. Syst.

Eppinger, S.D., Browning, T.R., 2012. Design Structure Matrix Methods and Applications. MIT Press, Cambridge, MA, USA.

Eppinger, S.D., Whitney, D.E., Smith, R.P., Gebala, D.A., 1994. A model-based method for organizing tasks in product development. Res. Eng. Des. 6, 1–13. https://doi.org/10.1007/BF01588087

Ericsson, A., Erixon, G., 1999. Controlling Design Variants: Modular Product Platforms 145.

Estrada, E., Knight, P., 2015. A First Course in Network Theory 272. https://doi.org/10.1145/2757218.2757223

Ethiraj, S.K., Levinthal, D., 2004. Modularity and Innovation in Complex Systems. Manage. Sci. 50, 159–173. https://doi.org/10.1287/mnsc.1030.0145

Evbuomwan, N.F.O., Sivaloganathan, S., Jebb, a., 1996. A survey of design philosophies, models, methods and systems. J. Eng. Manuf. 210, 301–320. https://doi.org/10.1243/PIME_PROC_1996_210_123_02

Fixson, S.K., 2005. Product architecture assessment: A tool to link product, process, and supply chain design decisions. J. Oper. Manag. 23, 345–369. https://doi.org/10.1016/j.jom.2004.08.006

Fortunato, S., 2010. Community detection in graphs. Phys. Rep. 486, 75–174. https://doi.org/10.1016/j.physrep.2009.11.002

Fricke, E., Schulz, A.P., 2005. Design for changeability (DfC): Principles to enable changes in systems throughout their entire lifecycle. Syst. Eng. 8, 342–359. https://doi.org/10.1002/sys.20039

Fussell, J.B., 1975. A Review of Fault Tree Analysis with Emphasis on Limitations. IFAC Proc. Vol. 8, 552–557. https://doi.org/10.1016/s1474-6670(17)67596-7

Gero, J.S., Kannengiesser, U., 2004. The situated function-behaviour-structure framework. Des. Stud. 25, 373–391. https://doi.org/10.1016/j.destud.2003.10.010

Gershenson, J.K., Prasad, G.J., Zhang, Y., 2003. Product modularity: Definitions and benefits. J. Eng. Des. 14, 295–313. https://doi.org/10.1080/0954482031000091068

Gershenson, J K, Prasad, G.J., Zhang, Y., 2003. Product modularity: Definitions and benefits. J. Eng. Des. 14, 295–313. https://doi.org/10.1080/0954482031000091068

Giffin, M., de Weck, O., Bounova, G., Keller, R., Eckert, C., Clarkson, P.J., 2009. Change Propagation Analysis in Complex Technical Systems. ASME J Mech Des. 131, 1–14. https://doi.org/10.1115/1.3149847

Glazier, T.J., Cámara, J., Schmerl, B., Garlan, D., 2015. Analyzing resilience properties of different topologies of collective adaptive systems. Proc. - 2015 IEEE 9th Int. Conf. Self-Adaptive Self-Organizing Syst. Work. SASOW 2015 55–60. https://doi.org/10.1109/SASOW.2015.14

Göhler, S., Eifler, T., Howard, T.J., 2016. Robustness Metrics: Consolidating the Multiple Approaches to Quantify Robustness. J. Mech. Des. 138, 111407. https://doi.org/10.1115/1.4034112

Goldman, S.L., 2004. Why we need a philosophy of engineering: a work in progress. Interdiscip. Sci. Rev. 29, 163–176. https://doi.org/10.1179/030801804225012572

Goodrum, C.J., Shields, C.P.F., Singer, D.J., 2018. Understanding cascading failures through a vulnerability analysis of interdependent ship-centric distributed systems using networks. Ocean Eng. 150, 36–47. https://doi.org/10.1016/j.oceaneng.2017.12.039

Goulter, 1987. Current and future use of systems analysis in water distribution network design. Civ. Eng. Syst. 4, 175–184. https://doi.org/10.1080/02630258708970484

Grant, M.J., Booth, A., 2009. A typology of reviews: An analysis of 14 review types and associated methodologies. Health Info. Libr. J. 26, 91–108. https://doi.org/10.1111/j.1471-1842.2009.00848.x

Griffin, M., 2010. HOW DO WE FIX SYSTEM ENGINEERING? Acta Odontológica Venez. 49, 1–14. https://doi.org/10.1890/09-2296.1.Abrams

Guo, F., Gershenson, J.K., 2007. Discovering relationships between modularity and cost. J. Intell. Manuf. 18, 143–157. https://doi.org/10.1007/s10845-007-0007-y

Guo, F., Gershenson, J.K., 2004. A comparison of modular product design methods based on improvement and iteration, in: ASME 2004 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. pp. 261–269.

Haley, B.M., Dong, A., 2014. Creating faulTable network models of complex engineered, in: Proceedings of the ASME 2014 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE 2014. New York, USA.

Haley, B.M., Dong, A., Tumer, I.Y., 2016. A Comparison of Network-Based Metrics of Behavioral Degradation in Complex Engineered Systems. J. Mech. Des. 138, 121405.

https://doi.org/10.1115/1.4034402

Helmer, R., Yassine, A., Meier, C., 2010. Systematic module and interface definition using component design structure matrix. J. Eng. Des. 21, 647–6. https://doi.org/10.1080/09544820802563226

Hölttä-Otto, K., Chiriaca, N.A., Lysy, D., Suh, E.S., 2012. Comparative analysis of coupling modularity metrics. J. Eng. Des. 23, 790–806. https://doi.org/10.1080/09544828.2012.701728

Holtta-Otto, K., de Weck, O., 2007. Degree of Modularity in Engineering Systems and Products with Technical and Business Constraints. Concurr. Eng. 15, 113–126. https://doi.org/10.1177/1063293X07078931

Hölttä-Otto, K., De Weck, O., 2008. Metrics for assessing coupling density and modularity in complex products and systems, in: 2007 Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DETC2007. pp. 343–352. https://doi.org/10.1115/DETC2007-34871

Holtta-otto, K., Weck, O. De, 2007. Metrics for Assessing Coupling Density and Modularity. pp. 1–10.

Hölttä, K., Eun Suk, S., de Weck, O., 2005. Tradeoff Between Modularity and Performance for Engineered Systems and Products. Int. Conf. Eng. Des. ICED 05 ME, 1–14.

Hvam, L., Herbert-Hansen, Z.N.L., Haug, A., Kudsk, A., Mortensen, N.H., 2017. A framework for determining product modularity levels. Adv. Mech. Eng. 9, 168781401771942. https://doi.org/10.1177/1687814017719420

IEEE and ISO/IEC, 2017. ISO/IEC/IEEE 24765 — Systems and software engineering — Vocabulary [WWW Document]. URL https://www.iso.org/standard/71952.html (accessed 3.9.20).

IEEE and ISO/IEC, 2015. ISO/IEC/IEEE 15288 —Systems and software engineering — System life cycle processes [WWW Document].

IEEE and ISO/IEC, 2011. ISO/IEC/IEEE 42010 — Systems and software engineering — Architecture description.

INCOSE, 2015. Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, Systems Engineering.

International Electrotechnical Commision, 2018. IEC 60812 Failure modes and effects analysis.

International Electrotechnical Commission, 2006. IEC 61025: Fault Tree (FTA), Analysis. https://doi.org/10.1111/j.1539-6924.1988.tb00510.x

Ishii, K., Yang, T.G., 2003. Modularity: International industry benchmarking and research roadmap. ASME 2003 Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf. 1–11.

Jacobs, M., Vickery, S.K., Droge, C., 2007. The effects of product modularity on competitive performance. Int. J. Oper. Prod. Manag. 27, 1046–1068. https://doi.org/10.1108/01443570710820620

Jansen, A.C., Duchateau, E.A.E., Kana, A.A., 2018. Towards a novel design perspective for system vulnerability using a Markov chain, in: Proceedings of the International Naval Engineering Conference and Exhibition (INEC). https://doi.org/10.24868/issn.2515-818x.2018.028

Jansson, D.G., Smith, S.M., 1991. Design fixation. Des. Stud. 12, 3–11. https://doi.org/10.1016/0142-694X(91)90003-F

Jin, C., Chen, Q., Jamin, S., 2000. Inet: Internet topology generator, Options.

Johnson, R.B., Onwuegbuzie, A.J., 2004. Mixed Methods: A Research Paradigm Whose Time Has Come. Educ. Res. 33, 14–26.

Jose, A., Tollenaere, M., 2005. Modular and platform methods for product family design: Literature analysis. J. Intell. Manuf. 16, 371–390. https://doi.org/10.1007/s10845-005-7030-7

Jung, S., Simpson, T.W., 2017. New modularity indices for modularity assessment and clustering of product architecture. J. Eng. Des. 28, 1–22. https://doi.org/10.1080/09544828.2016.1252835

Kaufman, S., 1995. At home in the universe. Search Laws Self-Organization.

King, A.M., Sivaloganathan, S., 1999. Development of a methodology for concept selection in flexible design strategies. J. Eng. Des. 10, 329–349. https://doi.org/10.1080/095448299261236

Kott, A., Abdelzaher, T., 2016. Resiliency and Robustness of Complex, Multi-Genre Networks. eprint arXiv:1601.06731.

Kress, V.E., Shoffiner, M.F., 2007. Focus groups: A practical and applied research approach for counselors. J. Couns. Dev. 85, 189–195. https://doi.org/10.1002/j.1556-6678.2007.tb00462.x

Kroese, D.P., Brereton, T., Taimre, T., Botev, Z.I., 2014. Why the Monte Carlo method is so important today. Wiley Interdiscip. Rev. Comput. Stat. 6, 386–392. https://doi.org/10.1002/wics.1314

Langford, J., McDonagh, D., 2003. Focus groups: supporting effective product development.

Latora, V., Marchiori, M., 2008. Vulnerability and Protection of Critical Infrastructures 11–14. https://doi.org/10.1103/PhysRevE.71.015103

Lau, A.K.W., Yam, R.C.M., Tang, E., 2007. The impacts of product modularity on competitive capabilities and performance: An empirical study. Int. J. Prod. Econ. 105, 1–20. https://doi.org/10.1016/j.ijpe.2006.02.002

Laurence, S., Hanlin, T., John F, S., Stephanie, P., 2010. Shared modular build of warships, Distribution. https://doi.org/10.1214/07-EJS057

Magee, C., Whitney, D., Moses, J., 2010. ESD.342 Network Representations of Complex Engineering Systems [WWW Document]. Massachusetts Inst. Technol. MIT OpenCourseWare. URL https://ocw.mit.edu/courses/engineering-systems-division/esd-342-network-representations-of-complex-engineering-systems-spring-2010/ (accessed 11.11.19).

Magee, C.L., 2010. ESD 342 class 14 decomposition Class 14 Lecture outline † Decomposition Taxonomy and examples Network-based Approaches to Quantitative Decomposition.

Maier, M.W., Rechtin, E., 1997. The Art of Systems Architecting, New York. CRC Press LLC.

Malak, R., Baxter, B., Hsiao, C., 2015. A decision-based perspective on assessing system robustness. Procedia Comput. Sci. 44, 619–629. https://doi.org/10.1016/j.procs.2015.03.069

Martelot, E. Le, Hankin, C., 2013. Multi-scale community detection using stability optimisation. Int. J. Web Based Communities 9, 323. https://doi.org/10.1504/IJWBC.2013.054907

Meehan, J.S., Duffy, A.H.B., Whitfield, I., 2007. Supporting “design for reuse” with modular design. Concurr. Eng. 15, 141–155.

Mehrpouyan, H., Haley, B., Dong, A., Tumer, I.Y., Hoyle, C., 2014. Resiliency analysis for complex engineered system design. Artif. Intell. Eng. Des. Anal. Manuf. AIEDAM 29, 93–108. https://doi.org/10.1017/S0890060414000663

Melvin, J., Deo, H., 2002. Axiomatically designed robustness. Am. Supplier Inst. 19th Annu. Taguchi Methods Symp.

Mhenni, F., Nguyen, N., Choley, J.Y., 2014. Automatic fault tree generation from SysML system models. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics, AIM 715–720. https://doi.org/10.1109/AIM.2014.6878163

Min, G., Suh, E.S., Hölttä-Otto, K., 2016. System Architecture, Level of Decomposition, and Structural Complexity: Analysis and Observations. J. Mech. Des. 138, 021102. https://doi.org/10.1115/1.4032091

Minai, A.A., Braha, D., Bar-Yam, Y., 2006. Complex engineered systems: A new paradigm.

Underst. Complex Syst. 2006, 1–21. https://doi.org/10.1007/3-540-32834-3_1

Ministry of Defense, U., 2015a. Key Threshold Requirements for Surface Ship Vulnerability.

Ministry of Defense, U., 2015b. How Defence Works 1–55.

Miraglia, S., 2014. Systems Architectures and Innovation : the Modularity-Integrality Framework, Cambridge Service Alliance Working Paper.

Moullec, M.-L., Bouissou, M., Jankovic, M., Bocquer, J.C., 2013. Toward System Architecture Generation and Performances Assessment Under Uncertainty Using Bayesian Networks. J. Mech. Des. 135, 041002. https://doi.org/10.1115/1.4023514

Moullec, M., Jankovic, M., Eckert, C., 2016. Selecting system architecture : What a single industrial experiment can tell us about the traps to avoid when choosing selection criteria 250–262. https://doi.org/10.1017/S0890060416000238

NATO, 2002. ANEP 43 Ship Combat Survivability.

Nepal, B., Monplaisir, L., Singh, N., 2005. Integrated fuzzy logic-based model for product modularization during concept development phase. Int. J. Prod. Econ. 96, 157–174. https://doi.org/10.1016/j.ijpe.2004.03.010

Newman, M., 2010. Networks: an introduction.

Newman, M.E.J., 2006. Modularity and community structure in networks. Proc. Natl. Acad. Sci. U. S. A. 103, 8577–82. https://doi.org/10.1073/pnas.0601602103

Newman, M.E.J., Girvan, M., 2004. Finding and evaluating community structure in networks. Phys. Rev. E 69, 026113. https://doi.org/10.1103/PhysRevE.69.026113

Office of the Assistant Secretary of Defense for Homeland Defense and Global Security, 2015. Space Domain Mission Assurance: A Resilience Taxonomy White Paper. Fed. Am. Sci. 1–10.

Okudan, G.E., Tauhid, S., 2008. Concept selection methods - a literature review from 1980 to 2008. Int. J. Des. Eng. 1, 243. https://doi.org/10.1504/ijde.2008.023764

Otto, K., Hölttä-Otto, K., Sanaei, R., Wood, K.L., 2020b. Incorporating Field Effects into Functional Product-System Architecting Methods. J. Mech. Des. Trans. ASME 142, 1–15. https://doi.org/10.1115/1.4044839

Otto, K.N., Wood, K.L., 2001. Product design: techniques in reverse engineering and new product development. Prentice Hall.

Page, J., 2012. Flexibility in Early Stage Design of U.S. Navy Ships: An Analysis of Options. J. Sh. Prod. Des. 28, 128–133. https://doi.org/10.5957/JSPD.28.3.120012

Pahl, G., Beitz, W., 1996. Engineering Design: A Systematic Approach, Second Edi. ed. https://doi.org/10.1007/978-1-84628-319-2

Paparistodimou, G., Duffy, A., Voong, C., Robb, M., 2017. System Architectures Assessment Based On Network Metrics, in: 19th International Dependency and Structure Modeling Conference, DSM. Aalto University Design Factory, Finland.

Paparistodimou, G., Duffy, A., Whitfield, R.I., Knight, P., Robb, M., 2020. A network tool to analyse and improve robustness of system architectures. Des. Sci. 6, e8. https://doi.org/10.1017/dsj.2020.6

Parker, M.C., 2014. A Contextual Multipartite Network Approach to Comprehending the Structure of Naval Design by.

Parraguez, P., Eppinger, S.D., Maier, A.M., 2015. Information Flow Through Stages of Complex Engineering Design Projects: A Dynamic Network Analysis Approach. IEEE Trans. Eng. Manag. 62, 604–617. https://doi.org/10.1109/TEM.2015.2469680

Parraguez, P., Maier, A., 2016. Using Network Science to Support Design Research: From Counting to Connecting, in: Experimental Design Research: Approaches, Perspectives, Applications. pp. 153–172.

Parraguez, P., Piccolo, S.A., Perisic, M.M., Storga, M., Maier, A.M., 2019. Process Modularity Over Time: Modeling Process Execution as an Evolving Activity Network. IEEE Trans. Eng. Manag. PP, 1–13. https://doi.org/10.1109/tem.2019.2935932

Pasqual, M.C., Weck, O.L. De, 2011. Multilayer network model for analysis and management of change propagation. https://doi.org/10.1007/s00163-011-0125-6

Pero, M., Stößlein, M., Cigolini, R., 2015. Linking product modularity to supply chain integration in the construction and shipbuilding industries. Int. J. Prod. Econ. 170, 1–14. https://doi.org/10.1016/j.ijpe.2015.05.011

Piccolo, S.A., Lehmann, S., Maier, A., 2018. Design process robustness: A bipartite network analysis reveals the central importance of people. Des. Sci. 4, 1–29. https://doi.org/10.1017/dsj.2017.32

Pidd, M., 2010. Why modelling and model use matter. J. Oper. Res. Soc. 61, 14–24. https://doi.org/10.1057/jors.2009.141

Pimmler, T.U., Eppinger, S.D., 1994. Integration analysis of product decompositions. ASME 1994 Int. Des. Eng. Tech. Conf. 6th Des. Theory Methodol. Conf. Minneapolis, USA, 11-14 Sept. 1994 343–351.

Piran, F.A.S., Lacerda, D.P., Camargo, L.F.R., Viero, C.F., Dresch, A., Cauchick-Miguel, P.A., 2016. Product modularization and effects on efficiency: An analysis of a bus manufacturer using data envelopment analysis (DEA). Int. J. Prod. Econ. 182, 1–13. https://doi.org/10.1016/j.ijpe.2016.08.008

Pumpuni-Lenss, G., Blackburn, T., Garstenaur, A., 2017. Resilience in Complex Systems:

An Agent-Based Approach. Syst. Eng. https://doi.org/10.1002/sys

Qinetiq, 2020. SURVIVE® [WWW Document]. https://www.qinetiq.com/What-we-do/Maritime/Marine-Design-Software?pagetab=Survive.

Rabiee, F., 2004. Focus-group interview and data analysis. Proc. Nutr. Soc. 63, 655–660. https://doi.org/10.1079/pns2004399

Raz, A.K., DeLaurentis, D.A., 2017. System-of-Systems Architecture Metrics for Information Fusion: A Network Theoretic Formulation. AIAA Inf. Syst. Infotech @ Aerosp. 1–14. https://doi.org/10.2514/6.2017-1292

Reichardt, J., Bornholdt, S., 2006. Statistical mechanics of community detection. Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys. 74, 016110. https://doi.org/10.1103/PhysRevE.74.016110

Rigterink, D.T., 2014. Methods for Analyzing Early Stage Naval Distributed Systems Designs, Employing Simplex, Multislice, and Multiplex Networks. The University of Michigan.

Rivkin, J.W., Siggelkow, N., 2007. Patterned Interactions in Complex Systems: Implications for Exploration. Manage. Sci. 53, 1068–1085. https://doi.org/10.1287/mnsc.1060.0626

Robinson, S., 2008. Conceptual modelling for simulation Part II: A framework for conceptual modelling. J. Oper. Res. Soc. 59, 291–304. https://doi.org/10.1057/palgrave.jors.2602368

Robinson, T.J., Borror, C.M., Myers, R.H., 2004. Robust parameter design: A review. Qual. Reliab. Eng. Int. 20, 81–101. https://doi.org/10.1002/qre.602

Roozenburg, N.F.M., Eekels, J., 1995. Product Design: Fundamentals and Methods.

Ross, A.M., Rhodes, D.H., Hastings, D.E., 2008. Defining changeability: Reconciling flexibility, adaptability, scalability, modifiability, and robustness for maintaining system lifecycle value. Syst. Eng. 11, 246–262. https://doi.org/10.1002/sys.20098

Roth, M., Wolf, M., Lindemann, U., 2015. Integrated matrix-based fault tree generation and evaluation. Procedia Comput. Sci. 44, 599–608. https://doi.org/10.1016/j.procs.2015.03.027

Ryan, M., Efatmaneshnik, M., 2014. Balancing Modularity : the Role of Uncertainty.

Sanaei, R., Otto, K., Holtta-Otto, K., Luo, J., 2015. Trade-Off Analysis of System Architecture Modularity Using Design Structure Matrix DETC2015-46403.

Sanaei, R., Otto, K.N., Hölttä-Otto, K., Wood, K.L., 2017. An algorithmic approach to system modularization under constraints, in: Understand, Innovate, and Manage Your Complex System! - Proceedings of the 19th International DSM Conference. The Design Society, pp. 15–24.

Sanaei, R., Otto, K.N., Hölttä-Otto, K., Wood, K.L., 2016. Incorporating constraints in system modularization by interactive clustering of design structure matrices, in: Proceedings of the ASME Design Engineering Technical Conference. ASME, p. V02BT03A042. https://doi.org/10.1115/DETC2016-60510

Sarkar, S., Dong, A., Henderson, J. a, Robinson, P. a, 2014. Spectral Characterization of Hierarchical Modularity in Product Architectures. J. Mech. Des. 136, 0110061–1100612. https://doi.org/10.1115/1.4025490

Saunders, A., Tosey, P., 2013. The Layers of Research Design. Rapport 14, 58–59. https://doi.org/08 jun 2015

Schofield, J., 2018. SCL’s purple fire survivability/lethality assessment tool, in: Warship 2018: Procurement of Future Surface Vessel. London,UK.

Schulz, A.P., Fricke, E., 1999. Incorporating flexibility, agility, robustness, and adaptability within the design of integrated systems - key to success?, in: AIAA/IEEE Digital Avionics Systems Conference - Proceedings. https://doi.org/10.1109/dasc.1999.863677

Selva, D., Cameron, B., Crawley, E., 2016. Patterns in System Architecture Decisions. Syst. Eng. 19, 477–497. https://doi.org/10.1002/sys.21370

Shanthikumar, J.G., 1987. Reliability Of Systems With Consecutive Minimal Cutsets. IEEE Trans. Reliab. R-36, 546–550. https://doi.org/10.1109/TR.1987.5222467

Sharman, D., Yassine, A., 2004. Characterizing complex product architectures. Syst. Eng.

Sheard, S.A., 2012. Assessing the impact of complexity attributes on system development project outcomes.

Siemaszko, D., Pittet, S., 2011. Impact of modularity and redundancy in optimising the reliability of power systems that include a large number of power converters. Microelectron. Reliab. 51, 1484–1488. https://doi.org/10.1016/j.microrel.2011.07.056

Sinha, K., 2014. Structural Complexity and its Implications for Design of Cyber Physical Systems. Massachusetts Institute of Technology.

Sinha, K., Han, S.Y., Suh, E.S., 2019. Design structure matrix-based modularization approach for complex systems with multiple design constraints. Syst. Eng. 1–10. https://doi.org/10.1002/sys.21518

Sinha, K., Suh, E.S., 2017. Pareto-optimization of complex system architecture for structural complexity and modularity. Res. Eng. Des. 29, 1–19. https://doi.org/10.1007/s00163-017-0260-9

Sinha, K., Suh, E.S., de Weck, O., 2018. Integrative Complexity: An Alternative Measure for System Modularity. J. Mech. Des. 140, 051101. https://doi.org/10.1115/1.4039119

Sinha, K., Suh, E.S., Weck, O. De, 2017. Correlating integrative complexity with system modularity. pp. 1–8.

Slagle, J., 2007. Architecting Complex Systems for Robustness. MIT.

Sosa, M., Mihm, J., Browning, T., 2011. Degree Distribution and Quality in Complex Engineered Systems. J. Mech. Des. 133, 101008. https://doi.org/10.1115/1.4004973

Sosa, M.E., Eppinger, S.D., Rowles, C.M., 2007a. A Network Approach to Define Modularity of Components in Complex Products. J. Mech. Des. 129, 1118. https://doi.org/10.1115/1.2771182

Sosa, M.E., Eppinger, S.D., Rowles, C.M., 2007b. A Network Approach to Define Modularity of Components A Network Approach to Define Modularity of Components. https://doi.org/10.1115/1.2771182

Sosa, M.E., Eppinger, S.D., Rowles, C.M., 2003. Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions. J. Mech. Des. 125, 240. https://doi.org/10.1115/1.1564074

Sosa, M.E., Mihm, J., Browning, T.R., 2013. Linking Cyclicality and Product Quality. Manuf. Serv. Oper. Manag. 15, 473–491. https://doi.org/10.1287/msom.2013.0432

Sparks, G., 2014. Enterprise Architect User Guide [WWW Document]. Sparx Syst. URL https://www.sparxsystems.com.au/bin/EAUserGuide.pdf

Sterbenz, J.P.G., Çetinkaya, E.K., Hameed, M.A., Jabbar, A., Qian, S., Rohrer, J.P., 2013. Evaluation of network resilience, survivability, and disruption tolerance: Analysis, topology generation, simulation, and experimentation: Invited paper. Telecommun. Syst. 52, 705–736. https://doi.org/10.1007/s11235-011-9573-6

Stone, R.B., Wood, K.L., Crawford, R.H., 2000. A heuristic method for identifying modules for product architectures. Des. Stud. 21, 5–31. https://doi.org/10.1016/S0142-694X(99)00003-4

Sudjianto, A., Otto, K., 2001. Modularization to Support Multiple Brand Platforms. Des. Eng. 1–14.

Suh, N.P., 1990. The principles of design. Oxford University Press.

Taguchi, G., 1987. Systems of Experimental Design: Engineering Methods to Optimize Quality and Minimize Cost.

Tamaskar, S., Neema, K., Delaurentis, D., 2014. Framework for measuring complexity of aerospace systems. Res. Eng. Des. 25, 125–137. https://doi.org/10.1007/s00163-014-0169-5

Tang, C.S., Zimmerman, J.D., Nelson, J.I., 2009. Managing New Product Development and Supply Chain Risks: The Boeing 787 Case. Supply Chain Forum An Int. J. 10, 74–86. https://doi.org/10.1080/16258312.2009.11517219

Tangmunarunkit, H., Govindan, R., Jamin, S., Shenker, S., Willinger, W., 2002. Network topology generators: Degree-based vs. structural. Comput. Commun. Rev. 32, 147–159. https://doi.org/10.1145/964725.633040

Tomiyama, T., Moyen, F., 2018. Resilient architecture for cyber-physical production systems. CIRP Ann. 67, 161–164. https://doi.org/10.1016/j.cirp.2018.04.021

Turner, A.J., Monahan, W., Cotter, M., 2017. Quantifying the ilities: a literature review of robustness, interoperability, and agility. 15th Annu. Conf. Syst. Eng. Res. 1035–1049.

Ullman, D.G., 2002. The Ideal Engineering Decision Support System. Retrieved March 10, 2009.

Ulrich, K., 1994. Fundamentals of Product Modularity, in: Sriram Dasu, C.E. (Ed.), Management of Design. Kluwer Academic Publishers, Boston/Dordrecht/London, pp. 219–231.

Ulrich, K., Eppinger, S., 1997. Product design and development, McGraw-Hil. ed. New York.

Ulrich, K., Tung, K., 1991. Fundamentals of product modularity, in: ASME (Ed.), ASME Winter Annual Meeting Symposium on Issues in Design/Manufacturing Integration. ASME, Atlanta, pp. 73–79.

van Oers, B.J., 2011. A Packing Approach for the Early Stage Design of Service Vessels.

Vickery, S.K., Bolumole, Y.A., Castel, M.J., Calantone, R.J., 2015. The effects of product modularity on launch speed. Int. J. Prod. Res. 53, 5369–5381. https://doi.org/10.1080/00207543.2015.1047972

Vickery, S.K., Koufteros, X., Dröge, C., Calantone, R., 2016. Product Modularity, Process Modularity, and New Product Introduction Performance: Does Complexity Matter? Prod. Oper. Manag. 25, 751–770. https://doi.org/10.1111/poms.12495

Walsh, H.S., Dong, A., Tumer, I.Y., 2019. An Analysis of Modularity as a Design Rule Using Network Theory. J. Mech. Des. Trans. ASME 141, 031102. https://doi.org/10.1115/1.4042341

Walsh, H.S., Dong, A., Tumer, I.Y., 2017. The role of bridging nodes in behavioral network models of complex engineered systems. Des. Sci.

Watts, D.J., Strogatz, S.H., 1998. Collectivedynamics of “small-world” networks. Nature 393, 440–442. https://doi.org/10.1038/30918

Whitfield, R.I., Smith, J.S., Duffy, a. H.B., 2002. Identifying component modules 15–17.

Whitfield, R.I., Wright, P.N.H., Coates, G., Hills, W., 1998. A robust design methodology suitable for application to one-off products. J. Eng. Des. 9, 373–387. https://doi.org/10.1080/095448298261507

Whitney, D.E., 2003. Chapter 17 Network Models of Mechanical.

Wilson, J., 2010. Essentials of Business Research - A Guide to Doing Your Research Project, SAGE Publication.

Wyatt, D.F., Wynn, D.C., Jarrett, J.P., Clarkson, P.J., 2012. Supporting product architecture design using computational design synthesis with network structure constraints. Res. Eng. Des. 23, 17–52. https://doi.org/10.1007/s00163-011-0112-y

Wynn, D.C., Clarkson, P.J., 2018. Process models in design and development. Res. Eng. Des. 29, 161–202. https://doi.org/10.1007/s00163-017-0262-7

Yassine, A.A., Naoum-Sawaya, J., 2016. Architecture, Performance, and Investment in Product Development Networks. J. Mech. Des. 139, 011101. https://doi.org/10.1115/1.4034673

Yazdani, A., Jeffrey, P., 2012. Applying network theory to quantify the redundancy and structural robustness of water distribution systems. J. Water Resour. Plan. Manag. 138, 153–161. https://doi.org/10.1061/(ASCE)WR.1943-5452.0000159.

Yin, R.K., 2003. Case study research: design and methods. Essent. Guid. to Qual. methods Organ. Res. 5, 204. https://doi.org/10.1097/FCH.0b013e31822dda9e

Youmans, R.J., Arciszewski, T., 2014. Design fixation: Classifications and modern methods of prevention. Artif. Intell. Eng. Des. Anal. Manuf. AIEDAM 28, 129–137. https://doi.org/10.1017/S0890060414000043

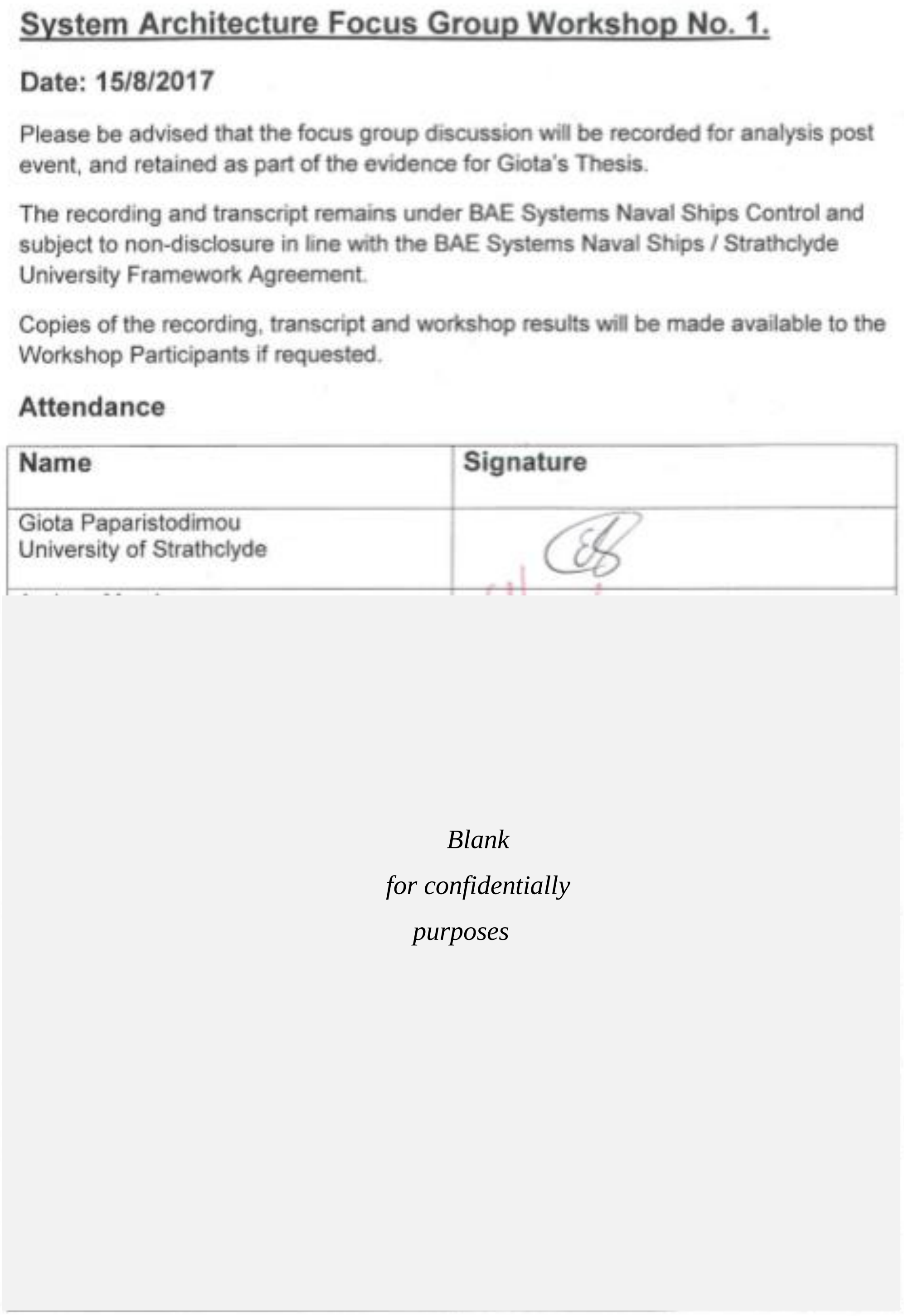
Yu, T.-L., Yassine, A. a., Goldberg, D.E., 2007. An information theoretic method for developing modular architectures using genetic algorithms. Res. Eng. Des. 18, 91–109. https://doi.org/10.1007/s00163-007-0030-1

Zakarian, A., Knight, J.W., Baghdasaryan, L., 2007. Modelling and analysis of system robustness. J. Eng. Des. 18, 243–263. https://doi.org/10.1080/09544820600804939

Zhang, W.Y., Tor, S.Y., Britton, G.A., 2006. Managing modularity in product family design with functional modeling. Int. J. Adv. Manuf. Technol. 30, 579–588. https://doi.org/10.1007/s00170-005-0112-z

Zhou, S., Mondragón, R.J., 2004. The rich-club phenomenon in the internet topology. IEEE Commun. Lett. 8, 180–182. https://doi.org/10.1109/LCOMM.2004.823426

Appendix I: Focus group discussion record



System Architecture Focus Group Workshop No. 1.

Date: 15/8/2017

Please be advised that the focus group discussion will be recorded for analysis post event, and retained as part of the evidence for Giota's Thesis.

The recording and transcript remains under BAE Systems Naval Ships Control and subject to non-disclosure in line with the BAE Systems Naval Ships / Strathclyde University Framework Agreement.

Copies of the recording, transcript and workshop results will be made available to the Workshop Participants if requested.

Attendance

Name	Signature
Giota Paparistodimou University of Strathclyde	

Blank

for confidentially

purposes

Appendix II: Type B and C-Case studies results

The Appendix II supplements the Chapter 8 Case studies provide the analysis stages calculated results of the Type B and C case studies.

Type B: Destroyer – Analysis stage results

In this section the analysis stage results of the nested redundancy, modularity and robustness loops are presented.

1st Redundancy loop: High redundancy

The paragraphs present the robustness results for high-level of redundancy

Combinatory disruption of components

Table 52: Combinatory disruptions of components Type B high redundancy architecture

	If removed component 1	and also removed component 2	A	B	C	D	E
1	HV Switchboard 1	HV Switchboard 2	N	N	Y	Y	N
2	HV Switchboard 1	Transformer2	N	Y	Y	Y	N
3	HV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
4	HV Switchboard 2	Transformer 1	N	Y	Y	Y	N
5	HV Switchboard 2	LV Switchboard 1	N	Y	Y	Y	N
6	Transformer 1	Transformer2	N	Y	Y	Y	N
7	Transformer 1	LV Switchboard 2	N	Y	Y	Y	N
8	Transformer2	LV Switchboard 1	N	Y	Y	Y	N
9	LV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
10	PM Converter 1	PM Converter 2	Y	Y	Y	N	N
11	PM Converter 1	Propulsion Motor 2	Y	Y	Y	N	N
12	PM Converter 1	Auxiliary CSW pump aft	Y	Y	Y	N	N
13	PM Converter 2	Propulsion Motor 1	Y	Y	Y	N	N
14	PM Converter 2	Auxiliary CSW pump forward	Y	Y	Y	N	N
15	Propulsion Motor 1	Propulsion Motor 2	Y	N	Y	N	N
16	Propulsion Motor 1	Auxiliary CSW pump aft	Y	Y	Y	N	N
17	Propulsion Motor 2	Auxiliary CSW pump forward	Y	Y	Y	N	N
18	Steering Gear Power Pack 1	Steering Gear Power Pack 2	Y	N	Y	Y	N
19	CW Manifold2	CW Manifold3	Y	Y	N	Y	N
20	Auxiliary CSW pump forward	Auxiliary CSW pump aft	Y	Y	Y	N	N
	Total		**9**	**3**	**1**	**9**	**20**

The combinatory results indicated 20 combinatory instances of two components will lead on a total loss of the robustness of the system.

Modularity intermediate loop: Low – medium – high modularity

Table 53: Modular configurations for high redundancy architecture Type B

First redundancy loop *high-level of redundancy in the system architecture Type B*			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
1	Diesel Generator 1, HV Switchboard 1, Transformer 1, HDE 5 -HV Filter1, HV Switchboard Interconnector2	Diesel Generator 1, Diesel Generator 3, HV Switchboard 1, HV Switchboard 3, Transformer 1, HDE 5 -HV Filter1	Diesel Generator 1, HV Switchboard 1, Transformer 1, HDE 5 -HV Filter1, HV Switchboard Interconnector1, HV Switchboard Interconnector2
2	Diesel Generator 2, HV Switchboard 2, Transformer2, HV Switchboard Interconnector1	Emergency Generator 1, Emergency Switchboard 1	Diesel Generator 2, Diesel Generator 3, HV Switchboard 2, HV Switchboard 3, Transformer2, HDE 5 -HV Filter2, CW Manifold3
3	Diesel Generator 3, HV Switchboard 3	Diesel Generator 2, HV Switchboard 2, HDE 5 -HV Filter2, HV Switchboard Interconnector1, HV Switchboard Interconnector2	Emergency Generator 1, Emergency Switchboard 1, LV Switchboard 2, EDC 10, EDC 12, LV Filter 2, LV Switchboard Interconnector2, Steering Gear Power Pack 2
4	LV Switchboard 1, EDC 1, EDC 3, LV Switchboard Interconnector1, LV Switchboard Interconnector2	EDC 2, ChilledWaterPlan1PF, ChilledWaterPlan1PF-Manifold	Gas Turbine Alternator 2, EDC 6, LP Seawater Pump 2, GTSW Pump 2, GT Cooler2, GT Module2
5	LV Switchboard 2, EDC 10, EDC 12, Steering Gear Power Pack 2	EDC 4, LV Filter 1, ChilledWaterPlan2SF, CW Manifold2, Fore mast	EDC 2, EDC 4, LV Filter 1, ChilledWaterPlan1PF, ChilledWaterPlan2SF, ChilledWaterPlan1PF-Manifold, CW Manifold2, Fore mast
6	EDC 9, ChilledWaterPlan4SA, ChilledWaterPlan4SA-Manifold	EDC 9, ChilledWaterPlan4SA, ChilledWaterPlan4SA-Manifold	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward
7	Emergency Generator 1, Emergency Switchboard 1, EDC 11, Steering Gear Power Pack 1	LV Switchboard 1, EDC 1, EDC 3, EDC 11, LV Switchboard Interconnector2, Steering Gear Power Pack 1	EDC 8, PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft, Aft mast
8	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward	LV Switchboard 2, EDC 10, EDC 12, LV Switchboard Interconnector1, Steering Gear Power Pack 2	LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11, LV Switchboard Interconnector1, Steering Gear Power Pack 1, ChilledWaterPlan4SA, ChilledWaterPlan4SA-Manifold

First redundancy loop
high-level of redundancy in the system architecture Type B

Module ID	**First modularity loop** *low-level of modularity*	**Second modularity loop** *medium level of modularity*	**Third modularity loop** *high-level of modularity*
9	PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft	Transformer2, LV Filter 2, ChilledWaterPlan3PA, CW Manifold3	Gas Turbine Alternator 1, EDC 7, ChilledWaterPlan3PA, LP Seawater Pump 1 , GTSW Pump 1 , GT Cooler1, GT Module1
10	EDC 2, ChilledWaterPlan1PF, ChilledWaterPlan1PF-Manifold	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward	
11	EDC 4, LV Filter 1, ChilledWaterPlan2SF, CW Manifold2, Fore mast	Gas Turbine Alternator 2, EDC 6, LP Seawater Pump 2, GTSW Pump 2, GT Cooler2, GT Module2	
12	LV Filter 2, HDE 5 -HV Filter2, CW Manifold3	Gas Turbine Alternator 1, EDC 7, LP Seawater Pump 1, GTSW Pump 1 , GT Cooler1, GT Module1	
13	EDC 7, ChilledWaterPlan3PA, LP Seawater Pump 1	EDC 8, PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft, Aft mast	
14	Gas Turbine Alternator 1, GTSW Pump 1, GT Cooler1, GT Module1		
15	Gas Turbine Alternator 2, EDC 6, LP Seawater Pump 2, GTSW Pump 2, GT Cooler2, GT Module2		
16	EDC 8, Aft mast		
Robustness metric	0.819	0.79	0.711
Modularity metric	0.66	0.69	0.72
Classification of modules (Modules ID)			
central modules	[15]	[1]	[5]
periphery modules	[3,9,16]	[2]	[6]

2nd Redundancy loop: Medium redundancy

The second redundancy loop represents a medium level of redundancy developed from the high redundancy configuration, with reduced generator redundancy. This represents an actual ship operating condition, unlike the semi-hypothetical medium and low redundancy architectures developed for Type A and C.

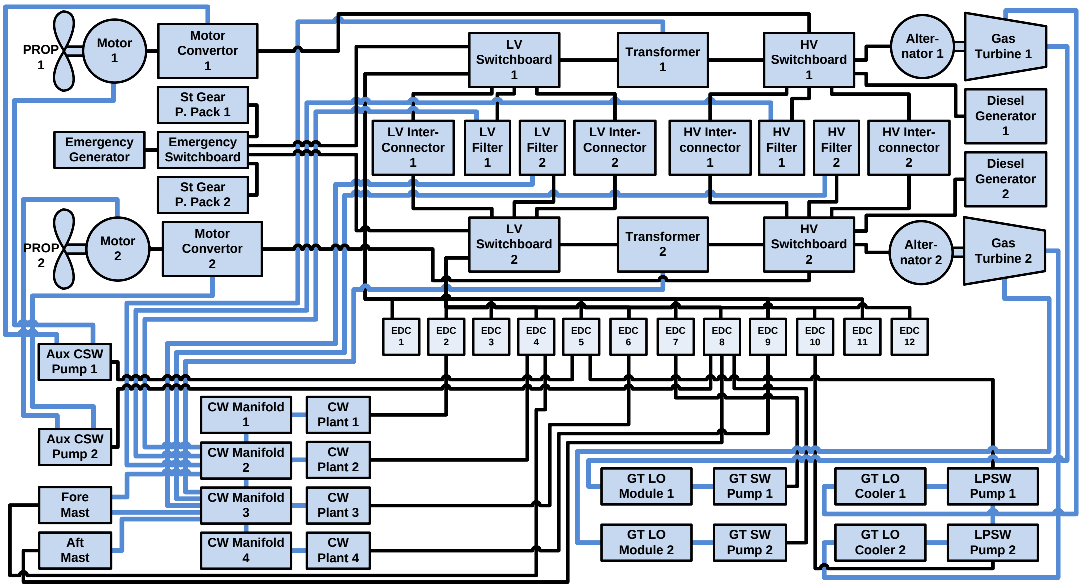


Figure 80: Type B medium redundancy schematic

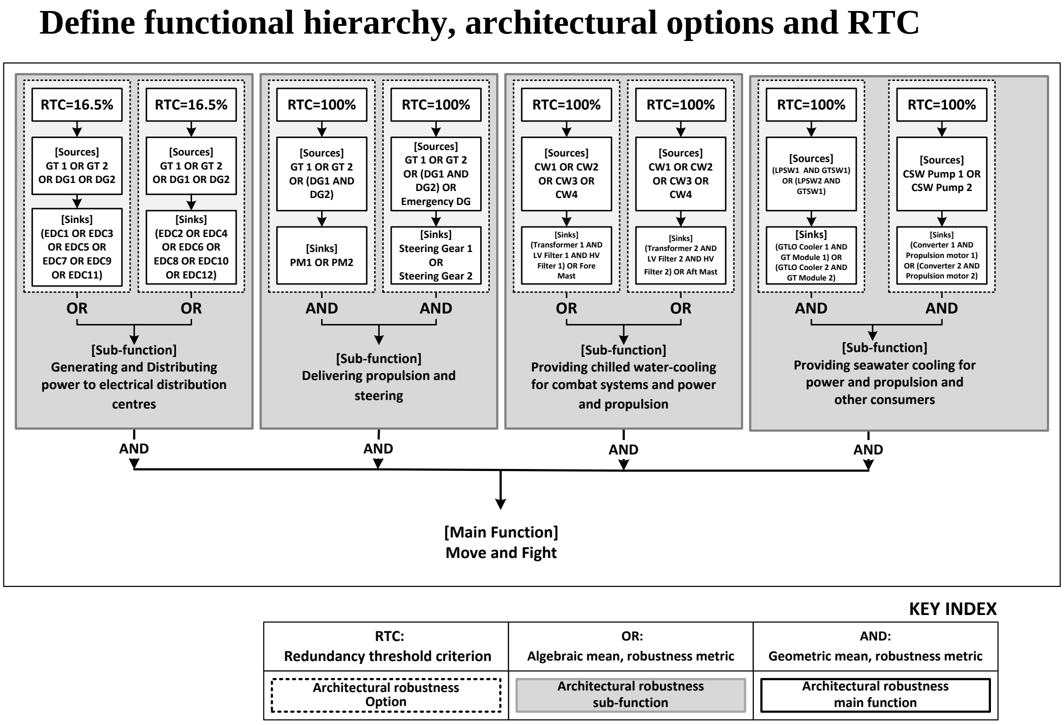


Figure 81: Functional hierarchy, architectural options and RTC for medium redundancy system Type B

Combinatory disruption of two components

An exhaustive combinatory disruptions scenario of a simultaneous disruption of two components was elaborated. All the possible combination of two components disruptions was simulated and the following Table 54 presents the results.

Table 54: Combinatory disruptions of components Type B medium redundancy architecture

	If removed component 1	**and also removed component 2**	**A**	**B**	**C**	**D**	**E**
1	Gas Turbine Alternator 1	HV Switchboard 2	Y	N	Y	Y	N
2	Gas Turbine Alternator 2	HV Switchboard 1	Y	N	Y	Y	N
3	HV Switchboard 1	HV Switchboard 2	N	N	Y	Y	N
4	HV Switchboard 1	Transformer2	N	Y	Y	Y	N
5	HV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
6	HV Switchboard 2	Transformer 1	N	Y	Y	Y	N
7	HV Switchboard 2	LV Switchboard 1	N	Y	Y	Y	N
8	Transformer 1	Transformer2	N	Y	Y	Y	N
9	Transformer 1	LV Switchboard 2	N	Y	Y	Y	N
10	Transformer2	LV Switchboard 1	N	Y	Y	Y	N
11	LV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
12	PM Converter 1	PM Converter 2	Y	Y	Y	N	N
13	PM Converter 1	Propulsion Motor 2	Y	Y	Y	N	N
14	PM Converter 1	Auxiliary CSW pump aft	Y	Y	Y	N	N
15	PM Converter 2	Propulsion Motor 1	Y	Y	Y	N	N
16	PM Converter 2	Auxiliary CSW pump forward	Y	Y	Y	N	N
17	Propulsion Motor 1	Propulsion Motor 2	Y	N	Y	N	N
18	Propulsion Motor 1	Auxiliary CSW pump aft	Y	Y	Y	N	N
19	Propulsion Motor 2	Auxiliary CSW pump forward	Y	Y	Y	N	N
20	Steering Gear Power Pack 1	Steering Gear Power Pack 2	Y	N	Y	Y	N
21	CW Manifold2	CW Manifold3	Y	Y	N	Y	N
22	Auxiliary CSW pump forward	Auxiliary CSW pump aft	Y	Y	Y	N	N
23	LP Seawater Pump 1	LP Seawater Pump 2	Y	Y	Y	N	N
24	LP Seawater Pump 1	GT Cooler2	Y	Y	Y	N	N
25	LP Seawater Pump 2	GT Cooler1	Y	Y	Y	N	N
26	GTSW Pump 1	GTSW Pump 2	Y	Y	Y	N	N
27	GTSW Pump 1	GT Module2	Y	Y	Y	N	N
28	GTSW Pump 2	GT Module1	Y	Y	Y	N	N
29	GT Cooler1	GT Cooler2	Y	Y	Y	N	N
30	GT Module1	GT Module2	Y	Y	Y	N	N
	Total		**9**	**5**	**1**	**17**	**30**

From the 1653 possible combinatory disruptions of two components in a system of 58 there were only 30 combinations found to cause total loss of the main function.

Modularity intermediate loop: Low – medium – high modularity

Table 55: Modular configurations for medium redundancy architecture Type B

Second redundancy loop ***Medium level of redundancy in the system architecture Type B***			
Modul ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
1	Emergency Generator 1, Emergency Switchboard 1, EDC 12, Steering Gear Power Pack 2	Emergency Generator 1, Emergency Switchboard 1	Gas Turbine 1, EDC 7, ChilledWaterPlan3PA, LP Seawater Pump 1, GTSW Pump 1, GT Cooler1, GT Module1
2	Diesel Generator 1, HV Switchboard 1, HV Switchboard Interconnector1, HV Switchboard Interconnector2	Diesel Generator 1, Diesel Generator 2, HV Switchboard 1, HV Switchboard 2, HV Switchboard Interconnector1, HV Switchboard Interconnector2	Emergency Generator 1, Emergency Switchboard 1, LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11, LV Switchboard Interconnector1, LV Switchboard Interconnector2, Steering Gear Power Pack 1, ChilledWaterPlan4SA, ChilledWaterPlan4SA-Manifold
3	Diesel Generator 2, HV Switchboard 2, Transformer2	LV Switchboard 1, EDC 1, EDC 3, EDC 11, LV Switchboard Interconnector1, Steering Gear Power Pack 1	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward
4	LV Switchboard 1, EDC 1, EDC 3, EDC 11, Steering Gear Power Pack 1	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward	Diesel Generator 2, HV Switchboard 2, HDE 5 -HV Filter2, HV Switchboard Interconnector1, HV Switchboard Interconnector2, PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft
5	EDC 4, ChilledWaterPlan2SF, Fore mast	LV Switchboard 2, EDC 10, EDC 12, LV Switchboard Interconnector2, Steering Gear Power Pack 2	EDC 2, ChilledWaterPlan1PF, ChilledWaterPlan1PF-Manifold
6	EDC 9, ChilledWaterPlan4SA, ChilledWaterPlan4SA-Manifold	EDC 2, ChilledWaterPlan1PF, ChilledWaterPlan1PF-Manifold	Diesel Generator 1, HV Switchboard 1, Transformer 1, EDC 4, LV Filter 1, HDE 5 -HV Filter1, ChilledWaterPlan2SF, CW Manifold2, Fore mast
7	LV Switchboard 2, EDC 10, LV Switchboard Interconnector1, LV Switchboard Interconnector2	Gas Turbine 1, EDC 7, ChilledWaterPlan3PA, LP Seawater Pump 1 , GTSW Pump 1, GT Cooler1, GT Module1	Transformer2, LV Switchboard 2, EDC 8, EDC 10, EDC 12, LV Filter 2, Steering Gear Power Pack 2, CW Manifold3, Aft mast

Second redundancy loop

Medium level of redundancy in the system architecture Type B

Modul ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
8	Transformer 1, LV Filter 1, HDE 5 -HV Filter1, CW Manifold2	Transformer2, EDC 9, LV Filter 2, HDE 5 -HV Filter2, ChilledWaterPlan4SA, CW Manifold3, ChilledWaterPlan4SA-Manifold	Gas Turbine 2, EDC 6, LP Seawater Pump 2, GTSW Pump 2, GT Cooler2, GT Module2
9	LV Filter 2, HDE 5 -HV Filter2, ChilledWaterPlan3PA, CW Manifold3	Gas Turbine 2, EDC 6, LP Seawater Pump 2, GTSW Pump 2, GT Cooler2, GT Module2	
10	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward	Transformer 1, EDC 4, LV Filter 1, HDE 5 -HV Filter1, ChilledWaterPlan2SF, CW Manifold2, Fore mast	
11	PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft	EDC 8, PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft, Aft mast	
12	EDC 2, ChilledWaterPlan1PF, ChilledWaterPlan1PF-Manifold		
13	Gas Turbine 1, EDC 7, LP Seawater Pump 1, GTSW Pump 1, GT Cooler1, GT Module1		
14	Gas Turbine 2, EDC 6, LP Seawater Pump 2, GTSW Pump 2, GT Cooler2, GT Module2		
15	EDC 8, Aft mast		
Robustness metric	0.805	0.72	0.68
Modularity metric	0.67	0.7	0.75
Classification of modules (Modules ID)			
central modules	**[13,14]**	**[8]**	**[2]**
periphery modules	**[11,15]**	**[1]**	**[3,5]**

3rd Redundancy loop: Low redundancy

In contrary with the high and medium-level redundancy system architectures instances, that for the case study Type B are realistic designs, the low-level redundancy presented in following schematic is a semi-hypothetical architecture, provide herein, as a point of comparison. It is advised that the definition of low redundancy is qualitative in nature and it is different for each case study system architecture design. For example, the Type A low-level redundancy system architecture had been proposed with no redundant power sources where in the Type B low-level redundancy are included a redundant source.

It is expected that also during the application on the proposed methodology the different levels of redundancy (high-medium-low) will relate with an original system architecture that is considered. Because in the Type B system architecture, the starting point of high redundancy was architecture with three diesel generators and two gas turbines, thus the low-level redundancy is one diesel generator and one gas turbine.

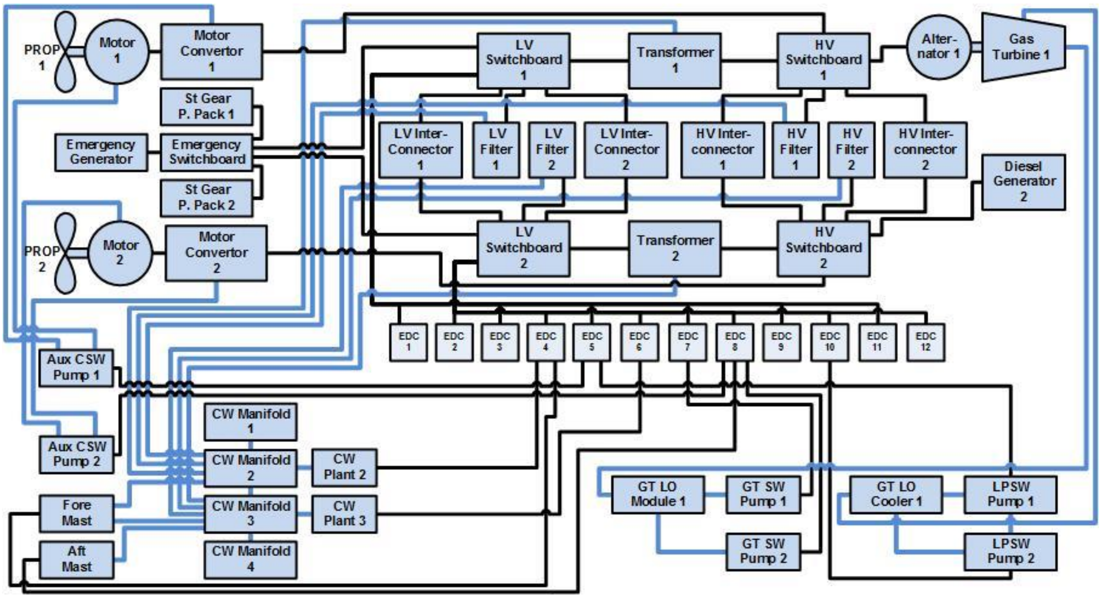


Figure 82: Type B low redundancy schematic

Define functional hierarchy, architectural options and RTC

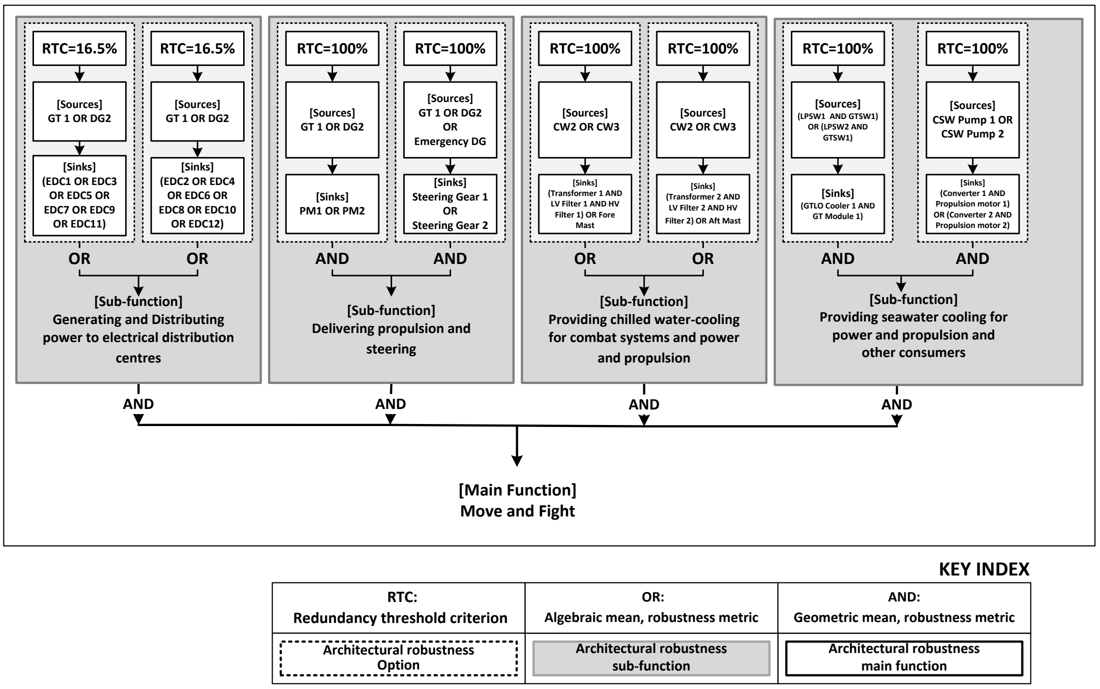


Figure 83: Functional hierarchy, architectural options and RTC for low redundancy system Type B

Combinatory disruption of components

Table 56: Combinatory disruptions of components Type B low redundancy architecture

	If removed component 1	and also removed component 2	A	B	C	D	E
1	Diesel Generator 2	Gas Turbine Alternator 1	N	N	Y	Y	N
2	Diesel Generator 2	HV Switchboard 1	N	N	Y	Y	N
3	Diesel Generator 2	LV Switchboard 2	N	Y	Y	Y	N
4	Diesel Generator 2	Propulsion Motor 1	Y	N	Y	Y	N
5	Gas Turbine Alternator 1	HV Switchboard 2	N	N	Y	Y	N
6	Gas Turbine Alternator 1	LV Switchboard 1	N	Y	Y	Y	N
7	Gas Turbine Alternator 1	Propulsion Motor 2	Y	N	Y	Y	N
8	HV Switchboard 1	HV Switchboard 2	N	N	Y	Y	N
9	HV Switchboard 1	Transformer2	N	Y	Y	Y	N
10	HV Switchboard 1	LV Switchboard 1	N	Y	Y	Y	N
11	HV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
12	HV Switchboard 1	LV Switchboard Interconnector2	N	Y	Y	Y	N
13	HV Switchboard 1	Propulsion Motor 2	Y	N	Y	Y	N
14	HV Switchboard 2	Transformer 1	N	Y	Y	Y	N
15	HV Switchboard 2	LV Switchboard 1	N	Y	Y	Y	N
16	HV Switchboard 2	LV Switchboard 2	N	Y	Y	Y	N
17	HV Switchboard 2	LV Switchboard Interconnector1	N	Y	Y	Y	N
18	HV Switchboard 2	Propulsion Motor 1	Y	N	Y	Y	N
19	Transformer 1	Transformer2	N	Y	Y	Y	N
20	Transformer 1	LV Switchboard 2	N	Y	Y	Y	N

	If removed component 1	and also removed component 2	A	B	C	D	E
21	Transformer 1	CW Manifold3	Y	Y	N	Y	N
22	Transformer2	LV Switchboard 1	N	Y	Y	Y	N
23	LV Switchboard 1	LV Switchboard 2	N	Y	Y	Y	N
24	LV Switchboard 1	HV Switchboard Interconnector1	N	Y	Y	Y	N
25	LV Switchboard 2	HV Switchboard Interconnector2	N	Y	Y	Y	N
26	LV Filter 1	CW Manifold3	Y	Y	N	Y	N
27	HDE 5 -HV Filter1	CW Manifold3	Y	Y	N	Y	N
28	PM Converter 1	PM Converter 2	Y	Y	Y	N	N
29	PM Converter 1	Propulsion Motor 2	Y	Y	Y	N	N
30	PM Converter 1	Auxiliary CSW pump aft	Y	Y	Y	N	N
31	PM Converter 2	Propulsion Motor 1	Y	Y	Y	N	N
32	PM Converter 2	Auxiliary CSW pump forward	Y	Y	Y	N	N
33	Propulsion Motor 1	Propulsion Motor 2	Y	N	Y	N	N
34	Propulsion Motor 1	Auxiliary CSW pump aft	Y	Y	Y	N	N
35	Propulsion Motor 2	Auxiliary CSW pump forward	Y	Y	Y	N	N
36	Steering Gear Power Pack 1	Steering Gear Power Pack 2	Y	N	Y	Y	N
37	ChilledWaterPlan2SF	ChilledWaterPlan3PA	Y	Y	N	Y	N
38	ChilledWaterPlan2SF	CW Manifold3	Y	Y	N	Y	N
39	ChilledWaterPlan3PA	CW Manifold2	Y	Y	N	Y	N
40	CW Manifold2	CW Manifold3	Y	Y	N	Y	N
41	Auxiliary CSW pump forward	Auxiliary CSW pump aft	Y	Y	Y	N	N
	Total		**20**	**10**	**7**	**9**	**41**

A total of 41 combinatory combinations where identified from a total possible 1326 combination amongst 52 components in the system architecture in Table 56. The combination of disruption of Chilled Water Plan 2 Manifold and 3 Manifold is illustrated in the following Figure 84.

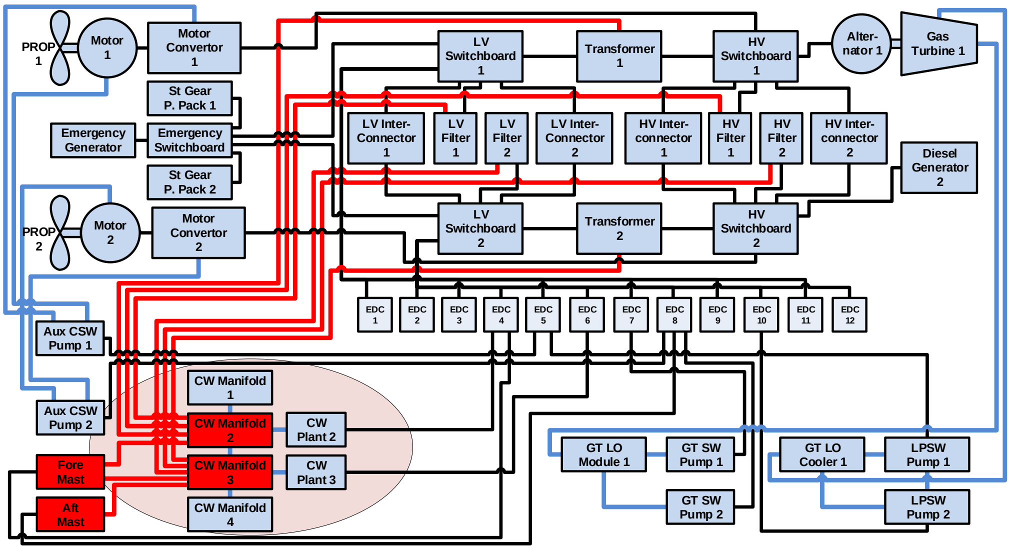


Figure 84: Type B medium redundancy schematic showing combined disruption of CW Manifold 2 and 3

Chilled Water Manifold 2 and in combination with a disruption LP 3 manifold lead on both fore and aft mast to be loss and also both power architectural options. This is a total loss failure of the main function of the system.

Modularity intermediate loop: Low – medium – high modularity

Table 57: Modular configurations for low redundancy architecture Type B

Third redundancy loop *Low-level of redundancy in the system architecture Type B*			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
1	Emergency Generator 1, Emergency Switchboard 1, EDC 12, Steering Gear Power Pack 2	HV Switchboard 1, Transformer 1, HDE 5 -HV Filter1, HV Switchboard Interconnector1	Diesel Generator 2, HV Switchboard 1, HV Switchboard 2, Transformer 1, HV Switchboard Interconnector1, HV Switchboard Interconnector2
2	HV Switchboard 1, Transformer 1, HDE 5 -HV Filter1, HV Switchboard Interconnector2	Diesel Generator 2, HV Switchboard 2, Transformer2, HDE 5 -HV Filter2, HV Switchboard Interconnector2	Emergency Generator 1, Emergency Switchboard 1, LV Switchboard 2, EDC 2, EDC 10, EDC 12, Steering Gear Power Pack 2
3	Diesel Generator 2, HV Switchboard 2, Transformer2, HV Switchboard Interconnector1	LV Switchboard 1, EDC 1, EDC 3, EDC 9, LV Filter 1, LV Switchboard Interconnector1	Gas Turbine Alternator 1, EDC 6, EDC 7, LP Seawater Pump 1, LP Seawater Pump 2, GTSW Pump 1 , GTSW Pump 2, GT Cooler1, GT Module1
4	LV Switchboard 1, EDC 1, EDC 3, EDC 9, LV Switchboard Interconnector2	EDC 8, PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft, Aft mast	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward
5	LV Switchboard 2, EDC 2, EDC 10, LV Switchboard Interconnector1	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward	EDC 8, PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft, Aft mast
6	EDC 6, LP Seawater Pump 2, GTSW Pump 2	Emergency Generator 1, Emergency Switchboard 1, EDC 11, Steering Gear Power Pack 1	LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11, LV Switchboard Interconnector1, LV Switchboard Interconnector2, Steering Gear Power Pack 1
7	EDC 5, PM Converter 1, Propulsion Motor 1, Auxiliary CSW pump forward	LV Switchboard 2, EDC 2, EDC 10, EDC 12, LV Switchboard Interconnector2, Steering Gear Power Pack 2	EDC 4, LV Filter 1, HDE 5 -HV Filter1, ChilledWaterPlan2SF, ChilledWaterPlan1PF-Manifold, CW Manifold2, Fore mast
8	PM Converter 2, Propulsion Motor 2, Auxiliary CSW pump aft	EDC 4, ChilledWaterPlan2SF, ChilledWaterPlan1PF-Manifold, CW Manifold2, Fore mast	Transformer2, LV Filter 2, HDE 5 -HV Filter2, ChilledWaterPlan3PA, CW Manifold3, ChilledWaterPlan4SA-Manifold
9	EDC 11, Steering Gear Power Pack 1	LV Filter 2, ChilledWaterPlan3PA, CW Manifold3, ChilledWaterPlan4SA-Manifold	

Third redundancy loop

Low-level of redundancy in the system architecture Type B

Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
10	LV Filter 1, ChilledWaterPlan2SF, ChilledWaterPlan1PF-Manifold, CW Manifold2	Gas Turbine Alternator 1, EDC 6, EDC 7, LP Seawater Pump 1, LP Seawater Pump 2, GTSW Pump 1 , GTSW Pump 2, GT Cooler1, GT Module1	
11	LV Filter 2, HDE 5 -HV Filter2, CW Manifold3		
12	EDC 7, ChilledWaterPlan3PA, ChilledWaterPlan4SA-Manifold		
13	Gas Turbine Alternator 1, LP Seawater Pump 1, GT Cooler1		
14	GTSW Pump 1, GT Module1		
15	EDC 4, Fore mast		
16	EDC 8, Aft mast		
Robustness metric	0.823	0.73	0.659
Modularity metric	0.6	0.71	0.75
Classification of modules (Modules ID)			
central modules	[4]	[10]	[3]
periphery modules	[8,9,14,15,16]	[5,9]	[4]

Type C: Modern Frigate – Analysis stage results

In this section the analysis stage results of the nested redundancy, modularity and robustness loops are presented.

1st Redundancy loop: High redundancy

The paragraphs present the robustness results for high-level of redundancy

Combinatory disruption of components

Table 58: Combinatory disruptions of components Type C high redundancy architecture

	If removed component 1	**and also removed component 2**	**A**	**B**	**C**	**D**	**E**
1	HV Switchboard 1 forward	HV Switchboard 2 aft	N	N	Y	Y	N
2	HV Switchboard 1 forward	Transformer2	N	N	Y	Y	N
3	HV Switchboard 1 forward	LV Switchboard 2	N	N	Y	Y	N
4	HV Switchboard 2 aft	Transformer 1	N	N	Y	Y	N
5	HV Switchboard 2 aft	LV Switchboard 1	N	N	Y	Y	N
6	Transformer 1	Transformer2	N	N	Y	Y	N
7	Transformer 1	LV Switchboard 2	N	N	Y	Y	N
8	Transformer 1	LPSW Manifold 2	Y	Y	Y	N	N
9	Transformer2	LV Switchboard 1	N	N	Y	Y	N
10	Transformer2	LPSW Manifold 1	Y	Y	Y	N	N
11	LV Switchboard 1	LV Switchboard 2	N	N	Y	Y	N
12	PM Converter 1	LPSW Manifold 2	Y	Y	Y	N	N
13	PM Converter 2	LPSW Manifold 1	Y	Y	Y	N	N
14	Propulsion Motor 1	LPSW Manifold 2	Y	Y	Y	N	N
15	Propulsion Motor 2	LPSW Manifold 1	Y	Y	Y	N	N
16	Steering Gear Power Pack 1	Steering Gear Power Pack 2	Y	N	Y	Y	N
17	CW Manifold1	CW Manifold2	Y	Y	N	Y	N
18	CW Manifold1	Radar	Y	Y	N	Y	N
19	CW Manifold2	CW Manifold4	Y	Y	N	Y	N
20	CW Manifold2	Computer room HVAC 2	Y	Y	N	Y	N
21	CW Manifold2	Sonar	Y	Y	N	Y	N
22	CW Manifold4	Computer room HVAC 1	Y	Y	N	Y	N
23	LPSW Manifold 1	LPSW Manifold 2	Y	Y	Y	N	N
24	Radar	Sonar	Y	Y	N	Y	N
25	Computer room HVAC 1	Computer room HVAC 2	Y	Y	N	Y	N
	Total		**9**	**10**	**8**	**7**	**25**

Table 58 above encapsulates the combined disruption of two components. The Type C system has a total of 53 components. 1378 combination of two components were simulated and only 25 were found to result in a total loss of robustness.

Modularity intermediate loop: Low – medium – high modularity

Table 59: Modular configurations for high redundancy architecture Type C

First redundancy loop ***high-level of redundancy in the system architecture Type C***			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
1	Diesel Generator 1, Diesel Generator 3, HV Switchboard 1 forward, HV Interconnector HVSW1 to HVSW2, HV Interconnector HVSW2to HVSW1	Diesel Generator 1, Diesel Generator 3, HV Switchboard 1 forward, Transformer 1, HV Interconnector HVSW1 to HVSW2, HV Interconnector HVSW2to HVSW1	Diesel Generator 2, Diesel Generator 4, HV Switchboard 2 aft, Transformer2, HV Interconnector HVSW1 to HVSW2, HV Interconnector HVSW2to HVSW1, ChilledWaterPlan2SF, ChilledWaterPlan4SA
2	Diesel Generator 2, Diesel Generator 4, HV Switchboard 2 aft, Transformer2	Diesel Generator 2, Diesel Generator 4, HV Switchboard 2 aft, Transformer2, ChilledWaterPlan2SF	LV Switchboard 1, LV Switchboard 2, EDC 1, EDC 2, EDC 3, EDC 4, EDC 9, EDC 10, EDC 11, EDC 12, LV interconnector LVSW1 to LVSW2, LV interconnector LVSW2 to LVSW1, Steering Gear Power Pack 1, Steering Gear Power Pack 2
3	LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11, LV interconnector LVSW2 to LVSW1	LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11, LV interconnector LVSW1 to LVSW2, LV interconnector LVSW2 to LVSW1	Diesel Generator 1, Diesel Generator 3, HV Switchboard 1 forward, Transformer 1, EDC 5, EDC 6, PM Converter 1, Propulsion Motor 1, LP Seawater Pump 1, LP Seawater Pump 2, LPSW Manifold 1
4	LV Switchboard 2, EDC 2, EDC 4, EDC 10, EDC 12, LV interconnector LVSW1 to LVSW2	LV Switchboard 2, EDC 2, EDC 4, EDC 6, EDC 10, EDC 12, Steering Gear Power Pack 1	EDC 7, EDC 8, Gas turbine, Gearbox, PM Converter 2, Propulsion Motor 2, LP Seawater Pump 3, LP Seawater Pump 4, LPSW Manifold 2, GT Lubricating & Fuel Oil Module 1
5	EDC 6, LP Seawater Pump 2	EDC 5, PM Converter 1, Propulsion Motor 1, LP Seawater Pump 1, LP Seawater Pump 2, LPSW Manifold 1	ChilledWaterPlan1PF, ChilledWaterPlan3PA, CW Manifold1, CW Manifold2, CW Manifold3, CW Manifold4, Radar, Computer room HVAC 1, Computer room HVAC 2, Sonar
6	EDC 7, LP Seawater Pump 3	EDC 8, LP Seawater Pump 4	
7	EDC 8, LP Seawater Pump 4	Gas turbine, Gearbox, GT Lubricating & Fuel Oil Module 1	
8	Gas turbine, Gearbox, GT Lubricating & Fuel Oil Module 1	PM Converter 2, Propulsion Motor 2, Steering Gear Power Pack 2, LPSW Manifold 2	
9	Transformer 1, PM Converter 1, Propulsion Motor 1, LPSW Manifold 1	ChilledWaterPlan3PA, ChilledWaterPlan4SA, CW Manifold3, CW Manifold4, Computer room HVAC 2	
10	PM Converter 2, Propulsion Motor 2, LPSW Manifold 2	EDC 7, LP Seawater Pump 3	

First redundancy loop ***high-level of redundancy in the system architecture Type C***			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
11	Steering Gear Power Pack 1	ChilledWaterPlan1PF, CW Manifold1, CW Manifold2, Radar, Computer room HVAC 1, Sonar	
12	Steering Gear Power Pack 2		
13	ChilledWaterPlan2SF, CW Manifold2, Radar, Computer room HVAC 1		
14	ChilledWaterPlan3PA, ChilledWaterPlan4SA, CW Manifold3, CW Manifold4, Computer room HVAC 2		
15	EDC 5, LP Seawater Pump 1		
16	ChilledWaterPlan1PF, CW Manifold1, Sonar		
Robustness metric	0.818	0.683	0.559
Modularity metric	0.64	0.66	0.74
Classification of modules (Modules ID)			
central modules	[3,4]	[11]	[4]
periphery modules	[5,6,7,10,15]	[6,10]	[1]

2nd Redundancy loop: Medium redundancy

The second redundancy loop experiment with the level of redundancy of the diesel generators. Given that redundancy is introduced in the systems for reliability reasons also, is suggested that medium redundancy system architecture has two DGs of higher performance, and reliability characteristics. Additionally, the revision of the Type C system architecture has two less CW plans.

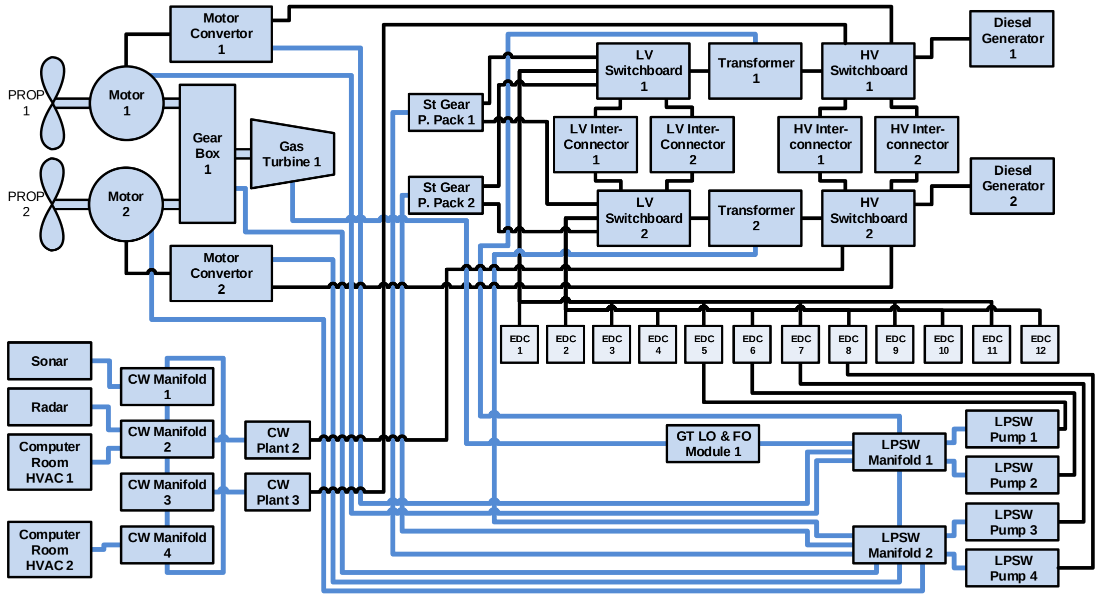


Figure 85: Type C medium redundancy schematic

Define functional hierarchy, architectural options and RTC

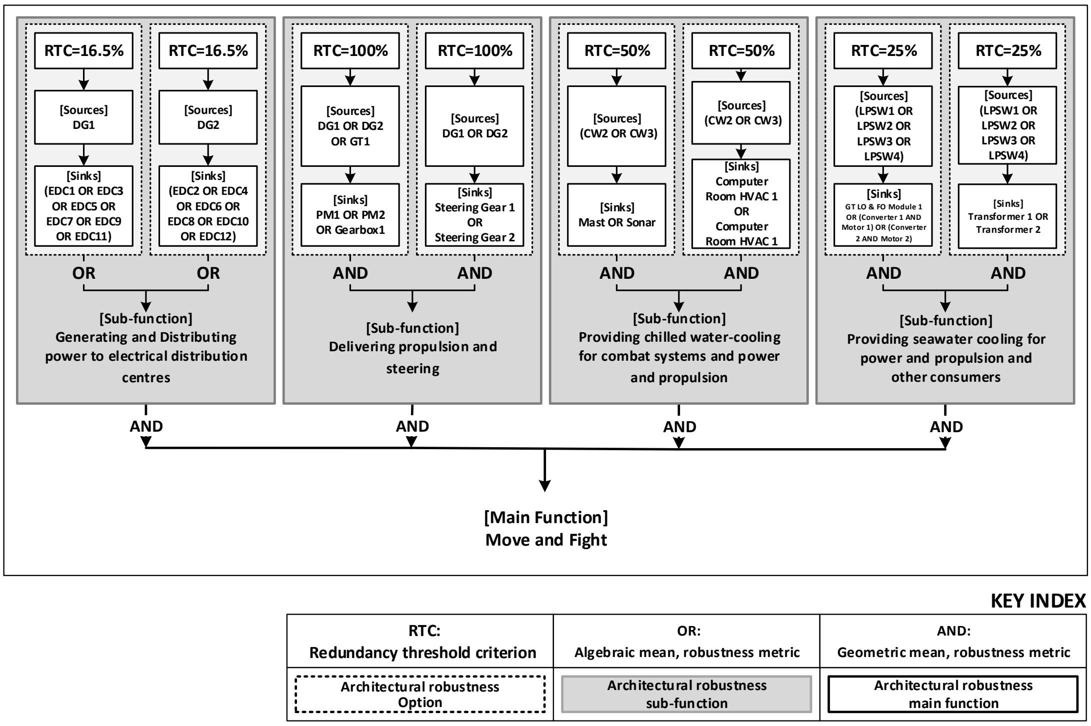


Figure 86: Functional hierarchy, architectural options and RTC for medium redundancy Type C

Combinatory disruption of components

The system Type C has a total of 44 components which lead on a possible of 1176 combination of two components.

Table 60: Combinatory disruptions of components Type C medium redundancy architecture

	If removed component 1	**and also removed component 2**	**A**	**B**	**C**	**D**	**E**
1	Diesel Generator 1	Diesel Generator 2	N	N	Y	Y	N
2	Diesel Generator 1	HV Switchboard 2 aft	N	N	Y	Y	N
3	Diesel Generator 1	Steering Gear Power Pack 2	Y	N	Y	Y	N
4	Diesel Generator 2	HV Switchboard 1 forward	N	N	Y	Y	N
5	Diesel Generator 2	Steering Gear Power Pack 1	Y	N	Y	Y	N
6	HV Switchboard 1 forward	HV Switchboard 2 aft	N	N	Y	Y	N
7	HV Switchboard 1 forward	Transformer2	N	N	Y	Y	N
8	HV Switchboard 1 forward	LV Switchboard 2	N	N	Y	Y	N
9	HV Switchboard 1 forward	Steering Gear Power Pack 2	Y	N	Y	Y	N
10	HV Switchboard 2 aft	Transformer 1	N	N	Y	Y	N
11	HV Switchboard 2 aft	LV Switchboard 1	N	N	Y	Y	N
12	HV Switchboard 2 aft	Steering Gear Power Pack 1	Y	N	Y	Y	N
13	Transformer 1	Transformer2	N	N	Y	N	N
14	Transformer 1	LV Switchboard 2	N	N	Y	Y	N
15	Transformer 1	PM Converter 2	Y	Y	Y	N	N
16	Transformer 1	Propulsion Motor 2	Y	Y	Y	N	N
17	Transformer 1	LPSW Manifold 2	Y	Y	Y	N	N
18	Transformer2	LV Switchboard 1	N	N	Y	Y	N
19	Transformer2	PM Converter 1	Y	Y	Y	N	N
20	Transformer2	Propulsion Motor 1	Y	Y	Y	N	N
21	Transformer2	LPSW Manifold 1	Y	Y	Y	N	N
22	LV Switchboard 1	LV Switchboard 2	N	N	Y	Y	N
23	PM Converter 1	PM Converter 2	Y	Y	Y	N	N
24	PM Converter 1	Propulsion Motor 2	Y	Y	Y	N	N
25	PM Converter 1	LPSW Manifold 2	Y	Y	Y	N	N
26	PM Converter 2	Propulsion Motor 1	Y	Y	Y	N	N
27	PM Converter 2	LPSW Manifold 1	Y	Y	Y	N	N
28	Propulsion Motor 1	Propulsion Motor 2	Y	Y	Y	N	N
29	Propulsion Motor 1	LPSW Manifold 2	Y	Y	Y	N	N
30	Propulsion Motor 2	LPSW Manifold 1	Y	Y	Y	N	N
31	Steering Gear Power Pack 1	Steering Gear Power Pack 2	Y	N	Y	Y	N
32	ChilledWaterPlan2SF	ChilledWaterPlan3PA	Y	Y	N	Y	N
33	ChilledWaterPlan2SF	CW Manifold3	Y	Y	N	Y	N
34	ChilledWaterPlan3PA	CW Manifold2	Y	Y	N	Y	N
35	CW Manifold1	CW Manifold2	Y	Y	N	Y	N
36	CW Manifold1	Radar	Y	Y	N	Y	N
37	CW Manifold2	CW Manifold3	Y	Y	N	Y	N
38	CW Manifold2	CW Manifold4	Y	Y	N	Y	N

	If removed component 1	and also removed component 2	A	B	C	D	E
39	CW Manifold2	Computer room HVAC 2	Y	Y	N	Y	N
40	CW Manifold2	Sonar	Y	Y	N	Y	N
41	CW Manifold4	Computer room HVAC 1	Y	Y	N	Y	N
42	LPSW Manifold 1	LPSW Manifold 2	Y	Y	Y	N	N
43	Radar	Sonar	Y	Y	N	Y	N
44	Computer room HVAC 1	Computer room HVAC 2	Y	Y	N	Y	N
	Total		**12**	**17**	**12**	**16**	**44**

Modularity intermediate loop: Low-medium-high

Table 61: Modular configurations for medium redundancy architecture Type C

Second redundancy loop *Medium level of redundancy in the system architecture Type C*			
Module ID	**First modularity loop** *low-level of modularity*	**Second modularity loop** *medium level of modularity*	**Third modularity loop** *high-level of modularity*
1	Diesel Generator 1, HV Switchboard 1 forward, Transformer 1, HV Interconnector HVSW1 to HVSW2	Diesel Generator 1, HV Switchboard 1 forward, Transformer 1, HV Interconnector HVSW2to HVSW1, ChilledWaterPlan3PA	Diesel Generator 1, Diesel Generator 2, HV Switchboard 1 forward, HV Switchboard 2 aft, Transformer 1, Transformer2, HV Interconnector HVSW1 to HVSW2, HV Interconnector HVSW2to HVSW1
2	Diesel Generator 2, HV Switchboard 2 aft, Transformer2, HV Interconnector HVSW2to HVSW1	Diesel Generator 2, HV Switchboard 2 aft, Transformer2, HV Interconnector HVSW1 to HVSW2, ChilledWaterPlan2SF	LV Switchboard 2, EDC 2, EDC 4, EDC 8, EDC 10, EDC 12, LV interconnector LVSW2 to LVSW1, LP Seawater Pump 4
3	LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11, LV interconnector LVSW1 to LVSW2	LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11, Steering Gear Power Pack 2	LV Switchboard 1, EDC 1, EDC 3, EDC 5, EDC 7, EDC 9, EDC 11, LV interconnector LVSW1 to LVSW2, LP Seawater Pump 1
4	LV Switchboard 2, EDC 2, EDC 4, EDC 10, EDC 12, LV interconnector LVSW2 to LVSW1	LV Switchboard 2, EDC 2, EDC 4, EDC 10, EDC 12, LV interconnector LVSW1 to LVSW2, LV interconnector LVSW2 to LVSW1	EDC 6, Gas turbine, Gearbox, PM Converter 1, Propulsion Motor 1, LP Seawater Pump 2, LPSW Manifold 1, GT Lubricating & Fuel Oil Module 1
5	EDC 5, LP Seawater Pump 1	EDC 7, LP Seawater Pump 3	PM Converter 2, Propulsion Motor 2, Steering Gear Power Pack 1, Steering Gear Power Pack 2, LP Seawater Pump 3, LPSW Manifold 2
6	EDC 6, LP Seawater Pump 2	EDC 8, LP Seawater Pump 4	ChilledWaterPlan2SF, ChilledWaterPlan3PA, CW Manifold1, CW Manifold2, CW Manifold3, CW Manifold4, Radar, Computer room HVAC 1, Computer room HVAC 2, Sonar
7	EDC 7, LP Seawater Pump 3	Gas turbine, Gearbox, GT Lubricating & Fuel Oil Module 1	

Second redundancy loop ***Medium level of redundancy in the system architecture Type C***			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
8	EDC 8, LP Seawater Pump 4	EDC 6, PM Converter 1, Propulsion Motor 1, LP Seawater Pump 2, LPSW Manifold 1	
9	Gas turbine, Gearbox, GT Lubricating & Fuel Oil Module 1	PM Converter 2, Propulsion Motor 2, Steering Gear Power Pack 1, LPSW Manifold 2	
10	PM Converter 1, Propulsion Motor 1, LPSW Manifold 1	CW Manifold1, CW Manifold2, CW Manifold3, CW Manifold4, Radar, Computer room HVAC 1, Computer room HVAC 2, Sonar	
11	PM Converter 2, Propulsion Motor 2, LPSW Manifold 2	EDC 5, LP Seawater Pump 1	
12	Steering Gear Power Pack 1		
13	Steering Gear Power Pack 2		
14	ChilledWaterPlan2SF, CW Manifold2, Radar, Computer room HVAC 1		
15	ChilledWaterPlan3PA, CW Manifold3, CW Manifold4, Computer room HVAC 2		
16	CW Manifold1, Sonar		
Robustness Metric	0.818	0.69	0.36
Modularity metric	0.63	0.68	0.7
Classification of modules (Modules ID)			
central modules	[3,4]	[10]	[6]
periphery modules	[5,6,7,8,10,11,16]	[5,6,11]	[5]

3rd Redundancy loop: Low redundancy

Type A and Type C system architectures share similarities as they are both designed for frigate ships. Type C is the novel design replacing the Type A. The approach that it was adopted in order to develop a semi-hypothetical example of low-level redundancy system architecture was the same as Type A. As was previously discussed this is presented here only to exemplify the proposed methodology and is not suggested as a realistic design.

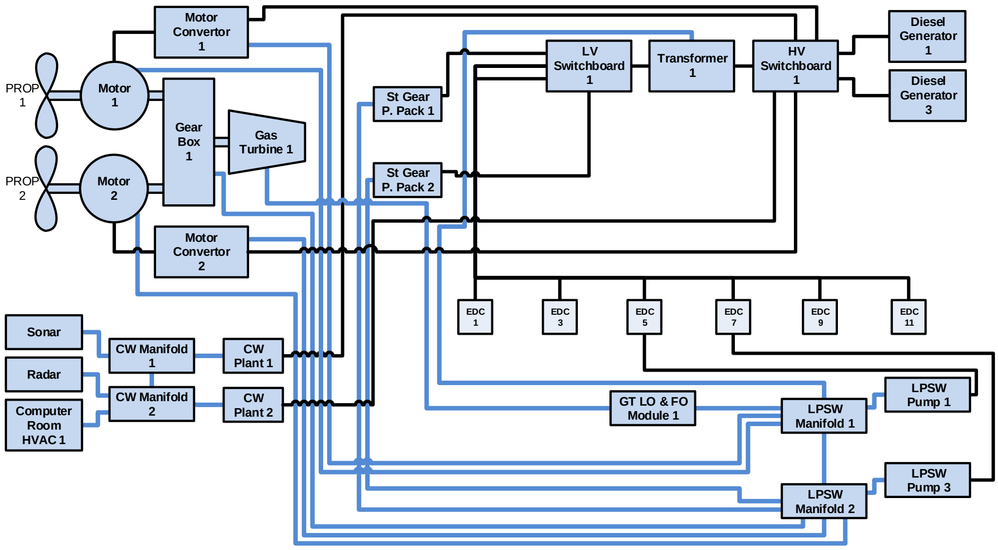


Figure 87: Type C low redundancy schematic

Define functional hierarchy, architectural options and RTC

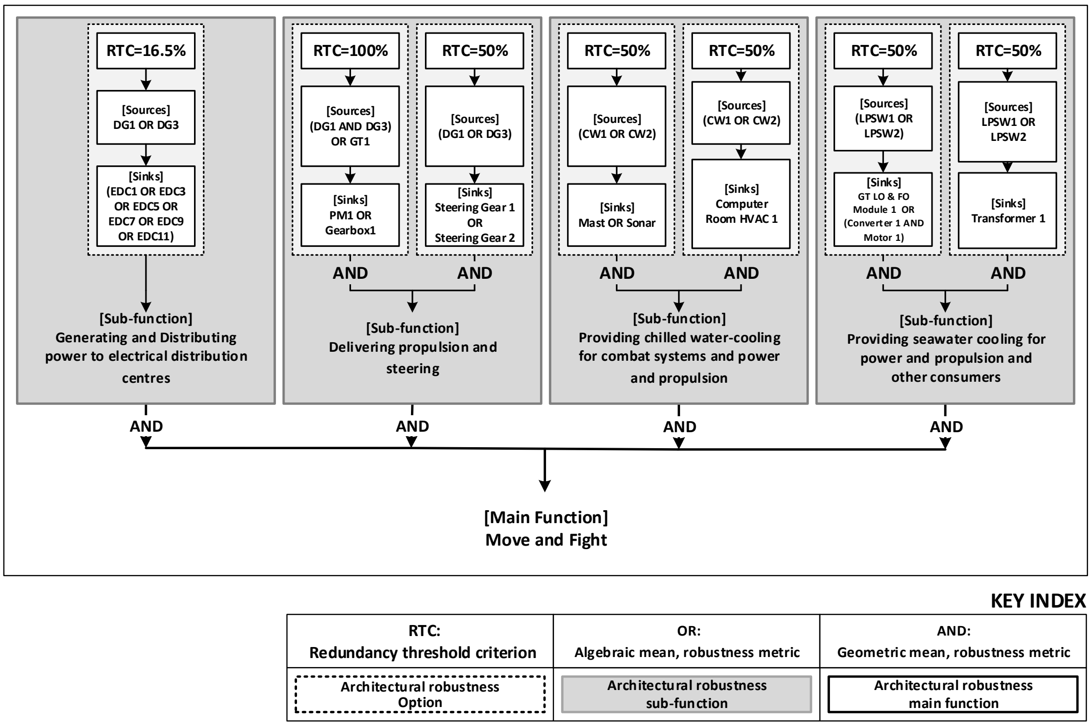


Figure 88: Functional hierarchy, architectural options and RTC for low redundancy system Type C

Combinatory disruption of components

Similar with the case study low redundancy system architecture Type A, herein only the total results are provided. It is apparent that the low redundancy system architecture is not a reasonable or appropriate architecture as there is a high combination of two components disruption that could lead in the system total loss. The proposed methodology results confirm the expected behaviour of such system architecture design.

Table 62: Combinatory disruptions of components Type C low redundancy architecture

If removed component 1	**and also removed component 2**	**A**	**B**	**C**	**D**	**E**
Total		**88**	**97**	**93**	**6**	**187**

Modularity intermediate loop: Low – medium – high modularity

Table 63: Modular configurations for low redundancy architecture Type C

Third redundancy loop ***low-level of redundancy in the system architecture Type C***			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
1	Diesel Generator 1, Diesel Generator 3, HV Switchboard 1 forward	Diesel Generator 1, Diesel Generator 3, HV Switchboard 1 forward, Transformer 1	Diesel Generator 1, Diesel Generator 3, HV Switchboard 1 forward, Transformer 1
2	Transformer 1	LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11, Steering Gear Power Pack 1	EDC 5, PM Converter 1, Propulsion Motor 1, LP Seawater Pump 1, LPSW Manifold 1
3	LV Switchboard 1, EDC 1, EDC 3, EDC 9, EDC 11	PM Converter 2, Propulsion Motor 2, Steering Gear Power Pack 2, LPSW Manifold 2	LV Switchboard 1, EDC 1, EDC 3, EDC 7, EDC 9, EDC 11, LP Seawater Pump 3
4	EDC 7, LP Seawater Pump 3	PM Converter 1, Propulsion Motor 1, LPSW Manifold 1	Gas turbine, Gearbox, GT Lubricating & Fuel Oil Module 1
5	Steering Gear Power Pack 1, Steering Gear Power Pack 2, LPSW Manifold 2	ChilledWaterPlan1PF, CW Manifold1, Sonar	PM Converter 2, Propulsion Motor 2, Steering Gear Power Pack 1, Steering Gear Power Pack 2, LPSW Manifold 2
6	PM Converter 1, Propulsion Motor 1, LPSW Manifold 1	EDC 5, LP Seawater Pump 1	ChilledWaterPlan1PF, ChilledWaterPlan2SF, CW Manifold1, CW Manifold2, Radar, Computer room HVAC 1, Sonar
7	PM Converter 2, Propulsion Motor 2	EDC 7, LP Seawater Pump 3	
8	ChilledWaterPlan2SF, CW Manifold2, Radar, Computer room HVAC 1	Gas turbine, Gearbox, GT Lubricating & Fuel Oil Module 1	
9	EDC 5, LP Seawater Pump 1	ChilledWaterPlan2SF, CW Manifold2, Radar, Computer room HVAC 1	

Third redundancy loop ***low-level of redundancy in the system architecture Type C***			
Module ID	**First modularity loop** ***low-level of modularity***	**Second modularity loop** ***medium level of modularity***	**Third modularity loop** ***high-level of modularity***
10	Gas turbine, Gearbox, GT Lubricating & Fuel Oil Module 1		
11	ChilledWaterPlan1PF, CW Manifold1, Sonar		
Robustness Metric	0.37	0.446	0.273
Modularity metric	0.64	0.72	0.8
Classification of modules (Modules ID)			
central modules	[3]	[2]	[6]
periphery modules	[4,6,7,9]	[4,6,7]	[4]

Appendix III: Structural patterns and key features

The technical network A and B are described in the Case Studies Chapter 8, specifically Type A in Section 8.2.1 and Type B Appendix II (medium redundancy system architecture option). This preliminary assessment of the technical systems as networks was performed prior to the evaluation part of the study. The empirical investigation allows to collect observation of the structural patterns and features of the networks that aid on informing the development of the generator in combination with the findings of the literature.

Both networks (Figure 89 & 90) are shown a cyclical configuration, reflecting the interdependency of subsystems components. For example, electrical power generation requires LPSW cooling, but the LPSW pumps require electrical power to function. Such interdependencies create the cyclical (loop) pattern. The cyclical pattern enables the functionality of these complex and robust Type A and B technical systems

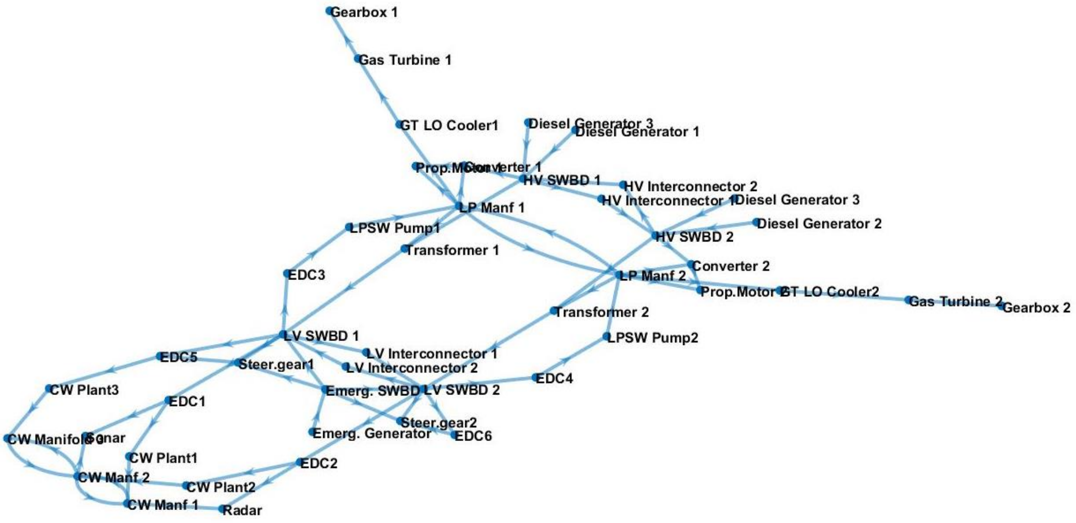


Figure 89: Technical network Type A

Figure 90: Technical network Type B

Robustness and modularity results

This section presents the results of the assessment of the two technical system architectures based on modularity and robustness metrics and methods. Figure 91 shows the robustness response curves for random and targeted disruptions. For random attacks, Type B is more robust than Type A, consistent with expert opinion. However, in targeted attacks, the overall response of the two systems falls rapidly as increasing numbers of nodes are removed. Type B displays better robust behaviour under attacks when up to four central components are removed, with five or more the robustness collapses. Large targeted disruptions suggest multiple distributed attacks; however, the systems are designed to survive a single event, meaning that multiple attacks are not representative of the operational environment.

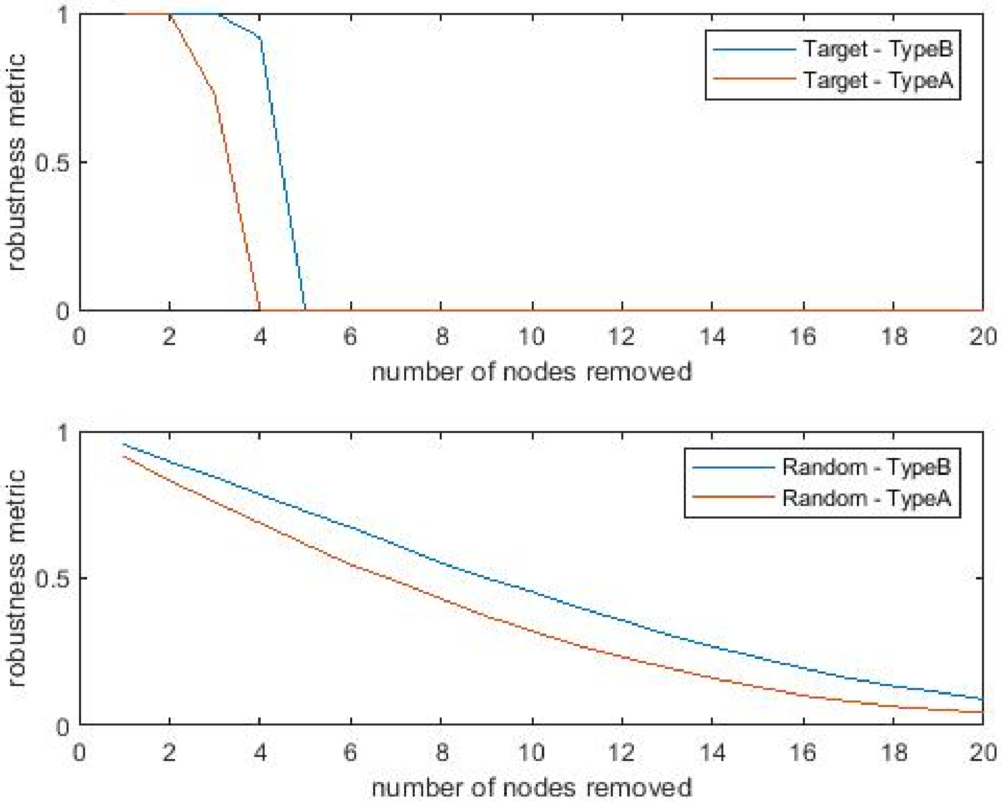


Figure 91: Robustness for Type A and B technical system architectures

In naval designs spatial separation of critical, duplicated components minimises the damage profile and limits the number of nodes lost. The development of future improved designs with a more linear robustness response to targeted attacks could provide improved survivability and operational capability in an evolving threat environment.

Table 64 shows the Newman modularity index for the two architectures. Type A has a higher degree of modularity, with simpler systems and fewer interconnections, whereas Type B has a lower modularity, with increased levels of interconnections (due to additional redundancy) designed to give improved robustness.

Table 64: Modularity for Type A and B technical system architectures

Technical systems	**Modularity**
Type A	0.63
Type B	0.48

It is observed that efforts to increase the level of robustness through additional redundancy in Type B lead to a more complex architecture compared to Type A.

The Type A design is a more compact and less complex distributed system allowing to be installed into a smaller platform (less costly solution). The Type B design requires considerably more volume and is a more complex system due to the need for higher automation to successfully operate, requiring a larger platform (higher cost). Even so Type B is a more robust solution, and is a more modern and advance design, the experts consider that the Type A network has considerable benefits in respect to cost and operation.

Empirical observations

The empirical observations presented in this section were derived in a meeting lead by the researcher, with participants from industry and academia. A network science academic, two engineering design academics, one senior power and propulsion engineer, research and technology engineering manager deliberated the network representations and the robustness and modularity results of Type A and Type B technical systems. While observation 1, 2 and 3 are also found to be mentioned in the literature, observations 4 was established based on the discussion. Following are the four key observations derived:

1) The network representations of the technical systems A and B are cyclical. This observation is consistent with identified literature covering other domains such mechanical assemblies, electrical circuits and system of system architectures. This suggests that a component-based system architecture can be designed based on a cyclical main structure pattern.

2) Main hub components have source and sink components connected to them and are critical for the robustness of the system. For example, electrical supply from the generators feed HV switchboards representing a source style hub pattern, whereas EDCs demand from LV switchboards representing a sink style hub pattern. This verifies that a source – hub – sink pattern exists in component-based system architectures.

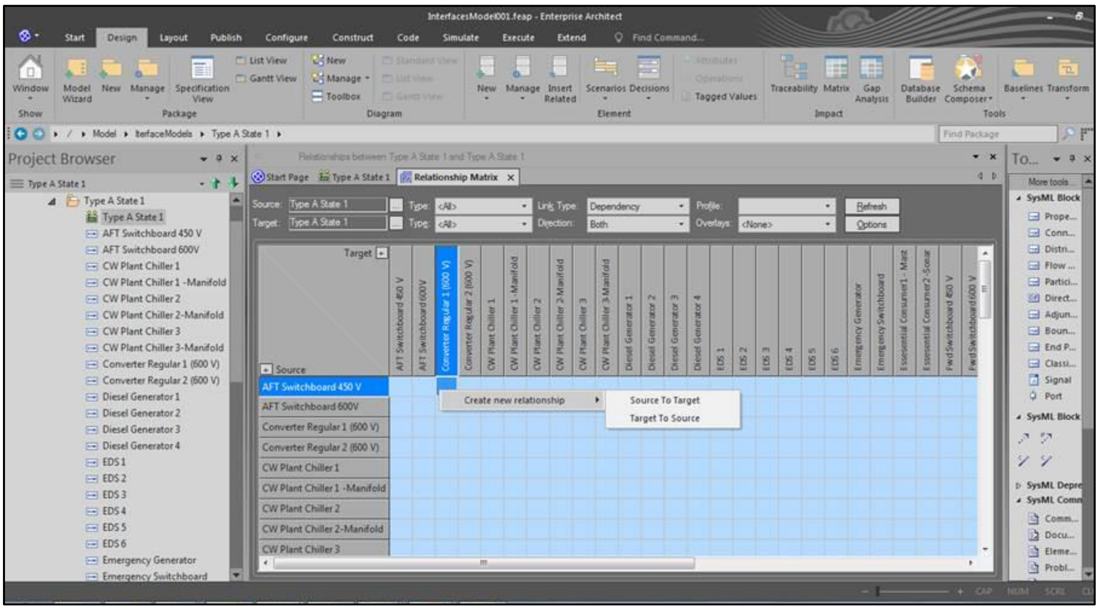
3) The level of connectivity between hubs is designed to maintain functionality after disruption. In Type A and Type B, the hub components have alternative path connectivity designed for redundancy. This implies that a hub connectivity pattern occurs in component-based system architecture.

4) At a lower level of decomposition, hubs display bus modular, star, or hierarchical style patterns. This suggests that structural patterns can be identified at a hub level.

The four empirical observations in combination with the literature findings on theoretical patterns and hubs, source and sink stimulated the development of the network tool (Chapter 7) and the experiments of Chapter 9.

Appendix IV: MBSE industrial design practise

This section entails investigating the potential of incorporating the proposed RoMoGA methodology in Sparx EA software environment that is currently used in the system engineering department of BAE Systems. The industrial design practice captures the current design approaches employed in the systems engineering department. EA is a modelling and design tool employed in the systems engineering department. EA is based on the SysML language which can visualise, analyse, model, test, and maintain systems, software, processes, and architectures. The researcher in cooperation with the senior systems engineer has worked on implementing the manual developed DSM's, presented in the Chapter 8 Case Studies, into the Sparx EA software. Together they worked through the software to identify whether it was capable of matrix representation like the design structure matrix representation used as the main input for the computational analysis section of the methodology. It was found that EA distinguishes between source and sink (termed: target components) which is like the concept used by the functional hierarchy diagrams that are required as inputs for the computational analysis.



Source: Type A State 1
Type: <All>
Link Type: Dependency
Profile:
Target: Type A State 1
Type: <All>
Direction: Both
Overlays: <None>

Figure 92: Sparx EA software relationship matrix

The above representation in Figure 92, which is a snapshot from the Sparx EA software, is identical to the Design Structure Matrix (DSM) which is the main means used to capture the input required for the proposed methodology. This means that the Sparx EA software is capable of generating the required input.

The Sparx EA relationship matrix, analogous to a DSM, allows the categorisation of the dependencies between elements as per the following (Sparks, 2014):

1. Categorise a component as source or target
2. Categorise type and direction of a relationship
3. Categorise the package where specific source and target components are saved

The direction represents the component that is the source and that is the sink. The observations from the industrial design practice indicate that EA facilitates the generation of data required as an input for the RoMoGA methodology, which reinforces its potential applicability to future projects. The relationship matrix capability of EA is not currently in used by the system engineering team. However, potentially a future work is the analysis part of RoMoGA methodology that is this study was performed in MATLAB environment, could be incorporated as code an extension of EA software, through future software coding.

It was observed that only by developing block definition diagrams in EA, automatically transforming them into a relationship matrix, and creating supplementary functional hierarchies (source and sink component per function), the proposed RoMoGA methodology could be applied in the system engineering design practice.

The establishment of inputs for applied the RoMoGA methodology can guide the architects to develop more elaborated block diagrams, by including information of the source and sink component. This will allow the development of a system data basis in the EA tool that could then inform analysis and assessment tools the analysis part of the RoMoGA or other analysis.