Automated Layout Generation and Design Rationale Capture to Support Early-Stage Complex Ship Design

le Poole, J.J.

DOI
10.4233/uuid:9ba03a3a-2bf1-44eb-a994-d497b30788f0

Publication date
2024

Document Version
Final published version

Citation (APA)
le Poole, J. J. (2024). *Automated Layout Generation and Design Rationale Capture to Support Early-Stage Complex Ship Design*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:9ba03a3a-2bf1-44eb-a994-d497b30788f0

Automated Layout Generation and Design Rationale Capture to Support Early-Stage Complex Ship Design

Automated Layout Generation and Design Rationale Capture to Support Early-Stage Complex Ship Design

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. T.H.J.J. van der Hagen,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 17 juni 2024 om 17:30 uur

door

Johannes Jacobus LE POOLE

scheepsbouwkundig ingenieur, Technische Universiteit Delft, Nederland,
geboren te Kesteren, Nederland.

Dit proefschrift is goedgekeurd door de

promotor: prof. ir. J.J. Hopman
copromotor: dr. A.A. Kana

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. ir. J.J. Hopman,	Technische Universiteit Delft, *promotor*
Dr. A.A. Kana,	Technische Universiteit Delft, *copromotor*
Onafhankelijke leden:	
Prof. dr. H.M. Gaspar,	Norwegian University of Science and Technology
Prof. dr. ir. F.E.H.M. Smulders,	Technische Universiteit Delft
Prof. dr. F.M. Brazier,	Technische Universiteit Delft
Prof. ir. C.H.C.F. Kaan,	Technische Universiteit Delft
Dr. ir. J.F.J. Pruyn,	Technische Universiteit Delft, reservelid
Overige leden:	
Dr. ir. E.A.E. Duchateau,	Commando Materieel en IT

This research was funded by the Netherlands Ministry of Defence. The views expressed in this dissertation are those of the author and do not reflect any official policy of the Netherlands Ministry of Defence, the Materiel and IT Command, or the Royal Netherlands Navy. Furthermore, the results presented here are for the sole purpose of illustration and do not have an actual relation with any past, current, or future warship procurement project at the Materiel and IT Command.

Keywords: complex ships; social-technical; early-stage design; feasibility; layouts; design rationale; rework

Printed by: ProefschriftMaken

ISBN 978-94-6384-575-5

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

They that go down to the sea in ships,
that do business in great waters;
These see the works of the LORD,
and his wonders in the deep.
For he commandeth, and raiseth the stormy wind,
which lifteth up the waves thereof.
They mount up to the heaven,
they go down again to the depths:
their soul is melted because of trouble.
They reel to and fro, and stagger like
a drunken man, and are at their wits' end.
Then they cry unto the LORD in their trouble,
and he bringeth them out of their distresses.
He maketh the storm a calm,
so that the waves thereof are still.
Then are they glad because they be quiet;
so he bringeth them unto their desired haven.
Oh that men would praise the LORD for his goodness,
and for his wonderful works to the children of men!
[...]
Whoso is wise, and will observe these things, even
they shall understand the lovingkindness of the LORD.

Psalm 107:23-43

CONTENTS

SUMMARY

During the early-stage design of complex vessels, such as naval vessels, a challenging problem needs to be solved. On the one hand, the design problem (i.e. requirements) needs to be formulated. On the other hand, via the stakeholder dialogue, this problem formulation is influenced by the generated solutions (e.g. concept designs). Such a problem can be described as a 'wicked' problem, which lacks a consensus on the problem and solution across stakeholders. To inform the stakeholder dialogue, designers need to gain insight into the technical feasibility, costs, and risks of these requirements and potential design solutions. These aspects need to be addressed early on because of the lock-in of the concept design by, mostly, early decisions. In this situation, rework is considered a challenge because of the high cost of late design changes and might be reduced by providing designers with more accurate information on technical feasibility and risk. Yet, wicked problems cannot be solved by technological solutions only, as there is an inter-related social aspect to be considered as well.

In this dissertation, layout design is selected as a prime example of an important aspect of ship design, for the following reasons. Firstly, the ship's layout represents the integration of all design aspects. Secondly, the layout is input to many design disciplines and is essential in the stakeholder dialogue. Typically, layouts are developed with increasing fidelity (i.e. level of detail) throughout early-stage design. In complex ship layout design, a principal challenge is the effort required to obtain insights into potential detailed sizing and integration issues and risks that might be encountered later in the design process. Underestimating such risks can cause costly and time-consuming rework. Current design methods lack the speed or detail to provide sufficient insight into these risks.

Besides the identification of physical integration issues, designers require design rationale (i.e. the justification of design decisions) to make informed design decisions (e.g. when rework is required). The challenge is that current manual design methods do not support the designer to capture and reuse design rationale in a meaningful way. In practice, design rationale may be documented (e.g. minutes, notes, reports). In addition, existing research shows that design rationale can be captured and reused by individual ship designers. However, no suitable design rationale method currently allows for integrated, in-situ documentation of design rationale during the complex ship layout design. However, this is essential to capture both the decision and its context, the concept design. Hence, it's currently unknown how the potentially intrusive activity of design rationale capture can be effectively integrated into the complex ship layout design process.

To fill the gaps identified above, this dissertation aims to fulfil the following research goal: *To reduce the effort required to identify and solve detailed layout integration issues during social-technical early-stage complex ship design via automated layout generation and design rationale capturing.*

The dissertation contributes to this goal in two ways. Firstly, a new layout generation method, called WARship GEneral ARrangement (WARGEAR) is proposed in the first part of this dissertation. WARGEAR allows designers to rapidly generate and evaluate detailed layout plans, based on a lower level of detail predefined functional arrangement comprising the ship's main building blocks. First, the designer provides the main input to WARGEAR. Subsequently, WARGEAR arranges passageways and staircases with a probabilistic placement algorithm. Then, it uses a network-based approach combined with probabilistic selection for the allocation of spaces to compartments. The allocated spaces are arranged using cross-correlation to enable a very fast arrangement of large layouts. Finally, a 'carving'-based approach is applied to ensure connectivity throughout the ship. The method is steered by a bi-level particle swarm optimisation code. The resulting detailed layouts and related performance data can then be further studied by the designer to obtain design insights.

WARGEAR is applied in four case studies. The first demonstrates how WARGEAR could be used to generate detailed layouts for a notional surface vessel. Results show that WARGEAR can be used to gain insight into sizing and integration issues in iterations of approximately 15 minutes. In the second case study, WARGEAR is applied to a realistic ship design problem. Its results are compared with earlier generated results by naval architects. The results indicate that WARGEAR could be used to provide insights into a wider range of variations (3 versus 8) in less time (2 weeks versus 2 days). The third case study extends WARGEAR's allocation algorithm to limit its dependence on a predefined functional arrangement. Also, data exploration is added and demonstrated on a small- and a large-scale (an Oceangoing Patrol Vessel, OPV) design problem. This way, insights into complex interrelationships between design parameters can be obtained. In the fourth case study, WARGEAR is combined with a queueing-based logistic performance assessment method to provide early insight into the logistic performance of a Landing Platform Dock design. The case studies prove WARGEAR can be used to get timely insight into detailed layout sizing and integration issues earlier in the design process.

Yet, designers can not capture design rationale in the context of the progressing concept design. Therefore, a new design rationale method (called the Ship Design Rationale Method or SDRM) is developed in the second part of this dissertation. The main aim is twofold: First, to develop a design rationale method to aid designers in the continuous capturing and reuse of design rationale during the collaborative concept design process and, second, to evaluate how the developed design rationale method benefits collaborative design decision-making such that it leads to better insight into design issues across design teams and better concept designs during a single design session and over time for realistic ship design problems.

To achieve this goal, firstly, a proof-of-concept is developed. The proof-of-concept allows designers to capture a limited scope of design rationale for simple layout problems (e.g. 2D layouts with 10 spaces). Subsequently, the proof-of-concept is evaluated in three-person teams comprising students and experts from industry. The results of the design experiment indicate that using a design rationale method while designing a layout can have both measurable and perceived benefits. An example of the former is that the design rationale method motivates teams to use 'network arrangement'. Such network arrangement of systems visually supports the team in sketching the initial arrangement of systems. Participants generally perceive the design rationale method to facilitate enrichment and negotiated knowledge, aspects aiding to provide a better understanding of the design problem within the entire design team. The results do not indicate that the design rationale method directly leads to qualitatively better concept designs compared to the baseline method. However, this could also be caused by the simple design problems used in this experiment.

Secondly, the scope of design rationale is extended. A small case study is conducted to demonstrate how the method could be used concurrently with designing an OPV in an existing 3D design tool (Surface Ship Design Tool, SSDT). The case study confirms that this would be possible, but that further integration is needed to reduce the designer's workload and to allow for better data management, for example.

Therefore, thirdly, the previous versions of the method are expanded and integrated into a GUI. Also, the SSDT is re-implemented and expanded to improve the integration with the SDRM and the flexibility of the concept design. Also, the SSDT allows designers to store the concept design at any point as a 'design instance', with an accompanying explanation. The SDRM enables low-intrusive capture of design rationale in the context of the concept design. Captured design rationales are linked to objects in the concept design. In combination with the design instances, the various states of the concept design can be reviewed at any point. Also, the SDRM supports designers in retrieving past design data via search functionalities and network representations of the captured design rationale. Furthermore, a three-stage process is developed to help designers build and maintain a knowledge base of the decisions taken during the development of the concept design. In a design session, designers using this process, first, look back on the current status of the concept design as well as the rationale behind this concept design. This aims to get designers up-to-date with the status of the concept design. In the main phase, the concept design is changed and design rationale is captured in response to emerging design issues, new insights, etc. In the final phase, designers look back on the work performed during the main phase to see if any important changes have not been documented yet, to ensure all important decisions and supporting design rationale are captured for future retrieval.

A case study is conducted to evaluate the long-term benefits of the SDRM. Two-person expert teams work on the design of a frigate, with a focus on topside design and machinery system arrangement. The experiment takes place over at least 3 weeks. Results indicate that the SDRM allows designers to refamiliarise with the concept design. Furthermore, the results indicate that the use of the SDRM stimulates designers to be more explicit in their design reasoning, which could improve overall communication and decision-making in the design team.

To summarise, the methods proposed in this dissertation allow designers, first, to identify potential sizing and integration issues in complex ship layouts, with less effort than required in current ship design practice. Second, the methods enable designers to capture their design reasoning in the context of the progressing concept design, which supports both current and future decision-making. However, due to the nature of early-stage complex ship design, the methods will not be *decisive* for decision-making but are nonetheless *supportive* to the human designer.

Samenvatting

Tijdens het vroegtijdig ontwerp van complexe schepen zoals marineschepen, moet een uitdagend probleem worden opgelost. Aan de ene kant moet het ontwerpprobleem (ofwel eisen) worden geformuleerd. Aan de andere kant wordt deze probleemformulatie, via het stakeholderdialoog, beïnvloed door de ontwikkelde oplossingen (de conceptontwerpen). Zo'n probleem kan worden omschreven als een 'wicked' probleem, waarin onder stakeholders een gebrek is aan consensus over zowel het probleem als de oplossing. Om het stakeholderdialoog te voeden moeten ontwerpers inzicht krijgen in de technische haalbaarheid, kosten en risico's van de gestelde eisen en mogelijke ontwerpoplossingen. Deze aspecten moeten vroegtijdig worden geadresseerd omdat het conceptontwerp door voornamelijk initiële besluiten wordt vastgelegd. In deze context wordt recursie gezien als een uitdaging vanwege de hoge kosten van latere ontwerpveranderingen. Recursie zou kunnen worden verminderd door ontwerpers van accuratere informatie over technische haalbaarheid en risico's te voorzien. Toch kunnen wicked problemen niet alleen door technische oplossingen worden opgelost, omdat gerelateerde sociale aspecten ook moeten worden meegenomen.

In deze dissertatie is het ontwerp van indelingen geselecteerd als een primair voorbeeld van een belangrijk scheepsontwerpaspect, vanwege de volgende redenen. Ten eerste representeert de indeling van het schip de integratie van alle ontwerpaspecten. Ten tweede is de indeling een input voor verschillende ontwerpdisciplines en is het essentieel in het stakeholderdialoog. Indelingen worden typisch ontwikkeld met toenemende precisie (d.w.z. detailniveau) tijdens het vroegtijdig design. Een primaire uitdaging voor het vroegtijdig ontwerp van indelingen van complexe schepen is de inspanning die nodig is om inzicht te krijgen in eventuele gedetailleerde schalings- en integratieproblemen en -risico's die later in het ontwerpproces zouden kunnen opspelen. Het onderschatten van zulke risico's kan leiden tot dure en tijdsintensieve recursie. Huidige ontwerpmethodes missen de benodigde snelheid of detailniveau om voldoende inzicht te geven in dergelijke risico's.

Naast het identificeren van fysieke integratieproblemen hebben ontwerpers ontwerprationale (d.w.z. de motivering van ontwerpbeslissingen) nodig om geïnformeerde ontwerpkeuzes te maken (bijvoorbeeld wanneer recursie nodig is). The uitdaging is dat huidige handmatige ontwerpmethoden de ontwerper niet ondersteunen om ontwerprationale in een betekenisvolle manier op te slaan en te hergebruiken. In de praktijk wordt ontwerprationale wel opgeslagen, bijvoorbeeld in notulen, aantekeningen en rapporten. Ook laat voorgaand onderzoek zien dat individuele scheepsontwerpers ontwerprationale kunnen opslaan en hergebruiken. Echter is er nog een geschikte methode om ontwerprationale in een geïntegreerde, in situ documentatie van ontwerprationale tijdens het ontwerp van indelingen voor complexe schepen. Dit is echter essentieel om zowel de beslissing als de bijbehorende context (het conceptontwerp) te vangen. Daarom is het momenteel onduidelijk hoe de, mogelijk intrusieve activiteit van ontwerprationale

opslaan op een effectieve manier kan worden geïntegreerd met het ontwerpproces voor complexe schepen.

Om de hierboven geïdentificeerde wetenschappelijke lacune te vullen, beoogt deze dissertatie het volgende onderzoeksdoel te vervullen: *Het verminderen van de benodigde inspanning voor het identificeren en oplossen van gedetailleerde indelingsintegratieproblemen tijdens sociaal-technisch vroegtijdig complex scheepsontwerp door geautomatiseerde indelingsgeneratie en het vangen van ontwerprationale.*

Deze dissertatie draagt op twee manieren bij aan dit doel. Ten eerste is er in het eerste deel van deze dissertatie een nieuwe methode voor het genereren van indelingen, genaamd WARship GEneral ARrangement (WARGEAR) voorgesteld. WARGEAR stelt ontwerpers in staat om snel gedetailleerde indelingstekeningen te genereren en te evalueren op basis van een minder gedetailleerde functionele indeling die de belangrijkste bouwblokken van het schip bevat. Ten eerste voorziet de ontwerper WARGEAR van de benodigde input. Vervolgens arrangeert WARGEAR gangen en trappen door middel van een probabilistische plaatsingsalgoritme. Vervolgens gebruikt het een combinatie van netwerken en probabilistische selectie om ruimtes naar compartimenten te alloceren. De gealloceerde ruimtes worden vervolgens ingedeeld met behulp van kruiscorrelatie. Dit stelt WARGEAR in staat om snel grote indelingen te genereren. Ten slotte wordt een 'kerftechniek' gebruikt om de benodigde connectiviteit in de indeling te waarborgen. Dit geheel wordt gestuurd door een bi-niveau particle swarm optimalisatiecode. De resulterende gedetailleerde indelingen en gerelateerd prestatiedata kan vervolgens verder worden bestudeerd door de ontwerper om ontwerpinzichten te verkrijgen.

WARGEAR is toegepast in vier casestudies. De eerste laat zien hoe WARGEAR kan worden gebruikt om gedetailleerde indelingen van een fictief schip te genereren. De resultaten laten zien dat WARGEAR kan worden gebruikt om inzicht te krijgen in schalings- en integratieproblemen in iteraties van ongeveer 15 minuten. In de tweede casestudie is WARGEAR toegepast op een realistisch ontwerpprobleem. Deze resultaten laten zien dat WARGEAR kan worden gebruikt om inzichten te verkrijgen in meer variaties (3 versus 8) in minder tijd (2 weken versus 2 dagen) in vergelijking met de eerdere resultaten van scheepsontwerpers. In de derde casestudie is het allocatie-algoritme van WARGEAR uitgebreid, om de afhankelijkheid van een vooraf gedefineerd functioneel ontwerp te verminderen. Tevens zijn er data-exploratietechnieken toegepast en gedemonstreerd op een klein- en grootschalig (een Oceangoing Patrol Vessel, OPV) ontwerpprobleem. Inzichten in complexe interacties tussen ontwerpparameters konden worden verkregen. In de vierde casestudie is WARGEAR gecombineerd met een queueing-gebaseerde logistieke prestatieanalysemethode om vroegtijdig inzicht te krijgen in de logistieke prestatie van een Landing Platform Dock ontwerp. Deze casestudies laten zien dat WARGEAR kan worden gebruikt om eerder in het ontwerpproces tijdig inzicht te krijgen in gedetailleerde schalings- en integratieproblemen.

Toch kunnen ontwerpers nog geen ontwerprationale vangen in de context van het ontwikkelende conceptontwerp. Daarom is er in het tweede deel van deze dissertatie een nieuwe ontwerprationalemethode ontwikkeld, de Ship Design Rationale Method (SDRM). Het hoofddoel is tweeledig. Ten eerste, om een ontwerprationalemethode te ontwikkelen die ontwerpers helpt om continue ontwerprationale te vangen en te hergebruiken tijdens het collaboratieve conceptontwerpprocess en, ten tweede, om te evalueren hoe de ontwikkelde methode zodanig bijdraagt aan dit ontwerpprocess dat het gebruik ervan leidt tot zowel beter inzicht in ontwerpuitdagingen in ontwerpteams als betere conceptontwerpen tijdens een ontwerpsessie en, voor realistische scheepsontwerpproblemen, over langere tijd.

Om dit doel te behalen is eerst een proof-of-concept ontwikkeld. Deze proof-of-concept stelt ontwerpers in staat om een beperkte scope aan ontwerprationale voor simpele indelingsproblemen (bijvoorbeeld 2D indelingen met 10 ruimtes) te vangen. Vervolgens is deze proof-of-concept getest in teams van drie personen (bestaande uit studenten en experts uit de industrie). De resultaten van dit ontwerpexperiment wijzen erop dat het gebruik van een dergelijke ontwerprationalemethode tijdens het ontwerp van scheepsindelingen zowel meetbare als bevonden voordelen oplevert. Een voorbeeld van een meetbaar voordeel is dat de methode teams stimuleert om 'netwerkindeling' te gebruiken. Hiermee worden teams visueel ondersteund in het schetsen van de initiële indeling van systemen. In het algemeen vonden deelnemers dat de ontwerprationalemethode bijdroeg aan verrijking van de dialoog en het opbouwen van overeengekomen kennis, aspecten die helpen een beter begrip te krijgen van het ontwerpprobleem in het ontwerpteam. De resultaten laten niet zien dat de methode direct leidt tot kwalitatief betere conceptontwerpen in vergelijking met een basismethode. Echter zou dit ook veroorzaakt kunnen zijn door de simpele ontwerpproblemen in het experiment.

Ten tweede is de scope van de ontwerprationale verbreed. Een kleine casestudie is uitgevoerd om te laten zien hoe de methode kan worden gebruikt terwijl er gelijktijdig een OPV ontworpen wordt in een bestaande 3D ontwerptool (Surface Ship Design Tool, SSDT). De casestudie bevestigt dat dit mogelijk is, maar dat verdere integratie nodig is om, bijvoorbeeld, de werklast van de ontwerper te reduceren en betere datamanagement mogelijk te maken.

Daarom zijn, ten derde, de eerdere versies van de methode uitgebreid en geïntegreerd in een GUI. Ook is de SSDT opnieuw geïmplementeerd en uitgebreid om de integratie met de SDRM te faciliteren én om de flexibiliteit van het conceptontwerp te vergroten. Verder maakt de SSDT het mogelijk dat ontwerpers het conceptontwerp op ieder ogenblik, met een bijbehorende verklaring, als een 'ontwerpexemplaar' kunnen opslaan. De SDRM maakt het mogelijk om op een laagintrusieve manier ontwerprationale in de context van het conceptontwerp te vangen. Opgeslagen ontwerprationales worden verbonden met objecten in het conceptontwerp. Hiermee, en in combinatie met de ontwerpexemplaren, kunnen ontwerpers de verschillende stadia van het conceptontwerp beoordelen. Verder ondersteunt de SDRM ontwerpers in het terughalen van eerdere ontwerpdata door middel van zoekfuncties en netwerkrepresentaties van de opgeslagen ontwerprationale. Daarnaast is een drie-staps proces ontwikkelt om ontwerpers te helpen met het opbouwen en onderhouden van een kennisbais van de gemaakte ontwerpbeslissingen. Tijdens een ontwerpsessie familiariseren ontwerpers in dit proces zich eerst met het huidige conceptontwerp en bijbehorende rationale. Daarna, in de hoofdfase, wordt het conceptontwerp aangepast en ontwerprationale gevangen naar aanleiding van verschijnende ontwerpuitdagingen, nieuwe inzichten, etc. In de laatste fase reflecteren ontwerpers op de laatste ontwerpwijzigingen om eventuele ongedocumenteerde ontwerpbeslissingen en bijbehorende rationales alsnog te documenteren voor toekomstig hergebruik.

Een casestudie is uitgevoerd om de langetermijnvoordelen van de SDRM te evalueren. Expertteams, bestaande uit twee personen, werken hierin aan het ontwerp van een fregat, met name aan het topzijde-ontwerp en de indeling van de voortstuwingsconfiguratie. Het experiment duurde minstens 3 weken. De resultaten laten zien dat de SDRM ontwerpers in staat stelt zich te herfamiliariseren met het conceptontwerp. Verder geven de resultaten de indicatie dat het gebruik van de SDRM ontwerpers stimuleert om explicieter te zijn in hun ontwerpredenering. Dit zou ten goede kunnen komen aan de communicatie en besluitvorming in het team.

Samenvattend, de methodes die in deze dissertatie ontwikkeld zijn stellen ontwerpers in staat om, ten eerste, potentiële schalings- en integratieproblemen in indelingen van complexe schepen met minder inspanning te identificeren dan in het huidige ontwerpproces. Ten tweede maken de methodes het mogelijk om ontwerpers hun ontwerpredenering in de context van het ontwikkelende conceptontwerp op te slaan, wat zowel de huidige als toekomstige besluitvorming ten goede komt. Echter, vanwege de aard van vroegtijdig complex scheepsontwerp, zullen de methodes niet *doorslaggevend* zijn aan besluitvorming, maar zijn desondanks *ondersteunend* aan de menselijke ontwerper.

1

INTRODUCTION

All correct reasoning is a grand system of tautologies, but only God can make direct use of that fact. The rest of us must painstakingly and fallibly tease out the consequences of our assumptions.

H.A. Simon (1969)

The design of large vessels has always been a collaborative undertaking. That is to say, historically the collaboration lay more in the field of ship construction, where an experienced shipbuilder oversaw the building process and gave guidance in how to build the ship. Limited theoretical knowledge of design issues required shipbuilders to rely on their own practical experience. In those early days, ships were designed while being constructed, "not on the basis of an engineer's calculations but through the master shipbuilder's active engagement in the building process on the yard" (Hoving & Wildeman, 2012).

Requirements outside the head shipbuilder's experience could lead to an expensive failed design. A classic example is the Swedish *Vasa* which capsized within the first nautical mile of her maiden voyage on 10 August 1628. This failure was caused by challenging and changing design requirements and limited theoretical knowledge of design issues among the shipbuilder and the Swedish king, who ordered the ship. Furthermore, hierarchical relations between the shipbuilder and the king complicated the decision-making process (Cederlund, 2006).

Although the complexity has largely increased since those early days, the design of today's ships (e.g. naval vessels or heavy lift vessels, such as shown in Figure 1.1) often involves challenging and changing requirements, limited upfront knowledge of the impact of innovative technologies on the overall design, and many stakeholders. On the one hand, the complexity of ships stems from, for instance, the many relationships between design parameters, resulting in a design space that can never be fully computed (Duchateau, 2016). To understand the relationship between design parameters and ship

(a) HNLMS *Zeven Provinciën* - an Air Defence and Command Frigate (LCF).

(b) Heerema's *Balder* - a Deepwater Construction Vessel.

Figure 1.1: Examples of complex ships - photos by author.

performance, *concept designs* are essential (Andrews, 2018b; van Oers, 2011b). On the other hand, complexity is also affected by human, social factors. For example, design decision-making is done by a wide range of interrelated stakeholders and design disciplines (Brown, 1986). Also, new design insights might lead to changing stakeholder preferences as well as changing trade-offs and justification of decisions (Duchateau, 2016). By capturing such *design rationale*, designers can more easily assess the potential impact of design changes (Bratthall et al., 2000; Burge & Brown, 2000; Poorkiany et al., 2016).

The remainder of this chapter will shed light on the nature of early-stage ship design, as well as the importance and challenges of the two interrelated research directions for this dissertation: concept design generation and design rationale capturing during early-stage ship design. Therefore, the next sections will explain the background, motivation, and objective of this dissertation. Regarding the research background, first, the nature of early-stage complex ship design is investigated. Second, complex ship layout design is investigated, as ship layout generation is both a challenging problem *and* essential in the design process, and a prime example of a complex problem during early-stage design. Third, the role of design rationale (i.e. capturing design reasoning) is investigated, as it explains and justifies design decisions but is challenging to capture in the context of the concept design.

1.1. RESEARCH BACKGROUND

1.1.1. EARLY-STAGE COMPLEX SHIP DESIGN

COMPLEXITY

Ships can be divided into classes. For instance, Van Oers (2011b) defines these as transport vessels (e.g. container ships and bulk carriers) and service vessels (e.g. heavy lift vessels, frigates, and cruise ships). Watson (1998, p.437) uses multiple ship types to cover transport vessels (e.g. container ships, bulk carriers, passenger ships). While transport vessels carry goods across the globe, service vessels perform missions at sea. Compared to transport vessels, service vessels are typically complex products (and hence called 'complex ships'), for three interrelated reasons:

1. **The formulation of the actual design problem**, i.e. determining and formulating the right set of (balanced) design requirements, is the main challenge during early-stage design. Therefore, early-stage complex ship design has been characterised as a 'wicked problem' (Andrews, 2018b) - a type of problem where there is no consensus on either the problem or solution (Roberts, 2000). As Rittel and Webber (1973) state,

   > "*setting up and constraining the solution space and constructing the measure of performance is the wicked part of the problem. Very likely it is more essential than the remaining steps of searching for a solution which is optimal relative to the measure of performance and the constraint system.*"

   Therefore, defining the engineering problem (i.e. setting the requirements) can be as, or even more, challenging than generating solutions (i.e. developing concept designs).

   Andrews (2012b) advocates a process of requirements elucidation to solve this challenge. Requirements elucidation involves a dialogue between all relevant stakeholders, supported by insights into technical and financial feasibility and risk (Andrews, 2012b; van Oers et al., 2018). Because the early-stage design problem can be very fluid, stakeholders need to settle on negotiated knowledge, i.e. an established negotiated basis of correctness of information to allow for interaction between stakeholders with different perspectives on that information (De Bruijn & ten Heuvelhof, 2008; le Poole et al., 2022a). This means that early-stage complex ship design is of a social-technical nature (Van Bruinessen, 2016). In other words, besides technical aspects, individual and collective human aspects play a role. As such, the ship design process is a very human process with false paths and recursive design, as well as factors inside (e.g. availability of information) and outside (e.g. legislation) that can disrupt the design process (Andrews, 1981; Wolff, 2000). As a consequence, early-stage design is highly iterative. As explained below, it's also the only phase that allows for such iterations to get the problem rightly understood and an appropriate solution defined.

2. **The complexity of the ship itself**, e.g. due to the large number of interrelated functions and supporting systems (Simon, 1996). These systems are required to enable the ship to perform missions. Since the relative importance of these functions is hard to express, the design process becomes a conversation between a wide range of stakeholders, ranging from end users, owners, constructors, naval architects, and other specialists (Brown, 1986). Based on the outcome of such dialogue, designers can prioritise certain negotiable requirements over others. Non-negotiable requirements, such as sufficient intact stability, should always be fulfilled (Van Oers, 2011b). Ship design is essential to provide insights for such dialogues (e.g. Andrews, 2018b; van Oers et al., 2018).

   Other levels of complexity can be identified as well, such as complexity related to the ship design process (e.g. perceptions of different designers), the shipyard (organisational structure), and the market (e.g. supply and demand dynamics)

(Ebrahimi et al., 2021; Ebrahimi et al., 2020; Shields & Singer, 2017). These other forms of complexity also apply to, for instance, transport vessels.

In this dissertation, the focus is on ship layouts. For complex ships, layouts are one of the main results of iterative early-stage design. Furthermore, layouts play an important role both in informing the stakeholder dialogue and in verifying the feasibility of the requirements and the design, as elaborated in Section 1.1.2.

3. **The evaluation of the performance of these vessels**, i.e. understanding and measuring the impact of design choices. This is especially challenging in the case of vessels designed for multiple functionalities (e.g. the Pioneering Spirit (Allseas, n.d.)). For instance, how to measure the effectiveness of a frigate operating in peace-time operations? The election of the appropriate performance metrics is one of the tasks for early-stage design (Andrews, 2018b).

   Also, uncertainty regarding the future operational context can play a role. For example, the sizing of heavy-lift vessels for the offshore wind market is a function of future, uncertain turbine sizes (Jiang, 2021; van Lynden et al., 2022; Zwaginga et al., 2021). Note that there are complex ship types for which determining the performance is less challenging. For example, the performance of a pipe-laying vessel can be expressed in produced pipe length per day, which is a function of the duration of the production cycle for a pipe segment joint and the efficiency of the production process (van Staalduinen, 2019). Yet, the actual weather is still part of the 'design equation' and is less predictable.

   Also, complex ships are often one-off designs or have very low production runs. This means that each produced vessel is, basically, a prototype. In the case of novel vessels, the accuracy of existing low-fidelity evaluation tools, typically used during early-stage design, might be insufficient (Charisi et al., 2022).

   To determine the performance, costs, and risks during the design phase, designers need to develop and analyse concept designs. Indeed, because the ship design problem is frequently technically complex, layouts need to be generated to get insight into the complexity and potential solutions. Hence, again, layout design is an important part of concept design. Furthermore, layout design cannot easily be captured in technical requirements, as will be shown in Section 1.1.2. However, there is a need to document design decisions (i.e. what was decided and why). Indeed, because of the iterative process, design decisions are frequently revisited. Currently, on-the-fly capturing of design decisions and underlying justification or rationale can be an intrusive activity, as elaborated in Section 1.1.3. This limits the evaluation of past concept designs.

The interrelations between these areas of complexity are visualised in Figure 1.2. This figure shows the systems engineering 'V' diagram, a tool used by designers for problem and solution decomposition (Duchateau, 2016). First, it describes how the problem description is incrementally detailed (left branch). Then, the corresponding design solution to this problem is developed (bottom). Finally, the performance and effectiveness of the developed solutions are evaluated (right branch). The figure illustrates how each

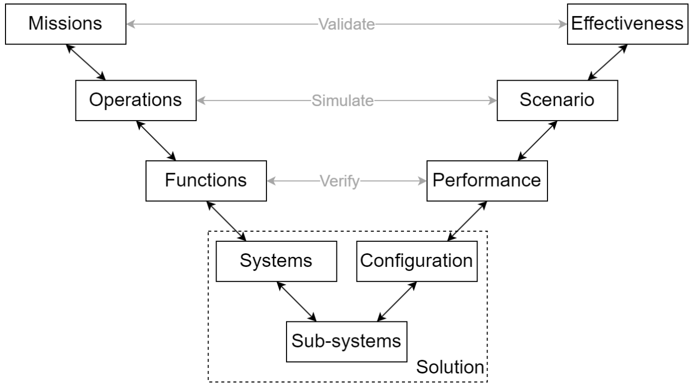


Figure 1.2: Systems engineering V diagram, adapted from (Duchateau, 2016).

step interacts with its predecessor and successor. Furthermore, high-level balancing activities, such as verification and validation need to be conducted. Note that this process is iterative, to ensure requirements, solutions, performance and budget remain in balance.

In summary, complex ship design is challenging due to the multi-aspect complexity described above. In this dissertation, the main focus will be on the design of complex vessels challenged by 'type 1' complexity, i.e. vessels for which the formulation of the design problem is the initial problem. What's currently lacking is, first, the ability to generate detailed concept designs in a limited time to support the stakeholder dialogue while reducing design risk. Second, there is a lack of design rationale capturing during the design activities. As a consequence, costly and time-consuming rework might be needed. This will be further investigated in the remainder of this chapter.

SHIP DESIGN PHASES

Above, the focus was on early-stage design. To place early-stage design into context, the various design stages are briefly discussed below. Although terminology and details might differ, the following phases can be identified (Andrews, 2018b; Duchateau, 2016; la Monaca et al., 2020; van Oers et al., 2018):

1. *Concept exploration phase.*
   The initial design efforts are often focused on understanding the design problem, identifying major design drivers (i.e. main design criteria with the highest size and cost impact), and identifying main solution options. Often the level of detail of concept designs is still limited to enable designers to explore a wide range of design options. Such a limited level of detail represents a higher level of uncertainty regarding the eventual feasibility of design options. Hence making assumptions to compensate for the lack of detail (or knowledge) is an important part of this phase. These assumptions are to be agreed upon between stakeholders. Once the relation between requirements and solutions is understood, the main requirements are finalised, and a set of promising concept designs might be selected.

2. *Concept definition phase.*
   During concept definition, the focus is on detailing one, or a limited number of, concept designs to ensure these are technically and financially feasible and to identify remaining risks in the concept design, as well as refining the assumptions made during concept exploration. In this phase, human control of the concept design is seen as favourable because 1) the increased level of detail requires decisions on details of the design, and 2) design changes are bespoke and implemented after elaborate (technical and non-technical) considerations among many stakeholders. As a consequence, different design tools might be chosen during concept definition compared to concept exploration. Design changes are typically a result of compromises between changing requirements, preferences, assumptions, etc., due to new insights gained through design work, parallel Research and Development (R&D), or constraints on the design process.

3. *Detailed design phase.*
   After signing a contract with a shipyard, the focus of the design efforts shifts to design for production, i.e. detailed engineering, as well as the acquisition of parts. This involves the generation of highly detailed production drawings and extensive documentation of design decisions. For shipyards, such design documentation, together with sea trials of the actual ship, are needed to demonstrate that the requirements from the contract are fulfilled.

4. *Construction phase.*
   Eventually, the ship is constructed. Frequently, the initial building steps are undertaken concurrently with the detailed design phase. Major design changes become very costly, but cannot always be avoided. A recent example is the 10m elongation of the four new Spanish S80 submarines to solve a weight unbalance due to an unnoticed calculation error. The required redesign and reconstruction resulted in a doubling of the program costs to 3.9B€ and a 10-year delay, as well as a dock enlargement costing 14M€ (BBC, 2018).

5. *Test and validation.*
   Once the ship is built, the shipyard needs to prove compliance with the requirements. This is done via system testing, sea trials, etc.

In this dissertation, the term early-stage design is used, which is assumed to comprise the first two design phases of the classification given above (i.e. concept exploration and concept definition), in line with Van Oers et al. (2018). The aim of early-stage design efforts is, firstly, to find a set of achievable requirements within an acceptable budget. Secondly, the aim is to find technically and financially feasible concept designs fulfilling these requirements (Van Oers, 2011b). Design work in later phases (i.e. post-contract) is (ideally) focused on engineering for production.

HUMAN ASPECTS

This section aims to give attention to the 'social' side of the design of complex ships. Although design tools are essential to allow the human designer to properly explore the technical and financial consequences of design decisions (Section 1.1.2), decision-making remains a human task. Indeed, the importance of the human designer in decision-making is underlined by the definition of *design*. According to Dorst (1997, p.35), design is "a thought process aimed at building a network of decisions that form the thought-construct 'product', which can be instantiated in the material world". Thus, decision-making is inherently part of design, and therefore the objective of the naval architect is to make the best decisions possible with the (frequently limited) information at hand (DeNucci, 2012). Thus, decisions are an integral part of design (Dorst, 1997) and hence designing is a human activity. Indeed,

> "*decisions help bridge the gap between an idea and reality. They serve as markers and units of communication to identify the progression of a design from initiation, through implementation to termination and they exhibit both domain-dependent and domain-independent features*" (Mistree et al., 1993).

Such decisions are usually made in a process involving multiple stakeholders (MacLean et al., 1991). Since complex ship design also involves a wide range of design disciplines (e.g. naval architects, structural engineers, marine engineering specialists) as well as customers (e.g. government, navy, and companies) or their representatives, it is worthwhile to investigate complex ship design decision-making from a multi-actor perspective.

Van Bruinessen (2016), besides focusing on the technical content of the interaction between actors working on individual system or ship design during co-evolving innovative ship design, also identified that the social dimension plays an important role in the ship design process. Social interactions can be identified on three levels (Drazin et al., 1999):

1. An intrasubjective (or individual) level.
2. An intersubjective level, between two or more individuals, representing shared frames of reference.
3. A collective level representing the unfolding of change across intersubjective levels.

Van Bruinessen (2016) used Smulders and Bakker (2012)'s model for intersubjective social interactions to show how consensus and shared understanding can be developed (Figure 1.3). For the sake of this dissertation, the three interaction-related aspects are briefly explained:

1. External cognitive objects (i.e. cognitive aspects) help align cognitive aspects between actors. A primary example in ship design is the layout drawing (see Section 1.1.2).
2. Structural aspects of the interaction (i.e. interaction aspects) relate to hierarchical structures between actors (e.g. project management and technical specialists)
3. Synchronising and testimonial activities. These comprise the activities that link the external cognitive objects and the structural aspects of the interaction. These

could include storytelling and dialogue mapping (e.g. Conklin, 2006) and aim to synchronise understanding across actors.

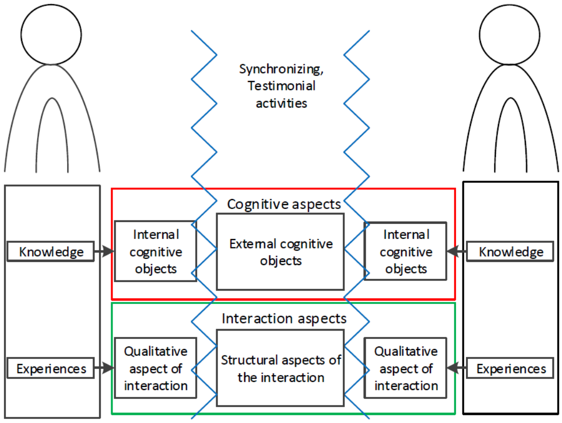


Figure 1.3: Disjuncture in the social dimension of interaction (Van Bruinessen (2016) based on Smulders and Bakker (2012))

To account for all relevant and different perceptions of stakeholders in wicked problems, the problem definition is often not fixed but only defined in broad terms. Such problem definition mainly considers the links between the views of stakeholders, i.e. it is used to describe the commonalities between the perceptions of stakeholders where the potential ground for consensus can be found. This creates room for win-win solutions. The fluidity of the problem definition can challenge the applicability of linear problem-solving approaches, e.g. problem specification, data gathering and analysis, solution formulation and implementation (De Bruijn & ten Heuvelhof, 2008; Roberts, 2000).

Therefore, by definition, a wicked problem implies decision-making involving multiple actors. These actors might have conflicting perceptions of the problem and thus might prefer different solutions. Proposed solutions might compete with other actors' preferred solutions (Head & Alford, 2015). Conklin (2006) explains that fragmentation (i.e. stakeholders see themselves more separated than united) occurs when wicked problems and social complexity (i.e. stakeholders have different, yet strong beliefs about what the problem is) are combined. To address this fragmentation of perspectives, understanding, and intentions, there is a need to build *shared* understanding and commitment. Shared understanding means that "the stakeholders understand each other's positions well enough to have intelligent dialogue about their different interpretations of the problem, and to exercise collective intelligence about how to solve it." Conklin (2006) proposes shared display as a solution. In shared display, a shared representation of the dialogue between stakeholders is created on a 'dialogue map', enabling stakeholders to focus and understand what they are doing. Shared commitment can be created when shared understanding is established.

Decisions in multi-actor decision-making are made based on negotiated knowledge, that is, common knowledge is created through interaction and discussion among stakeholders (De Bruijn & ten Heuvelhof, 2008). Creating negotiated knowledge can therefore be seen as a learning process of individual stakeholders (Partidario & Sheate, 2013). According to Partidario and Sheate (2013), knowledge is constructed by learning through an active, mental process of development. Knowledge is formed when information is interpreted by individual humans and is related to a context and anchored in the beliefs and commitments of those individuals. As a consequence, decision-makers need to learn what the impact of their requirements and decisions is, and cannot simply be transferred from an expert to stakeholders (e.g. via informing or consulting strategies (Michaels, 2009)). Learning is a fundamental human process that helps update and evolve experimental knowledge of previous designs, processes, events, actors, and environmental concerns (Duffy, 1997). Duffy continues "learning alters a human's state of knowledge and hence directly influences the human ability to solve problems". Enrichment is a form of learning that can occur in multi-actor decision-making (De Bruijn & ten Heuvelhof, 2008). Therefore, approaching design from a multi-actor decision-making perspective is essential to tailor design support methods to this learning process that underlies the creation of mutual consensus.

However, there is no single solution to address wicked problems (Conklin, 2006). This is because wicked problems have no 'root cause' that can be dealt with to remove the wickedness (Head & Alford, 2015). As a consequence, different perspectives on the main problem in wicked problems lead to different 'solutions'. For example, if stakeholder disagreement is seen as a main cause, this implies a preferred solution of dialogue to reduce conflicts. If insufficient knowledge is seen as the main cause, the preferred solution is further research to fill knowledge gaps and improve the information base for decision-making (Head & Alford, 2015).

A practical example of the social nature of complex ship design is the discussion regarding the position of an ammunition store on a naval vessel. A vulnerability specialist might want this store below the waterline, a safety specialist far from the accommodation, and a user close to the weapon. Also, a logistics specialist prefers easy access and the naval architect considers space and weight. Although this example could be classified as a complex problem (since it is rather well-defined) (Roberts, 2000), it might also be an example of a wicked problem if multiple actors are involved and if these actors are not yet in line (e.g. because they do not understand the implications of design decisions on other design disciplines and aspects yet). Indeed, complex ship design involves many design disciplines and specialists working on potentially conflicting or competing design aspects, and thus actors need to align and reach a consensus (Van Oers et al., 2018). The development and proposal of solutions might change perceptions of the problem and can lead to the identification of new interdependencies between solutions. These interdependencies might be hard to identify beforehand, for example, because design decisions for one class of ships can impact another class that is concurrently being designed. From a multi-actor decision-making perspective, it's therefore preferred to consider the 'why' behind stakeholders' preferred solutions. Although solutions might conflict, there might be commonalities in the underlying preferences and considerations. Therefore, Andrews (2018b) states that proper early-stage ship concept design should reveal, at least the major, hidden implications.

1

Besides describing the nature of early-stage complex ship design as a wicked problem, one studying this field should consider the relation between science and wicked problems. Indeed, Iijima (2022) discusses how the nature of wicked problems conflicts with the key scientific prepositions of reductionism, repeatability, and refutation:

1. Reductionism assumes an experiment to be conducted in a perfectly controlled environment and the subject can be understood by its elements only. However, in wicked problems, the whole cannot be understood by the properties of its constituent elements, but also needs the consideration of the interactions between elements. Hence, the small-scale case studies in this dissertation by definition fail to be representative of real-life wicked problems in ship design.

2. Repeatability is the ability of a method to produce similar results for repeated use of the method. Iijima (2022) points out that "in the case of a system including human activities, the presence of an observer influences the behaviour of the observed object, which in turn changes the results of the observation". Hence, complete repeatability cannot be reasonably assumed for wicked problems.

3. Refutation entails that only refutable statements are truly scientific statements. Because wicked problems involve negotiated knowledge, interpretation of the problem itself plays a large role. Thus, the problem description is not refutable.

Thus, wicked problems, such as concept design during early-stage complex ship design, can be studied. However, it might be challenging to draw decisive conclusions from experimental studies. This observation needs to be considered in the remainder of this dissertation.

In conclusion, complex ship design is of an intertwined technical and social nature, especially during early-stage design. Therefore, any method development to support early-stage design efforts needs to align with the technical and social aspects of complex ship design.

CHALLENGE OF RECURSION IN EARLY-STAGE DESIGN

Mavris and DeLaurentis (2000) show that the majority of the costs of the design are locked in during the early design stages (see Figure 1.4). This figure shows that these costs are committed due to decisions made, which subsequently result in lower design freedom. At the same time, these decisions are made when there is limited knowledge of the design. However, this figure does not show the iterative nature of ship design (Duchateau, 2016). Because ships cannot be simply represented by a set of equations, a direct solution cannot be derived. Instead, designers need to make educated guesses, work out the consequences of these guesses by calculations and drawings, update assumptions when new information becomes available, and iterate (Lamb, 2003).

Although all design decisions can be changed at any time in theory, there are multiple reasons why this is not feasible in practice. Changing the design becomes increasingly expensive throughout the design stages (Figure 1.5). This is mainly because the level of detail of the design increases over time, and thus, a major change at a later stage requires much more additional design changes, increasing the time and effort required to make

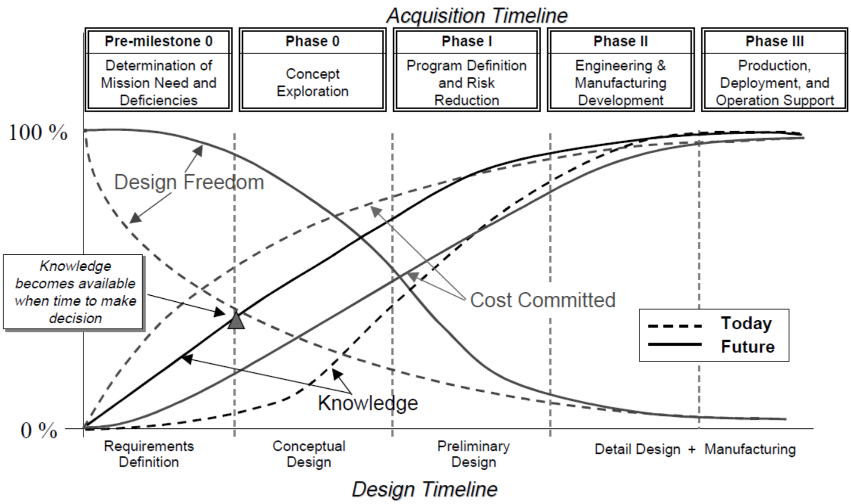


Figure 1.4: The relationship of design freedom, knowledge, and cost committed (Mavris & DeLaurentis, 2000).

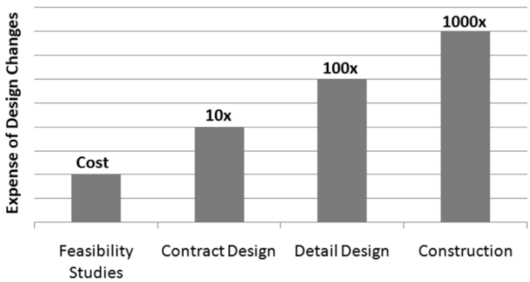


Figure 1.5: Estimates of change costs during different stages of design for naval vessels (Kana et al., 2016; Keane & Tibbitts, 1996)

these changes. Since ship design projects, as most projects, are often subject to time and budget constraints, there is a limit to the number of design changes possible.

Figure 1.6 shows a generic design timeline, and how recursive (i.e. iterative) design work could take place. Often, designers cannot evaluate all design options available at a given moment. The choice for a specific design path might be made implicitly or explicitly, and designers might not be aware of some other feasible design directions. Early-stage design decision-making is challenged by a high degree of uncertainty, which is both a cause for design iterations (because of incorrect assumptions or not recognised relationships between pieces of information (Wolff, 2000), or to balance the many aspects of ship design from an initial assumption (e.g. Evans, 1959)) and a reason to perform design iterations (to understand as much of the design problem as possible).

Ships are highly complex objects, in which many aspects need to be in balance. This was exemplified by the quote at the start of this chapter. To give an appreciation of this complexity, consider the following example. Suppose a ship needs to transport a pay-

1

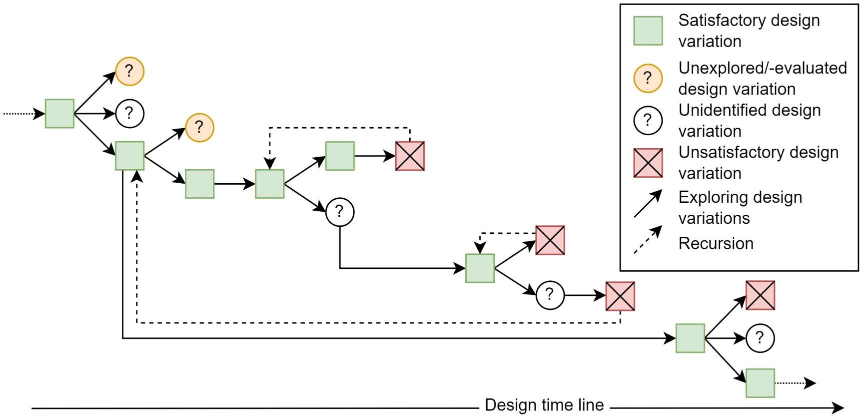


Figure 1.6: Recursion in design.

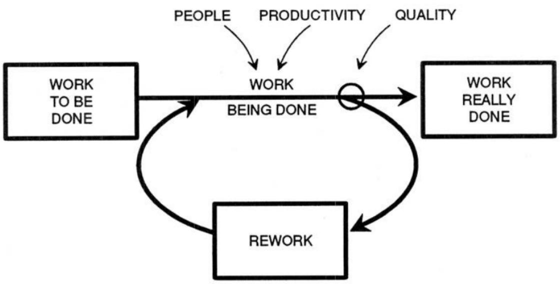
load at a certain speed and range. Then, the payload (partially) determines the hull size and displacement. Combined with the required speed, the displacement determines the ship's resistance, and hence which propulsion system is needed to satisfy the speed requirement. In turn, the selected propulsion system determines the fuel consumption and, with the required range, the amount of fuel can be determined. Both the propulsion system and fuel add to the ship's weight, which requires an update of the ship's displacement and subsequent assumptions and choices, etc. Besides this balance, naval architects need to consider the ship's stability and structural subdivision, to name but a few of the many design aspects.

Cooper (1993) sees inadequate accounting for iterative, recursive design (in his terms, rework) as a major driver for delays in complex development processes. Such delays could lead to a violation of time constraints for a project. Indeed, he states that "rework can account for the *majority* of work content (and cost) on complex development projects." In contrast to more linear project management methods, he proposes the Rework Cycle model (Figure 1.7) that explicitly includes rework. Rework comprises both known rework and undiscovered rework. The model accounts for the *quality* of work performed. Since rework can be a significant issue, it needs to be reduced. Based on the Rework Cycle model, this can be achieved by:

1. Improving designers themselves via, for instance, education and training. This is considered a role for educational institutes or companies.

2. Addressing the productivity of the designers by, for example, enhancing design tools or the working environment. However, this could lead to a higher frequency of design errors, without reducing the errors themselves.

3. Enhancing the quality of design decisions by, for instance, providing more insight into the consequences of decisions (Wolff, 2000). This option is considered to have the most potential. Indeed, besides improving the quality of decisions, it does not necessarily lead to more work for designers, as with the production improvement.



(a) Rework Cycle, complete with Undiscovered Rework (UR).

(b) Simulated project performance, showing Rework (RWK) versus UR.

Figure 1.7: The Rework Cycle (Cooper, 1993)

Therefore, based on the Rework Cycle model, this dissertation focuses on addressing the quality of design decisions, reducing the need to reconsider 'wrong' decisions (i.e. reduce costly and time-consuming rework). Specifically, the focus is on enabling designers to make more informed design decisions, via providing insight both into the consequences of design decisions and into the justification of past decisions.

Indeed, the iterative nature of design decision-making can also be observed in ship layout design, which is one of the important aspects that need to be addressed in the ship design process. One issue of particular interest is the design challenges posed by increasing the level of detail, especially the identification of sizing and integration issues downstream of the design process. This will be elaborated in Section 1.1.2. Because of the iterations in complex ship design, past design decisions often need to be reconsidered. This requires designers to know both the context and justification of these past design decisions, i.e. the design rationale. However, traceability of design decisions is only partially enabled in early-stage ship layout design, as will be shown in Section 1.1.3.

1.1.2. COMPLEX SHIP LAYOUT DESIGN

NATURE OF COMPLEX SHIP LAYOUT DESIGN

One of the key tasks for designers of complex ships is the generation of ship layout drawings. This activity is part of the concept design process and is needed to understand the relation between the design and performance spaces (Duchateau, 2016; Lamb, 2003). Some use the term 'synthesising' for 'designing' (e.g. Andrews, 2018b; Duchateau, 2016). Design insights and importance of specific decisions can typically only be determined post-priory or interactively due to the complex relationship between the design and performance space by designing (Duchateau, 2016; Pawling, 2007), as depicted in Figure 1.8. Because of this relationship, complex ship design is an example of 'satisficing' (Simon, 1996), where the objective is to find *good* concept designs, in contrast to 'optimising', where the objective is to find the *best* concept design in relation to one or more metrics (Pawling, 2007). The reason is that not all design performances might be expressed in a measurable utility function (Simon, 1996). Still, if individual design efforts are not constrained by overall design solution implications (e.g. costs, or conflicts with other design aspects), the congregated set of design solutions can be over-budget and over-ambitious. Exploring the problem and solution spaces, with respect to need and affordability, is the typical goal for early-stage design efforts. To develop a mutually satisfying concept design, conflicting and competing design aspects need to be resolved (Habben Jansen, 2020; Roberts, 2000; Wolff, 2000), for instance via the stakeholder dialogue (Van Oers et al., 2018).

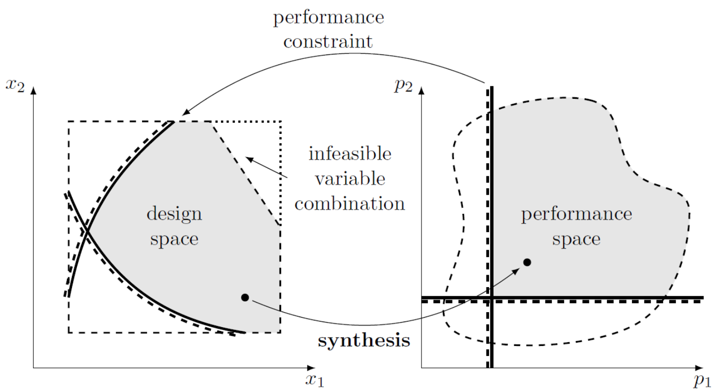


Figure 1.8: Complex interactions between design and performance space (Duchateau, 2016).

In line with the design phases discussed in Section 1.1.1, ship layouts are also developed in phases. The following two aspects describing the layout design process can be identified:

1. Designers use two-step design activities in response to (potentially changing) design problems: a) *ideation* and generation of design solutions and b) *evaluation* of these solutions and decision-making to select the preferred solution (McCall, 2010; la Monaca et al., 2020). These activities might be explicit or implicit. Designers can use rough sketches (e.g. Figure 1.9a) during these activities, for example, to communicate ideas (van der Lugt, 2005; Pawling & Andrews, 2011).

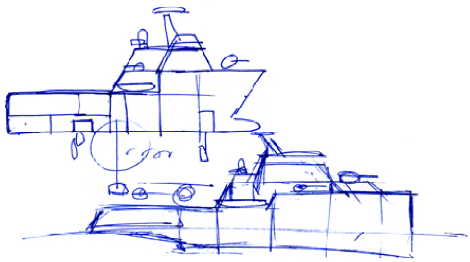
2. Layouts are developed with *incrementally increasing fidelity* - potentially with different tools over time (e.g. Andrews, 2018b; van Oers et al., 2018):

    (a) During the concept exploration phase, low-level-of-detail concept designs are generated (e.g. Figure 1.9b). Such designs typically comprise only major building blocks but already comprise decisions, for instance on 'design style' (Andrews, 2018a).

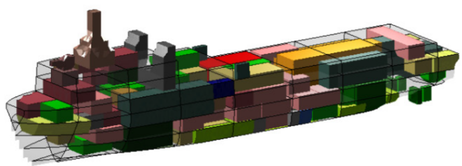
(b) During the concept definition phase, the fidelity is increased to medium-level-of-detail (e.g. Figure 1.9c). At the end of this phase, a high-level-of-detail General Arrangement Plan (GAP) is generated (e.g. Figure 1.9d) and can be very time-consuming to generate (see also Section 2.1). This GAP is, among others, a key document during contract negotiations.

(c) During the detailed design phase, the high-level-of-detail GAP is further matured and kept consistent with other detailed engineering deliverables.

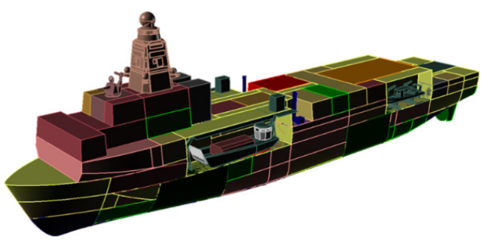
What level of fidelity is needed depends on the current design questions and issues. With increasing fidelity, the effort and time required to process design changes increase as well (as discussed in Section 1.1.1). This indicates that early-stage design efforts have the most significant impact on the final layout.



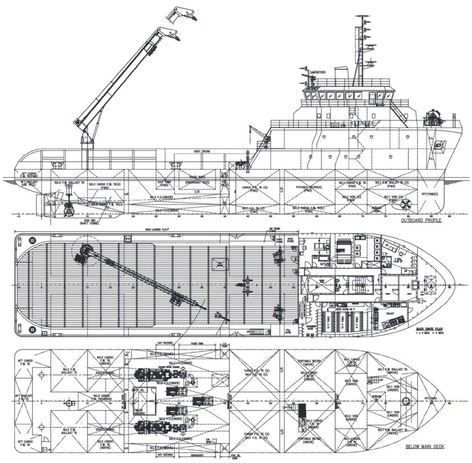
(a) Design sketch (Pawling & Andrews, 2011)



(b) Low level of detail 3D arrangement (Van Oers, 2011a)



(c) Medium level of detail 3D arrangement (Van Oers, 2011a)



(d) High level of detail 2D GAP (Seaboats.net, n.d.)

Figure 1.9: Examples of arrangement drawings of ships.

To summarise, ship layouts are important in the ship design process for, amongst others, the following four reasons (Andrews, 2012b; Carlson & Fireman, 1987; DeNucci, 2012; Watson, 1998):

1. A ship is a complex integrated system of individual functions. Incorrect configurations of systems can lead to unsatisfactory system performance. Yet, trade-offs often need to be made to achieve satisfactory performance.

2. Many ships, including naval ships, are space-critical, and thus their size and costs are governed by spatial requirements. These spatial requirements are difficult or even impossible to put into pure textual requirements due to the many trade-offs. Indeed, "drawings and written specifications are both integral and intertwined parts of the design. They are meant to explain each other" (Lamb, 2003). As such, a layout or GAP represents these spatial requirements and underlying, decided trade-offs.

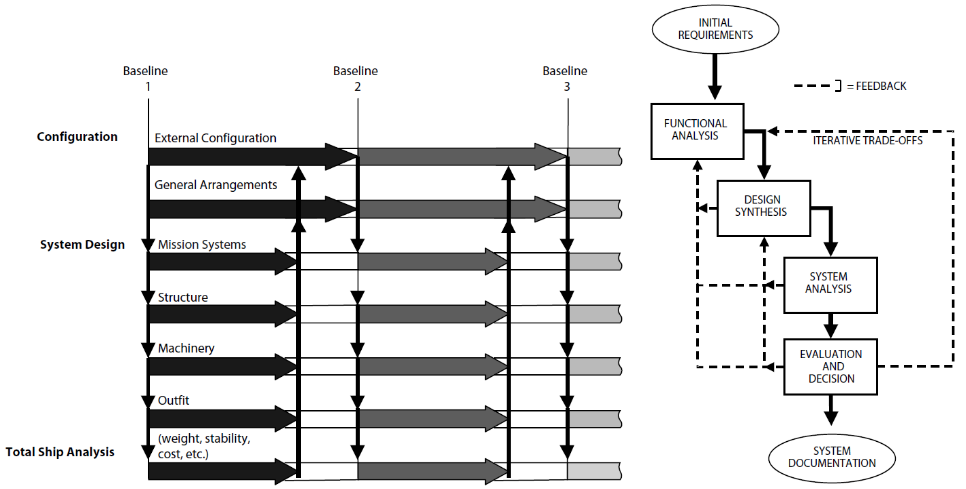
3. The ship arrangement is input for many other ship design tasks, such as weight calculations, structural calculations, etc. This is also visualised in Figure 1.10a, which shows that a ship's arrangement drawings form a baseline design for other design disciplines.

4. Layouts are necessary to help elucidate stakeholder preferences and requirements, as well as to provide stakeholders insight into the impact of their preferences on other design disciplines and the necessary trade-offs.

In conclusion, the concept design of ships (including layout design) is key to determining functional requirements and system performances, as depicted in Figure 1.10b. See Andrews (2018b) for an example of an elaborate ship synthesis process description. The development of concept designs is essential to inform the stakeholder dialogue on technical feasibility, affordability, and risks of the requirements and design solution (Van Oers et al., 2018). Hence, concept designs, and thus layouts, need to be defined at a sufficient fidelity. Figure 1.10b shows that the concept design needs to be updated whenever information from higher-level functional design, or lower-level system design and evaluation is updated.

CHALLENGES OF COMPLEX SHIP LAYOUT DESIGN

While being highly relevant in the ship design process, ship layout generation is a complex task. It requires the naval architect to identify, assign, evaluate, integrate, adjudicate, and control the space for each shipboard function and the physical relationships among these functions, as well as to position the equipment of a subsystem concerning their physical and functional relationships (Carlson & Fireman, 1987). This requires an architectural approach that considers the physical, logical, and operational organisation and integration of system components (Andrews, 2012a; Brefort et al., 2018; Duchateau, 2016; van Oers, 2011b). Indeed, the fine balance between, on the one hand, the required area for and positioning of systems and, on the other hand, the available area in a hull and superstructure cannot be found without actually generating arrangement plans. For instance, Gillespie (2012) focused on a network representation of the ship, which could be used to identify design drivers and constrained spaces. However, the allocation of



(a) Multi-disciplinary design development and position of configuration, i.e. layout, design.

(b) The position of design synthesis within the design process.

Figure 1.10: Ship design process and role of layout generation (Lamb, 2003).

systems to compartments (i.e. in rough arrangements) was necessary to determine the required ship size. The technical feasibility of these arrangements was not considered enough though, as the available and required area of systems was not taken into account. Further, his approach relied on a predefined set of system relations, which might not be readily available for different classes of ships, or require different weighing for different classes or designs. This then would require naval architects to spend much time defining such input, rather than altering relations while designing and analysing solutions to design problems. This dissertation argues that the latter scenario is more promising from a design knowledge perspective. Indeed, this is where the core of designer learning (i.e. understanding the relation between requirements and design performance) occurs, see also Chapter 2.

To help naval architects in generating layout designs various design tools and methods have been developed. Generally, one tool does not solve the whole problem. Hence, tools are developed for specific goals. For example, Packing automatically generates thousands of low-level-of-detail designs within hours to get insight into the relationship between design and performance spaces early in the design process (Van Oers, 2011b). Another example is the Design Building Block approach (DBB), which allows designers to manually generate concept designs with various levels of detail within days (Andrews & Dicks, 1997). Also Functional Integrated Design Exploration of Ships (FIDES) (Takken, 2009) is an example of a manual ship synthesis design tool. Such manual tools allow for better human control of the design progress and conscious consideration of design changes (Dicks, 2000; van Oers et al., 2018). This becomes especially important when the design matures, the level of detail increases, and the concept design is prepared for detailed and construction design.

However, manual tools are often time-intensive when designing at a higher level of detail, causing the designer to only evaluate a few design options (Le Poole et al., 2019). At the same time, a "delicate balance needs to be struck" between the fidelity (i.e. accuracy) of design models and the effort to build and use these models (Keane & Tibbits, 2018; Reinertsen, 1997). Especially during early-stage design, design decisions involve uncertainty about the feasibility of design changes. In other words, a design change might lead to a dead end, in which new design problems arise but cannot be solved without changing the overall design to a large extent. In such situations, one has to (partially) start over again, as illustrated by Figure 1.6. Thus, relatively small changes to the concept design might eventually have larger cascading effects.

To make such risk more acceptable, Reinertsen (1997) proposes the following three options: "First, we can decrease the magnitude of the downside. Second, we can reduce the probability of the downside. Third, we can increase the magnitude of the upside." Keane and Tibbits (2018) add: "The biggest opportunity to control risk in ship design usually lies in decreasing the magnitude of the downside, rather than reducing the probability of failure. However, this requires identifying the risk early enough in the design process when the downside can be readily addressed."

Thus, to help avoid design recursion, designers need to have insight into possible sizing and integration issues later in the design process. Initial layouts require "generous"[1] space margin or (empty) space to solve detailing problems without becoming infeasible later in the design, while too much margin can cause a design to become unbalanced or noncompetitive (Andrews, 2022a). While a limited level of detail in concept designs is desired during early-stage design to enable flexible changes in the overall design, a higher level of detail is needed to identify these issues. This higher fidelity could be provided by a GAP. However, generating a GAP can require up to 150 working hours (Le Poole et al., 2019) - which is problematic for early-stage design. This challenge might be overcome by supporting design tools. Such tools need to be fast and provide the required insights without extensive human evaluation to ensure timely insight into potential design issues. As will be further demonstrated in Part I, currently there is a gap in design tools to provide naval architects using manual tools the capability to quickly generate (i.e. in the order of minutes) more detailed layouts to evaluate the technical feasibility, risks, and integration issues of potential design paths, before committing to such path. This layout design problem was identified at the Netherlands Defense Materiel Organisation (DMO)[2] and initiated the research leading into this dissertation.

Developing such a capability has the following benefits, aligning with the nature of early-stage ship design:

1. A reduction in effort required for the creation of detailed arrangement plans enables the generation and analysis of a larger number of layout variations. This increases the amount of design insight to support the stakeholder dialogue.

2. Earlier insight into sizing and integration problems enables designers to make necessary changes to the design earlier in the process, reducing the cost of design rework.

[1] But how much?
[2] Currently renamed to Materiel and IT Command (COMMIT).

3. More design variations can be generated and analysed. This in turn enables a more thorough investigation of possible trade-offs.

Thus, such capability could help to identify risks earlier during complex ship layout design (Keane & Tibbits, 2018) and thus improve the quality of design decisions, reducing the need to reconsider 'wrong' decisions (i.e. reduce costly and time-consuming rework), as discussed in Section 1.1.1. However, although the development of a fast layout design method is a promising way to support designers in reducing recursive design work, it does not provide the designer with the essential rationales for past design decisions.

1.1.3. CAPTURING DESIGN RATIONALE DURING EARLY-STAGE COMPLEX SHIP LAYOUT DESIGN

CONTEXT OF DESIGN RATIONALE IN SHIP DESIGN

The design rationale behind a concept design is important because it explains and justifies the decisions leading to that design. One of the issues with generating layouts is that naval architects might find themselves in circular reasoning when trying to solve the many trade-offs in a layout problem (Duchateau, pers. comm., October 14, 2020). If an issue in the layout (e.g. a sub-optimal placed space) is solved, it might introduce multiple new issues, possibly for other design disciplines. If the consequences or impact of the latter are higher than those of the initial issue, the changes need to be reversed. In order to explain why the design is as it is, it is essential to capture those 'secondary' issues, corresponding trade-offs, the rationale behind the solutions, and the temporal order of the decisions (McCall, 2010). Indeed, interdependence between decisions can be temporal (Kana et al., 2016; Shields et al., 2016) and path-dependent, i.e. meaning the sequencing of decisions can change the final outcomes (Kana et al., 2016; Page, 2006). This is the case when these sequential decisions lead to incrementally more information for later decisions. The consequences of decisions are typically assigned to the initial choice (e.g Zandstra et al., 2016) - as illustrated in the example below.

> **Example of assigning consequences to decisions**
> A designer works on the design of a frigate. First, a helicopter hangar is arranged, which causes the ship to be lengthened. The associated increase in cost is assigned to the hangar. However, this lengthening has created empty space below the hangar, which can be used for some 'free' additional cabins. However, if the designer had focused on arranging these cabins first, the hangar might have been 'free' instead. Or should the cost increase be assigned to both the hangar and cabins?

However, design teams tend to be less experienced than required to effectively express rationales on designs during design reviews (DeNucci, 2012). For the same reason it can be challenging for relatively inexperienced naval architects to express why the design they came up with is a good, or even the best, compromise between conflicting or competing requirements.

As said above, drawings and written specifications are meant to explain each other (Lamb, 2003). Throughout the various design phases, many drawings, calculations, and plans need to be delivered. Although dozens of items need to be delivered, design rationale is not explicitly listed (DeNucci, 2012; Lamb, 2003). However, according to Lamb (2003):

> "*the design effort may also include the preparation of written materials, which aid in conveying the ideas of the designer and in explaining the working of the device or system. These written explanations may take the form of simple or extensive notes on the drawing or written specifications in booklet or book form.*"

For example, a requirement might demand the ship to operate up to an outside temperature of 45° Celsius. The supporting contextual explanation could be: 'The ship is to operate in warm regions. Furthermore, aspects such as "global warming" need to be considered.'

However, design rationale is not an explicit deliverable and the available time to complete the required work is frequently limited during early-stage design. Therefore, the level of design explanation documented during early-stage design can be limited in practice. This is a problem.

Indeed, when design rationale is not expressed or captured, it is easily lost over time (in part because the design time can be long, requirements might be easily changed during early-stage design, and design teams can change) which could lead to repeated discussions on the same design issues. This is especially the case in navies as naval officers have job placements which typically are shorter than the design phases. Hence a handover of knowledge is key. Andrews and Pawling (2007) show how the design of a Joint Support Ship evolved over several design iterations, see Figure 1.11. For instance, the increased density of the layouts shows that the design progressed over time. However, the rationale that went into these concepts or in the down-selection is unclear. Also, infeasible design solutions (e.g. concepts that do not meet requirements or cannot be technically integrated) are not shown. However, "recording a designer's line of reasoning makes it possible to revisit it later in order to assess it, to approve it, or more simply to learn from it, regarding either the system being designed or the decision process itself" (Falessi et al., 2013). Hence, such information is essential to avoid already encountered pitfalls or design challenges later in the design process.

BENEFITS AND CHALLENGES OF DESIGN RATIONALE IN GENERAL

The availability of the reasoning behind design considerations and decisions (i.e. design rationale) has various benefits, (e.g. DeNucci, 2012; Fischer & Shipman, 2011; Horner & Atwood, 2006; Lee, 1997; MacLean et al., 1991; McCall, 2010; van Oers et al., 2018; Wolff, 2000):

1. It can improve *communication* and cooperation between designers, other specialists and stakeholders. Indeed, explicit design rationale allows a wider range of stakeholders to understand the concept design.

2. It can enhance *documentation.* Indeed, expressed design rationale makes tacit knowledge explicit and thus more tangible. This is also useful for long-term storage and training within organisations.

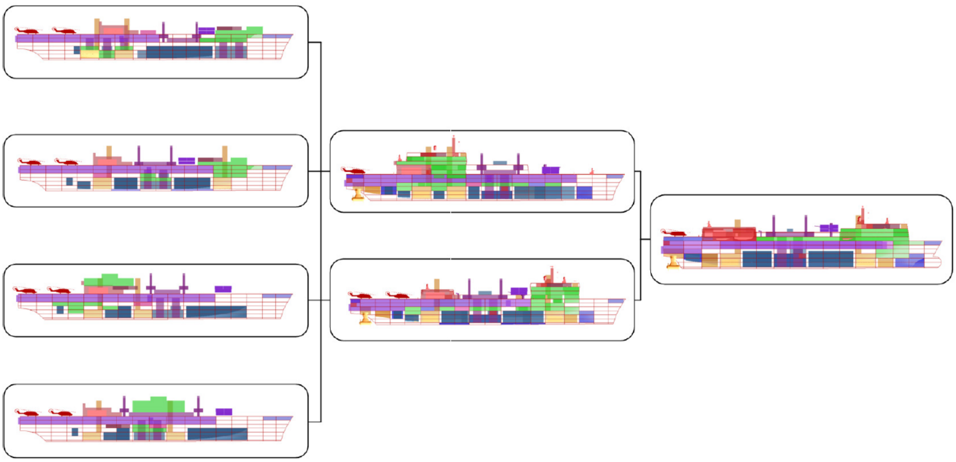


Figure 1.11: Evolving concept designs of a Joint Support Ship, (Andrews & Pawling, 2007).

3. Being explicit about the rationale behind an idea can improve *argumentation* by triggering critical thought and reflection on that idea. For instance, this can reduce logical jumps in thought processes.

However, besides the benefits of design rationale methods, there are various documented challenges for design rationale methods, (e.g. Ball et al., 2001; Burge & Brown, 2000; Conklin & Yakemovic, 1991; Fischer et al., 1991; Fischer & Shipman, 2011; Lee, 1997; le Poole et al., 2022c):

1. Design rationale methods may be perceived to be not cost-effective when the designers bearing the costs are not the same as the benefiting persons.
2. Designers can be reluctant to take the time to document the decisions they did not take or took and then were rejected. That is, they are less willing to spend time on ideas considered invaluable. However, the reason why these ideas are considered 'invaluable' can be 'valuable' information at a later stage (e.g. rework of the design).
3. Such methods can be intrusive in the design process. As a consequence, designers can be hesitant to use a design rationale tool besides other design tools. Regarding the individual designer, capturing all design rationale is not possible (e.g. because decisions can be taken unconsciously based on tacit knowledge) and not desirable (design issues and their solutions can be obvious).
4. There can be a social barrier to documenting (potentially wrong) decisions (e.g. job security, accountability, and liability).

DESIGN RATIONALE APPLICATIONS IN SHIP DESIGN

While design rationale has been researched frequently in the fields of software design (e.g. Aladib and Lee, 2019; Jarczyk et al., 1992) and aerospace (e.g. Aurisicchio et al., 2016; Bracewell et al., 2009; Kuofie, 2010), the application of design rationale in ship design to trace design decisions has been limited so far.

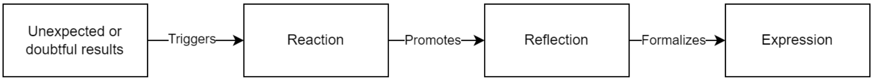
In ship design practice, concept designs often need to comply with pre-defined design rules, such as classification society rules and international regulations. Many of these rules lack the actual design rationale, i.e. the rules reflect the implicit, underlying rationale of why that rule was needed and how it was developed (Derbanne, 2022). For example, a design rule is that ships should have double lifeboat capacity, spread over the port and starboard sides. The implicit rationale is that if the ship capsizes to one side, still sufficient lifeboat capacity is available on the other side. As a result, often the explicit rationale behind concept designs is missing. Hence, if a trade-off must be made, there is no underlying rationale to see if other solutions could have a similar functional effect. For example, by making sure the design does not heel through other means. This example shows the difference between rule-based (i.e. based on design rules) and performance-based design (i.e. based on required performance) (e.g. IMO, n.d.).

Design rationale in ship design can be distinguished at different levels throughout the iterative design process. During early design synthesis, the designer will make decisions (with corresponding design rationale) on aspects such as the overall style of the ship (including, for instance, level of survivability, hull type, and propulsion concept) and generate concept designs comprising major building blocks (see for instance Andrews & Dicks, 1997; van Oers, 2011b; Takken, 2009). Similarly, when the design is further developed in more detail, decisions and rationale are more related to details, such as accurate arrangement and sizing of systems. It should be noted that the evolution of the concept design is not a goal in itself. For example, Baker (1956) used his “stylised design” (restricting the allocation of single functions to specific areas of the layout) to ensure that stakeholders beyond the designer were constrained from “interfering” in the design (Andrews, 2022b). However, further development and detailing of concept designs are often necessary. For instance, requirements can change, assumptions might need revision, and a higher level of detail might be required to ensure technical feasibility and identify risks (Van Oers et al., 2018).

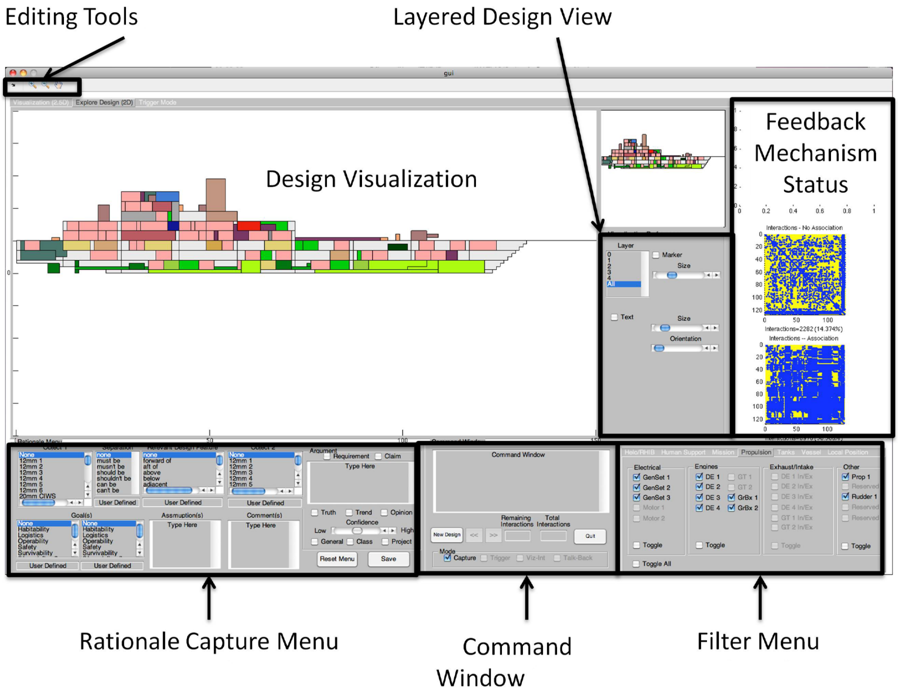
In ship design, research into design rationale is limited. The primary example is DeNucci (2012), who developed a design rationale method to trigger individual designers to express and capture design rationale. Figure 1.12 shows DeNucci’s principle process and the implemented tool. His Reactive Knowledge Capture (RKC) tool presented automatically generated unconventional, unexpected concept designs to designers to elucidate what they *did not like* about the layout of these designs, in terms of global positions of systems and relative positions between systems (DeNucci, 2012). Although this proved to be an acceptable way to elucidate design rationale for ship layout design, reversing the captured logic will not automatically result in an acceptable concept design because:

1. A ship is “a mass of compromises” (Forester, 1942). Therefore, design rationales in ship design might, and are likely to, conflict. Thus, compromises need to be made (DeNucci, 2012). Conflicting design aspects often require a dialogue between multiple stakeholders. DeNucci (2012) did not capture these resulting trade-offs.

2. Design rationale can be situation or project dependent. Therefore, the captured rationale might not be sufficient to make a fully informed trade-off.



(a) RKC process (DeNucci, 2012).



(b) RKC implementation (DeNucci, 2012).

Figure 1.12: Reactive Knowledge Capture (RKC) (DeNucci, 2012).

3. These new concept designs might, in turn, trigger designers to express additional preferences which were not triggered by the original 'wrong' designs.

4. Time and budget availability are typically low during concept design, compared to detailed design phases (Tupper & Rawson, 2001, p.634), and therefore may be questioned whether designers are willing or able to spend time on expressing what's *not* wanted (Conklin & Yakemovic, 1991), before trying to implement the reverse logic in a feasible and balanced concept design. Yet, stakeholders might be better able to express what they do not like, as opposed to what they like (DeNucci, 2012).

It should be noted that various tools and methods enable traceability of ship design aspects. For instance, tools such as Dynamic Object-Oriented Requirements System (DOORS) (IBM, n.d.) and Shipbuilder (Shipbuilder, n.d.) enable detailed traceability of requirements if implemented and documented by the user. Recent developments allow designers to trace the relation between requirements and the concept design using Shipbuilder as well (Van der Weg, 2020). Also, Model-Based Systems Engineering (MBSE) is aimed at the traceability of requirements and concept designs (Droste & Hage, 2022; Kooij, 2022; Tepper, 2010; Zech et al., 2022). Although design rationale can be captured within requirement traceability tools and MBSE software, design rationale-specific issues (such as: what to capture and how, see Section 5.1) still need attention (e.g. Do et al., 2014). Therefore, this dissertation specifically focuses on design rationale.

Concluding, being explicit about the reasoning behind design decisions, i.e. design rationale, is essential to explain and understand the development of concept designs over time. In practice, design rationale may be documented and also existing research shows that design rationale can be captured and reused. However, there is currently no suitable design rationale method that allows for integrated, in-situ documentation of design rationale during the complex ship layout design, described in Section 1.1.2. Therefore, it's currently unknown how the potentially intrusive activity of design rationale capture can be effectively integrated into the complex ship layout design process. Hence, this dissertation aims to generate and evaluate a tool to allow for in-situ documentation of design rationale during ship layout design. This will be elaborated in Part II.

1.2. RESEARCH OBJECTIVE AND QUESTIONS

In conclusion, Section 1.1 shed light on the nature of early-stage ship design, as well as the importance and challenges of the two interrelated research directions for this dissertation: concept design generation and design rationale capturing during early-stage ship design.

Section 1.1.1 has identified that design decision-making during early-stage complex ship design inherently is of an iterative nature. This iterative nature is caused by the 'wicked problem' of early-stage design. This requires a concurrent problem definition and solution generation in a social-technical context. Because of evolving insights, preferences, and requirements, the human designer frequently needs to reconsider past design decisions. A key question is how designers can be supported to make more informed design decisions, potentially increasing the quality of design decisions, and therefore reducing the need to reconsider 'wrong' decisions. This could reduce the need for costly rework later in the design process. Such timely design insights can be crucial during the ongoing stakeholder dialogue during the early design phases. Specifically, the focus is on the interrelated problems of layout generation and design rationale capturing. The reasoning is summarised below.

In Section 1.1.2, the importance of layout design in the complex ship design process was described. While many design tools and methods have been developed, a lack of design tools to provide sufficiently fast insights into potential detailed sizing and integration issues was identified. This gap leads to a late, or time-consuming, identification of the consequences of layout design decisions during lower level of detail manual layout design during early-stage complex ship design. By enabling timely insight into the technical feasibility of layouts, the risk of time-consuming, costly design recursions can, again, be reduced.

Section 1.1.3 approached complex ship design from a design rationale perspective. The availability of the justification behind design decisions is important to enable informed reconsideration of these decisions when rework needs to be performed. It was identified there is a lack of methods to support design decision documentation during the complex ship layout design activities, potentially leading to a loss of context of documented design decisions. By enabling designers to capture design rationale in a non-intrusive way, designers will be better supported to understand and explain the development of concept designs over time.

Taking these aspects together, the overall research objective for this dissertation is:

To reduce the effort required to identify and solve detailed layout integration issues during social-technical early-stage complex ship design via automated layout generation and design rationale capturing.

To achieve this research objective, the following research questions (RQs) are answered in this dissertation:

RQ1. To what extent can automated layout generation methods support real-time design decisions during early-stage complex ship design? (Part I)

This dissertation seeks to *support designers by reducing the effort required to investigate technical feasibility and potential integration issues of detailed ship layouts.* To reduce the need to reconsider design decisions, designers can benefit from insight into the possible consequences of future design decisions. Such consequences can be related to technical feasibility, for instance. However, obtaining these insights can be a time-consuming task in itself. The availability of a (semi-) automated layout generation tool is expected to reduce the required design efforts, enabling earlier design insights.

RQ2. To what extent can design rationale methods support real-time design decisions during early-stage complex ship design? (Part II)

This dissertation aims to *support designers in documenting and evaluating the context and justification of design decisions.* To make informed design decisions, the context and intention of past design decisions need to be available. Since such design rationale frequently resides in designers' memories and notebooks, design timelines for complex ships are often long, and design teams change over time, essential design rationale is not always available or retrievable when design decisions need to be reconsidered. Since current design rationale methods in ship design are insufficient, there is a need for a new design rationale method to support early-stage complex ship design decision-making.

1.3. CONTRIBUTIONS AND DISSERTATION OUTLINE

The research presented in this dissertation leads to an enhanced way of supporting design decision-making during early-stage complex ship layout design. Specifically, this research contributes to the field in the following ways:

- *Development of a layout design tool to provide real-time insight into layout sizing and integration issues during early-stage complex ship design.*
  Specifically, this tool enables designers to rapidly generate and evaluate concept designs for ship layouts at a higher level of detail. This is enabled through the first

use of cross-correlation to enable a very fast arrangement of large layouts on a space level. In addition, a probabilistic staircase placement algorithm, a network-based approach combined with probabilistic selection for allocation of spaces to compartments, and a 'carving'-based approach to ensure connectivity have been used to create realistic detailed arrangement plans.

- *Development of a design rationale method to enable design decision traceability during early-stage complex ship layout design.*
  This design rationale method is integrated with a ship layout design tool to enable designers to concurrently work on the concept design, and capture and retrieve design rationale. An important effect of such an integrated approach to design rationale is that it triggers designers to be more explicit about their design reasoning.
- *The experimental evaluation of short and long-term benefits of design rationale capture and reuse for complex ship layout design.*
  Two case studies are executed. In the first case study, a small design problem is used in three-person teams to investigate the role of design rationale within design sessions. In the second case, a partial frigate arrangement problem is used in two successive design sessions (with an interval of multiple weeks) to investigate the value of design rationale over longer periods of time.

The structure of this dissertation, and the relation between the research objective, research questions, contributions, and dissertation parts is visualised in Figure 1.13. The reader will find more elaborate gap analyses in Chapter 2 (Part I) and Chapter 4 (Part II).

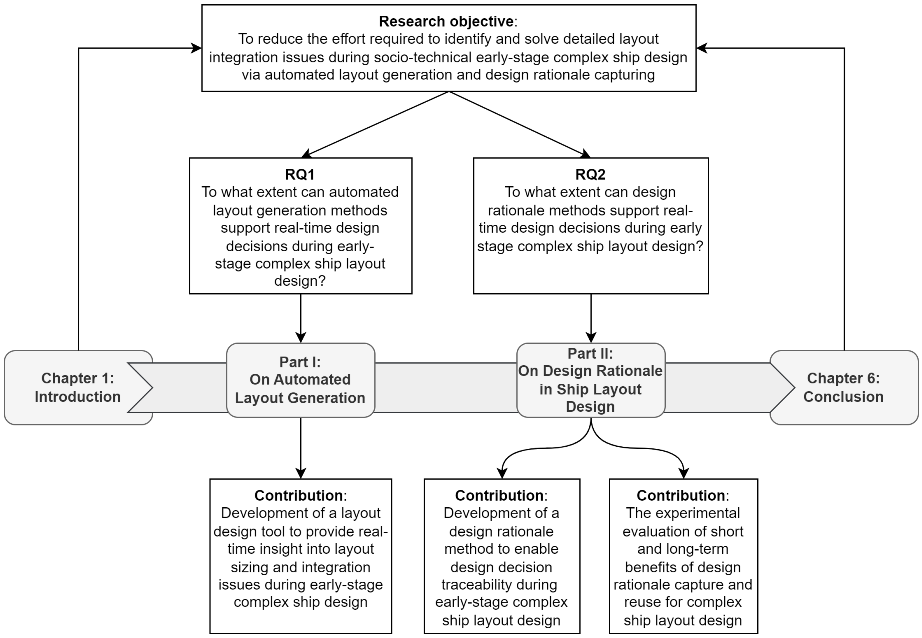


Figure 1.13: Structure of this dissertation

I

On Automated Ship Layout Generation

This part focuses on the generation of detailed layouts for complex ships. Chapter 2 proposes a new layout generation method, called WARship GEneral ARrangement (WARGEAR). Subsequently, this method is evaluated in Chapter 3. The aim is to reduce the effort required to obtain insights into detailed layout sizing and integration issues during early-stage design.

2

A NEW AUTOMATED LAYOUT GENERATION METHOD

The architectural arrangement of a ship, its layout, greatly affects its cost, capability and the convenience with which the crew carry out their tasks, but architecture is difficult to set out in guidelines and impossible to specify in a contract.

D.K. Brown (1991)

In Chapter 1, the need for a new ship layout design method was identified. Such a design method should be able to generate sufficiently detailed arrangements in a limited time to support designers in real-time design decision-making. Therefore, the main question for Part I is: *To what extent can automated layout generation methods support real-time design decisions during early-stage complex ship design?* To answer this question, first, this chapter specifies the research problem and research gap in Section 2.1. Subsequently, a literature review of layout modelling techniques will be provided in Section 2.3. Then, the mathematical details of the new ship layout design method, called WARGEAR (WARship GEneral ARrangement) will be extensively discussed in Section 2.4. Chapter 3 will provide extensive test cases using the new method.

2.1. GAP ANALYSIS

2.1.1. PROBLEM IDENTIFICATION AND METHOD REQUIREMENTS

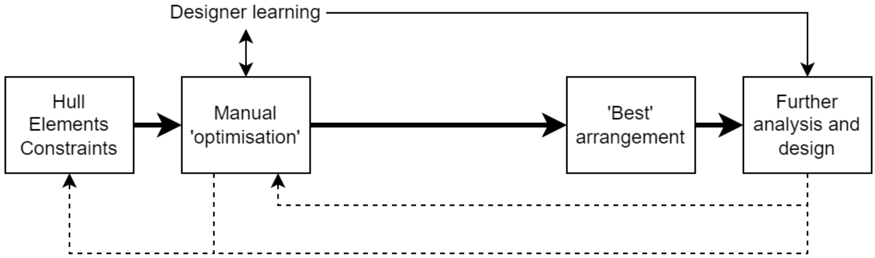
In Section 1.1.2 the essence and importance of ship design layout design were elaborated. As described, manual design tools are mainly deemed necessary to allow for better human control of the design progress and conscious consideration of design changes.

To achieve the 'best' concept design designers have traditionally relied on an iterative process of manual 'optimisation' through the identification of risks and challenges

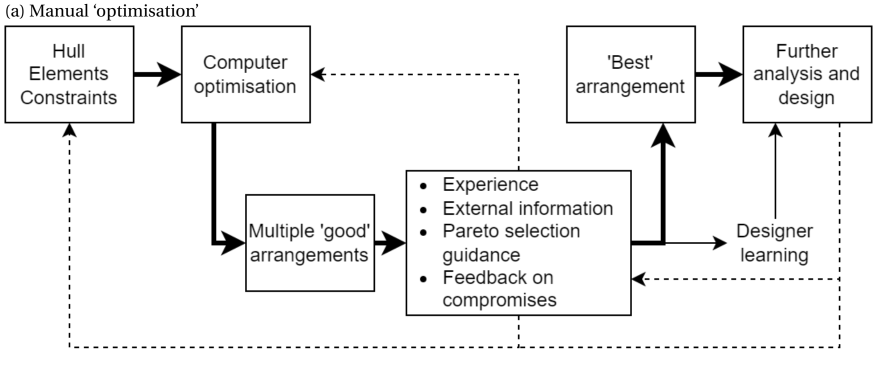
Parts of this chapter are based on Le Poole et al. (2019, 2022d).

and subsequent evolution of the design to address these issues (Cort & Hills, 1987; Gillespie, 2012). For layout design, this process involves, amongst others, drawing arrangement plans by hand or using a Computer-Aided Design (CAD) tool (e.g. Rhinoceros (McNeel, n.d.)). Although this involved process offers great learning opportunities for the designer, it is also a tedious and time-consuming process. As a consequence, designers might select design options before higher levels of fidelity are investigated (Singer et al., 2012). However, this higher level of fidelity will still be investigated in later design stages and might disclose sizing and integration issues. This late identification of issues can lead to time-consuming iterations of the overall design. This process is visualised in Figure 2.1a.

To address this issue, automated layout design tools were developed, able to generate hundreds to thousands of feasible design options (e.g. Nick, 2008; van Oers, 2011b). A major goal of early-stage design is to understand the relation between requirements and the technical and financial impact on the design (Section 1.1.2). Therefore, post-processing of data is essential to facilitate both designer learning and selection of the 'best' design (Gillespie, 2012). This down-selection can be performed via, for instance, Pareto-front selection or can be based on designer experience. This process is visualised in Figure 2.1b.



(a) Manual 'optimisation'



(b) Computer-based optimisation with designer selection

Figure 2.1: Manual versus computer-based optimisation with designer selection (adapted from Gillespie, 2012). Dashed lines indicate feedback loops, note that these also comprise 'designer learning'.

Despite the advantages of the computer-based optimisation process, if the level of detail required to ensure technical feasibility goes beyond the capabilities of design tools, designers fall back to the manual optimisation process with it's limitations (Le Poole et al., 2019). On a side note, this is also a reason for the switch between design tools in different design phases (Duchateau, 2016). This problem can be addressed by modifying existing design tools to extend their capabilities or by developing new tools or methods to bridge the capability gap. As described in Chapter 1, bridging this capability gap has three main advantages:

1. A reduction in effort required for the creation of detailed arrangement plans enables the generation and analysis of a larger number of layout variations. This increases the amount of design insight to support the stakeholder dialogue. Increased design insight is especially useful during concept exploration and early concept definition when the most important design decisions need to be taken.
2. Timely insight into sizing and integration problems enables designers to make necessary changes to the design earlier in the process, reducing the cost of design rework.
3. More design variations can be generated and analysed. This in turn enables a more thorough investigation of possible trade-offs and identification of potential risks. Hence, the designer is presented with a more elaborate set of design options to choose from. A larger set of design options enhances the designer's ability to identify trends compared to a few design options.

Regardless if a tool or method is manual or automated, some high-level characteristics need to be fulfilled (Andrews, 2012b). He proposes five characteristics for early-stage concept design tools, see Table 2.1. For the purpose of this dissertation, an indication is given if these characteristics can be achieved by manual and automated design tools. The following two observations can be made:

1. Manual design tools are most suitable during early-stage design, except for generating revelatory design insights, for which automated tools could be more useful.
2. Achieving all requirements with an automated design tool (especially for detailed layouts) can be challenging, as multiple aspects need to be addressed (Section 1.1.1).

Based on these observations, a fully automated design tool to fulfil the capability gap described above seems hard to achieve - and might also not be accepted (Reinertsen, 1997). Also, it might not be needed, considering that early-stage design is not only layout design. A promising idea is to integrate the use of manual and automated design tools, to speed up parts of the layout generation process while keeping the essential human control over the concept generation process.

As noted in Section 1.1.2, this general problem was identified at the Netherlands DMO as well, and led to the research presented in this dissertation. Hence, the DMO case will be further elaborated to illustrate the higher-level problem described above. During the concept definition design phase at the DMO, FIDES (Takken, 2009) is used as the principle ship synthesis tool. FIDES has many similarities with the Design Building Block approach (Andrews & Dicks, 1997). Using FIDES, designers can generate functional arrangements (such as shown in Figure 1.9c), with various levels of detail. Functional arrangements can comprise various Functional Building Block (FBB)s. The level

2

Table 2.1: Characteristics for design tools (adopted from Andrews, 2012b; Duchateau, 2016).

Characteristic	Explanation	Manual tools	Automated tools
Believable solutions	Solutions should be sufficiently descriptive and technically balanced (e.g., they must obey the laws of physics, the basic principles of naval architecture, and the necessary rules and regulations)	Very suitable	Suitable - However, stakeholder acceptance can be an issue
Coherent solutions	Solutions should be presented in a format that can be understood by stakeholders (Kossiakoff et al., 2003). For complex ship designs, the availability of 2D or 3D layout plans is essential, since much of the information, requirements, and performance is based on spatial data, and problem-solving often involves not only numbers but also architectural aspects (Andrews & Dicks, 1997; Duchateau, 2016; van Oers et al., 2018; van Oers, 2011b; Pawling, 2007). Solutions should be believable enough to prompt stakeholders into conversations on (rather than scrutinising) these designs	Very suitable	Very suitable
Open methods	No 'black box' but responsive to issues that matter to stakeholders, in a timely manner (Van Oers et al., 2018). Indeed, slow tools can damage the dynamic design process, resulting in stakeholders relying on quicker, but not necessarily better, tools to keep up and support decision-making by providing insight into design drivers (Duchateau, 2016)	Very suitable	Can be challenging by the very nature of automated tools
Revelatory insights	Able to support the identification of likely design drivers early on	Suitable - However, the required effort to generate a sufficient number of design options can be an issue	Very suitable - Although the required effort for data analysis can be considerable
A creative approach	Allowing for radical solutions to push requirements elucidation boundaries	Very suitable	Can be challenging - Although it's often the designer to set the tool's boundaries

of detail of FBBs can differ. For instance, an engine room might include detailed objects like engines, gearboxes and propulsion shafts to ensure this equipment actually fits inside the hull. Other FBBs, like accommodation areas, are often modelled with less detail and only represent the estimated required area, for instance based on naval standards and the required manning. A margin might be used to account for area required for access means, i.e. hallways and staircases. FBBs can also be volume driven, like fuel and water ballast tanks, or represent specific systems like cranes or sensor systems.

Van Oers et al. (2018) explain that the manual FIDES tool is preferred during the concept definition phase to ensure human control of concept design evolution (see also Section 1.1.2). Although certain parts of the concept design might be defined at high fidelity (e.g. engines), typically, designers aim to maintain a flexible (i.e. low fidelity) concept design to allow for a timely response to changes in requirements, preferences, and compromises. However, a major disadvantage of such lower-fidelity concept designs is that some sizing and integration issues might only be identified later in the design process, as elaborated above. To solve this issue, designers manually develop high-fidelity GAPs in generic CAD software. However, developing such GAPs can require up to 150 working hours (Le Poole et al., 2019) and, therefore, only a limited number of high-level of detail design variations can be studied (see also Section 3.2). This limits the amount of feedback to the functional arrangement as well as the number of identified sizing and integration issues, often resulting in time-consuming iterations of the functional arrangement later in the design process. Late changes are an issue, as shown in Section 1.1.1.

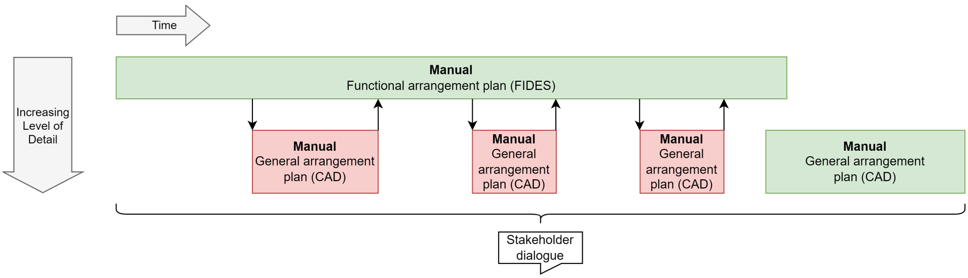
This process is visualised in Figure 2.2a.

Although this problem can be solved by increasing margins (resulting in larger concept designs), this is not an appropriate solution. Indeed, this solution increases the probability of overestimating the required space and therefore the probability of oversizing designs, which in turn might lead to poor cost estimates. Therefore, a more elaborate and location-specific estimation of the space required for accessibility is required. The following three uses of the GAP justify the focus on gaining insight into detailed layouts *in a limited time*:

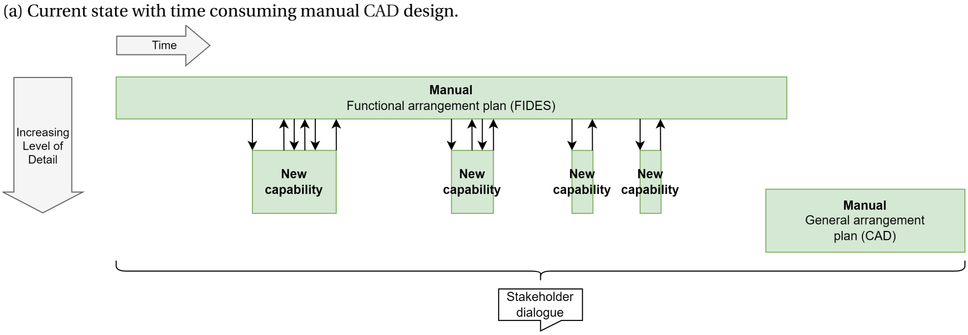
1. The GAP is used to validate the functional arrangement, i.e. to ensure the functional arrangement will lead to a feasible detailed design.

2. The insight obtained, both from the generation process of the GAP and the GAP itself, is used to support the stakeholder dialogue during early-stage design.

3. The GAP is used for further analysis of the design, e.g. for logistic performance assessments such as these developed by Droste et al. (2018).



(a) Current state with time consuming manual CAD design.



(b) Future state with new capability.

Figure 2.2: Current and future envisioned state (applied to DMO case). Arrows indicate iterations.

To improve the current state, a new capability (i.e. reducing the workload necessary to study detailed arrangement aspects) needs to be introduced to the design process. This might be an existing design tool, an extension thereof, or a new solution. Which option to pursue will be investigated in Section 2.1.2. However, before such an investigation can be conducted, the envisioned future state and method requirements will be defined. The future state is visualised in Figure 2.2b. In the future process, designers rely less on manually generated GAP but use a solution that generates sufficiently detailed arrangement plans in a very limited time to provide rapid insight into potential sizing and integration issues in low-level of detail functional arrangements. Such a solution should fulfil the following requirements[1]:

LRQ1. **The method should provide insight into potential sizing and integration issues, as well as insight into ways to resolve these issues.** This is the main aim, as identified above.

LRQ2. **Its speed should be in the order of minutes**. To provide designers with real-time insight into the consequences of design decisions, a rapid feedback loop needs to be possible. This requires the method to be fast, i.e. not be the bottleneck in the decision-making process. Indeed, as discussed in Section 1.1.1, decision-making itself is a very human act.

LRQ3. **The starting point is a predefined functional arrangement**. In line with the gap identified above, the method should support manual concept design development. The level of detail of these concept designs typically is of functional building blocks, i.e. functional arrangements (Andrews & Dicks, 1997; van Oers et al., 2018; Takken, 2009).

LRQ4. **The level of detail of generated layouts should be high**. The new method should provide more insight than is practically achievable with the functional arrangement serving as a starting point. As a consequence, the method is to arrange individual systems (e.g. cabins and workshops). While a functional arrangement can contain these spaces too, this will quickly reduce its flexibility and adaptability to changing requirements.

LRQ5. **The main driver for layouts to be considered is area**. The method is focused on 2D deck layouts because volume blocks (e.g. fuel tanks) can be easily modelled in a functional arrangement. Furthermore, the focus is on systems that vary in size (e.g. cabins), in contrast to fixed-size systems (e.g. engines). Also the latter can be modelled in a functional arrangement (Takken, 2009) and does not require less flexibility regarding potentially changing requirements.

LRQ6. **The number of diverse solutions should range from a few to hundreds**. The principle question to be answered by the required method is: ‘Does it fit, and why (not)?’. To answer this question, the method should be able to be able to generate a sufficiently large set of design alternatives, to provide insight into the feasibility and risks of the predefined functional arrangement. As such, the set of design alternatives can be small, and hence visual inspection can be feasible. If the set is large, appropriate means of analysing the set should be used.

[1]LRQ: Layout method Requirement.

2.1.2. EXISTING SHIP DESIGN TOOLS

Above, a set of requirements is given to address the identified problem. In this section, five major tools and methods used in ship design will be investigated regarding compliance with these requirements. The characteristics of the five tools and methods is summarised in Table 2.2. Intelligent Ship Arrangement (ISA) (Daniels et al., 2009; Nick, 2008) comes closest to meeting the requirements. However, results need to be provided in a more limited time. ISA is designed to support the naval architect in developing general arrangements and takes the definition of the hull, decks, and bulkheads as input. In contrast, the identified problem requires the method to take a medium level of detail functional arrangements as input (e.g. from FIDES (Takken, 2009) or DBB, which originated functional arrangements (Andrews, 2003)). However, predefined functional arrangements are no input to ISA. Hence, the overall conclusion is that there is currently no ship design tool that fulfils the requirements set in Section 2.1.1.

Thus, a choice between extending the capabilities of an existing tool and developing a new tool or method arises. Besides suitability, availability of the tools is a major discriminator for the former. Only Packing and FIDES are available. However, these tools are both volume-driven, while the requirement is to focus on area. FIDES is a manual tool with, at the time of conducting the research, limited possibilities for automation. Automation of the arrangement generation is seen as the only possibility to meet requirements LRQ2 and LRQ6. Furthermore, although automated, Packing would require a two-step fidelity increase as well as the ability to take a predefined functional arrangement as a starting point (LRQ3). Although this might be possible, this would require a complete redevelopment of the Packing source code, i.e. it would basically be a new tool. Hence, a new method will be developed in Section 2.4.

Table 2.2: Characteristics of ship layout design tools, adapted from (Duchateau, 2016; Gillespie, 2012).
[1]: estimated as minutes

Method	Driver	Architectural	Diversity	Speed	Num. of solutions	Level of detail
DBB (Andrews, 2003, 2018b; Andrews & Dicks, 1997)	volume	3D full ship	overall	hours-days/ manual	few	low to high
ISA (Daniels et al., 2009; Nick, 2008)	area	2D deck	arrangement	hours/ automated	hundreds	high
Packing (Van Oers, 2011b)	volume	2.5D/ 3D full ship	overall	hours/ automated	thousands	low
FIDES (Takken, 2009)	volume	3D full ship	overall	days/ manual	single	medium
Gillespie (Gillespie, 2012)	adjacency	network	arrangement	unknown[1]	few	low
Requirements	area	2D deck	arrangement	minutes	few to hundreds	high

2

2.2. DEFINITIONS

The following terms are widely used in the remainder of this dissertation and have the following meaning:

1. *Space*: A space is defined as a room (e.g. a cabin or a galley) in the ship. A space has properties such as Required Area (RA) and allowed Aspect Ratios (ARs).

2. *Compartment*: A compartment is defined as a volume inside the ship, enclosed by decks and bulkheads.

3. *Functional Building Block*: An Functional Building Block (FBB) is defined as a low-level of detail representation of one or more spaces serving a similar functionality (e.g. an accommodation FBB might represent multiple cabins). An FBB can overlap (partially) multiple compartments, see Figure 2.3. FBBs are derived from the design building blocks used in the Design Building Block approach (DBB) (Andrews, 2003; Takken, 2009).

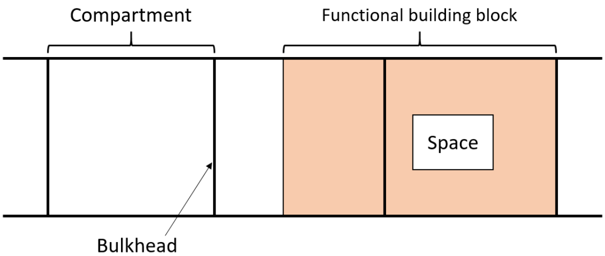


Figure 2.3: Visual explanation of the subdivision of a deck, presented in a top-down view. An FBB overlapping multiple compartments is shown (Le Poole et al., 2019).

2.3. LAYOUT MODELLING TECHNIQUES

In this section, existing ship layout design methods and tools, as well as a range of arrangement techniques used in building architecture will be reviewed to identify potentially useful ways to model detailed complex ship layouts for the purpose of developing the new layout design method.

2.3.1. INSPIRATION FROM SHIP LAYOUT DESIGN METHODS AND TOOLS

Although none of the tools investigated in Section 2.1.2 meets all requirements, the following components are considered valuable for the development of the new method:

1. *Visualisation and exploration of automatically generated layouts* can be supported via scatter plots (Duchateau, 2016; van Oers, 2011b), but also requires full layouts to be generated for detailed insights (Daniels et al., 2009; Duchateau, 2016; Nick, 2008; van Oers, 2011b; van Oers et al., 2008).

2. *The use of networks to allocate spaces to compartments* has been investigated by Gillespie (2012) and proved to be a powerful way to deal with the multitude of system adjacency and global location requirements that need to be satisfied in a feasible layout. However, the available area in compartments is not considered, which can cause over-utilised compartments. Also, defining all relationships between systems for each ship design (project) can be a time-consuming activity itself (Gillespie, 2012). For example, the network used by Gillespie (2012) contains 103 nodes (i.e. spaces) and 1017 edges (i.e. relative location constraints and relationships). Note that this network originated from a specific ship. Hence, the network needs to be redefined, or at least checked, for different ship types as well as concept designs (e.g. a change in manning concept might impact the set of spaces to be arranged, Section 3.2). Networks can also be used to evaluate ship designs, see for instance Gillespie (2012), Pawling and Andrews (2018), and Roth (2016).

3. *Space arrangement optimisation* has been approached in different ways. For instance, ISA (Daniels et al., 2009; Nick, 2008) uses a growth-based approach, while Packing (Van Oers, 2011b) utilises an overlap detection and removal approach. Van Oers (2007) identified that, besides overlap detection and removal, parametrisation of space positions can be done based on the sequence in which systems are arranged. For an example of the latter, see Lee et al. (2005).

   Both Packing and ISA used a Genetic Algorithm (GA) to perform the optimisation, but also other optimisation algorithms, such as Particle Swarm Optimisation (PSO), are being used (Cui & Turan, 2010).

4. *The sequence in which systems are arranged is important in both manual and automated design approaches.* For instance, Andrews and Pawling (2008) and Brown (1987) discuss arrangement sequences for manual approaches. Examples of arrangement sequences in automated approaches can be found in Gillespie et al. (2013), Nick (2008), and van Oers (2011b).

   - Brown (1987) proposes to start with arranging access routes and subsequently hierarchically arrange systems considering size and importance.
   - Andrews and Pawling (2008) propose to:
     (a) Commence with those blocks already seen as causing design unbalance or conflict;
     (b) Select the largest blocks first before tackling smaller blocks;
     (c) Select the most constrained blocks before those less constrained;
     (d) Start with the FLOAT blocks, then the MOVE blocks, followed by the FIGHT blocks, and, finally, the INFRASTRUCTURE blocks.
   - Van Oers (2011b) proposes the following sequence for Packing: hull, decks, (optionally, very large or constrained objects), bulkheads, other objects "in the order from *large and/or constrained*, to *small and/or less constrained*" [emphasis added].

- ISA (Nick, 2008) commences with the arrangement of staircases and spaces in compartments on the Damage Control Deck (DCD), and subsequently iteratively arranges spaces within all compartments.
- Gillespie et al. (2013) allocates communities of spaces based on descending global location preference, i.e. most restricted spaces are allocated first.

Generalising, sequential approaches tend to first tackle (perceived) large, constrained, or constraining systems before arranging systems with less impact on the overall design.

2.3.2. Inspiration from building architecture

In the field of building architecture, automated layout generation has been pursued for decades. Although ships are typically more constrained than buildings, it is expected that existing architectural layout generation techniques can benefit the development of the new method.

The generation of building layouts typically starts with an architectural program, which is specified by the architect in dialogue with the customer. Such architectural programs capture requirements for the building and typically include items as the total floor area, the building's footprint, a set with rooms, the required area per room, required aspect ratios per room, required adjacency between rooms, and type of adjacency between rooms (Merrell et al., 2010). Others divide their layout generation approach into two steps (Guo & Li, 2017; Medjdoub & Yannou, 2000). First, a topology is created to generate a rough layout that satisfies connectivity requirements from the architectural program. Second, the topology is refined into spatial layouts. In this way, global layout decisions are made before commencing with more detailed decisions. This way, the need for an exhaustive generation of detailed layouts is eliminated. Indeed, only topologically superior layouts need further investigation.

To generate spatial layouts a number of approaches can be taken. Four approaches are discussed below.

1. *Tile placement.* Peng et al. (2014) presents predefined templates of irregular shapes with allowed shape variation, combined with a two-step approach to tile a predefined domain: 1) a discrete step to select approximate template positions and 2) a continuous step to refine and reshape the templates.

2. *Treemap algorithms.* A treemap is a way of visualising elements in hierarchical structures, in which an area is subdivided into smaller pieces, where the size of each piece is related to the importance of the pieces in the hierarchy. A special treemap algorithm is the squarified treemap algorithm to generate layouts, which attempts to subdivide the domain in pieces that have an aspect ratio close to 1 (Marson & Musse, 2010). The main drawback of this method is that it is only able to subdivide squares or rectangles, which do not well resemble the shape of a ship.

3. *Growth based arrangement algorithms.* Inspired by the growing of crops, growth-based arrangement methods start with populating a domain with 'seeds' which are

then iteratively grown to the required size. Examples are can be found in (Camozzato, 2015; Inoue & Takagi, 2008; Lopes et al., 2010). A ship design example using a growth-based algorithm is ISA (Nick, 2008).

4. *Inside out arrangement methods.* While the methods described above start with a predefined domain, inside-out arrangement methods start with arranging the spaces and 'wrapping' the outer wall around these spaces (Merrell et al., 2010).

Although vertical connectivity of buildings is necessary, i.e. vertical adjacent floors need to be connected, not all layout generation tools are able to generate multi-floor layouts. Examples of tools that generate single floor layouts can be found in Baušys and Pankrašovaite (2005), Camozzato (2015), Inoue and Takagi (2008), and Marson and Musse (2010). Multi-floor layout plan generators that include vertical connections such as staircases can be found in Guo and Li (2017), Lopes et al. (2010), and Merrell et al. (2010).

Since all spaces need to be accessible, architectural programs include requirements on adjacency, i.e. connectivity, between spaces (Merrell et al., 2010). Therefore attention needs to be paid to ways in which connectivity can be modelled. Two approaches are discussed:

1. *Inclusion of passageways in the architectural program*, and subsequent arrangement of these passageways in a similar way to space arrangement (Baušys & Pankrašovaite, 2005; Merrell et al., 2010). In the case of Baušys and Pankrašovaite (2005), passageways or hallways need to meet a minimum required area requirement.

   In addition, doors can be placed according to a topology (e.g. Guo & Li, 2017; Lopes et al., 2010). This ensures hallways are connected with adjacent spaces. Connectivity for spaces not adjacent to hallways can be assured via other spaces acting as a pass-through. In ships, some spaces might be used as a pass-through, but generally, each space should be directly connected to passageways. Therefore, for ship designs, all passageways should be defined in the architectural program or topology. However, the exact number and connectivity of passageways is function of the exact arrangement of spaces. Therefore, this approach seems less suitable for ship design.

   This approach eliminates a pre-arrangement decision on the number of passageways that is to be included in the layout. A major drawback, however, is that this approach can result in too small spaces, since area is carved away from spaces to create passageways. This could be solved by readjusting the floor plan if the final area of any room is smaller than the minimum required area (Marson & Musse, 2010).

2.4. WARSHIP GENERAL ARRANGEMENT (WARGEAR) METHOD

Section 2.1 identified a gap in current ship layout design methods. Specifically, a method to generate space-level arrangements based on a predefined functional arrangement in a limited time is lacking. Also, it was found that extending an existing method seems less promising than developing a new method. Therefore, a new method, called WARship GEneral ARrangement (WARGEAR) is proposed. This section will elaborate on the new method's working mechanisms in detail.

2

2.4.1. METHOD OVERVIEW

Figure 2.4 shows a high-level overview of the WARGEAR method. The method starts with a query from a designer. Next, a three-part method is executed: 1) input generation, 2) arrangement of spaces, and 3) post-processing of the results. Step 3 results in the required insight to answer the query or a change in input values to generate alternative layouts.

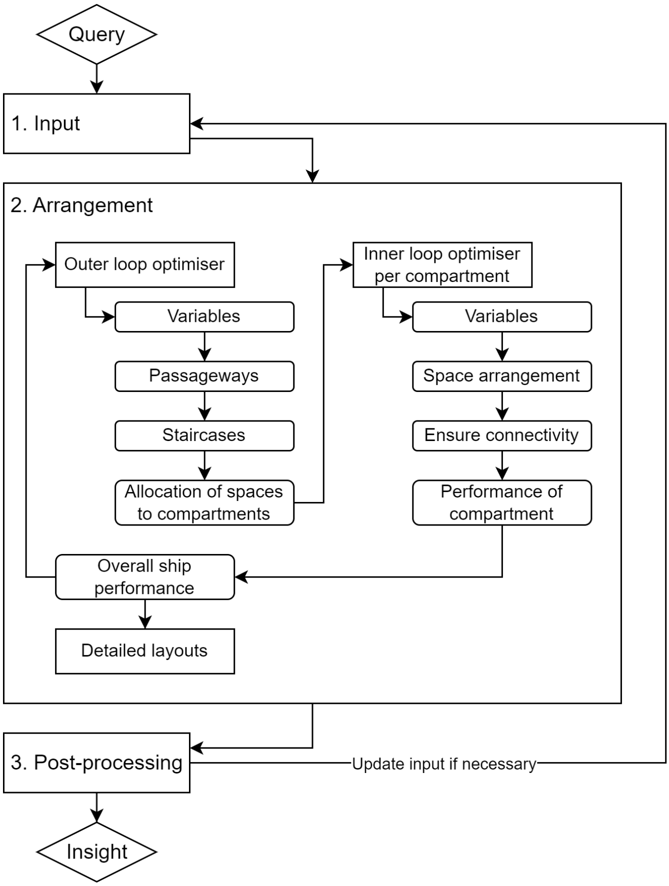


Figure 2.4: High-level overview of the WARGEAR method.

1. The *input* required for WARGEAR will be discussed in Section 2.4.2.
2. The mathematics used in the *arrangement* phase is elaborated on next. Section 2.4.3 explains how main passageways are generated. Subsequently, Section 2.4.4

elaborates on the arrangement of staircases. Next, Section 2.4.5 discusses the allocation of spaces to compartments. Section 2.4.6 introduces a novel space arrangement method. Then, Section 2.4.7 explains how connectivity is assured throughout the ship. Finally, Section 2.4.8 explains how the mathematical method is integrated and steered by a nested optimisation approach.

3. The *post-processing* of the resulting detailed arrangement is discussed in Section 2.4.9.

The order of arrangement steps in WARGEAR is based upon the idea that, first, global ship-level decisions are taken (i.e. the arrangement of passageways and staircases and allocation of spaces to compartments), where global is defined as influencing the arrangement of multiple compartments. These global decisions are most constraining to the design, as well as allow for the most accurate estimation of available area for allocating spaces to compartments. Second, compartment-level decisions are taken (i.e. the arrangement of spaces and ensuring connectivity). The first reason for this differentiation is that it is based upon other approaches to global and local arrangement problems (e.g. Medjdoub and Yannou (2000), Michalek et al. (2002), and Nick (2008)). The second reason is that compartment-level decisions have less impact on the total ship arrangement. The third reason for separating ship-level decisions from compartment-level decisions is to improve the arrangement of spaces at a compartment-level (Section 2.4.6), i.e. confined compartment arrangement problems are mathematically easier solvable than holistic ship arrangement problems. Overall, the arrangement procedure in WARGEAR compares well to arrangement sequences mentioned in Section 2.3.1, which tend to first tackle (perceived) large, constrained, or constraining systems before arranging systems with less impact on the overall design.

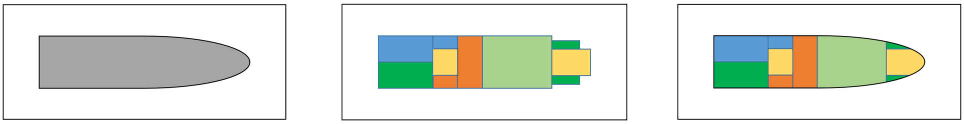
2.4.2. INPUT

The input required for WARGEAR consists of the following items:

- *Functional arrangement*. Functional arrangements are volume block-based arrangements. At the DMO, the FIDES tool (Takken, 2009) is used by naval architects to generate functional arrangements, describing the arrangement of FBBs. However, other tools could be used as well to create a low-to-medium level of detail 2D or 3D arrangement. This research does not aim to develop a tool that can be used by naval architects to develop a GAP from scratch. This is similar to ISA, which only requires a hull form and compartmentation, amongst others (Daniels et al., 2009; Nick, 2008). Instead, the goal is to overcome the challenges faced in using current tooling, namely by helping to speed up derisking of layouts during concept development, while keeping the flexibility of using low level of detail tools concurrently to keep up with the pace of the design process.

  The functional arrangement is used by WARGEAR to set the shape of the ship's hull and superstructure, as well as the rough internal arrangement divided up into different functional needs (cabin spaces, machinery spaces, operational spaces, etc.). To determine the net available positioning area, WARGEAR considers both the floor and ceiling of functional blocks. Typically, WARGEAR is used to arrange a set of spaces (as specified below) into a predefined set of functional blocks, for example, an accommodation block is detailed by WARGEAR by arranging a set of cabins. WARGEAR translates the functional arrangement into a positioning matrix, as shown in Figure 2.5.

1. The deck shapes are derived from the 3D CAD model created in FIDES. In WARGEAR, each deck shape is converted into a positioning matrix, as shown in Figure 2.5a. This positioning matrix is a discretisation of the functional arrangement plan, of which the grid size can be selected by the designer and is fixed for the whole layout. The positioning matrix is used for the subsequent arrangement of spaces, staircases and passageways.
2. All FBBs are described by rectangular or cubical coordinates (and trimmed within the hull in FIDES). This FBB data is also translated to a matrix (Figure 2.5b).
3. A simple matrix multiplication between Figure 2.5a and Figure 2.5b provides the full positioning matrix including the actual FBB shapes and locations (Figure 2.5c).



(a) Deck-shaped positioning matrix derived from the functional arrangement.

(b) Matrix containing the rectangular functional building blocks.

(c) Positioning matrix containing deck shape and functional building block locations.

Figure 2.5: Construction of the positioning matrix as input for WARGEAR.

- *Space list.* A standardised spreadsheet is used to create the space list for WARGEAR. For each space, the naval architect specifies the required area, the minimum and maximum allowed aspect ratio, the type of FBB the space is assigned to, as well as one or more specific FBBs that the space should be arranged in. Note that for many spaces this information is fixed and based on rules and regulations (e.g. accommodation standards for cabins). An example of such a space list is shown in Table 2.3. During the development of functional arrangements, naval architects often have an allocation of spaces to FBBs in mind (see Section 3.2.2). This envisioned allocation is a result of considerations of spatial, operational, and cultural aspects, for instance. Hence, WARGEAR does not explicitly decide on inter-space relationships but relies on the naval architect expressing preferences for decisions on these inter-space relationships via allocation to FBBs.

- *Staircase types and arrangement options.* The naval architect needs to specify the required staircases and overall staircase arrangement options for each compartment. By specifying that a certain compartment should contain a staircase, the naval architect connects that compartment with the overhead compartment.

Table 2.3: An example of the space list generated for WARGEAR.

ID	Name	Area	aspect ratio [low, high]	FBB name	FBB IDs		
1	Officer's cabin	15	[0.5, 1]	Accommodation cabins	25	26	27
2	Rating's cabin	20	[0.5, 1]	Accommodation cabins	27	28	29
3	Officer's day room	40	[0.5, 1]	Accommodation dayrooms	23	24	
4	Workshop	25	[0.5, 1]	Workshop areas	1	5	

Currently, up to three staircase types can be defined in WARGEAR. Within each staircase type an unlimited number of sizing variations can be specified. For instance, the naval architect might specify the following:

- Type: *(Escape) ladder*
  Size: 1x1m, 0.8x1.2m
- Type: *Standard staircase*
  Size: 3x1.5m, 2.8x1.2m, 2.6x1.1m
- Type: *Stairwell*
  Size: 3x3m, 2.8x2.8m

If, for example, a ladder needs to be arranged, WARGEAR will attempt to arrange a 1x1m ladder. If this attempt fails, the code will attempt to place a 0.8x1.2m ladder. If this also fails, WARGEAR alerts the naval architect that placing the required ladder is not possible.

Besides specifying the types and sizes of staircases, the naval architect may determine rough locations in which a staircase should be placed inside a compartment. Four options are available to choose from, namely:

1. Port side (PS). This option will place staircases in allowed positions towards the port side of the ship.
2. Starboard side (SB). This option will place staircases in allowed positions towards the starboard side of the ship.
3. Centre line (CL). This option will place staircases in allowed positions close to the CL of the ship.
4. No preference. This default option will use the general rules for staircase placement as specified in Section 2.4.4.

Further, specific functional blocks can be blocked for use in the staircase arrangement. For instance, the naval architect may specify that no staircases can be placed in storage rooms.

- *Run settings*. A variety of settings to run WARGEAR need to be specified. Examples are:

  1. Settings for the optimisation algorithm.
  2. File paths to relevant input files.
  3. Grid size, to control the resolution of WARGEAR's position matrices.
  4. Connectivity options.

2.4.3. PASSAGEWAY ARRANGEMENT

Since horizontal and vertical connectivity through the ship requires significant area, the placement of passageways and staircases (see Section 2.4.4) needs to be taken into account in generating detailed layouts. WARGEAR could decide upon the locations of all passageways. However, Le Poole et al. (2019) found that initial passageway routing in WARGEAR could yield unrealistic and unacceptable results. Also, it was found that including the main passageways in the functional arrangement improves the initial area estimation (also on the functional arrangement level), and thus leads to less technical risk. Furthermore, fixed main passageways reduce the calculation time, as the placement of main passageways is one of the main drivers for a good layout of spaces and staircases. Since design issues such as structural integrity are not considered by WARGEAR but are taken into account by the naval architect in the functional arrangement, predefined passageways yield more realistic results.

Additionally, WARGEAR offers naval architects the option to include a single longitudinal passageway in user-defined compartments. This option was included as naval architects might not model each main passageway in the functional arrangement. This might happen on less centralised decks or in small vessels, for instance. For each user-defined compartment that should include a main longitudinal passageway *j*, WARGEAR uses a variable y_j to determine the transverse position of that passageway. y_j is used to select between transverse positions that result in the longest passageway possible. For instance, some area might be blocked, and this procedure helps to route main passageways such that blocked area is avoided to the maximum extent possible. This procedure is visualised in Figure 2.6 for the case of two compartments.

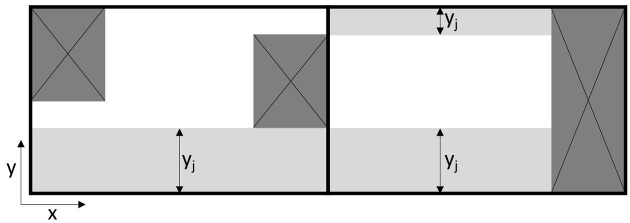


Figure 2.6: Top view of a two-compartment example on main passageway selection. Legend: *dark grey, black cross*: blocked area; *light grey*: available area for the main longitudinal passageway. The *two-headed arrow* indicates feasible y_j.

2.4.4. STAIRCASE ARRANGEMENT

While passageways tend to influence the arrangement of horizontally adjacent compartments, the staircase arrangement influences the arrangement of vertically adjacent compartments. Contrary to the arrangement of passageways, WARGEAR is used to arrange all required staircases (although the functional arrangement might include fixed staircases), because of the smaller footprint. A generic set of rules to determine the positioning of staircases was required to enable WARGEAR to find believable solutions (Section 2.1). Therefore, various existing GAPs of naval vessels and layouts generated by WARGEAR have been analysed and compared. This analysis led to the definition of the following rules for staircase positioning:

1. A long single staircase is preferred over split individual staircases, to ensure structural integrity as well as to improve deck area utilisation.
2. Staircases are typically placed directly adjacent to passageways, due to logistics and structural reasons. For instance, longitudinal bulkheads tend to be alongside main passageways. A higher probability for the arrangement of staircases along longitudinal passageways than transverse passageways is used.
3. If no passageways are available, staircases are placed in lobby-like areas, to avoid arrangement in functional spaces above or below, which is typically prohibited by regulations.
4. If no passageways or lobbies are available on a deck, any position can be chosen.
5. The preferred locations on all decks that need to be connected need to be considered in the staircase position selection.

These rules could become too restrictive for certain arrangement problems. However, these cases have not yet been encountered in the case studies described in Chapter 3. When WARGEAR is not able to place a staircase, a warning message is dropped.

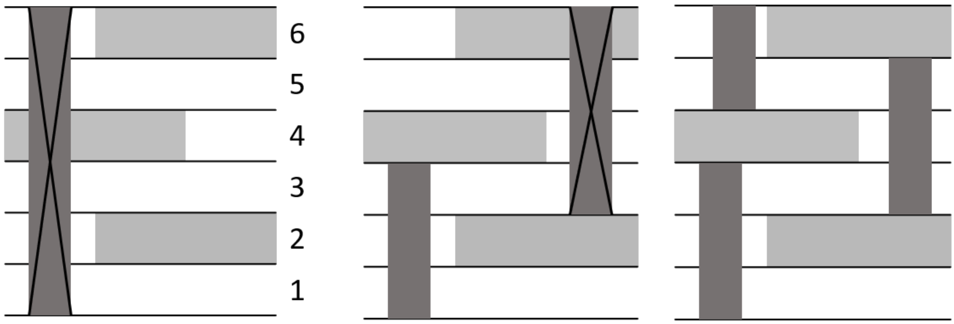
SPLITTING STAIRCASES

Following the first rule for staircase positioning, WARGEAR should arrange staircases such that staircases are as long as possible. The procedure used to determine how staircases are split into multiple staircases when a long staircase cannot be arranged is visualised in Figure 2.7. The figure shows a side view of six decks, where it is not possible to place one staircase across all decks due to the blocked area on deck 4. Two splitting operations are necessary to generate staircases such that decks 1 to 6 are connected. The splitting procedure is given in Algorithm 1. In the example shown in Figure 2.7, z_{split} is 4 and 6 respectively for the first and second splitting operations.

DETERMINING PREFERRED STAIRCASE POSITIONS

For each staircase to be arranged, the positioning guidelines outlined above are used to determine preferred positions. A staircase positioning (*SPM*) matrix is created for each deck based on these guidelines and is subsequently used in the final staircase position selection, which is discussed below. The $(i,j)^{th}$ element of SPM_z matrix for deck z is defined as follows:

$$SPM_z(i,j) = \begin{cases} 0, & \text{if the position cannot be used for staircases} \\ 1, & \text{if the position is directly adjacent to a} \\ & \text{longitudinal passageway} \\ 1, & \text{if the position is inside a lobby} \\ 0.75, & \text{if the position is directly adjacent to a} \\ & \text{transverse passageway} \\ 0.5, & \text{otherwise} \end{cases} \tag{2.1}$$



(a) The required staircase cannot be placed and needs to be split, as indicated by the black cross.

(b) The first splitting operation results in one placeable and one unplaceable staircase.

(c) The next splitting operation results in two additional staircases that are placeable.

Figure 2.7: Splitting one staircase into three staircases eventually to connect six partially blocked decks. Legend: *light grey* represents blocked area; *dark grey* represents staircases; a *cross* indicates an infeasible staircase.

Input: {S} = set of unplaced staircases, initially containing all staircases;
Input: positioning matrices for all decks;
Output: {S_u} = set of unplaceable staircases, initially empty;
Output: position matrices for all decks;
while *{S} is not empty* **do**
  1) Attempt to place the current first unplaced staircase i in the position matrix;
  if *Attempt is successful* **then**
    Remove staircase i from {S};
  else
    2) Attempt to split staircase i:
    for $z_{split} = z_{min}$ **to** z_{max} **do**
      if z_{split} *causes failure of placement of current staircase* **then**
        Split current staircase into a staircase that should run from z_{min} to $z_{split} - 1$ and one that should run from $z_{split} - 1$ to z_{max};
        Remove staircase i from {S};
        Add both new staircases to {S};
        Return to 1);
      else if $z_{split} = z_{max}$ **then**
        Remove staircase i from {S};
        Add staircase i to {S_u};
      end
    end
  end
end
for *{S_u}* **do**
  Drop warning
end

Algorithm 1: Pseudo code for the arrangement of staircases

To determine the preferred locations of staircases, the *SPM*s for all relevant decks need to be taken into account. For a given staircase x that needs to run from deck z_m to deck z_n, the final staircase positioning matrix for staircase x, $FSPM_x$, is generated by merging the staircase positioning matrices for decks z_m till z_n. This is done such that:

$$FSPM_x(i,j) = \begin{cases} 0, & \text{if } \prod_{z=m}^{n} SPM_z(i,j) = 0 \\ \sum_{z=m}^{n} SPM_z(i,j), & \text{otherwise} \end{cases} \tag{2.2}$$

An example of the combination of two staircase matrices into one *FSPM* is provided in Figure 2.8.

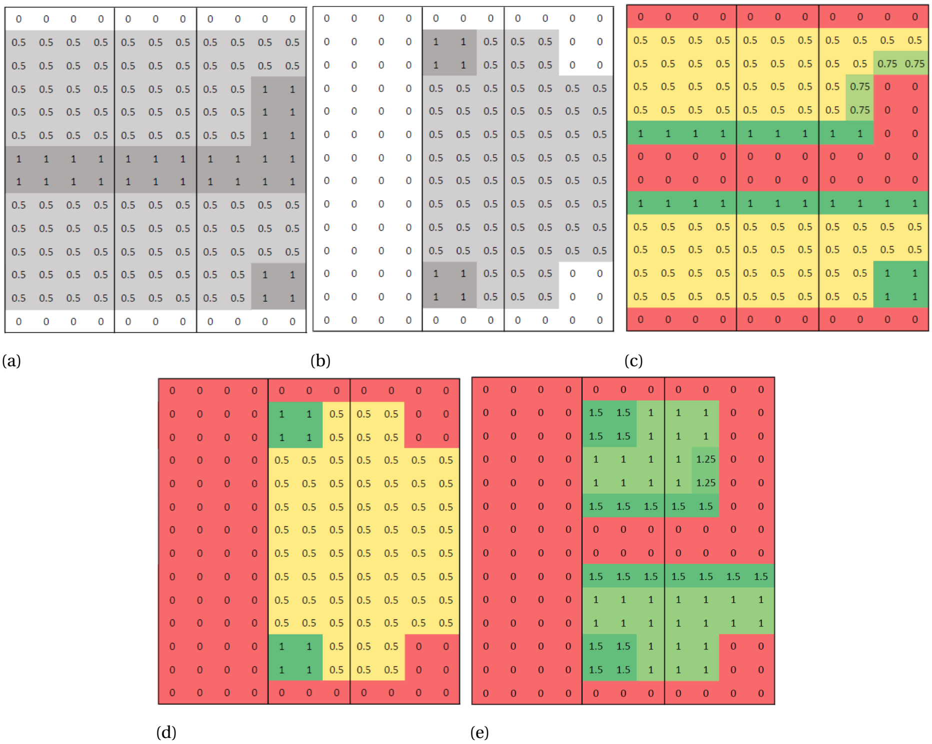


Figure 2.8: Creating a staircase positioning matrix for a two-deck, three-compartment test case. (a) Deck 1 with an L-shape passageway and a small lobby. (b) Deck 2 with an irregular shape and two lobbies. (c) SPM_1 for deck 1. (d) SPM_2 for deck 2. (e) Aggregated FSPM for the two decks, where higher values represent more preferred positions for staircases.

SELECTING STAIRCASE POSITIONS

The final step is to select a position from the generated FSPM. Obviously, the more preferred locations should have a higher probability of being chosen. At the same time, the code should be able to select less preferred locations if this appears to be necessary to, for instance, arrange spaces more efficiently. Furthermore, predefined preferred staircase positions are taken into account, e.g. a specific staircase might therefore be placed towards the starboard side of the ship, while the most preferred positions in the FSPM are located around the CL of the ship. The selection steps are:

1. Identify all available grid positions and their preference value.
2. Sort the available grid positions according to preference, which can be either the naval architect's preference or the numerical values in the FSPM.
3. Use a probability density function and a variable x to select the location of the staircase.

Referring to the example FSPM, Figure 2.8e results in 56 feasible positions of which twenty have an equal highest preference, i.e. preference value of 1.5, two positions follow closely with a preference value of 1.25, and 34 positions have a preference value of 1. The example shows that already for very coarse grids many positions with equal preference can exist. This is even more true for higher fidelity positioning grids as used fir realistic ship designs. Therefore a selection function is required that 1) provides roughly equal probabilities for early, i.e. more preferred, positions, and 2) low probabilities for late entries, i.e. less to non-preferred, positions. In this research, the probability density functions are based on Equation 2.3, since this function, depending on the n-value, possesses the required characteristics.

$$f(x) = b(1 - x^n) \tag{2.3}$$

Where $b = \frac{n+1}{n+2}$, which can be derived by integrating $f(x)$, and recognising that the cumulative probability of $f(x)$ equals 1.

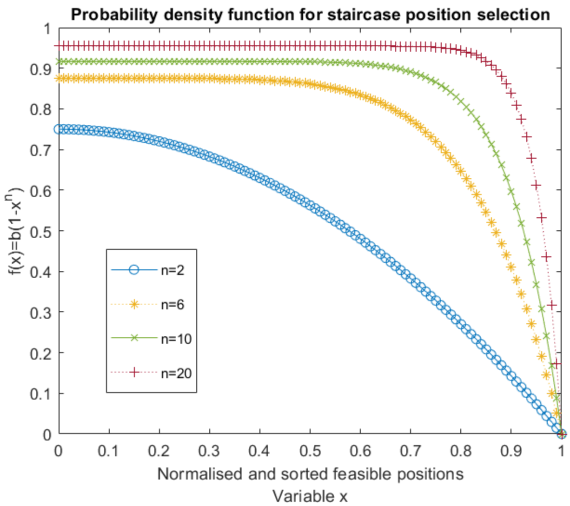


Figure 2.9: Four probability density functions for staircase selection for different n-values based on the function $f(x) = b(1 - x^n)$.

Four variations of such probability density functions are visualised in Figure 2.9. In WARGEAR, $n = 10$ is used as it proved to sufficiently promote the use of preferred positions. For instance, $n = 2$ is less suited as it does not provide equal probabilities for earlier positions in the list. The x-axis represents the sorted set of feasible grid positions and is related to a variable x. x can be randomly generated or obtained from an optimisation algorithm. The latter is used in this research to allow for steering of the layout generation process and repeatability.

2.4.5. ALLOCATION OF SPACES TO COMPARTMENTS

The meaningful allocation of spaces to compartments has been recognised as one of the key problems in early-stage ship design (DeNucci, 2012; Gillespie et al., 2013). The location of spaces and systems in a ship design impacts various performances, e.g. logistic performance (Droste et al., 2018; le Poole, 2018), and impacts other systems, e.g. the routing of interconnections between distributed systems (Duchateau et al., 2018).

One of the inputs of WARGEAR is an allocation of spaces to FBBs by the naval architects. Since FBBs in the functional arrangement can span multiple compartments and spaces are arranged per compartment, an allocation of spaces to compartments is required (Le Poole et al., 2019). Previously, only the allowed FBBs were considered for allocation, without considering the required area of spaces and the available area in compartments. This led to infeasible allocations, i.e. too many spaces could be allocated to compartments (Le Poole et al., 2019).

Since the functional arrangement is assumed to already satisfy the major relationships, e.g. accommodation should not be placed adjacent to main machinery spaces, and naval architects already have an allocation of spaces to compartments in mind, as will be discussed in Section 3.2, the aim is to develop a method that considers this envisioned allocation, as well as the available area in each relevant compartment and the required area for the spaces, such that, from an area perspective, the best possible allocation of spaces to compartments is obtained. To improve the probability that allocation of spaces is possible, the order in which spaces are allocated and compartments are used needs to be considered carefully, as elaborated below. Actual arrangement needs to be done to check whether spaces actually fit in the possibly irregular positioning matrix (see Section 2.4.6). The available area for spaces in compartment j is defined by Equation 2.4.

$$A_{available,j} = A_{compartment,j} - A_{blocked,j} - A_{staircase,j} - A_{passageways,j} - A_{allocated\,spaces,j} \tag{2.4}$$

In which $A_{compartment,j}$ is the total usable area between two bulkheads within the ship's hull or superstructure. The usable area is defined as the minimum of the floor and ceiling area in a compartment, resulting from flare or tumblehome hull or superstructure shapes. $A_{blocked,j}$ is the area of functional building blocks in compartment j that cannot be used for space arrangement (e.g. exhaust casings and HVAC rooms). The area used by staircases and passageways in compartment j is given by $A_{staircase,j}$ and $A_{passageways,j}$ respectively. $A_{allocated\,spaces,j}$ is the total required area, RA, of spaces already allocated to compartment j. Compartments with more available area are more likely to be used than compartments with less available area, since the probability that spaces fit is higher for the former.

The success of the allocation of spaces is also dependent on the size of spaces and the order in which spaces are allocated. Indeed, the probability that a large space fits in a compartment that is already partly used by other spaces is lower than the probability that a small space fits in the same compartment.

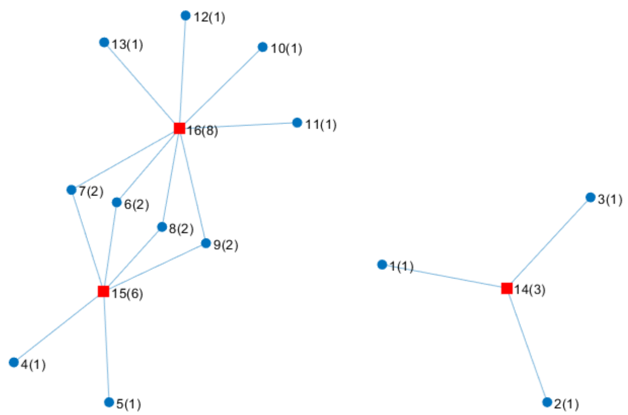


Figure 2.10: Network of the relation between spaces and compartments. Spaces are represented by round nodes and compartments by square nodes. Numbers indicate node IDs and (degree).

In WARGEAR, the list of allowed FBBs per space is translated into a network that represents the relations (edges) between spaces (nodes) and the compartments (nodes) that comprise the allowed FBBs (Figure 2.10). For all nodes, the degree is calculated and given in parentheses. The degree of a node is defined as the number of edges that connect to that node. For example, the degree of space 1 is 1, as it is only connected to compartment 14, whereas the degree of compartment 14 is 3, as it is connected to spaces 1 to 3. Subsequently, the degree of the nodes representing compartments and spaces is used to determine 1) which spaces should be allocated first and 2) which compartments should be used first. Although a more thorough study has been performed, the example in Figure 2.10 is used to explain the two rules used by WARGEAR:

1. Consider the case that the available area in compartment 16 is insufficient for spaces 6 to 13, but the area of compartments 15 and 16 is sufficient for spaces 4 to 13. If spaces 6 to 9 are allocated to compartment 16 before spaces 10 to 13, then spaces 10 to 13 cannot be allocated to compartment 16. Therefore, comparing the degrees of the spaces 10 to 13 with the degrees of the spaces 6 to 9, one can find that *allocating spaces with a lower degree prior to spaces with a higher degree is desirable*.

2. Using the rule above, spaces 1 to 5 and 10 to 13 can be allocated. Subsequently, spaces 6 to 9 need to be allocated. Although the example is not very complex, it can be argued that *it is desirable to use compartments with a lower degree prior to compartments with a higher degree*, since the probability that compartments contain spaces that still need to be allocated is less for compartments with a smaller

degree. Therefore, the probability that spaces cannot be allocated is also smaller. In the example, spaces 6 to 9 will be allocated to compartment 15 first, and the spaces that do not fit will be allocated to compartment 16.

Summarising, the following four statements are considered when allocating spaces to compartments, to maximise the probability that spaces are successfully allocated from an area point of view:

1. Large spaces are to be allocated prior to smaller spaces.
2. Compartments with more unused area are to be used prior to compartments with less unused area.
3. Spaces that are allowed in only a few compartments are to be allocated prior to those that are allowed in more compartments.
4. Compartments that are connected to fewer spaces are to be used prior to those connected to more spaces.

A roulette wheel selection method is used to select between available compartments for space i. In general, roulette wheel selection assumes that the probability of selection is proportional to the fitness of an individual. If N individuals are considered, each with a fitness $w_i > 0 (i = 1, 2, ..., N)$, then the selection probability of individual i is given by Equation 2.5 (Lipowski & Lipowska, 2012).

$$p_i = \frac{w_i}{\sum_{i=1}^{N} w_i} \quad (i = 1, 2, ..., N) \tag{2.5}$$

Subsequently, the roulette wheel is constructed with sectors whose sizes are proportional to $w_i\,(i = 1, 2, ..., N)$. Selection of an individual i is done by randomly selecting a point x at the roulette wheel and identifying the corresponding sector (Lipowski & Lipowska, 2012). In this dissertation, the fitness is defined as the probability $P_{allocation,ij}$ that space i is allocated to compartment j. $P_{allocation,ij}$ is given by Equation 2.6, provided that the allocation i to j, $Alloc_{ij}$ is allowed. The first fraction in this equation relates the statements 1 and 2 listed above. The second fraction relates to statements 3 and 4. The cumulative roulette wheel selection probability $P_{sel,ij}$ that compartment j is selected for space i is given by Equation 2.7. Subsequently the cumulative selection probability vector for the allocation of space i $P_{cum,sel}$ is given by Equation 2.8.

$$P_{allocation,ij} = \begin{cases} \frac{A_{available,j}}{RA_i} \cdot \frac{Degree_{space,i}}{Degree_{comp,j}} & \text{if } A_{available,j} \geq RA_i \text{ and } Alloc_{ij} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{2.6}$$

$$P_{sel,ij} = \frac{P_{allocation,ij}}{\sum_{j=1}^{N_{comp}} P_{allocation,ij}} \quad (j = 1, 2, ..., N_{comp}) \tag{2.7}$$

In which N_{comp} is the number of compartments.

$$P_{cum,sel}(j) = \sum_{n=1}^{j} P_{sel,ij} \quad (j = 1, 2, ..., N_{comp}) \tag{2.8}$$

The selection of a compartment on the roulette wheel would be usually done by generating a random number x. However, to assure traceability and to allow for regeneration of each layout, in WARGEAR variables provided by the optimisation algorithm are used, see Section 2.4.8. After each allocated space, $A_{available,j}$ is updated. The order in which spaces are allocated is determined via multi-level sorting. Spaces are sorted by ascending degree first (statement 3.) and then by descending RA (statement 1.).

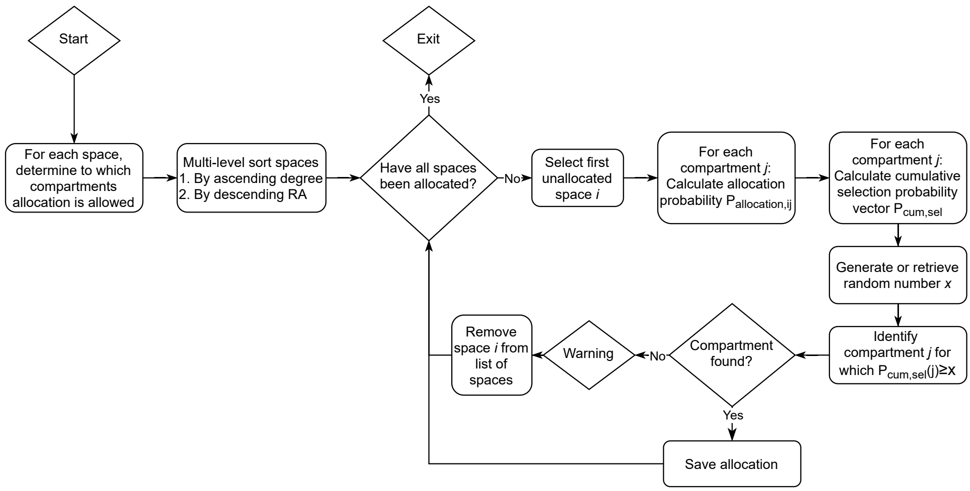


Figure 2.11: Flow chart of the allocation of spaces to compartments.

Table 2.4: Example of ten spaces and their allocation to FBBs.

			Compartments						
Space	**RA**	**Allocated to FBB**	**1**	**2**	**3**	**4**	**5**	**Degree space**	**Order index**
1	15	A, B	1	1	0	0	0	2	1
2	15	A, B	1	1	0	0	0	2	2
3	15	A, B	1	1	0	0	0	2	3
4	15	C, D	0	0	1	1	1	3	7
5	15	C, D	0	0	1	1	1	3	8
6	50	C, D	0	0	1	1	1	3	4
7	30	A, B, C, D	1	1	1	1	1	5	10
8	20	B, C, D	0	1	1	1	1	4	9
9	20	C, D	0	0	1	1	1	3	5
10	20	C, D	0	0	1	1	1	3	6
		Degree compartment	4	5	7	7	7		

The method for allocating spaces to compartments is summarised in Figure 2.11. To illustrate the allocation method a small example will be elaborated on. Assume a functional arrangement with four FBBs (named A-D) spanning respectively 1, 1, 2, and 1 compartments (named 1-5). The area of the FBBs is respectively 40, 40, 100, and 50 m^2. The area of the compartments is respectively 40, 40, 50, 50, and 50 m^2. Ten spaces are allocated to the four FBBs, see Table 2.4. The corresponding allocation of spaces to compartments also provides the degrees of the spaces and compartments. After determining the degree of the spaces, the order in which spaces will be allocated can be established, by sorting by degree space first, followed by sorting by RA. Subsequently, each space is allocated. Following the steps in Figure 2.11, for each space $P_{allocation,ij}$, $P_{sel,ij}$, and $P_{cum,sel}$ are calculated. Then a random number is drawn, and the corresponding compartment is determined. Before allocating the next space, the available area in the selected compartment is updated, see Table 2.5. For space 7, the selected compartment is 0, which means this space could not be allocated because there is no compartment available with sufficient area available to fit space 20.

In Appendix A.1 the allocation method is tested. This test supports the conclusion that the proposed allocation method performs equally or better than variations to the method. Hence, the method presented above remains to be used in WARGEAR. Section 3.3 extends the allocation method to consider interrelationships between systems.

Table 2.5: Allocation of ten spaces (see Table 2.4) to five compartments.
[1]: Spaces are sorted based on degree and RA. [2]: $A_{available,j} - RA_i = A_{available,2} - RA_1 = 40 - 15 = 25$. [3]: No compartment with sufficient area is available.

Space[1]	RA	Randomly selected compartment	$A_{available,j}$ 1	2	3	4	5
-			40	40	50	50	50
1	15	2	40	25[2]	50	50	50
2	15	2	40	10	50	50	50
3	15	1	25	10	50	50	50
6	50	4	25	10	50	0	50
9	20	3	25	10	30	0	50
10	20	5	25	10	30	0	30
4	15	3	25	10	15	0	30
5	15	3	25	10	0	0	30
8	20	5	25	10	0	0	10
7	**30**	**0**[3]	**25**	**10**	**0**	**0**	**10**

2.4.6. SPACE ARRANGEMENT

In this section, a novel space arrangement method based on cross-correlation is proposed, tested, and compared to the seed and growth algorithm used in WARGEAR previously (Le Poole et al., 2019). Indeed, it was found that the seed and growth algorithm is time-consuming due to its iterative nature. Also, the optimisation algorithm lacks control over space arrangement via initial seed locations, because spaces grow 'randomly' around the seed. As shown in Figure 2.12, different seed locations might result in the same arrangement of spaces. Therefore a feasible, and even optimal, solution might be easily found for simple arrangement problems, but more complex arrangement problems can be more challenging for the seed and growth method to solve. Indeed, large changes in input parameters might only result in small changes in layouts. Therefore, the optimisation algorithm needs to be robust enough to explore the whole design space, since early convergence might lead to stopping in local optima.

Instead of applying a method of iterative growth of spaces to identify how spaces could best be arranged in a given positioning area, a new method is proposed that can quickly assess which positions in a position matrix are suitable for a given space. Growth-

2

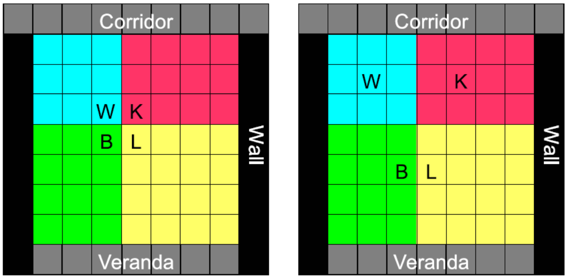


Figure 2.12: Different initial positions (indicated by [W,K,B,L]) can lead to the same layout using seed and growth space arrangement algorithms (Inoue & Takagi, 2008)

based approaches start with a single point in the positioning matrix and grow spaces till these meet their required area. In contrast, the new proposed space arrangement method uses cross-correlation, a mathematical operation that expresses how one function is correlated to another function (Bourke, 1996). This proposed method attempts to directly place the most preferred feasible space shape in the available area and thus requires fewer iterations. Therefore, the main reason to change the space arrangement method is the speed advantage of cross-correlation over seed and growth-based arrangement methods.

To the best of the researcher's knowledge, this method is new for layout design, although a variant can be found in the Packing methodology (Van Oers, 2011b) and ISA (Daniels & Singer, 2012). Inspiration for this new arrangement method was found in the fields of signal processing (Burrus & Parks, 1985; Najafi et al., 2020), neural network image recognition (Li et al., 2019; Lo et al., 1995), and probability theory (Pruinelli et al., 2019).

Since the space arrangement problem for WARGEAR focuses on a 2D deck plan, the mathematical expression for 2D cross-correlation has been adopted. The expression of the 2D cross-correlation between m_Axn_A-matrix A and m_Bxn_B-matrix B has been given in Equation 2.9. Matrix C is a m_Cxn_C-matrix that contains to what extent matrix B can be placed at position $(j-1,k-1)$ in A, with $m_C = m_A - m_B + 1$ and $n_C = n_A - n_B + 1$. Note that other dimensions of C are also possible, for instance, m_Axn_A. However, here m_Cxn_C is used as defined above as it provides only feasible positions within A. If $C(j+1,k+1) = \sum_{i=1}^{n_{\text{elements in B}}} B_i$, then matrix B can be placed at position $(j+1 : j+m_B, k+1 : k+n_B)$ in A.

$$C(j+1,k+1) = \sum_m \sum_n A(m,n) \cdot \bar{B}(m-j,n-k) \tag{2.9}$$

Where:

$$j = 0 : m_A - m_B \tag{2.10}$$

$$k = 0 : n_A - n_B \tag{2.11}$$

$$m = 1 : m_A \tag{2.12}$$

$$n = 1 : n_A \tag{2.13}$$

$$1 \leq m - j \leq m_B \tag{2.14}$$

$$1 \leq n - k \leq n_B \tag{2.15}$$

$$j, k, m, n \in \mathbb{Z}^{\geq} \tag{2.16}$$

Further, $\bar{B}$ denotes the complex conjugate of B, although for the arrangement problem $\bar{B} = B$, because of the absence of complex numbers in B.

2

Matrix A represents the positioning matrix and contains ones when a position is available and zeros otherwise. Matrix B represents a space and contains just ones since WARGEAR arranges rectangle spaces only (Le Poole et al., 2019). However, arranging irregular shapes is also possible using this method. Let B_i represent matrix B for space i. The area of B_i is equal to the size of space i, while its dimensions are based on the allowed aspect ratio of space i. Since a space might not fit in A, it is necessary to vary the aspect ratio of B_i. Since the objective is to satisfy the area and aspect ratio requirements for all spaces, the dimensions of B_i need to be varied such that both requirements are met to the maximum extent possible. To achieve this, first the table $B_{range,dimensions}$ is created for each space i. It contains all lengths and widths that satisfy Equations 2.17 to 2.19 and therefore might be considered for space i.

$$Area_i = Length_i \cdot Width_i \leq RA \cdot \left(1 + \frac{AOP}{100}\right) \tag{2.17}$$

Where AOP is a constant Area Overshoot Percentage, which allows spaces to be AOP % larger than RA. $AOP = 20\%$ is assumed.

$$min(AR) \leq AspectRatio_i \leq max(AR) \tag{2.18}$$

$$\frac{1}{max(AR)} \leq AspectRatio_i \leq \frac{1}{min(AR)} \tag{2.19}$$

Where $AspectRatio_i$ is given by Equation 2.20 and AR is a two-element vector containing the range of allowed aspect ratios for space i. Typically $AR = [0.5\ 1]$.

$$AspectRatio_i = \frac{Length_i}{Width_i} \tag{2.20}$$

Subsequently table $B_{range,dimensions}$ is sorted by the following sorting method:

1. The first subset contains the rows in which $Area = RA$.
2. The second subset contains the rows in which $Area > RA$. These rows are sorted based on *increasing* size.
3. The third subset contains the rows in which $Area < RA$. These rows are sorted based on *decreasing* size.

Each subset is sorted by increasing aspect ratio.

Following the generation of table $B_{range,dimensions}$, the arrangement method outlined in Algorithm 2 is used to arrange space i. The sorting method used ensures that 1) the aspect ratio requirement is always met, and 2) the area requirement is met to the maximum extent possible while limiting exceeding the required area.

2

```
index = 0;
while space i is not arranged do
    index = index+1;
    select L_index and B_index from B_range,dimensions;
    create L_index x B_index-matrix B_i;
    cross-correlate matrices A and B_i to matrix C, using Equation 2.9;
    if at least one position is available then
        select an available position;
        arrange space i at the selected position;
        update matrix A;
    else
        return
    end
end
```

Algorithm 2: Pseudo code for the arrangement of space i

To clarify how the method outlined above leads to the identification of feasible positions for a given positioning matrix A and a space B, consider the following matrices A and B, shown respectively in Matrices 2.21 and 2.22. Then the cross-correlated matrix C of the matrices A and B is given by Matrix 2.23. The only positions in C that satisfy $C(j+1,k+1) = \sum_{i=1}^{n_{\text{elements in B}}} B_i$ are $(1,1)$ and $(3,3)$. Therefore the only positions where B can be placed in A are $A(1:2,1:2)$ and $A(3:4,3:4)$. The accuracy of this answer is clear from comparing Matrices 2.21 and 2.22.

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \tag{2.21}$$

$$B = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \tag{2.22}$$

$$C = \begin{bmatrix} 4 & 2 & 0 \\ 2 & 2 & 2 \\ 0 & 2 & 4 \end{bmatrix} \tag{2.23}$$

Having developed a way to quickly identify the most feasible positions for spaces, three issues regarding space arrangement still remain to be addressed.

1. *In which order should multiple spaces be arranged?* There are three variations possible: 1) large spaces are arranged prior to small spaces, 2) small spaces are arranged prior to large spaces, and 3) spaces are, from a space size perspective, arranged in a mixed order. The order of the latter is to be selected via variables provided by the optimisation algorithm.

2. *How to choose between available positions for space i?* Indeed, the simple example shown above already gives two possible positions for space B. In a larger positioning matrix, such as one representing a compartment or a deck in a ship, the number of feasible positions for a given space will likely be large. Four options for choosing from feasible positions are proposed in this dissertation, which will be elaborated on.

   (a) *Choose the first available position (A)*: This causes spaces to be asymmetrically arranged in a compartment, which is not preferred from a ship's stability point of view, Figure 2.13a. This also causes spaces to be arranged in the aft SB of a compartment, resulting in limited to no void area between spaces. Such void area is best used to arrange other spaces or to place passageways to ensure connectivity. In the case such void area is absent between spaces, substantial area needs to be taken from spaces or spaces need to be rearranged. Note that void area can be sensible in ship design, for instance, to increase the ship's length for seakeeping (Keuning & Pinkster, 1995) or for future growth (Ferreiro & Stonehouse, 1991), although such void area usually is a product of design margins and not a surprising result from the arrangement process.

   (b) To improve the symmetry of the arrangement *choose positions as close to the ship's CL as possible (B)*. This causes spaces to be arranged aft in a compartment predominantly, Figure 2.13b. Again, this option does not allow passageways to 'emerge' from the arrangement of spaces.

   (c) *Choose positions as far from the centre of the compartment as possible (C)*. This method first arranges spaces in the corners of the compartment, and subsequently closer to the centre of the compartment, Figure 2.13c. Generally, this proves to result in a more useful distribution of void area to be used for connectivity.

   (d) *An optimisation algorithm selects a position (D)*. The previous three options don't require variables. Although that might lead to a fast answer as no optimisation is required, optimalilty is not guaranteed. To allow WARGEAR to find more optimal designs, the fourth option is to use variables provided by an optimisation algorithm to select from available positions, Figure 2.13d. However, this might also require space overlap detection and removal.

3. *How to orient spaces?* The orientation in which a space is arranged can significantly impact the success of the arrangement of other spaces. In some cases, it would be beneficial to change the orientation of spaces. For instance, it might be useful to arrange space i in transverse direction although in table $B_{range,dimensions}$ the first row contains values for a longitudinally arranged space as this could allow other spaces to be more effectively arranged. To allow WARGEAR to change the orientation of spaces, an additional variable might be used to select whether a space is arranged in longitudinal or transverse direction.

In Appendix A.2 the cross-correlation-based space arrangement method is tested. Also, the various options to address the three issues described above, are evaluated. The

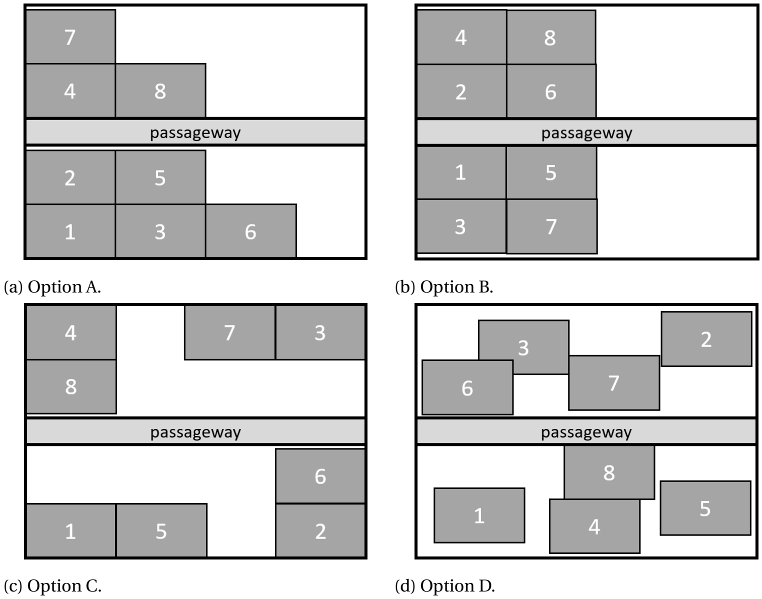


Figure 2.13: Visualisation of space position selection options in an example of a compartment with a central passageway

test showed that the cross-correlation method is, on average, 20 times faster than WARGEAR's original seed and growth method. Also, the quality (in terms of the difference between the required and achieved areas of spaces) of the layouts generated by the cross-correlation method is better than that generated by the seed and growth algorithm. In addition, an optimisation algorithm-selected arrangement order of spaces, combined with optimisation algorithm-selected orientation of spaces was found to perform well. From a layout quality perspective, the position selection options A and C outperform options B and D. Selecting an option by the optimisation algorithm (D) was found to perform similarly to options A and C. When opting for option D, the optimisation algorithm typically chooses between options A and C as well, underlining the performance of these position selection options.

2.4.7. Ensuring connectivity

Although the main passageways are taken from the functional arrangement, or generated by WARGEAR (see Section 2.4.3), and staircases are placed prior to space arrangement (see Section 2.4.4), an additional step is required to ensure all spaces are connected and can be properly accessed. This additional step checks the connectivity between passageways, staircases, and spaces in a compartment, and corrects connectivity if it is found to be insufficient. The following connectivity checks are implemented by default, but can be turned off by the naval architect:

1. *Passageway-passageway.* Multiple passageways, e.g. two parallel passageways, can be connected if necessary. For instance, to generate escape routes between

port side and starboard side passageways.

2. *Passageway-staircase.* Staircases need to be accessible from passageways, if passageways and/or staircases exist.

3. *Staircase-staircase.* In the case of multiple staircases in one compartment, these staircases are to be connected, again to create escape routes.

4. *Space-passageway.* Basic feasibility requires spaces to be accessible from the main passageways.

5. *Space-staircase.* If main passageways are absent, e.g. below the damage control deck, spaces are to be connected to staircases, again because of basic feasibility requirements.

Connectivity is ensured *after* spaces have been arranged and is based on an existing carving method (Marson & Musse, 2010), discussed in Section 2.3.2. Depending on the selection of connectivity checks demanded by the naval architect, the algorithm arranges additional passageways to connect passageways, staircases, and spaces. The width of these passageways is controlled via a single parameter set by the naval architect. Using a network representation of the arrangement of passageways, staircases, and spaces in each compartment, the algorithm uses the walls of spaces as potential locations of additional passageways. This reflects the author's observations made in a study into GAPs generated by naval architects, namely that passageways tend to share walls with spaces. The algorithm uses the following steps to ensure connectivity in a compartment:

1. A network representation of passageways, staircases, and spaces, in the form of an adjacency matrix A is created. $A(i,j)$ is 1 if node i and j are connected, and 0 otherwise. Edges and nodes located at bulkheads are removed, because additional passageways located at bulkheads lead to a large reduction in space size. Indeed, these passageways can only use the area on one side of the bulkhead.

2. A matrix D containing the distances between connected nodes is generated. $D(i,j)$ is defined as follows

$$D(i,j) = \begin{cases} 0.5d_{ij}, & \text{if edge } ij \text{ is part of a passageway or staircase} \\ d_{ij}, & \text{if edge } ij \text{ is not adjacent to a space} \\ \frac{d_{ij}}{f(\text{obj}_{\text{connected spaces}})}, & \text{if edge } ij \text{ is shared by one or more spaces} \\ 0, & \text{if edge } ij \text{ is non-existing.} \end{cases} \tag{2.24}$$

Where d_{ij} is the Manhattan distance between nodes i and j, defined as:

$$d_{ij} = |x_i - x_j| + |y_i - y_j| \tag{2.25}$$

And $\text{obj}_{\text{connected spaces}}$ is defined by Equation 2.26 in which connected spaces are represented by subscripts $1:n$.

$$\text{obj}_{\text{connected spaces}} = \left[max\left(100, \frac{AA_1}{RA_1} \cdot 100\right), ..., max\left(100, \frac{AA_n}{RA_n} \cdot 100\right) \right] \tag{2.26}$$

2

Because spaces do not always meet their required area, and it is preferred that as little area as possible is lost due to carving additional passageways, an exponential function $f(\overline{\text{obj}_{\text{connected spaces}}})$ is used, and is defined by Equation 2.27. The growth factor gf used in WARGEAR is 0.06.

$$f(\overline{\text{obj}_{\text{connected spaces}}}) = (1 + gf)^{\overline{\text{obj}_{\text{connected spaces}}} - 100} \qquad (2.27)$$

3. A matrix E containing information on which side of a wall is preferred to place an additional passageway if this wall is selected. The preferred side is determined based on the objective value (see Section 2.4.8) of the spaces sharing walls. Matrix E is not utilised in the routing, but only to make the final decision where to carve, i.e. from which spaces to take area if necessary. Spaces that meet their required area are more likely to be carved from than spaces that are too small.

4. Each connectivity check is performed by routing between a pair of systems, e.g. passageway and space, using Dijkstra's shortest path algorithm (Dijkstra, 1959).

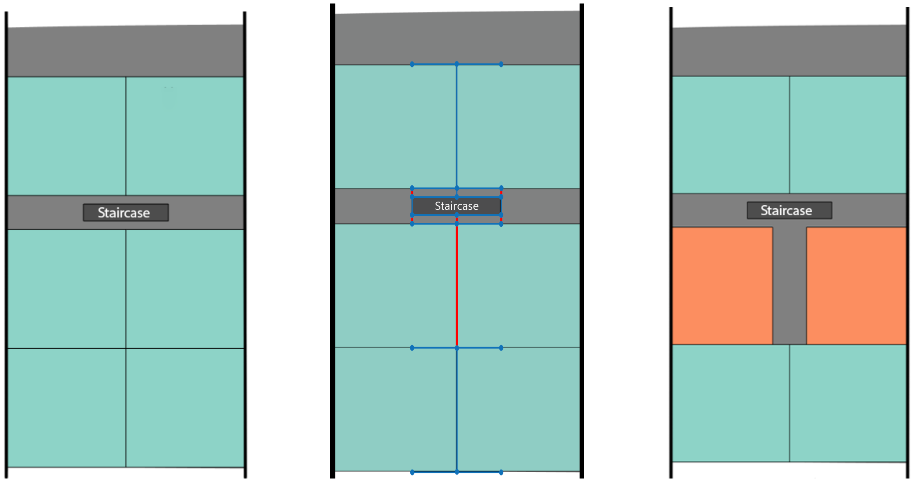
5. At the location of each found path an additional passageway is carved.

A visual explanation of this approach is provided in Figure 2.14. Figure 2.14c shows that, after carving an additional passageway, two spaces do not meet their required area anymore. An overall readjustment of the allocation of spaces, position of staircases, and arrangement of spaces might lead to an improved arrangement. The performance of the carving method is investigated in Appendix A.3 using the case study with the integrated methodology presented in Section 3.1.

2.4.8. OPTIMISATION PROBLEM

The previous sections described the major mathematical methods in WARGEAR. In Figure 2.15 a flow chart of the *arrangement* phase of the tool is presented. This section will elaborate on how a nested optimisation approach is used in WARGEAR to steer the arrangement of detailed arrangements. A nested optimisation (or bi-level) approach is an operation research technique to solve hierarchical decision-making problems (Oduguwa & Roy, 2002), simplifying a large optimisation problem into smaller optimisation problems. In WARGEAR, the outer optimisation loop steers the placement of passageways, staircases, and the allocation of spaces to compartments. The inner optimisation loop controls the arrangement of spaces in individual compartments. The optimisation algorithms provide variables for the various elements of WARGEAR. These elements (e.g. the staircase arrangement algorithm) use the provided variables to choose from feasible options (e.g. to choose from feasible staircase positions).

To assess each layout, the achieved area (AA) of each space is compared with its required area (RA). A total score for each generated layout is given by Equation 2.28. In this equation, spaces are not allowed to compensate for spaces that don't meet their RA. For example, if two spaces with $RA = 10$ are arranged with $AA_1 = 5$ and $AA_2 = 15$, then space 2 does not compensate for the insufficient area of space 1. The score for this design would be $F = 5$, instead of $F = 0$ when the area is compensated (Droste &



(a) Compartment with six spaces and a staircase. All spaces meet their required area (indicated by green).

(b) Network representation overlaying the space arrangement of this compartment. Red lines indicate edges that need to be carved to ensure connectivity. Note that edges and nodes at bulkheads (left and right) are removed.

(c) The compartment layout after carving required additional passageways. Two spaces don't meet their required area now (indicated by orange).

Figure 2.14: Visual explanation of the carving process for a compartment with six spaces and a staircase.

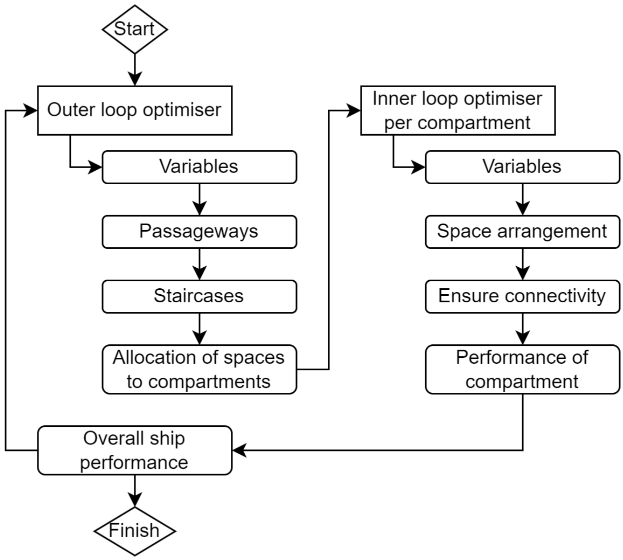


Figure 2.15: Organisation of WARGEAR's mathematical methods inside a nested optimisation approach.

le Poole, 2020). The objective function F for a complete layout is given by Equation 2.28 and is minimised. This equation is used to steer the outer optimisation loop.

$$\text{Minimise } F = \sum_{i=1}^{n_{space}} max(0, RA_i - AA_i) \tag{2.28}$$

2

All constraints, for instance, that spaces should be within allowed aspect ratios and space positions should be within the ship's hull and superstructure, are controlled by the WARGEAR code. Variables $x_k \in [0,1]$, generated by the optimisation algorithm, are used to select from feasible options for decision k. For instance, if a space can be allocated to four compartments, a variable is used to select between one to four only.

Variables x_k are used for:

1. Passageways
2. Staircases
3. Allocation of spaces to compartments

The optimisation problem for the inner optimisation loop is defined as follows. The objective function F_j for compartment j is given by Equation 2.29.

$$\text{Minimise } F_j = \sum_{i=1}^{n_{space,j}} max(0, RA_i - AA_i) \tag{2.29}$$

Variables x_j are used for space arrangement only, as the algorithm used to ensure connectivity does not use variables.

The best arrangements of individual compartments generated in the inner optimisation loop are combined into a single layout for each attempt in the outer optimisation loop. For instance, referring to the case study presented in Section 3.1 and Table 3.1, a total of $(NumIt + 1) \cdot PopSize = (20 + 1) \cdot 10 = 210$ arrangement attempts will be made by the outer optimisation loop. For each attempt, the inner optimisation loop will arrange each compartment in $(5 + 1) \cdot 3 = 18$ attempts. Only the best arrangement of each of these 18 attempts will be used to construct a full layout. Therefore, only 210 layouts will be presented to the naval architect. In contrast, suppose six compartments need to be arranged. Then, evaluating each possible combination would result in $210 \cdot 18^{n_{\text{compartments}}} = 210 \cdot 18^6 = 7.14 \cdot 10^9$ layouts, but yield no more insights into layout sizing and integration issues.

The optimisation algorithm used for both the inner and outer optimisation is a Particle Swarm Optimisation (PSO). A PSO has been chosen because it was readily available and is easy to use. Also, it was found that the PSO implementation used in WARGEAR can find sufficiently feasible solutions in a limited time. Hence, no effort was spent on finding a more sophisticated optimisation algorithm. Although better-performing algorithms might find slightly better solutions faster, it is not expected that using a more sophisticated optimisation algorithm will lead to significantly better or more insights into space sizing and integration issues, which is the principle goal of WARGEAR. Refer to Coello et al. (2004), Kennedy (2010), Poli et al. (2007), and Shi et al. (2001) for a more detailed description of PSOs.

2.4.9. POST-PROCESSING AND VISUALISATION

After generating a set of layouts, the results are presented to the naval architect for further analysis. First, a scatter plot of the scores of all generated layouts is shown. Scatter plots were found to be a simple and familiar way of visualising the relation between two (or more) characteristics of a set of designs (Duchateau, 2016). Second, the naval architect selects a design from the scatter plot for detailed analysis. This will be discussed in more detail in Section 3.1.

To allow for quick recognition of the context of the spaces arranged by WARGEAR, the functional arrangement is used as a basis for the visualisation (Section 3.2). All functional blocks from the functional arrangement that are not further detailed by WARGEAR are plotted using the same colours as the functional arrangement. This helps identify areas in a compartment that cannot be used for space arrangement, such as exhaust stacks.

To support direct insight into the quality of the arrangement, spaces arranged by WARGEAR are coloured in accordance with the objective value of that space (Le Poole et al., 2019) and Section 3.2. Green is used to indicate that a space meets its required area, while red shades indicate to which extent a space did not meet its required area.

Also, main space properties, such as space name, number, required area, and achieved area, are provided in the detailed layout plan. Especially the information on required and achieved area supports the naval architect in better understanding to which extent a space did not meet its required area (Section 3.2). It was considered to provide a measure based on $\frac{AA}{RA}$ instead of providing both RA and AA. Such fraction would provide quick insight but is limited in giving insight into absolute numbers. For instance, consider three spaces with a required area of 10, 20, and $40m^2$. Assume these spaces have an achieved area of respectively 8, 18, and $32m^2$. Then, a fraction-based measure would yield 0.8, 0.9, and 0.8 respectively. This result already tells that the second space has been best arranged, despite it lacks 10% of its required area. However, the result does not convey the serious problems with the third space, as it misses $8m^2$ in absolute numbers, which might be harder to solve. Therefore, both RA and AA are provided, as these give insight into both relative and absolute (possible) lack of required area. As a result, the naval architect will likely start to address the arrangement issue of the third space, using the detailed arrangement together with the detailed textual and numerical information provided by WARGEAR. Chapter 3 elaborates on detailed case studies, which will also show WARGEAR's visualisations.

2.5. CONCLUSION

The main question for this chapter was: *To what extent can automated layout generation methods support real-time design decisions during early-stage complex ship design?* By the end of this chapter, this question can be partially answered. First, from the nature of ship layout design, Section 2.1 established that such an automated layout generation method should be able to generate arrangement drawings of sufficient detail (i.e. space level) at sufficiently high speed (i.e. in the order of minutes). Also, it should be responsive to the designer and provide feasible and believable results. Second, a literature review of layout generation methods and tools revealed that no existing layout design method fulfils all requirements. Third, the mathematical working principles of a new fast ship layout generation method, called WARGEAR, was described in Section 2.4. The mathematical principles of WARGEAR should make it capable of generating space-level arrangements in a matter of minutes, meeting the method requirements. This fact will be demonstrated in Chapter 3.

Table 2.6: WARGEAR: compliance with method requirements

ID	Requirement	Compliance WARGEAR
LRQ1	The method should provide insight into potential sizing and integration issues and risks and ways to solve these issues and reduce the risks.	Partially fulfilled (Section 2.4) - Case studies need to confirm the intended capabilities.
LRQ2	Its speed should be in the order of minutes	Partially fulfilled (Section 2.4) - Case studies need to confirm the intended capabilities.
LRQ3	The starting point is a predefined functional arrangement	Fulfilled (Section 2.4.2)
LRQ4	The level of detail of generated layouts should be high	Fulfilled (Sections 2.4.6 and 2.4.7)
LRQ5	The main driver for layouts to be considered is area	Fulfilled (Section 2.4.6)
LRQ6	The number of diverse solutions should range from a few to hundreds	Fulfilled (Section 2.4.8)

Table 2.6 summarises WARGEAR's compliance with the method requirements given in Section 2.1.1. Most of the method's requirements are fulfilled by the modelling discussed above. However, a few issues remain to be addressed. First, a demonstration and verification of WARGEAR needs to be performed to check compliance with requirements LRQ1 and 2. Second, WARGEAR needs to be compared to human design efforts to see how designers can benefit from such an automated method. Finally, designers need to be supported to analyse WARGEAR's (possibly) extensive output in real-time. These issues will be addressed in Chapter 3.

3
AUTOMATED LAYOUT DESIGN CASE STUDIES

If it looks right, it probably is right, if it looks wrong, then check it out.

R. Baker (1995)

The previous chapter introduced the WARGEAR method in detail but concluded that a demonstration and verification of the complete method is still necessary. Therefore, this chapter will present three case studies demonstrating, as well as extending various aspects of the method.

1. Section 3.1 demonstrates how WARGEAR can be used to generate detailed layouts for a notional surface vessel[1].

2. Section 3.2 presents a design study performed using WARGEAR in parallel with human design work[2].

3. Section 3.3 provides an extension of WARGEAR's allocation algorithm, as well as an allocation method. This method can be used to obtain design insights from WARGEAR's potentially large output data set[3].

4. Section 3.4 provides an additional demonstration of WARGEAR, applied to a Landing Platform Dock (LPD) design case[4].

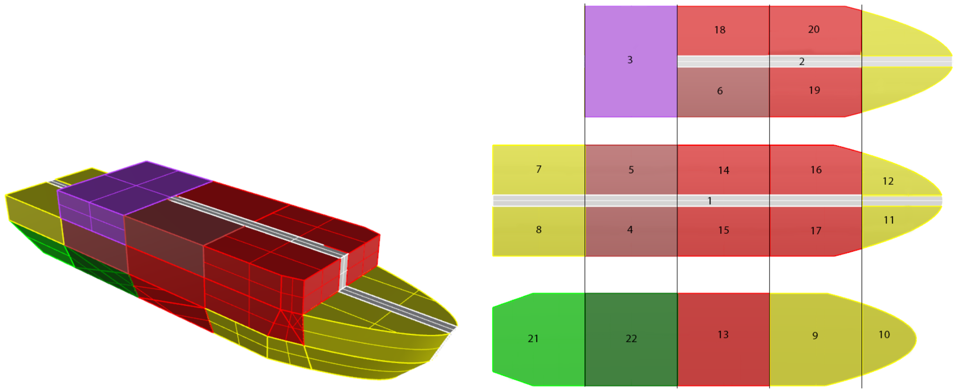
[1]This section is based on Le Poole et al. (2022d).
[2]This section is based on Le Poole et al. (2020).
[3]This section is based on Le Poole et al. (2022b).
[4]This section is based on Droste and le Poole (2020).

3.1. CASE STUDY 1 - GENERATING DETAILED LAYOUTS

In Chapter 2 the mathematical principles of WARGEAR have been discussed. In this section, WARGEAR is used to generate detailed layouts for a functional arrangement of a notional surface vessel. The aim of this case study is twofold:

1. To demonstrate and test the integrated mathematics of the WARGEAR method.
2. To demonstrate how the WARGEAR method can be used to generate and analyse detailed layouts and to derisk functional arrangements, i.e. demonstrate WARGEAR's usefulness.

The functional arrangement of the notional surface vessel is shown in Figure 3.1. Note that this vessel does not reflect a particular vessel (to demonstrate WARGEAR is generally applicable) or contain systems such as engines or weapon systems. The main reason for the latter is that WARGEAR would not be used to arrange these systems. As such, adding such systems would not add to this case study. In this case study the arrangement of cabins, galley, mess, and stores will be investigated. The available area in each compartment and functional building block and the full space list can be found in Appendix B.2. Note that based on the available area in the compartments ($450m^2$), the required area of spaces ($535m^2$), and the initial allocation of spaces to compartments, all spaces should fit. WARGEAR will be used to check whether all spaces indeed fit when staircases and additional local passageways are taken into account.



(a) 3D visualisation. (b) Deck plans.

Figure 3.1: Arrangement of the notional surface vessel used in Case Study 1. See Appendix B.2 for details.

3.1.1. CASE STUDY SETTINGS

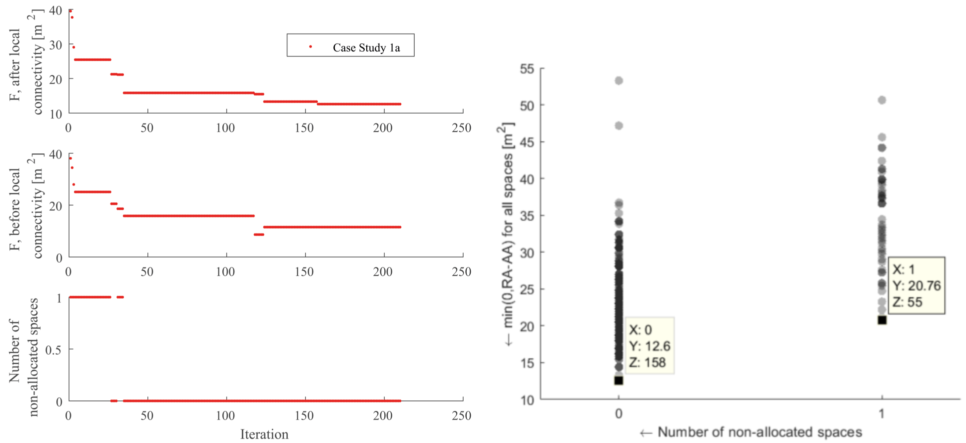
The case study consists of two parts. First, insights gained from layouts generated by WARGEAR in Case Study 1a will be analysed. Second, these insights will be used to update the input for Case Study 1b. In Appendix A.3, Case Study 1a is used in parallel with two additional cases to validate the combination of the space arrangement and passageway carving approaches in the context of the complete method. The settings for the optimisation algorithm are kept constant in each case, as these prove to lead to optimisation convergence and generate useful layouts, as further discussed in Section 3.1.2 and 3.1.3. Table 3.1 summarises the settings for the optimisation algorithms. Note that the inner optimisation loop (PSO2) might not run toward convergence. This was deliberately chosen to reduce overall calculation time and still yields sufficient results.

Table 3.1: PSO settings.

	PSO 1	PSO 2	Explanation
NumIt	20	5	Number of iterations
PopSize	10	3	Population size
w	0.5	1	Inertia weight
w_{damp}	0.9	0.4	Inertia Weight Damping Ratio
c1	0.5	2	Personal Learning Coefficient
c2	2.5	2	Global Learning Coefficient

3.1.2. CASE STUDY 1A: INITIAL RESULTS

The results of executing WARGEAR with above settings are summarised in Figure 3.2a and Table 3.2. The graph shows the convergence of the *F*-value (see Equation 2.28) across all iterations. Also, the number of non-allocated spaces, the average compartment area utilisation, and the objective value *F* before local connectivity is ensured, are shown for the best-performing layout.



(a) Convergence plot.

(b) Scatter plot, in which each dot represents a layout. The two selected layouts are discussed in this case study.

Figure 3.2: High-level results of Case Study 1a.

Next, two layouts generated in Case Study 1a will be compared to identify possible sizing and integration issues. Indeed, across all PSO generations, the allocation of one space appeared to be challenging, pointing to possible issues. This might be caused by the available area in compartments, the initial allocation of spaces to functional building blocks, and/or required space size. Furthermore, no layout has been generated that meets all spatial requirements for all spaces. Figure 3.2b shows a scatter plot with all

3

Table 3.2: Summary of results of Case Study 1a and 1b.

	A	B			
Case number	**Mimimum F obtained [m^2]**	**Minimum number of spaces not-allocated**	**#B**	**A|B [m2]**	**Run time [s]**
1a	12.60	0	163	12.60	737.05
1b	3.69	0	210	3.96	1065.41

layouts generated in Case Study 1a, their objective score (y-axis), and the number of non-allocated arranged spaces. The two layouts that will be evaluated are 1) the overall best-performing layout (ID 158, Figure 3.3a) and 2) the best-performing layout that misses one space (ID 55, Figure 3.3b), as indicated in the scatter plot.

The following observations can be made from these generated 2D deck plans:

1. The local passageway carving method ensured connectivity on the lower deck in both Figures 3.3a and 3.3b. On the other decks, connectivity was already in place after space arrangement, since all spaces were directly accessible from the main passageway. Although this demonstration is limited in size, it shows that the carving method works as intended.

2. The space that could not be allocated in Figure 3.3b is 'Store 2' on the middle deck. The starboard arrangement of a staircase in the second compartment from the aft results in insufficient available space to allocate this store.

3. If 'Store 2' is successfully allocated, it is still not possible to properly arrange the three stores in one compartment, as can be seen in Figure 3.3a. This insight could, for example, be used to question the spatial requirements for the stores.

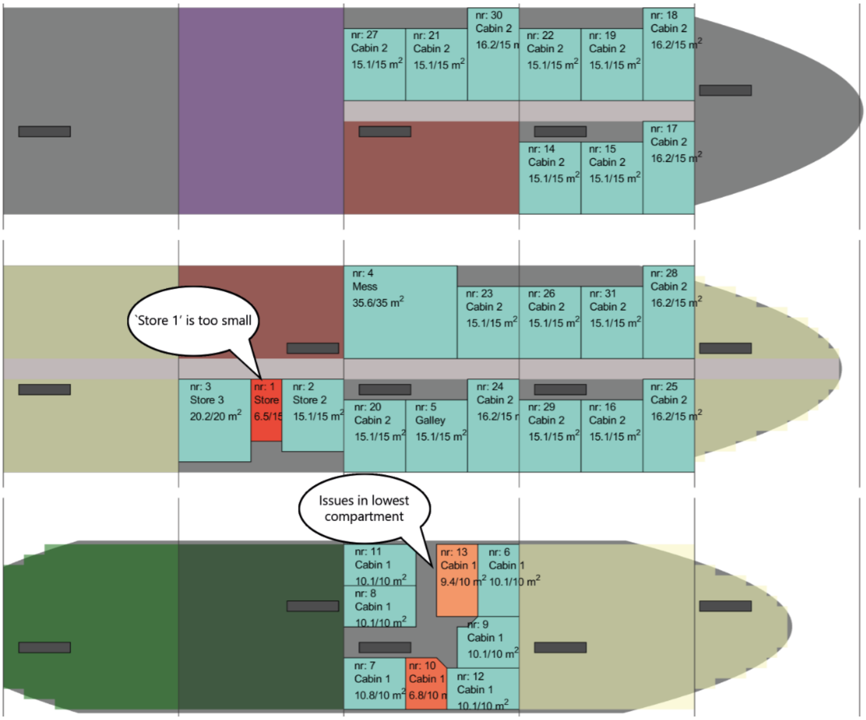
4. Although the allocation of spaces to compartments is not completely equal for the two layouts under consideration, most compartments have been arranged similarly. The main difference is found in the arrangement of the lowest arranged compartment. It appears that in Figure 3.3a this compartment is arranged less realistically than in Figure 3.3b. Also, it seems that one cabin of $10m^2$ cannot be properly arranged.

5. The discretisation of the functional arrangement into positioning matrices with a grid size of $0.6\text{x}0.6m^2$ results in quite substantial differences between the predefined deck shape in the functional arrangement and the detailed layout generated by WARGEAR, see Figures 3.3a and 3.3b. This is especially visible in the bow. In reality, the available area will also be smaller due to structural elements and insulation along bulkheads and the hull. However, this level of detail is not considered in WARGEAR.

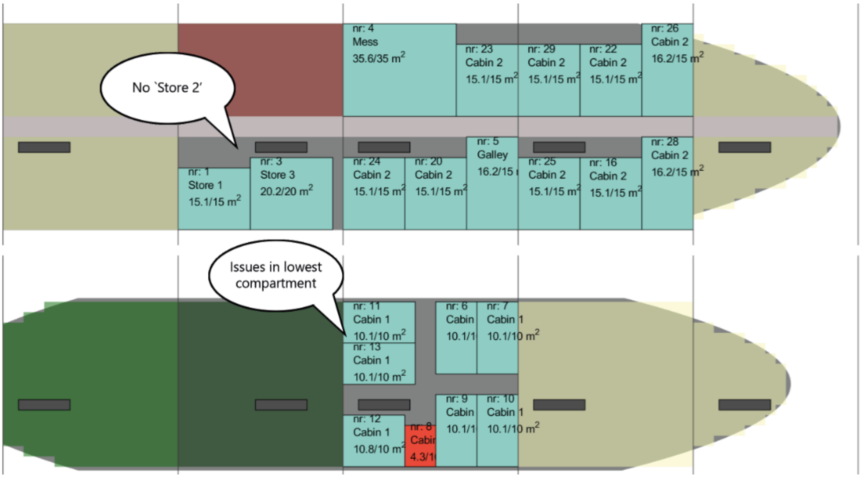
The results of the quick investigation of two layouts lead to the following proposed changes to the input for Case Study 1b:

1. One cabin with a required area of $10m^2$ will be removed from the space list.

2. The required area of 'Store 3' is reduced to $15m^2$, such that three similar stores will be arranged in the notional surface vessel.



(a) Best performing layout from Case Study 1a (ID 158)



(b) Lowest two decks of the best-performing layout that misses one space from Case Study 1a (ID 55). The upper deck is arranged similarly as layout ID158, Figure 3.3a.

Figure 3.3: Two selected layouts from Case Study 1a.

3. The grid resolution will be increased to a grid size of 0.3x0.3m^2, to increase the resemblance of the functional arrangement in the detailed layouts.

3.1.3. CASE STUDY 1B: RESULTS FROM UPDATED INPUT

The proposed changes to the input of Case Study 1b yield the following results, see Table 3.2 and Figure 3.4.

1. The increased resolution by a factor of four of the positioning matrices lead to an increase of the running time (1065 seconds) by a factor of 1.45 (compared to 737 seconds for Case Study 1a).
2. The increased resolution did not solve the grid shape error in the second compartment.
3. No more issues regarding the allocation of spaces have been encountered.
4. The best-performing layout has already been found in iteration 58 of 210, which indicates that the PSO is indeed a sufficient optimisation algorithm for WARGEAR.
5. The three stores can be properly arranged in one compartment.
6. Six spaces don't meet their required area by $0.4m^2$, but this can be easily corrected by the naval architect as sufficient space is available in corresponding compartments to slightly adjust the shape of other spaces to create additional useful area.

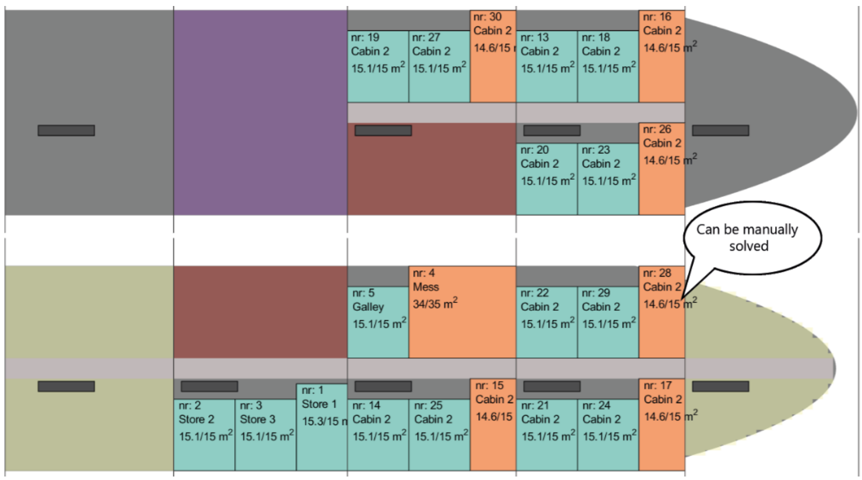


Figure 3.4: Upper two decks of the best performing layout from Case Study 1b (ID 58). The lowest deck was properly arranged

3.1.4. CONCLUSIONS FROM CASE STUDY 1

The main aim of the case study into a notional surface vessel presented above was used to demonstrate and test the integrated mathematics of the WARGEAR method. The results showed that WARGEAR functions as expected, i.e. the integration of the several pieces presented in Chapter 2 is successful.

Secondly, the case study aimed to demonstrate the usefulness of WARGEAR by studying two generated layouts in more detail to identify possible layout sizing and integration issues. This was indeed possible and led to two changes in requirements of the notional surface vessel and one change of the run settings. These changes led to a detailed layout that met almost all spatial requirements and could be used by a naval architect as a starting point for GAP development. These results show that WARGEAR is indeed a useful addition to the naval architect's toolset to reduce the risk of spatial requirements earlier in the design process.

3.2. CASE STUDY 2 - COMPARATIVE STUDY

As mentioned in Section 1.1.2, the layout design problem addressed by WARGEAR was identified at the Netherlands DMO and subsequently led to the research leading into this dissertation. Hence, this section will focus on the first application of WARGEAR at the DMO, aimed at method validation and user acceptance. Therefore this case study consists of the following two parts:

1. **Case Study 2a: Design review**. The method was tested on a realistic warship design case for the purpose of **method validation**. In Section 3.2.1 insight will be given into the type of questions WARGEAR can answer. Also, feedback from naval architects on the results will be elaborated on.

2. **Case Study 2b: Familiarisation**. To enhance **user acceptance**, a comprehensive presentation of the method, why it has been developed, its working mechanism, and the envisioned use in the design process of future naval vessels, has been given to several naval architects and senior management at the DMO. Also, feedback from this session has been recorded. Results are discussed in Section 3.2.2.

3.2.1. CASE STUDY 2A: DESIGN REVIEW

CONTEXT

For a warship design project at the DMO, naval architects were asked to give insight into the possible feasible manning decompositions and the options for feasible accommodation standards for a fixed functional arrangement. The manning decomposition comprises the number of officers, Non-Commissioned Officer (NCO), and ratings to be accommodated in the ship. Following from the manning decomposition is a list of required cabins and corresponding cabin sizes. Depending on the accommodation standard, the required cabin size can be reduced by separating sanitary spaces from the cabins into shared sanitary blocks. On the one hand, this will increase the number of spaces to be arranged, since the space list contains not only various cabins but also sanitary blocks. On the other hand, separating sanitary spaces from cabins might lead to a feasible arrangement of spaces, when larger cabins with a higher accommodation standard cannot be arranged, depending on the overall layout of the vessel and the available area and shape of the available area. In Figure 3.5 this breakdown of choices is summarised.

In the current design process, as described in Section 2.1, naval architects have to spend significant effort investigating the feasibility of the various manning compositions and accommodation standards. Ideally, multiple different GAPs are developed to inves-

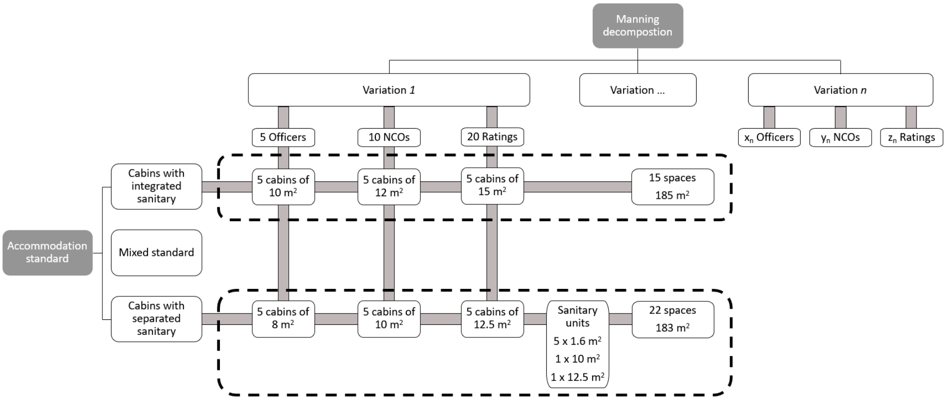


Figure 3.5: Simplified breakdown of the design variations for Case Study 2a. Note that numbers are indicative. The dashed boxes show the spaces resulting from different accommodation standards for the same manning decomposition.

tigate the different options and to better support the stakeholder dialogue. However, due to capacity limitations, typically only a few variations can be studied in detail during early-stage design. For this case study the GAPs generated by the naval architects were used to validate the WARGEAR method and thus to increase the confidence in the method at the DMO.

In order to produce the required input for WARGEAR, one of the naval architects, who had approximately four years of experience in naval ship design and who had worked on the design project, was consulted. This led to clear insight into the variations of manning compositions and accommodation standards to study. After preparing the code to handle the specific functional arrangement, approximately 15-30 minutes were required to produce the space list for the first variation. This space list was already partially available from the standard design process. For each variation, WARGEAR needs approximately 15 minutes to complete the calculations. With an additional 15-30 minutes to review the produced detailed layouts, the total time required to complete one variation is approximately 1 to 1.5 hours. In total eight variations were studied in a period of two days. This is significantly less than the time required by naval architects (approximately two weeks) to find whether a certain manning decomposition and accommodation standard fits into the design, and why (not). Table 3.3 summarises the indicative duration for WARGEAR and naval architect's GAP generation.

Table 3.3: Comparison between indicative duration for WARGEAR and traditional GAP generation.

	GAP	**WARGEAR**
Per variation:		
Input		15-30 min.
Layout generation		15 min.
Post-processing		15-30 min.
Total duration per variation		1-1.5 hours
Number of variations	≈3	8
Total duration	2 weeks	2 days

Note that naval architects still have to develop a GAP, but with the additional insight gained by the WARGEAR study naval architects have a better starting point regarding both the probable feasible manning decomposition and accommodation standards, as well as a rough feasible arrangement of spaces in the ship. The insight provided by WARGEAR can be used to better define spatial requirements earlier in the design process. WARGEAR can be used to assess whether spatial requirements, e.g. requirements for manning decomposition and accommodation standards, are feasible from a technical point of view to inform decision-makers and, if necessary, to challenge requirements. The main purpose of WARGEAR, however, is to inform the naval architects, such that they are better informed in the stakeholder dialogue.

As mentioned above, the naval architects had already developed insight into the possible feasible manning configurations and had developed a GAP for the design case. Therefore, the WARGEAR results of some of the variations could be compared to the naval architect's efforts[5]. The detailed layouts generated by WARGEAR compared well to the existing GAP at various points, which gave confidence that naval architects could use the detailed layouts as a starting point for GAP development in future design tasks. The following three observations could be made.

1. The naval architects were able to arrange cabins in such a way that all furniture would fit, but not all cabins would meet their required area. Since WARGEAR is not able to arrange the furniture inside cabins, only the area of spaces between the detailed layout and the GAP could be compared. Like the naval architects, WARGEAR was able to arrange all spaces, but not regarding their required area. In fact, the difference between the total arranged area by naval architects and WARGEAR was minimal: WARGEAR was able to arrange an additional $7m^2$, which is less than 1% of the total arranged area.

2. Although WARGEAR was able to arrange slightly more square meters, the cabin sizes in the existing GAP were more balanced. Indeed, some of the cabins in the detailed layout were clearly too small to fit all furniture, while most cabins met their required size or were slightly larger[6].

3. WARGEAR only arranges rectangular spaces, whereas naval architects can use more creativity and elaborate shapes, such as L-shapes, to make more efficient use of available area.

EXAMPLE COMPARTMENT ARRANGEMENT

In Figure 3.6a a representative section of a larger detailed layout generated by WARGEAR is given. The figure shows two compartments on one deck, a T-shaped passageway, three staircases, and several arranged spaces. Most spaces have met their required area (RA), i.e. the achieved area (AA) is larger than or equal to the RA, which is indicated by the green colour. Some spaces have not met their area requirement, which is indicated by the red colour shades. The total quality of layout j is assessed using Equation 2.28, which

[5] Because WARGEAR was used on a wider range of design variations.

[6] Spaces in WARGEAR are allowed to overshoot their required area by 20%, or any value set by the user, to meet their allowed aspect ratio, see Section 2.4.6.

is to be minimised. Additionally, the layout is assessed by visual inspection by naval architects, in a manner such as demonstrated in Section 3.1.

Spaces are allocated to the two compartments based on the spaces' required area and the available area in each compartment. The available area equals the total area in a compartment minus the area used by passageways and staircases. Therefore spaces should theoretically fit from an area point of view. Indeed, the total required area is $242.1m^2$ and the available area is $299m^2$. However, not all spaces meet their required area, due to two reasons:

- The two staircases in the aft compartment have been separated from the passageway, which is caused by the arrangement of spaces on a lower deck. The arrangement of staircases resulted in a restricted and irregular shaped positioning matrix, in which not all cabins can be arranged properly, e.g. spaces 40 and 42.
- To connect all spaces and staircases with the main passageway, WARGEAR reserves area to construct local passageways to ensure space connectivity, e.g. from spaces 14, 29, 33, and 45. Spaces 14 and 45, for instance, do not meet their area requirement because of this effect.

Naval architects might consider removing one of the staircases in the aft compartment to create additional area for spaces to be arranged. Also, non-rectangular space shapes might be adopted to solve sizing issues of space 47 for instance. In Figure 3.6b an improved layout has been manually drawn, which is based on the detailed layout developed by WARGEAR. Table 3.6c shows the difference between Figure 3.6a and Figure 3.6b. Three spaces still do not meet their area requirement after manual improvement, although only by $0.2m^2$, while space 46 still needs significant attention.

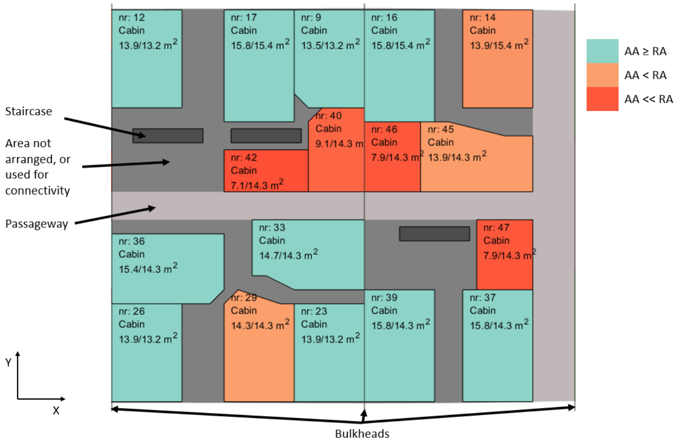
RESULTS

During the case study, the results found by WARGEAR were presented and explained to several naval architects. In the following paragraphs, the reactions to WARGEAR's detailed layout plans are described.

One of the first comments given was on the validation of the detailed layouts, i.e. are the answers given by WARGEAR correct, can we trust the results? This is a valid point, and indeed the main objective of the case study. Detailed layouts can be validated by comparison with GAPs. Unfortunately, there was no GAP available for every variation studied. Therefore not all results could be validated. However, when a GAP was available, the results of WARGEAR were very comparable, see the Context described above.

Further comments were mostly related to the visualisation of detailed layouts to naval architects. Improvements in this area help obtain more design insights and thus help support the stakeholder dialogue, see Chapter 1:

- In previous versions of the code, the visualisation of the detailed layout only contained the arranged spaces and a grey deck shape, as shown in Figure 3.7a. Later, certain FBBs which could not be used for space arrangement, e.g. exhaust casings, were shown as black areas. FBBs are the main elements of functional arrangements, refer to Andrews and Dicks (1997) and Takken (2009). However, this still gave limited insight into the total layout of the vessel. Therefore it was decided to



(a) The detailed layout as generated by WARGEAR, scoring $F = 27.1$

(b) A manually improved layout, scoring $F = 2.5$

Nr.	RA [m²]	AA_{DA} [m²]	AA_{IL} [m²]
9	13.2	13.5	13.8
12	13.2	13.9	13.7
14	15.4	13.9	16.1
16	15.4	15.8	15.2
17	15.4	15.8	17.9
23	13.2	13.9	13.9
26	13.3	13.9	13.9
29	14.3	14.3	15.5
33	14.3	14.7	15.3
36	14.3	15.4	17.1
37	14.3	15.8	15.7
39	14.3	15.8	15.8
40	14.3	9.1	14.1
42	14.3	7.1	14.1
45	14.3	13.9	15.8
46	14.3	7.9	12.3
47	14.3	7.9	16.9

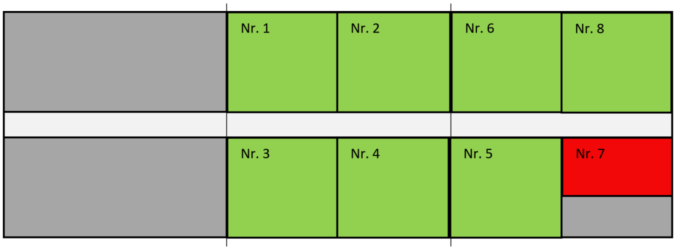
(c) Comparison between required area (RA) and achieved area (AA) in both the detailed (DA) and the improved layout (IL)

Figure 3.6: Example layout of two compartments on one deck as generated by WARGEAR and a manually improved layout.

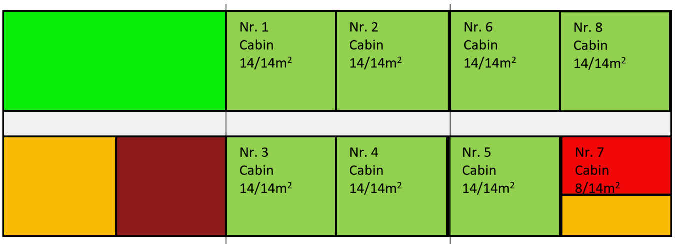
improve the visualisation by including all FBBs which were not further arranged by WARGEAR. To support the comprehension of the detailed layout, the colours from the functional arrangement were used.

- In the past, area success has been defined as missing grid cells regarding achieved and required area (Le Poole et al., 2019). However, this gives limited insight into the actual square meters missing, since the score has to be multiplied by the squared grid size (typically $0.6mx0.6m = 0.36\frac{m^2}{\text{gridcell}}$), which is inconvenient. To better inform the naval architect, WARGEAR now calculates the actual missing square meters.

3



(a) Layout visualisation as in Le Poole et al. (2019).



(b) Updated layout visualisation.

Figure 3.7: Comparison between layout visualisation in Le Poole et al. (2019) and the updated visualisation. The example comprises three compartments, eight spaces (Nr. 1-8), and four fixed FBBs (grey area in (a)).

- The use of green and red colour shades gives direct insight into which spaces have or haven't met their required area (Le Poole et al., 2019). However, it proved difficult to communicate the extent to which spaces did not meet their required area via various shades of red because the shades provided insufficient information on the actual achieved and required area of spaces. Therefore it was proposed to present the actually achieved area and required area for each space in the detailed layout as well. After discussing a layout with a naval architect the space name was also added to enhance insight.

To provide a visual comparison between the past and current visualisation of detailed layouts, see Figure 3.7. These Figures show three compartments of which only two compartments contain spaces arranged by WARGEAR. In the third compartment, one FBB cannot be used for space arrangement. While this is immediately clear from Figure 3.7b, Figure 3.7a will likely make naval architects wonder why space 7 cannot be fully arranged, due to the missing context. Note that this updated visualisation was also used in Section 3.1.

Therefore, these changes to the visualisation of the detailed layout allow naval architects to get more direct insight into the success of the layout. Based on an investigation of the detailed layout by naval architects, additional variations to the space list or adaptions to the functional arrangement can be made. Naval architects can also accept the detailed layout as a starting point for further detailing into a GAP, even if the detailed layout has not met the required area of all spaces. If, for example, one space has achieved 14.5 of the required $15m^2$ and some free area is available, naval architects are likely able to solve this by rearranging spaces, or by using more elaborate shapes to arrange spaces, such as L-shaped spaces. As noted earlier, WARGEAR gives a good initial starting point including an analysis of where potential area problems arise.

LESSONS LEARNED

This section will elaborate on lessons learned from the first application of WARGEAR at the DMO. The lessons learned can be divided into two categories, 1) method-related and 2) process-related. These are elaborated on below:

1. *Method-related*
   Some issues were identified related to the input of the functional arrangement into WARGEAR.

   - The designation of decks is used to identify the DCD, amongst others. To identify the DCD, WARGEAR tries to find the deck called ‘DCD’, a name which had been used in functional arrangements studied for the setup of WARGEAR. However, in this case study WARGEAR was not able to find the DCD, because the deck was designated ‘DCD-deck’. This issue can be solved by a mandatory check of deck names at the input phase of the method, or by a user input.
   - The curvature of the ship’s keel caused errors during the generation of the positioning matrices (see Le Poole et al. (2019)). Since WARGEAR can only arrange flat decks, only decks which are not affected by the curvature of the keel line can be loaded into the method. However, in most cases, this is no problem as mainly tanks and engine rooms are allocated below the waterline, which falls outside the scope of WARGEAR. In order to allow the arrangement of curved parts of the hull, these parts could be vertically projected to a flat deck, as shown in Figure 3.8.

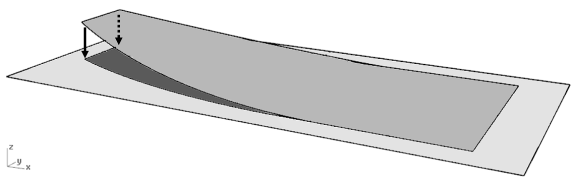
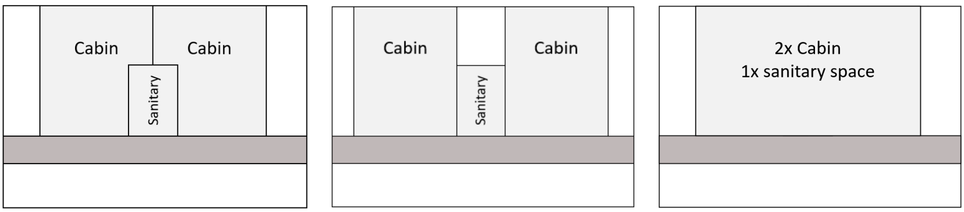


Figure 3.8: Projecting a curved deck to a flat deck plane.

   - Currently, the floor plan of each deck is used for the arrangement of spaces. However, the ceiling area of a deck might be smaller than its floor area, which is, for example, often the case in naval ship’s superstructures to reduce radar cross-section. Therefore both the floor and ceiling plans of each deck need to be considered when creating the positioning matrix for spaces.

2. *Process-related*
   In preparation for the case study, a generalised input file was developed to improve usability. The case study also revealed the following issues:

- A reporting method for traceability is currently missing. Therefore design variations need to be stored in separate input files. However, the results from the calculations and the rationale behind the variations cannot be stored in an integrated data set at the moment. Such an integrated data set is essential for traceability and documentation of results, as lessons learned will be easily lost otherwise (see Part II).
- The case study also showed that very small spaces, such as single sanitary units of approximately $2m^2$, are difficult to arrange correctly. This is particularly the case when an efficient arrangement requires large and small spaces to be alternately placed. If naval architects already envision a particular solution, such as shown in Figure 3.9a, they might congregate a larger and smaller space to guide WARGEAR and thus to improve the quality of the overall layout, as shown in Figure 3.9b and Figure 3.9c. The disadvantage of this approach is that these larger congregated spaces can be harder to arrange in irregular shaped compartments.



(a) Envisioned arrangement of two cabins and one sanitary space.

(b) Theoretical best solution for WARGEAR, with more scattered void area.

(c) One functional block to be arranged by WARGEAR, representing Figure 3.9a.

Figure 3.9: Congregation of multiple spaces to improve detailed layout quality.

3.2.2. CASE STUDY 2B: FAMILIARISATION

Following the execution of Case Study 2a, the WARGEAR method was presented to eight naval architects and senior officers and managers at the DMO. The aim of this presentation was to strengthen the acceptance of the method into the DMO ship design process by providing insight into the method and its capabilities. The main topics were a recapitulation of the research goal, the requirements developed for the method, the principle mathematical working mechanisms, and the envisioned use of WARGEAR in the DMO design process. The latter was illustrated and supported by the results of the validation case study described in Section 3.2.1. The feedback was not recorded during the presentation itself but documented immediately after the presentation, and subsequently further processed. In total twenty comments were recorded. Processing these comments, three categories could be distinguished, namely: Usability, Reliability, and Application. In the remainder of this section, the main comments for each category will be discussed. In general, the naval architects were positive about the possibilities WARGEAR provides to rapidly generate detailed layout plans. The presentation was seen as informative, both on the side of the mathematical background of the method, as well as on the presentation of the results and possible applications of WARGEAR.

Usability

The usability category concerns how willing naval architects are to use the method in their work and what additions to the method will be required to improve usability. The comments provided in this category mainly concerned the generation of the required input for WARGEAR. Other comments relate to the ability of naval architects to positively influence the quality of the detailed layouts.

Input generation

Several comments addressed the generation of the space list. During the presentation, some concerns were already taken away by explaining that only 30 minutes were required to generate the first variation of the space list for Case Study 2a. Later variations took even less effort. However, more efficient generation of the space list is possible, for instance via a default sheet or by making use of recent research carried out at the DMO to generate the space list based on requirements (Van der Weg, 2020).

At the moment only FBBs can be used to specify the preferred location of spaces, which is sufficient in general. However, naval architects might want to allocate spaces into a specific compartment. Indeed, while arranging FBBs, higher level of detail aspects (e.g. which spaces are placed in which compartment or FBB) are also considered. Because FBBs can overlap with multiple compartments, a compartment-specific allocation is currently not possible. To efficiently translate the naval architect's mental picture of the allocation of spaces to compartments to WARGEAR input, existing tools and methods at the DMO, such as FIDES, and WARGEAR need to be coupled. It was determined that the DMO will be responsible for the proper coupling between tools and methods. Therefore, this topic is out of scope for the remainder of this dissertation. Efficient generation of the space list is considered to be sufficient for WARGEAR, while tool integration is considered out of scope.

Quality of layouts

The naval architects would like to have increased control over the dimensions of spaces. Currently, space sizes (length and width) are determined by required area and a range of allowed aspect ratios. However, some discrete sizes might not be preferred because the efficient arrangement of furniture can be challenging or impossible. In the current setup of the method, the implementation of this feature is relatively easy. However, the workload in the Input phase of the WARGEAR process needs to be carefully considered, i.e. naval architects should not be overloaded with checks upfront as this would be considered a hindrance to using the method. The amount of required input and the quality of the layouts needs to be balanced properly.

To properly test the usability of the WARGEAR tool, future test cases should also be performed by DMO naval architects. Only by hands-on experience and use of the method, the bottlenecks in the workflow will emerge. Subsequently, improvements to the usability of the method can be implemented. An example could be a list of standard cabin sizes, to reduce the effort required for WARGEAR input generation.

3

Reliability

The reliability category covers comments concerning the method's ability to generate accurate detailed layouts. One of the naval architects addressed the implementation of staircases in WARGEAR, as the sizing of a staircase at the lowest and upper deck is equal to the sizing of the staircase on intermediate decks. In reality, the area around the staircase on the lowest and upper deck can be used more efficiently, as indicated in Figure 3.10. By simplifying staircases, WARGEAR provides a slightly conservative estimation of the total area required for staircases.

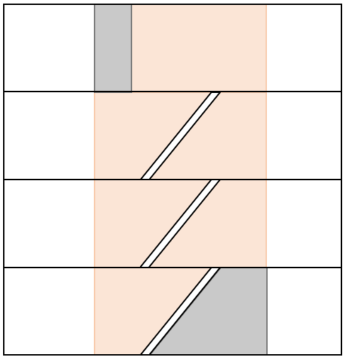


Figure 3.10: Difference between open area required around ship staircases in reality (orange) and area arranged by WARGEAR (orange and grey).

During the presentation, the aim of WARGEAR was discussed. WARGEAR has not been developed to replace naval architects but to support them in their work by providing quick insight into layout sizing and integration problems. However, the work of the naval architects is certainly taken as a benchmark for the quality of the detailed layouts during the validation of WARGEAR. Indeed, to be used extensively at the DMO, WARGEAR should perform at least at a comparable level as naval architects as the results should be reliable and give sufficient insight into layout risks. The results of Case Study 2a were sufficient proof to the audience that WARGEAR is sufficiently able to answer the questions of interest.

It was also stressed that the WARGEAR method should be able to generate detailed layouts that comply to *rules*, and that therefore not all *exceptions* should be modelled. It is believed that modelling exceptions will decrease the effectiveness of the code, as it will increase the number of requirements to which detailed layouts should comply. Also, the number of exceptions will increase with every new ship type. One attendant put this as: 'Experience tells us that there will always be something that we had not thought of'. Therefore, WARGEAR needs to use the set of rules that will result in reliable layouts for as many ship types as possible. Exceptions are then dealt with when manually translating the WARGEAR results into a GAP.

Application

The application category relates to the usability category and covers the comments on the integration of WARGEAR in the DMO design process and the types of design problems WARGEAR can be used for.

In FIDES, FBBs can be sized based on the required area of spaces and an area margin for additional logistic systems, such as staircases. The attendants would like to see a feedback loop between WARGEAR and FIDES to improve the initial area estimations and margins. This could be done by relating the area required for spaces and logistic systems in the detailed layouts to the initial estimated area and margins in the functional arrangement. However, before altering the FIDES input, a manually generated GAP needs to be generated for a detailed assessment of all aspects of a layout. Nonetheless, the difference in required and anticipated areas will be communicated to the naval architects so they can use this information to improve the sizing of the functional arrangement, which is a main objective of WARGEAR in the first place.

All attendants agreed that the detailed layouts provide significant insight in layout sizing issues. Notably, the detailed layouts prompted the naval architects into spontaneous discussions about design variations, such as the benefits of single versus double passageways. Although the presentation was not aimed to prompt these discussions, it indicates that the detailed layouts can be used for other purposes than just validation of functional arrangements. One of these roles is to provide a starting point for the generation of GAPs, which could support the stakeholder dialogue later in conceptual design, see Chapter 1.

Another possible use that was identified, is the validation of designs with an even lower level of detail than functional arrangements. For example, at the DMO another ship design method, Packing (Van Oers et al., 2018; van Oers, 2011b), is used to generate low-detail designs with the aim to explore initial requirements. A rough translation of a ‘Packing’ design to a functional arrangement might allow the generation of detailed layouts in a limited time, and thus enable more in-depth validation of initial requirements. This would be beneficial to improve initial budget estimations, for example. However, the translation of a ‘Packing’ design to a functional arrangement might be challenging because the internal layout of the former lacks a significant level of realism, which could be time-consuming to solve.

A question was raised about whether WARGEAR can be used to arrange machinery spaces. This falls outside the scope but would be an interesting study. WARGEAR cannot readily arrange machinery, since the arrangement of machinery differs from the arrangement of spaces at various points. For instance, space size is more flexible than machinery size, positioning of machinery is more restricted by ship stability constraints, and access to machinery needs to be taken into account as well as pipe routing and length (Van der Bles, 2019; Poulis, 2022).

3.2.3. CONCLUSIONS FROM CASE STUDY 2

This case study had a twofold goal, namely 1) method validation, and 2) user acceptance.

1. **Method validation**
   A comparative test case showed that WARGEAR is able to generate detailed layout plans that compare well to GAPs previously generated by naval architects. However, naval architects require days to provide the same insight, which could be provided by WARGEAR in a matter of hours. This will enable naval architects to

spend their time on solving sizing and integration issues, rather than identifying these issues.

2. **User acceptance**
   To strengthen user acceptance, the results of the test case were shared with multiple naval architects and senior officers and managers at DMO. The attendees were generally positive but also provided several comments and questions regarding the case study and the WARGEAR method.

This case study shows that WARGEAR is indeed a useful addition to the naval architect's tool set to reduce the risk of spatial requirements earlier in the design process.

3.3. Case Study 3 - Extending WARGEAR's allocation algorithm

3.3.1. Introduction

The two case studies described above focus on the validation of the WARGEAR method. The results show that WARGEAR can be used iteratively to gain insights into detailed layout sizing and integration issues in a limited time. In Case Study 3, a part of WARGEAR will be used and extended to investigate if a higher speed can be achieved compared to Case Study 2, for a larger and more complex input set. In addition, an attempt will be made to relax WARGEAR's need for sufficiently developed functional arrangements as input. This could support the use WARGEAR earlier in the design process to identify potential sizing and integration issues earlier, for instance by coupling to Packing (Van Oers, 2011b).

Hence, this case study will further answer this Part's main research question: *To what extent can automated layout generation methods support real-time design decisions during early-stage complex ship design?*

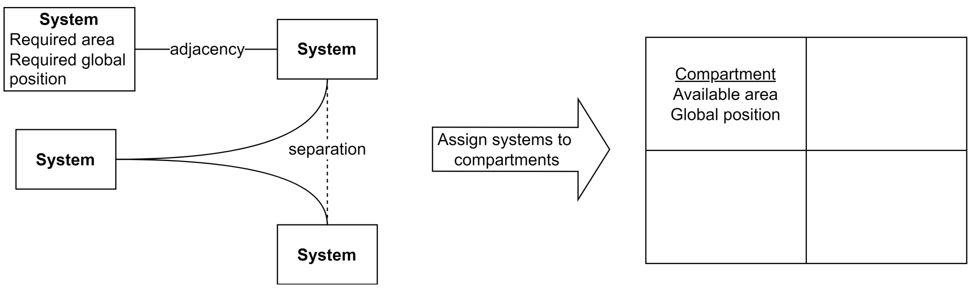


Figure 3.11: Visual explanation of the systems-to-compartments allocation problem

To answer the question posed above, the real-time use of automated design tools is broken into two sub-questions: 1) *how can such tools be used to generate concept designs in real-time?*, and 2) *how can these concept designs be analysed in real-time?* Answering these questions for all design tools is infeasible for the scope of this dissertation. Therefore, a representative design problem is used, namely the allocation of systems to compartments. This design problem, as illustrated in Figure 3.11, is highly dimensional (due to the high number of design options (Duchateau, 2016)), is subject to many spatial constraints (e.g. compartment sizing), and has an impact on the ship's performance (e.g. logistics, stability, and vulnerability of distributed systems), and is input to many design disciplines. Additionally, design parameters can be highly interdependent (e.g. available area in compartments versus required area and global position for systems as well as relative positions between systems). Finally, the allocation of systems to compartments can be a starting point for more detailed layout design (e.g. Medjdoub & Yannou, 2000; Nick, 2008; le Poole et al., 2022d). Hence, this problem is considered to be a suitable example of overall ship layout design.

Various research investigated the system-to-compartment allocation problem, (e.g. Gillespie, 2012; Nick, 2008; Stevens, 2016). In addition to these examples, WARGEAR has a system-to-compartment allocation algorithm (Section 2.4.5). This algorithm is used in Case Study 3 to answer the questions posed above.

3.3.2. METHOD - OVERVIEW

As introduced above, in this case study the problem of allocating systems to compartments is studied. To solve such problems, this dissertation proposes the three-step method shown in Figure 3.12:

1. *Input.* This step is human-centric, as it requires the designer to decide on the constraints and requirements for the allocation problem, as described in Section 3.3.1.

2. *Allocation.* In this step, an automated allocation algorithm is used to generate concept designs. This allocation method is based on the algorithm discussed in Section 2.4.5.

3. *Analysis.* This step is human-centric again, as it is aimed to let the designers gain insight into potential sizing and integration issues. This requires substantial data analysis efforts, which typically is a very involved activity (Duchateau, 2016).

The human-centric steps are expected to be most time-consuming, but cannot be eliminated because it's also in these steps (especially during Analysis) that most learning occurs (Section 2.1). However, also these steps can be supported by, for example, appropriate visualisations. The three steps are further elaborated below.

When used in an actual design process, the availability of a dedicated database for storing data related to the three steps (i.e. input, allocation, and analysis) is considered to be important. Indeed, Duchateau (2016) notes that, on the one hand, if "the user has to wait for long periods between each iteration [...] he or she will likely lose focus or fail to keep track of the decision steps in each consecutive iteration." On the other hand, Duchateau mentions that problems (e.g. fatigue and loss of focus) may be caused when human-computer "interaction moments follow in quick succession, especially when dealing with a large amount of complex results." In real-time collaborative decision-making processes, both types of interaction frequencies will appear - between and within design sessions respectively. Hence the storage of design data (and supporting rationale) for later retrieval is expected to benefit the designer and the overall design process. See also DeNucci (2012) and Part II.

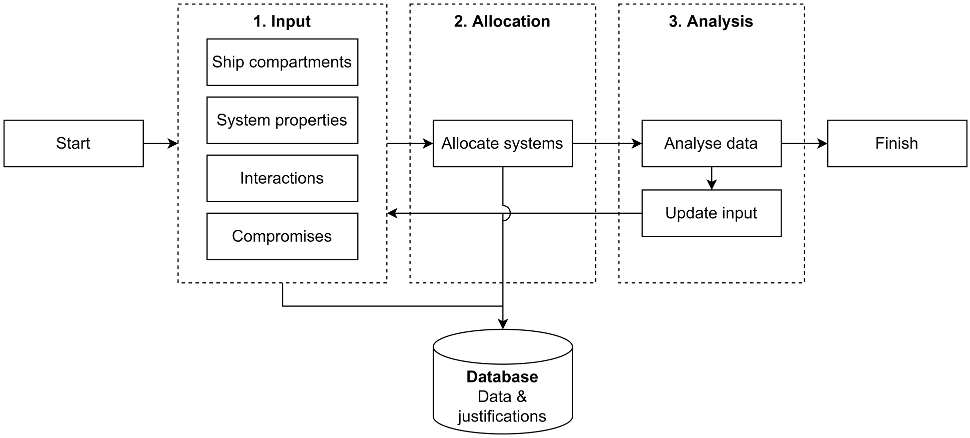


Figure 3.12: High level overview of the proposed method

3.3.3. METHOD - INPUT

The first step of the allocation method is to generate the input. This input is subsequently used by the allocation algorithm to allocate required systems to the available compartments. The input comprises:

- *Ship compartments.* Ship compartments are enclosed by transverse bulkheads and decks. Positions of transverse bulkheads in naval ships are often driven by damage length considerations and required space for larger systems such as engine rooms, or main sensor masts which require sufficient structural support. The compartmentation of the concept design is generated via bulkhead and deck positions, as well as deck area per compartment. This could be extended to include, for instance, available volume per compartment. Each compartment is assigned a vertical and longitudinal global position, describing where the compartment is situated in the ship.

- *System Properties.* System Properties are captured in a list of systems and their respective properties. These properties are, for example, required area and volume, or preferred global positions. Currently, the method considers required area and global positions of systems. The latter are expressed in terms of the global positions of compartments.

- *Interactions.* Interactions are preferred or required spatial relationships between systems or System Properties (DeNucci, 2012; le Poole et al., 2022c). Originally, WARGEAR required designers to link systems to particular FBBs or compartments, see Section 2.4.2. Based on these relationships, WARGEAR would assign systems to compartments. However, WARGEAR is not able to group or spread systems based on interactions between systems. Table 3.4 presents the five interaction types that have been implemented for this Case Study. These interaction types are visualised in Figure 3.13.

Table 3.4: Interaction types

ID	Description	Explanation
1	Compartment adjacency	systems need to be in the same compartment.
-1	Compartment separation	systems need to be in different compartments.
2	Maximum Manhattan distance	systems can be separated by a maximum Manhattan distance. the Manhattan distance is calculated between compartment centroids since a precise position for each space is not available yet.
-2	Minimum Manhattan distance	systems should be separated by a minimum Manhattan distance.
-3	Minimum radial separation	systems should be separated by a minimum number of compartments.

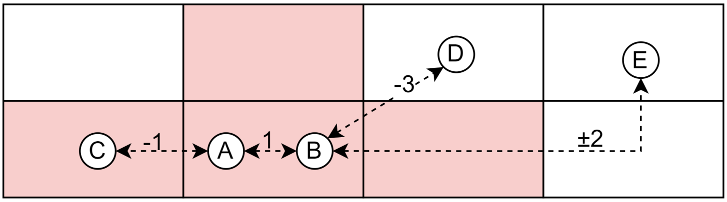


Figure 3.13: Visual representation of implemented interaction types. A–B: Compartment adjacency; A–C: Compartment separation; B-D: minimum radial separation; B–E: minimum/maximum Manhattan distance.

- *Compromises.* Compromises form the preferred solutions to a set of conflicting or competing interactions (DeNucci, 2012) or System Properties (Le Poole et al., 2022c). Currently, compromises are not implemented in the tool but are considered to be a useful feature. Indeed, this would allow the tool to make trade-offs in line with what the human designer prefers.

System Properties, Interactions, and Compromises also comprise a justification. An example of an interaction is *the ammunition store should be adjacent to the gun* [relation], *to reduce dangerous transport of ammunition through the ship* [justification] (Le Poole et al., 2022c). In this case study, the justification for the input is not explicitly used by the tools but might be useful for retrieval during actual decision-making, as mentioned in Section 3.3.2. See also Part II.

3.3.4. METHOD - ALLOCATION

As mentioned before, the algorithm used to allocate systems to compartments is an extension of the algorithm presented in Chapter 2. For a detailed overview of the allocation algorithm, refer to Section 2.4.5. In this section, only the relevant parts and extensions will be discussed. A flowchart of the adapted allocation algorithm is provided in Figure 3.14. Elements with grey shading have been added or adapted from the original version. In short, the extensions and adoptions comprise:

1. The inclusion of global position and interaction constraints.
2. Relaxation of these constraints in case these are too restrictive.
3. The option to use multiple system sorting algorithms, to enable different allocation sequences.
4. The option to perform the allocation multiple times, to achieve a more precisely defined design space.

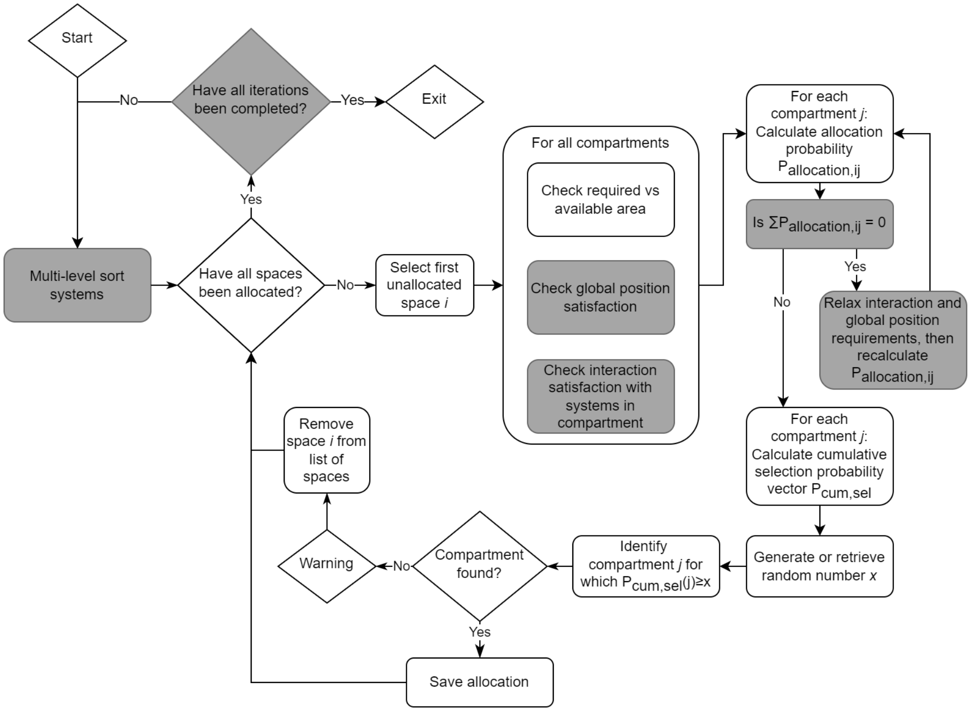


Figure 3.14: Flowchart of the allocation calculations, extended from Figure 2.11. Grey-shaded elements represent extensions and adaptions for this case study.

To determine which compartments are available for allocation of system i, the following three aspects are considered:

1. The available area in a compartment needs to be sufficient to accommodate system i. The available area $A_{available,j}$ in compartment j is defined by Equation 3.1. This equation is a simplification of Equation 2.4 and takes only into consideration that the available area decreases when systems get allocated to compartment j.

$$A_{available,j} = A_{compartment,j} - A_{allocated\,systems,j} \tag{3.1}$$

2. A compartment needs to fulfill specified System Properties, such as global positions. If a system needs to be high up in the ship, compartments that are located high up are preferred over compartments situated at the bottom of the vessel.

3. Systems need to be allocated in the same compartment as other systems with which these share adjacency relationships. Similarly for other types of interactions, compartments that fulfill relative position constraints are preferred.

To differentiate between available and preferred compartments the probability $P_{allocation,ij}$ that system i is allocated to compartment j was used in Section 2.4.5. This probability takes into account the considerations for preferring or ignoring compartments.

$P_{allocation,ij}$ is given by Equation 3.2 and has been adapted from the original formulation (Equation 2.6) to include global positions and interactions. Subsequently, the updated $P_{allocation,ij}$ is used in Equations 2.7 and 2.8 (see Section 2.4.5).

$$P_{allocation,ij} = \begin{cases} \frac{A_{available,j} \cdot N_{intsat,j}}{Degree_{comp,j}} & \text{if } A_{available,j} \geq RA_i \text{ and } GP_j = GP_i \text{ (if } GP_i \text{is specified)} \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

Where:

$N_{intsat,j}$ is the number of interactions between already allocated systems and system i, that will be satisfied if system i is allocated to compartment j. If there are no such interactions, $N_{intsat,j} = 1$.

GP_i and GP_j are the global position of system i and compartment j respectively.

$Degree_{comp,j}$ is the number of systems a compartment is connected to, based on GP_i and GP_j.

If $P_{allocation,ij} = 0$ for all compartments, no compartment is available that satisfies required area, global position, and interaction requirements. In such cases, the global position and interaction requirements are relaxed such that all compartments adjacent to initially preferred compartments are now also available. Subsequently, $P_{allocation,ij}$ is recalculated. Figure 3.15 shows that adjacent compartments to compartments that would satisfy global positions or interactions become preferred compartments after relaxation. Note that other relaxation rules are possible, e.g. to only extend to compartments at the same deck. As in Chapter 2, a roulette wheel selection method is then used to select between available compartments for system i.

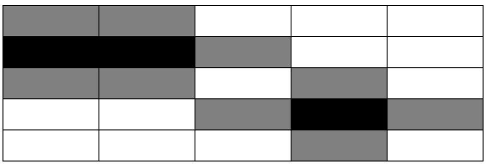


Figure 3.15: Compartment preference before (black) and after relaxation (grey)

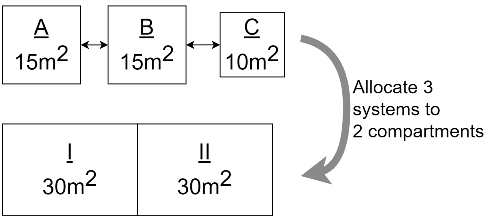


Figure 3.16: Setup of three systems to be allocated to two compartments.

To illustrate this procedure, consider three systems A ($15m^2$), B ($15m^2$), and C ($10m^2$) need to be allocated to two compartments I ($30m^2$) and II ($30m^2$), shown in Figure 3.16. Two interactions are defined between systems A-B and B-C, meaning systems in these pairs need to be adjacent, i.e. in the same compartment. The available area in none of the compartments is sufficient to accommodate all three systems. Table 3.5 summarises the calculation of $P_{allocation}$ for the two compartments. The systems are allocated in the order A, B, C. For system A, any compartment can be chosen with equal probability. Assume compartment I is selected for system A. Consequently, system B will be allocated to compartment I as well to satisfy interaction A-B. Finally, the allocation of system C fails because of the need to satisfy interaction B-C and insufficient available area in compartment I. Relaxation of the interaction requirement allows system C to be allocated to compartments adjacent to preferred compartments, i.e. compartment II.

Table 3.5: Allocation of systems A, B, and C to compartments I and II.
1): Underlined text indicates selected compartments. 2): $P_{allocation} = 0$ for both compartments, hence the interaction requirement is relaxed. 3): due to relaxation, compartment II becomes available for system C.

	Step 1. Allocate system A		Step 2. Allocate system B		Step 3. Attempt to allocate system C		Step 4. Allocate system C (after relaxing interaction B-C)	
	Compartment		Compartment		Compartment		Compartment	
	I	II	I	II	I	II	I	II
$A_{available}$	30	30	15	30	0	30	0	30
$Degree_{comp}$	1	1	1	1	1	1	1	1
$N_{intsat,j}$	1	1	1	0	1	0	1	$1^{3)}$
$A_{available} \geq RA$	1	1	1	1	0	1	0	1
$GP_j = GP_i$	1	1	1	1	1	1	1	1
$P_{allocation}$	$\underline{30}^{1)}$	30	$\underline{15}$	$0^{2)}$	$0^{2)}$	0	0	$\underline{30}$

The output of the allocation phase is a set of allocations, i.e. preliminary layouts of compartments with allocated systems, and data describing the performance of these layouts with respect to the input. For instance, the data describes which system properties and interactions have been satisfied.

3.3.5. METHOD - ANALYSIS

The analysis of the data generated by the allocation algorithm can lead to insights, which can be used in subsequent design decision-making. The analysis process is very much human-centric and involves exploring and working with the data (i.e. data *exploration*) (Duchateau, 2016). To guide the analysis process, the following aspects need to be investigated (Duchateau, 2016):

1. Identify how, when, and why design parameters relate. This includes the identification of positive (i.e. re-enforcing) interdependencies as well as conflicting relationships.

2. Identify how potential conflicts might be resolved or avoided.

To support the exploration to answer these questions, designers might make use of (dynamic) visualisation and filtering of the data (Duchateau, 2016; Gaspar et al., 2014; van Oers, 2011b). The following three-step analysis process is proposed:

1. *Identify nature of design parameter relationships*. That is, the naval architect is to identify whether design parameters are likely to conflict and to what extent. One means to quantify such a relationship between design parameters is correlation. In terms of the general arrangement of ships, a high correlation between two design parameters means that these parameters can both likely be satisfied. For example, suppose two systems with each a parameter 'area'. A high positive correlation between these two area parameters indicates that, across the set of concept designs, the two systems often satisfy these design constraints. Therefore, correlation can be a powerful means to get insight into the relationships between all pairs of design parameters. However, this will require appropriate visualisation due to the large dimensionality of the data set. In this Case Study, binary performance is investigated, i.e. parameters are satisfied or not satisfied. To quantify relationships between parameters, the ϕ coefficient of correlation (Garrett, 1958, p389) can be used to calculate the correlation between the binary satisfaction of all pairs of design parameters across the (potentially filtered) set of generated concept designs. The ϕ coefficient is given by Equation 3.3 (Garrett, 1958), in which $A-D$ refer to the four quadrants in Table 3.6.

$$\phi = \frac{AD - BC}{\sqrt{(A+B)(C+D)(B+D)(A+C)}} \tag{3.3}$$

Table 3.6: Matrix for calculation of coefficients of correlation between two binary items.
A-D: number of observations in the data set. $\phi = -0.58$, $r_t = -0.05$ and $r_c = 0.43$ for the example (right)

		Item 1	
		No	**Yes**
Item 2	**Yes**	B	A
	No	D	C

		Item 1	
		No	**Yes**
Item 2	**Yes**	6	10
	No	1	4

Although the ϕ-coefficient of correlation provides a measure of the correlation between two binary items, it provides only limited insight into the *extent* that both items can be satisfied. For example, ϕ does not communicate the balance between A and D. Hence, ϕ cannot be used to inform the designer whether two items can generally be satisfied (i.e. A is larger than D) or if they can generally not be met simultaneously (i.e. A is smaller than D). Therefore, a new two-item correlation metric has been developed. It quantifies to which extent two items can be satisfied relative to the extent that both or one of the items needs to be compromised.

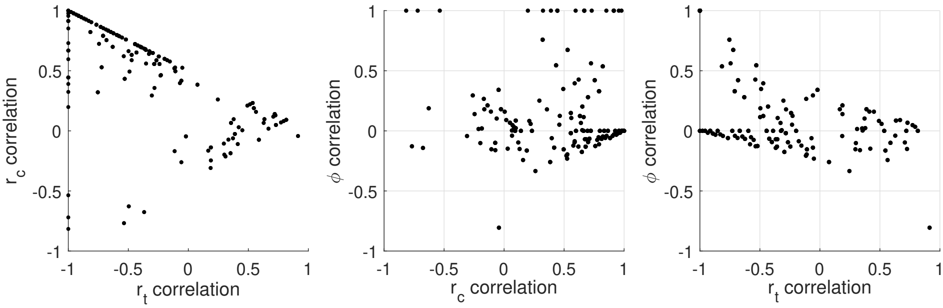
The first coefficient, r_t, is provided by Equation 3.4 and describes in how many cases two items need to be traded off against each other, i.e. one can choose only item 1 or only item 2. If $r_t = -1$, there are no cases in which there is a strict trade-off necessary. If $r_t = 1$, there is a conflict between the two items in all generated concept designs. If $r_t < 0$, less than half of the cases comprise a conflict. The remaining cases comprise either cases where both items are satisfied or cases where neither of the cases is satisfied.

$$r_t = 2\frac{B+C}{A+B+C+D} - 1 \tag{3.4}$$

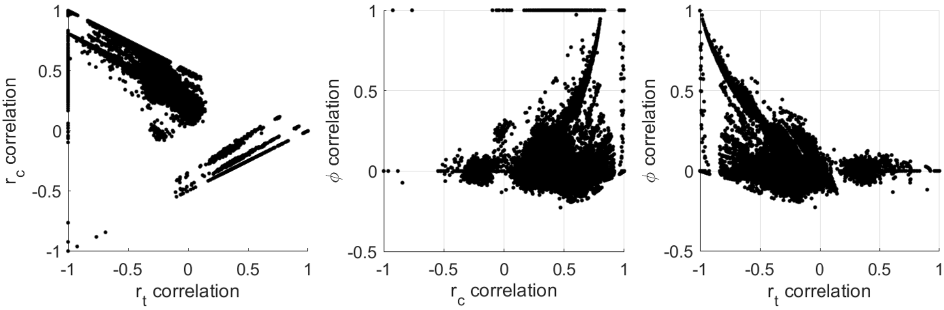
The second coefficient, r_c, is provided by Equation 3.5 and describes the balance between the number of concept designs where both items are satisfied (i.e. A) and the number of cases where neither of the items is satisfied (i.e. D). If $r_c > 0$, A is larger than D. The maximum value of $r_c = 1$, which means that both items are satisfied in all cases. Similarly, $r_c < 0$ if A is smaller than D, and $r_c = -1$ if both items are never satisfied at the same time.

$$r_c = \frac{A - D}{A + B + C + D} \tag{3.5}$$

Hence, the Utopian point for the two new correlation coefficients is $r_t = -1$ and $r_c = 1$.



(a) Based on 19 parameters in Case Study 3a.



(b) Based on 739 parameters in Case Study 3b.

Figure 3.17: Comparison between the ϕ (Garrett, 1958), and the new r_t and r_c coefficients of correlation. Each dot represents the correlation value between two design parameters across 1000 designs. The left figures show the r_t and r_c correlation for all pairs of design parameters. The middle figures show the ϕ and r_c correlation and the right figures show the ϕ and r_t correlation for the same parameters.

Figure 3.17 shows the relation between Garrett (1958)'s ϕ, and the new r_t and r_c coefficients of correlation, based on 19 and 739 parameters across 1000 concept designs generated in the case studies presented in Sections 3.3.6 and 3.3.7 respectively. The left figures show the relation between r_t and r_c. It clearly shows that many pairs of design parameters can often be met simultaneously (i.e. $r_t \approx -1$

and $r_c \approx 1$), or need to be traded off (i.e. $r_t \approx 1$ and $r_c \approx 0$). There appears to be a slight negative correlation between r_t and ϕ (right figures). However, there is no clear correlation between r_c and ϕ (middle figures). Therefore, ϕ does indeed provide some information on whether two parameters need to be traded off but does not provide information on whether two parameters can be met simultaneously. Both the small and large cases show similar correlations. This indicates the metric can be generally used. Hence, both r_c and r_t are used instead of ϕ in the remainder of this case study to allow designers to get clear insights into the relationship between pairs of design parameters.

2. *Use exploratory filtering.* Although r_c and r_t can be used to inform the designer of the nature of the relationship between *pairs* of parameters, additional effort is required to identify how larger sets of parameters relate. For example, if $A < B$, $B < C$, and $C < A$, a pairwise comparison does not immediately reveal the inconsistency in the relationships. Indeed, besides dependence between pairs of design parameters, a designer needs to know the dependencies between *all* parameters, for instance, to evaluate which combinations of parameters are most restrictive to the design space. A concrete example is the question if all specified global positions can be satisfied in a single concept design, and if not, which global positions cannot be satisfied and why they cannot be satisfied. Eventually, interactive, exploratory filtering of the design data helps the designer to identify how, when and why design parameters relate, but also to identify potential promising concept designs (Duchateau, 2016).

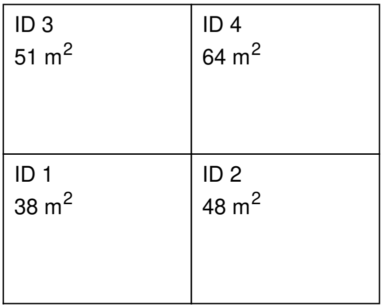
3. *Generate and analyse selected concept designs.* Studying individual concept designs might yield additional insights into possible solutions to address identified conflicts. Additionally, it might be used to identify which parameters need to be adapted in subsequent iterations. Generally, concept designs are less abstract than numerical representations of design data (such as the developed correlation coefficients). Thus, individual concept designs might be of good use during collaborative design sessions, i.e. to identify additional design rationale (DeNucci, 2012) or as a familiar representation of the design (Van Oers et al., 2018).

3.3.6. CASE STUDY 3A: SMALL SCALE DEMONSTRATION

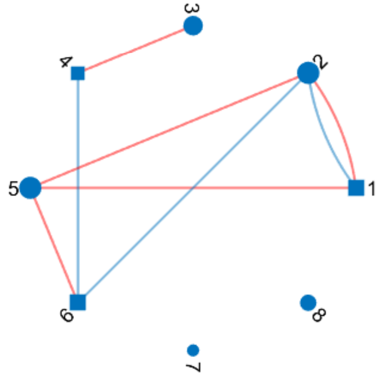
This section describes a small case study that demonstrates the principle working mechanisms of the allocation algorithm as well as the data exploration process. The case study comprises 4 compartments of various sizing, 8 systems with various sizing and positioning requirements, and 8 interactions between these systems. This input is visualised in Figure 3.18. The case study comprises 19 design parameters (i.e. system size and position, and interactions) in total. Details of the input can be found in Appendix B.3.

The total available area in the four compartments ($201.6m^2$) is larger than the required area by the eight systems ($185m^2$). The developed method will be used to check whether a feasible distribution of the systems across the compartments is possible. The required interactions contain one directly conflicting, non-resolvable pair of interactions between systems A and B. The feasibility of either of these interactions and the impact on other design parameters will be evaluated. To investigate possible allocation

3



(a) Compartmentation for Case Study 3a



(b) Network of systems and interactions. Increasing node size corresponds to increasing system size. Square nodes indicate systems with global position and area requirements, while round nodes represent systems with an area requirement only. Blue and red edges indicate adjacency and separation interactions between connected systems respectively.

Figure 3.18: Visualisation of input to Case Study 3a

configurations the developed method is used to generate a set of 1000 concept solutions. The generation time is in the order of 6 seconds. This indicates that, for small design problems, solutions can be generated in real-time.

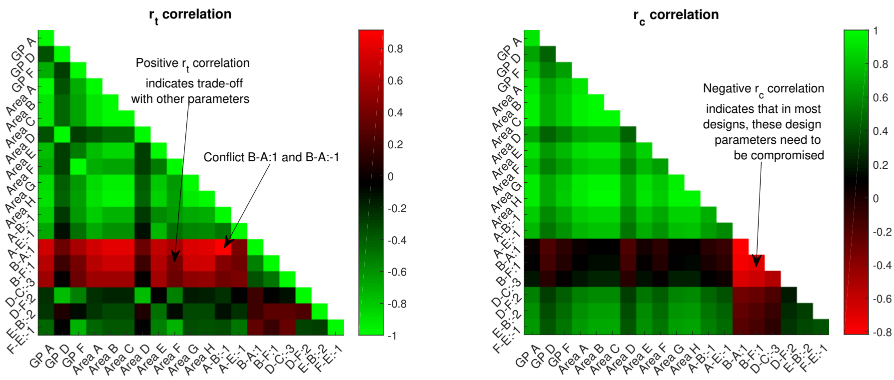
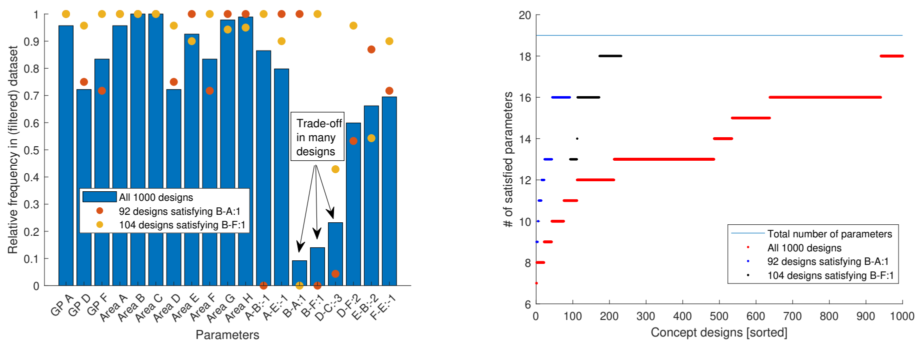


Figure 3.19: r_t and r_c correlation between 19 design parameters for Case Study 3a.

Figure 3.19 shows the r_t and r_c correlation between 19 design parameters. Most design parameter pairs are characterised by a negative r_t and a positive r_c correlation. This means that parameters in these pairs can likely both be satisfied. In contrast, there is a high positive r_t correlation between interactions B-A:1[7], B-F:1, and D-C:-3 and all other parameters. That is, there is likely a conflict between these parameters and all other

[7] Interaction format: system x-system y: interaction type, see Table 3.4.



(a) Relative frequency of design parameter satisfaction.

(b) Number of design parameters satisfied.

Figure 3.20: High level results for Case Study 3a.

parameters. Also, there is a significant negative r_t correlation between these three interactions, i.e. there is a conflict in the designs where these interactions are satisfied, that is, only one of the interactions in each pair is satisfied. Similarly, there is a strong negative r_c correlation between these parameters. That is, in most designs, these three interactions cannot be satisfied, regardless if considered individually (diagonal values) or in pairs (non-diagonal values). This is also shown in Figure 3.20a, where the length of each bar corresponds to the number of designs in which a design parameter is satisfied. For example, interaction B-F:1 is satisfied in only 14% of the designs.

These two observations indicate that these three interactions are most restrictive for the design space if these interactions need to be satisfied. Therefore, the consequences of satisfying these interactions are investigated further. For the sake of brevity, only interactions B-A:1 and B-F:1 are taken into consideration, yet the procedure would be similar for D-C:-3.

As indicated above, there is a conflict between interactions B-A:1 and B-F:1. The interactions require systems A, B, and F to be allocated to the same compartment. However, the total required area for these three systems is $90m^2$, which is larger than any available compartment. Hence, these two interactions can never be simultaneously be satisfied, unless the area requirements are compromised or the available space enlarged.

Although meeting any of these two interactions is a challenge, let's investigate the impact on the design space if B-A:1 and B-F:1 are to be satisfied separately. Figure 3.20a shows the relative frequency of design parameter satisfaction for all 1000 designs, as well as for the filtered set satisfying interaction B-A:1 (containing only 92 designs) and the filtered set satisfying interaction B-F:1 (containing only 104 designs).

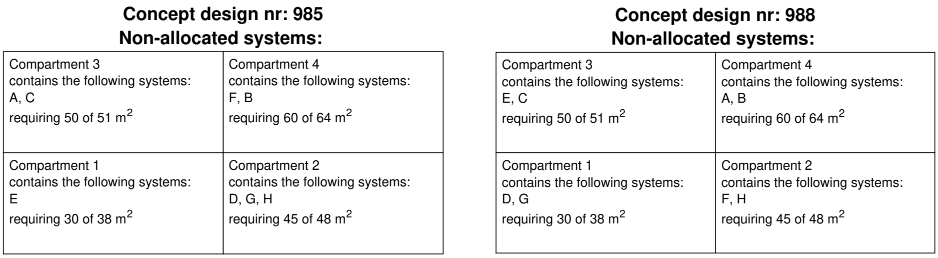
All designs satisfying interaction B-A:1, meet 9 of 19 parameters. Besides the conflict with interaction B-F:1, a conflict with interaction A-B=-1 becomes apparent, since there are no designs satisfying A-B:-1. This conflict can also be noted by the high r_t correlation in Figure 3.19. Note this was the conflict that was deliberately included in the input. Also, there are still a few designs in which interaction D-C:-3 is satisfied.

All designs satisfying interaction B-F:1, meet 8 of 19 parameters. Yet, selecting B-F:1 seems to be less restrictive on the overall design space than selecting B-A:1. This can be seen by the relative position of the data points in Figure 3.20a, where for 13 parameters the relative frequency is equal or higher if B-F:1 is selected.

Figure 3.20b shows the number of design parameters satisfied for all 1000 designs, as well as for the filtered set satisfying interaction B-A:1 and the filtered set satisfying interaction B-F:1.

- At maximum, 18 of 19 design parameters can be met. This is due to the deliberate (and unsolvable) conflict between interaction A-B:-1 and B-A:1.
- If B-A:1 needs to be satisfied, at maximum 16 design parameters are met, i.e. 3 design parameters cannot be met (interactions A-B:-1 and B-F:1, and one other parameter).
- If B-F:1 needs to be satisfied, at maximum still 18 design parameters are met, only interaction B-A:1 needs to be compromised. Hence, the selection for B-F:1 seems to be more promising, and therefore it could be decided to compromise interaction A-B:1.
- The selection of either of these interactions shows also positive trends, e.g. the global position of system D (GP D) is satisfied in relatively more designs. In other cases, parameters are relatively less frequently satisfied (e.g. GP F for B-A:1).

Finally, two of the 1000 concept designs are reviewed, which respectively satisfy interaction B-A:1 (Figure 3.21b) and B-F:1 (Figure 3.21a). Both designs satisfy the maximum number of satisfied design parameters found for these cases, i.e. 16 and 18 respectively. Based on these two concept designs, the available area in Compartment 1 seems relatively large, compared to Compartment 3. The former has $8m^2$ left, while the latter only has $1m^2$ after the allocation of systems.



(a) Layout ID 985, satisfying interaction B-F:1 and 18 parameters in total.

(b) Layout ID 988, satisfying interaction A-B:1 and 16 parameters in total.

Figure 3.21: Two layouts for Case Study 3a

Layout ID 985 satisfies all parameters, except for interaction B-A:1. This problem is not resolvable with the current compartment sizing, but can be resolved if a compartment is enlarged to $90m^2$, as explained above.

Layout ID 988 does satisfy interaction B-A:1 but does not satisfy the interactions A-B:-1 (not resolvable), B-F:1 (only resolvable with a sufficiently large compartment), and D-C:-3. The latter interaction requires systems C and D to be separated by a minimum radial distance of 1 compartment. This is not satisfied, since these systems are allocated in adjacent compartments. The most promising solution is to swap systems D and H. This would require compartment 2 to be $50m^2$, which is only $2m^2$ larger than its current size. Yet, this would keep Layout ID 988 inferior to Layout ID 985, because it would let the other two interactions unsatisfied.

As said above, both designs satisfy the maximum number of satisfied design parameters found by the allocation algorithm. The evaluation of the two selected layouts shows that the allocation algorithm indeed found the maximum *possible* number of satisfied design parameters for this case study.

In practice, concept designs that do not fulfill all requirements might not pass a design review. However, during early-stage design, the goal of design work is also to get insight into design drivers, feasibility, and risk (see Chapters 1 and 2). From that perspective, non-perfect concept designs (e.g. because of lack of detail, or because not all requirements are met) can still be useful to support the early stage stakeholder dialogue.

Lessons learned

There are three main lessons to be learned from Case Study 3a. First, the time required to analyse the data is significantly larger than the time required to generate the data. While the generation time is in the order of seconds, one can spend hours on the analysis of the data. One of the main reasons is that the case study had been conducted without a clear starting question. Additional iterations (e.g. to investigate the impact of enlarging a compartment or the impact of removing one or more unsolvable constraints) will likely be faster due to the more limited scope. During early-stage design, such iterations to find design insights are typical (Duchateau, 2016).

Second, understanding the meaning of the new correlation metrics takes time. However, it is likely that training and experience using the metrics will reduce this effort. Also, the new correlation metrics were found to be useful in identifying likely conflicts between design parameters.

Thirdly, the generation of the appropriate visualisations takes considerable effort and time. However, these can be reused when other data sets are analysed, as will be observed in Case Study 3b.

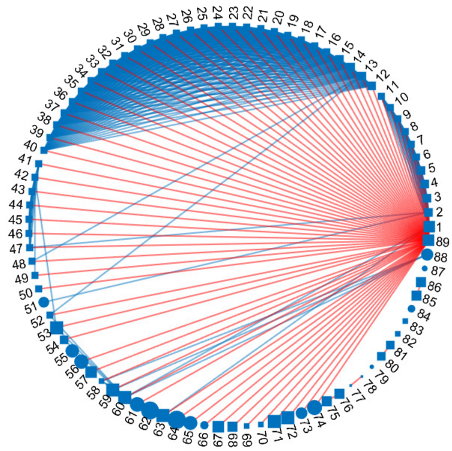
3.3.7. CASE STUDY 3B: OCEANGOING PATROL VESSEL

This section describes a full ship size allocation problem, to evaluate if and how large-scale allocation problems can be handled by the proposed method. The case study comprises 35 compartments, 89 systems (comprising 89 area and 75 global position requirements), and 575 interactions. This input is visualised in Figure 3.22. Details of the input can be found in Appendix B.3. The list of systems is based on the notional Oceangoing Patrol Vessel (OPV) presented in Chapter 5.

Table 3.7 summarises two runs of the allocation algorithm. The first run generated 1000 concept designs and the second run generated only 50 concept designs. What is clear, is the difference in required calculation time (decrease from 209 to 14 seconds), as well as accuracy (decrease from 92 to87%). In contrast to the first run, the second run might be representative of interactive design work from the perspective of calculation time. However, does the reduction of accuracy also yield a reduction of insight into

ID 29 60 m^2	ID 30 96 m^2	ID 31 96 m^2	ID 32 96 m^2	ID 33 96 m^2	ID 34 96 m^2	ID 35 60 m^2
ID 22 60 m^2	ID 23 96 m^2	ID 24 96 m^2	ID 25 96 m^2	ID 26 96 m^2	ID 27 96 m^2	ID 28 60 m^2
ID 15 60 m^2	ID 16 96 m^2	ID 17 96 m^2	ID 18 96 m^2	ID 19 96 m^2	ID 20 96 m^2	ID 21 60 m^2
ID 8 45 m^2	ID 9 72 m^2	ID 10 72 m^2	ID 11 72 m^2	ID 12 72 m^2	ID 13 72 m^2	ID 14 45 m^2
ID 1 30 m^2	ID 2 48 m^2	ID 3 48 m^2	ID 4 48 m^2	ID 5 48 m^2	ID 6 48 m^2	ID 7 30 m^2

(a) Compartmentation for Case Study 3b



(b) Network of systems and interactions. Node size is related to system size. Square nodes indicate systems with global position requirements. Blue edges indicate adjacency interactions. Red edges indicate separation interactions.

Figure 3.22: Visualisation of input to Case Study 3b

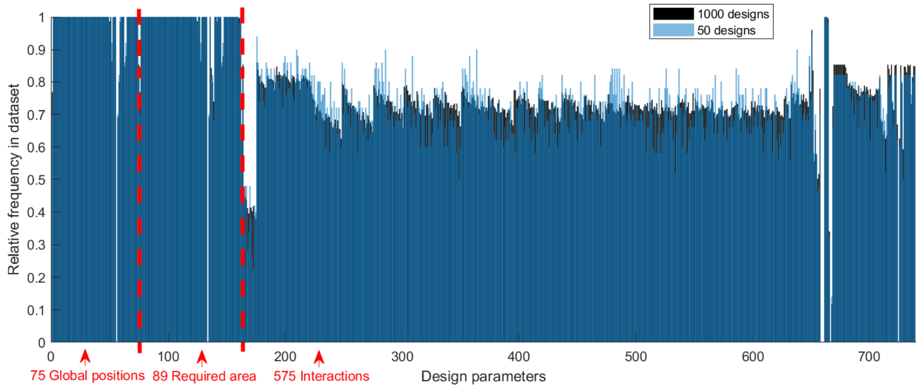


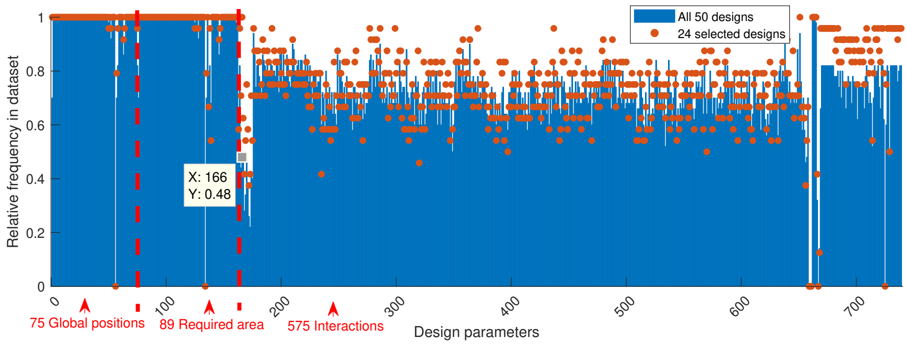
Figure 3.23: Relative frequency of design parameter satisfaction for Case Study 3b, for 1000 and 50 designs. For the sake of readability, the parameters haven't been labelled.

Table 3.7: Summary of results for Case Study 3b

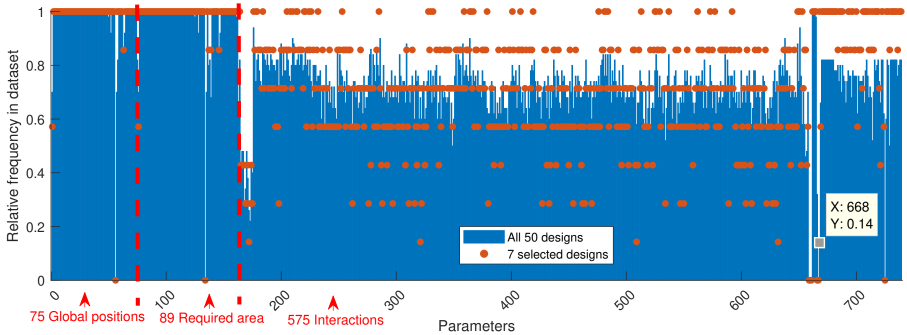
	Run 1	Run 2
Number of concept designs	1000	50
Calculation time [s]	209	14
Maximum number of parameters met	678	644
Percentage of total number of parameters (739) [%]	92	87

constraining design parameters?

This seems not to be the case. Indeed, Figure 3.23 shows the relative frequency of design parameter satisfaction for Case Study 3b for both runs. Assuming the first run



(a) Design parameter 166: interaction Commanders Cabin - 1 person Officers cabin:2



(b) Design parameter 668: interaction Waste store - Mess:1

Figure 3.24: Impact of two design parameters on design space. A grey box indicates the relative frequency of selected parameters. Orange dots indicate the relative frequency in filtered design space.

is most accurate, Figure 3.23 clearly shows for which parameters the second run overestimated (the light blue bar is visible) or underestimated (the black bar is visible) the satisfaction of design parameters. Although there are differences, the overall trend for each design parameter is comparable between the two runs. Hence, this is an indication that faster, lower-accuracy models might be used (although carefully) as a basis for collaborative design decision-making.

Next, the impact of two design parameters on the design space is evaluated. The two selected design parameters are 166 (interaction Commanders Cabin - 1 person Officers cabin:2) and 668 (interaction Waste store - Mess:1). The filtered design space for these interactions is shown in Figures 3.24a and 3.24b respectively. Some of the observations that can be made are:

1. Selecting parameter 668 yields the largest reduction of the design space, to 7 designs. Parameter 166 yields 48% of the original design space.

2. The spread in design parameter satisfaction is larger for parameter 668 than for parameter 166.

3. There is no direct conflict between these two parameters since both filtered sets contain designs that still satisfy the other parameter.

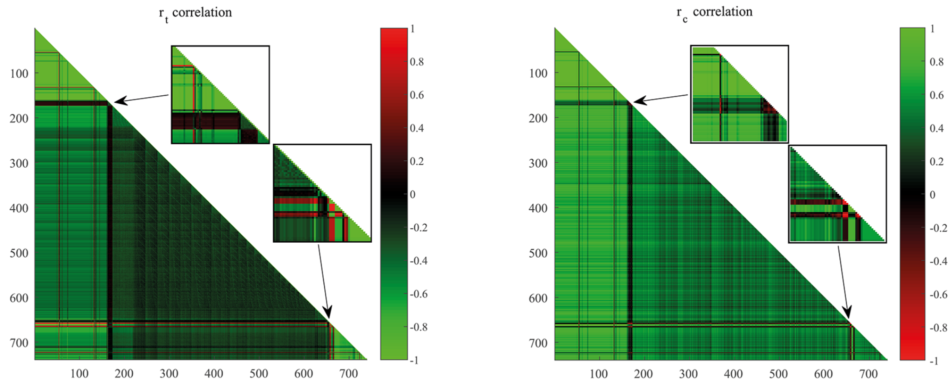


Figure 3.25: r_t and r_c correlation between 739 design parameters for Case study 3b.

Figure 3.25 shows the r_t and r_c correlation between the design parameters in Case Study 3b. The insets show more detailed views of particular parts of the design space. What stands out from the overall correlation map is the correspondence with Figure 3.24. For instance, areas where $r_t \approx 1$ and $r_c \approx -1$ (i.e. visible line patterns) coincide with low relative frequency areas in Figure 3.24. This gives confidence that the new correlation metrics are also applicable to more elaborate design problems.

Figure 3.26 shows one of the generated concept designs in Case Study 3b. Such allocation of systems might be used by a naval architect as a starting point for the further development of a detailed General Arrangement Plan, or be used in WARGEAR to automatically generate a 2D layout plan.

Lessons learned

There are three lessons to be learned from Case Study 3b. First, setting up the design problem requires significant effort. This is not expected to be a major issue in the context of real-time collaborative design sessions. Indeed, designers will likely prepare models, etc. *prior* to the sessions (Bandecchi et al., 2000). It has been seen that the time required for the generation of concept designs is still relatively low, which indicates that automated design tools might be useful, even for large design problems.

Second, the new correlation metrics provide results in line with other data derived from system allocation. However, the metrics do not provide insight into the extent that parameters are met. For instance, can a system, which currently cannot be allocated, be allocated with 95% of its currently required area? Currently, these variations are not evaluated in the allocation process. Instead, the designer is required to alter the input to investigate such variations.

Third, the availability of the visualisations in Case Study 3a led to a perceived decrease in time and effort required for analysing the substantially larger data set obtained in Case Study 3b. A more elaborate use of dedicated data exploration tools might help to

Compartment 29 contains the following systems: Intel room, NBCD filter rooms requiring 17 of 60 m^2	Compartment 30 contains the following systems: Officers cabins , Engineering office, HVAC rooms requiring 84 of 96 m^2	Compartment 31 contains the following systems: Officers cabins , Officers cabins , Workshop mechanical and welding, Workshop electrical requiring 89 of 96 m^2	Compartment 32 contains the following systems: Dayroom officers, Medical area requiring 52 of 96 m^2	Compartment 33 contains the following systems: Officers cabins, Radio central, Briefing room requiring 68 of 96 m^2	Compartment 34 contains the following systems: Computer room, Main switchboard requiring 96 of 96 m^2	Compartment 35 contains the following systems: Sanitary & showers , Fresh water maker 1 requiring 22 of 60 m^2
Compartment 22 contains the following systems: Meeting room requiring 48 of 60 m^2	Compartment 23 contains the following systems: Officers cabins , Officers cabins , Officers cabins , Compass room, Fresh water maker 2 requiring 67 of 96 m^2	Compartment 24 contains the following systems: Officers cabins, Dry stores requiring 86 of 96 m^2	Compartment 25 contains the following systems: Officers cabins, Fitness room, Baggage storage, Fire fighting room requiring 68 of 96 m^2	Compartment 26 contains the following systems: Command central requiring 96 of 96 m^2	Compartment 27 contains the following systems: Commanders Cabin, Officers cabins , NBCD filter rooms requiring 75 of 96 m^2	Compartment 28 contains the following systems: Bridge requiring 60 of 60 m^2
Compartment 15 contains the following systems: Chiller unit aft requiring 20 of 60 m^2	Compartment 16 contains the following systems: Rating cabins, Computer room requiring 78 of 96 m^2	Compartment 17 contains the following systems: Rating cabins, Dayroom petty officers requiring 70 of 96 m^2	Compartment 18 contains the following systems: Petty officers cabins, Petty officers cabins, Petty officers cabins requiring 42 of 96 m^2	Compartment 19 contains the following systems: Petty officers cabins , Petty officers cabins, Petty officers cabins, Petty officers cabins, Petty officers cabins, Petty officers cabins requiring 85 of 96 m^2	Compartment 20 contains the following systems: Petty officers cabins , Petty officers cabins, Petty officers cabins, Petty officers cabins, Petty officers cabins, Compass room requiring 83 of 96 m^2	Compartment 21 contains the following systems: Petty officers cabins , Petty officers cabins, Petty officers cabins, Petty officers cabins requiring 57 of 60 m^2
Compartment 8 contains the following systems: Rating cabins, Emergency switchboard requiring 20 of 45 m^2	Compartment 9 contains the following systems: Mess requiring 67 of 72 m^2	Compartment 10 contains the following systems: requiring 0 of 72 m^2	Compartment 11 contains the following systems: Petty officers cabins, Rating cabins, Sanitary & showers , Laundry requiring 68 of 72 m^2	Compartment 12 contains the following systems: Petty officers cabins, Petty officers cabins, Petty officers cabins requiring 42 of 72 m^2	Compartment 13 contains the following systems: Petty officers cabins , Petty officers cabins, Petty officers cabins, Petty officers cabins, Petty officers cabins requiring 71 of 72 m^2	Compartment 14 contains the following systems: Petty officers cabins, Petty officers cabins, Hydrophore requiring 38 of 45 m^2
Compartment 1 contains the following systems: Rating cabins, Bakey requiring 21 of 30 m^2	Compartment 2 contains the following systems: Rating cabins, Main switchboard requiring 42 of 48 m^2	Compartment 3 contains the following systems: Engine room requiring 48 of 48 m^2	Compartment 4 contains the following systems: requiring 0 of 48 m^2	Compartment 5 contains the following systems: Rating cabins, Sanitary & showers requiring 25 of 48 m^2	Compartment 6 contains the following systems: Dayroom rating, HVAC rooms requiring 42 of 48 m^2	Compartment 7 contains the following systems: Chiller unit forward requiring 20 of 30 m^2

Figure 3.26: Example of allocation of systems to compartments in Case Study 3b (Layout ID 942).

get easier insight into the vast amount of data produced in the allocation process.

3.3.8. CONCLUSIONS FROM CASE STUDY 3

This case study aimed to investigate how design tools for ship layout design can be used in a real-time manner. That is: 1) how can such tools be used to generate concept designs in real-time, and 2) how can these concept designs be analysed in real-time? As an example problem, the allocation of systems to compartments was considered. WARGEAR's space allocation algorithm was adapted and extended. A new two-item correlation metric was developed to support designers in identifying conflicts, and hence necessary trade-offs, between design parameters.

Based on two case studies, the calculation time or accuracy of the allocation algorithm does not seem to be the main issue for collaborative design decision-making. Most effort is required for the analysis of the data - which is not a problem for real-time collaborative design as such but needs to be considered when selecting tools and methods for such design sessions. However, it is beneficial to use design tools with a specific goal or inquiry in mind, as this will enhance the search for insights into the design space. Hence, the development of interactive data exploration and decision tracing seems to be a promising and essential research direction to support collaborative design decision-making.

3.4. CASE STUDY 4 - DESIGN OF A LANDING PLATFORM DOCK

This section provides an additional application of WARGEAR. The case study was part of a combined study, aiming at integrating detailed layout generation and logistic performance assessment (Droste & le Poole, 2020). The detailed layout generation was achieved by using WARGEAR, while the logistic performance assessment was conducted via Droste et al. (2020)'s queueing-based method. To achieve the integration, WARGEAR was extended such that, for selected layouts, it semi-automatically generates a network of queueing architecture elements for logistic performance assessment.For the sake of brevity, the logistic performance assessment is not further elaborated on here but can be found in Droste and le Poole (2020).

3.4.1. CASE STUDY SETUP

To demonstrate how this method supports naval architects during the early-stage design of internal layout and process-driven ships the design of a notional LPD is considered as a case study. Specifically, the test case was aimed to provide insight into possible bottlenecks in the operational processes and the relation of these bottlenecks to the layout. LPDs are used to transport marines with their vehicles, weapons, and equipment over large distances, and eventually deploy them to a designated location (e.g. a beach) via smaller craft and helicopters. Hence, LPDs are considered to be internal layout and process-driven, because of the movement of marines during transit (e.g. they need to get their meals three times a day) and the deployment (e.g. also involving the transport of equipment and vehicles).

Some of the overall requirements for the notional LPD, which include sizing and manning requirements, are listed in Table 3.8. Also sizing requirements for logistic systems, i.e. staircases and passageways are provided, as well as additional sizing requirements for individual spaces. The manning requirements combined with the space sizing and capacity requirements define the required number of cabins aboard the LPD.

Table 3.8: Requirements for the notional LPD.
[1] Commanding Officer. [2] Officers. [3] Non-Commissioned Officers. [3] marked spaces are *not* arranged by WARGEAR. [4] The number after the abbreviation represents the capacity of the cabin.

Sizing requirements		
L_{oa}	120 m	
B_{oa}	24 m	
Compartment length	15 m	
Manning requirements		
Rank	Number	
CO [1]	1	
OFF [2]	16	
NCO [3]	22	
Ratings	60	
Marines	360	
Logistic system requirements		
	Length	*Width*
Main passageway	[-]	2 m
Secondary passageway	[-]	1.2 m
Main staircase	3 m	3 m
Secondary staircase	3 m	0.8 m

Space name	**Length**	**Width**	**Area**
Medical facilities			
Operating room (2x)	[-]	[-]	50
Triage	[-]	[-]	40
Storage (2x)	[-]	[-]	30
Changing room	[-]	[-]	25
Operational spaces			
Briefing room	[-]	[-]	130
Weapon hand out	[-]	[-]	50
Dock [3]	30	13.9	[-]
Vehicle deck [3]	60	13.9	[-]
Helicopter deck [3]	60	13.9	[-]
Helicopter hangar [3]	60	13.9	[-]
Accommodation cabin area[4]			
CO-1	[-]	[-]	20
OFF-2	[-]	[-]	12
NCO-2	[-]	[-]	10
RAT-4	[-]	[-]	15
MAR-8	[-]	[-]	20
Accommodation service area			
Galley	[-]	[-]	115
Mess	[-]	[-]	125

Based on the two sets of requirements an initial functional arrangement for the notional LPD has been generated, and is shown in Figure 3.27. Note that this functional arrangement is still incomplete, e.g. engine rooms, exhaust stacks, and fuel tanks, amongst others, have not been implemented, and not all available area has been utilised. How-

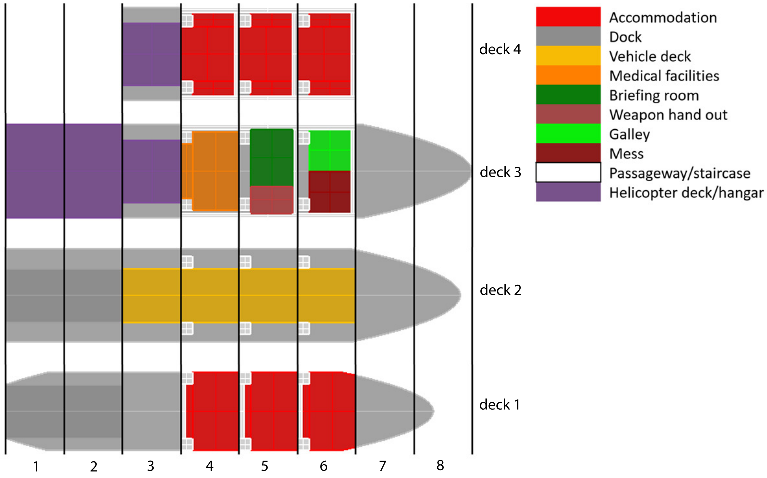


Figure 3.27: A functional arrangement of the notional LPD.

ever, since the purpose is to demonstrate an integrated detailed layout generation and evaluation method, this functional arrangement was found to provide sufficient information to enable operational analysis and to enable insight into the interrelations between the layout and operational processes.

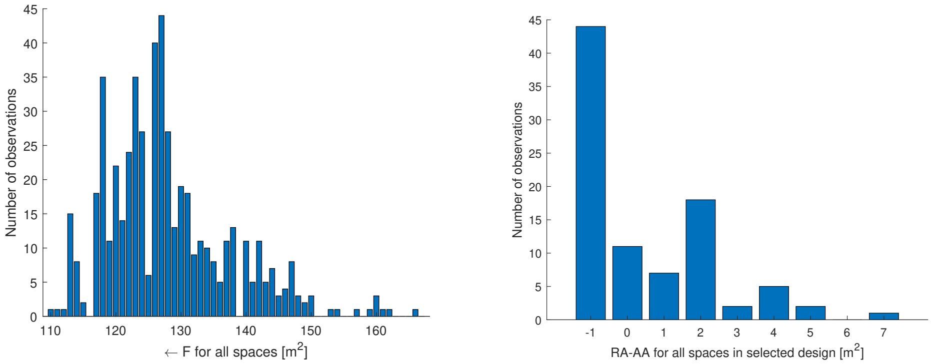
3.4.2. LAYOUT GENERATION

In total 520 detailed layouts have been generated in approximately 20 minutes by WARGEAR. WARGEAR had to generate 90 spaces in each layout. In Figure 3.28a a histogram of the objective scores F for all generated layouts, see Section 2.4.8, is given. Since the objective function is minimised, the most preferred detailed layout, from an area performance point of view, corresponds to the most left bin in the histogram. In this run, the most left bin contains only one detailed layout. Although this detailed layout still falls 110 m^2 short of the total required area of all spaces, this layout will be further investigated because only 1.2 m^2 per space is missing on average.

Additionally, a histogram of the discrepancy between the RA and Achieved Area (AA) for each space in this layout is given in Figure 3.28b. This histogram shows that most spaces meet or even overshoot their required area, while only a few spaces are problematic, e.g. one space falls 7m^2 short of its RA. The latter spaces might prove challenging to be corrected manually.

3.4.3. LAYOUT EVALUATION

Subsequently, each deck is further analysed, starting at the uppermost deck. Since the deck 2 has not been further arranged by WARGEAR, this deck is not shown here. At deck 4, shown in Figure 3.29a, most spaces meet their RA. However, to ensure connectivity of all spaces to passageways (shown in light grey), area is reserved from spaces. For instance, area is reserved from spaces 51 to 56 in compartment 4. Since there is also area available and some spaces are redundantly connected, e.g. spaces 53 and 54, naval ar-



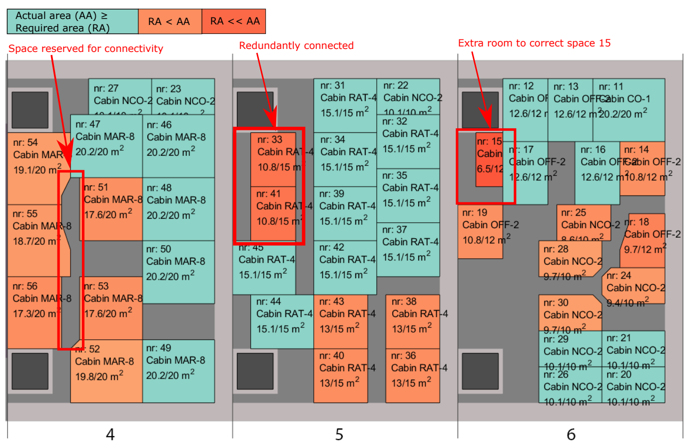
(a) Histogram of the objective scores for all 520 generated designs

(b) Distribution of the discrepancy between required area (RA) and actual area (AA) for the selected design

Figure 3.28: Performance of layouts in Case Study 4.

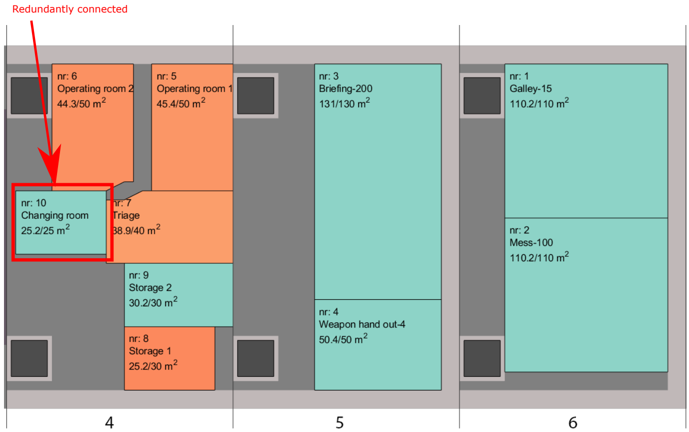
chitects are likely able to manually change the arrangement of the aft compartment such that all spaces meet their required area. Similarly, in compartment 5, spaces 33 and 41 are redundantly connected, and the available area aft of these spaces can be used to manually correct the layout generated by WARGEAR. Likewise, compartment 6 has significant area available to solve the insufficient area of space 15, for instance. To show how naval architects might translate a detailed layout generated by WARGEAR into a more feasible GAP, Figure 3.30 shows the manual arrangement of deck 4.

The detailed arrangement of deck 3 is shown in Figure 3.29b. Although some medical facilities fail to meet their RA, sufficient area is available to manually produce an

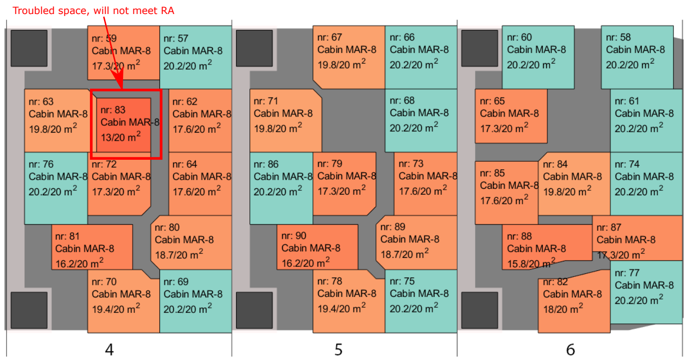


(a) Deck 4

Figure 3.29: A more elaborate study of each arranged deck of the selected detailed layout.



(b) Deck 3



(c) Deck 1

Figure 3.29: A more elaborate study of each arranged deck of the selected detailed layout - continued.

improved layout. Also, the amount of available area indicates that the Medical Facilities functional block, as shown in Table 3.8 and Figure 3.27, has been oversized. Therefore naval architects might utilise some of the area in compartment 4 for other purposes, e.g. to arrange exhaust stacks.

Figure 3.29c shows the arrangement of deck 1. Compartments 4 and 5 are almost equally arranged. Although it is not immediately clear how to manually change the arrangement of spaces such that all spaces meet their required area, space 83 in compartment 1 will not meet its RA. The arrangement on deck 4 allows additional spaces to be arranged, certainly in compartment 6. Therefore space 83 might be reallocated to deck 4 to improve the overall objective score of the layout. Thus, the layout generation provides insight that on the lowest deck, a maximum of eleven MAR-8 cabins of $20m^2$ can be arranged. To further improve the detailed layout plans for the given functional arrangement, the gained insight can be used to improve the allocation of spaces to compartments, which is part of WARGEAR's input.

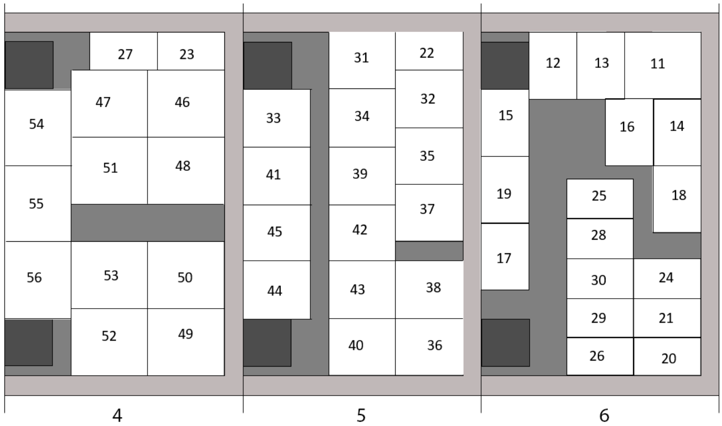


Figure 3.30: A manually created layout plan of deck 4, based on the layout generated by WARGEAR. All spaces meet their required area in the improved layout.

Based on the queueing method results, logistical waiting time takes place in roughly the same passageways for all simulations. To help naval architects understand where this logistical waiting time is created, the network representation of the layout has been visualised overlaying the actual layout in Figure 3.31. The locations in the network where waste is created are highlighted, while the marker size indicates how much waiting time is created. Also, the functional spaces are indicated. The following observations can be made using using this figure:

1. Most functional and logistical waiting time is created on deck three, in compartments 5 and 6.
2. The entrance of the briefing room and the weapon handout are located at the same location. This might cause a logistical bottleneck.
3. Significant functional waiting time is created in the mess itself. This might indicate that the capacity of the mess is insufficient to serve the 360 marines in the LPD.
4. Most logistical waiting time is created in the transverse passageway near the mess. This might be caused by the limited capacity of the mess, but also by insufficient space in the passageway itself.

Besides demonstrating the proposed method, the test case was aimed to provide insight into possible bottlenecks in the operational processes and the relation of these bottlenecks to the layout of the notional LPD. These insights point towards several improvements to the design. However, such changes should be thoroughly analysed to assess whether such changes would lead to other issues in the execution of the operational processes. The insight gained from the generation and analysis, as well as some design improvements, are summarised below:

1. On deck 1 only eleven MAR-8 cabins can be arranged per compartment. This insight helps improve the allocation of spaces to increase the feasibility of the layout

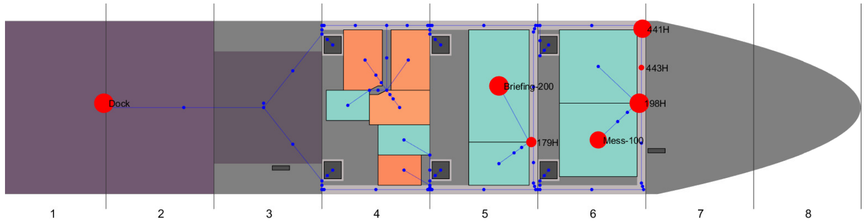


Figure 3.31: Arrangement of deck 3 in the selected layout with overlaying network representation, as well as an indication of locations where functional or logistical waiting time is created (red circles), based on the output of the queueing method.

from an area point of view.

2. The entrance of key spaces, such as the briefing room, should be separated to reduce the logistical load on the passageways around these spaces.

3. The capacity of the mess, 100 marines, is insufficient to serve all 360 marines at once, which causes the briefing to be delayed and leads to a high logistical load on passageways. The capacity of the mess could be increased, or the processes of entities could be planned such that the load on the mess is reduced.

4. Passageways on deck 3 around the mess are logistical bottlenecks. This is caused by the limited capacity of the mess, although the width of passageways might have an impact as well. The latter should be thoroughly studied, by varying the width of passageways. However, this might impact the feasibility of the layout from an area perspective. Hence, the interrelation between the processes and the layout.

3.4.4. CONCLUSIONS FROM CASE STUDY 4

The goal of this case study was to demonstrate how integration between WARGEAR and a queueing-based logistic performance assessment supports naval architects during the early-stage design of internal layout and process-driven ships. The design of a notional LPD is considered as a case study. WARGEAR was used to generate a set of detailed layouts in approximately 20 minutes. These layouts could be used to identify sizing and integration issues. Furthermore, the generated layouts could be used as a basis for further manual arrangement. In addition, the results of the queueing-based method could be combined with the detailed layouts to provide insight into possible bottlenecks in the operational processes and the relation of these bottlenecks to the layout of the LPD. These results support the earlier conclusions about WARGEAR's ability to support early-stage complex ship layout design.

3.5. CONCLUSION

This chapter described three case studies applying WARGEAR and one case study extending WARGEAR's space allocation algorithm. Using the outcomes of these case studies the second research question: *To what extent can automated layout generation methods support real-time design decisions during early stage complex ship design?* can be further answered.

1. Case Study 1 (Section 3.1) shows that WARGEAR can be used to gain insight into sizing and integration issues. Also, these design insights can be used in subsequent design iterations - i.e. WARGEAR can be used in a rapid interactive manner. This was demonstrated via a notional surface vessel. After generating a set of design solutions using WARGEAR, a detailed layout was selected and analysed. Based on the analysis, the input to WARGEAR was adapted and an improved set of layouts was generated in approximately 15 minutes.

2. Case Study 2 (Section 3.2) shows that WARGEAR is able to generate design insights in a limited time (in the order of minutes) for a realistic design case. This exercise also showed that, in less time, more design variations can be studied using WARGEAR compared to a human designer using regular CAD software. Yet, the outcomes of the WARGEAR and CAD-based design studies were comparable. This suggests WARGEAR can be used in an iterative, supportive manner, as discussed in Section 2.1.1 and shown in Figure 3.32.

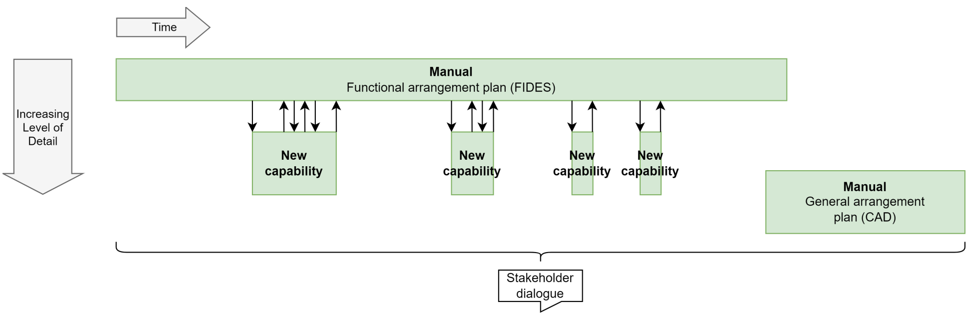


Figure 3.32: Future state with new capability (applied to the DMO case). Copied from Figure 2.2b.

3. While the first two case studies focused on validation of WARGEAR, Case Study 3 (Section 3.3) extended a part of the WARGEAR algorithm to investigate how automated design tools for ship layout design can be used in a real-time manner. The results show how insight can be gained into complex and incomprehensible interrelationships between systems and the overall ship layout in a relatively limited time via a new correlation metric. In addition, it was found that the calculation time can be minimal, even for complex problems, but most effort is required for input generation and post-processing of the results. Tailor-made visualisation tools can help designers to identify items of interest faster.

4. Case Study 4 provided an additional application of WARGEAR. It was integrated with a queueing-based logistic performance assessment to support naval architects during the early-stage design of internal layout and process-driven ships. An application to a LPD confirmed WARGEAR can be used to identify sizing and integration issues early on. In addition, the results of the queueing-based method

could be combined with the detailed layouts to provide insight into possible bottlenecks in the operational processes and the relation of these bottlenecks to the ship's layout.

Table 3.9 summarises WARGEAR's compliance with the method requirements given in Section 2.1.1. The case studies presented in this chapter contributed to the fulfilment of requirement LRQ1 and 2, as well as additional confirmation of the other requirements.

Table 3.9: WARGEAR: compliance with method requirements.

ID	Requirement	Compliance WARGEAR
LRQ1	The method should provide insight into potential sizing and integration issues and risks and ways to solve these issues and reduce the risks.	Fulfilled (Sections 2.4, 3.1 and 3.2)
LRQ2	Its speed should be in the order of minutes	Fulfilled (Sections 2.4, 3.1-3.3)
LRQ3	The starting point is a predefined functional arrangement	Fulfilled (Sections 2.4.2 and 3.1)
LRQ4	The level of detail of generated layouts should be high	Fulfilled (Sections 2.4.6, 2.4.7, 3.1 and 3.3)
LRQ5	The main driver for layouts to be considered is area	Fulfilled (Sections 2.4.6, 3.1 and 3.3)
LRQ6	The number of diverse solutions should range from a few to hundreds	Fulfilled (Sections 2.4.8 and 3.1)

Chapter 1 pointed out that documentation of design decisions is essential to understand the development of concept designs. However, WARGEAR does not facilitate the documentation of design decisions. Therefore, designers rely on means to capture design rationale separately from the design tool, potentially leading to a lack of the context of decisions (Pawling, 2007, p.130). On-the-fly design rationale capture and reuse will be the topic of Part II of this disseration.

II

On Design Rationale in Ship Layout Design

This part focuses on capturing design rationale during early-stage complex ship layout design. Chapter 4 proposes and evaluates a proof-of-concept design rationale method. Subsequently, this method is extended in Chapter 5. The aim is to enable designers to capture and reuse design rationale on-the-fly during early-stage layout design.

4

PROOF OF CONCEPT DESIGN RATIONALE METHOD

It is this closely linked chain of connection between the whole of the principles on which the qualities desirable for a ship depend, that renders naval construction so essentially a science of analogies and comparisons. Scarcely any one point in the design of a ship can be considered for perfecting abstractedly, without also involving a compensating sacrifice in some other point, equally essential in its nature to the perfection of the whole. Our object has been to endeavour to combine, in as great a degree as the present extent of knowledge will admit, the requisites which we have enumerated as forming an efficient ship-of-war.

The Chatham Committee of Naval Architects (1842)

In Chapter 1 the need for a design rationale method, suitable for on-the-fly design rationale capture and reuse, was identified. Hence, the main research question for Part II is "*To what extent can design rationale methods support real-time design decisions during early-stage complex ship design?*" To answer this question, this chapter presents a literature review of design rationale methods for layout design and identifies the research gap in Section 4.1. Then, Section 4.2 presents a new design rationale method for ship layout design. This method is subsequently tested in an experiment. The experimental setup is described in Section 4.3, results are presented in Section 4.4 and discussed in Section 4.5.

4.1. GAP ANALYSIS

4.1.1. PROBLEM IDENTIFICATION AND METHOD REQUIREMENTS

Design rationale was introduced as an essential, yet challenging design aspect in Chapter 1. In this section, first, several definitions for design rationale are presented to gain a

Parts of this chapter are based on Le Poole et al. (2022c) and Le Poole et al. (2023).

better understanding of what design rationale entails.

MacLean et al. (1989) state:

> "*A design rationale is not a record of the design process – it is a co-product of the design along with the artefact and itself has to be designed.*"

Klein (1993) describes design rationale as:

> "*The underlying intent and logical support [...] for the decisions.*"

Lee (1997) states:

> "*Design rationales are important tools because they can include not only the reasons behind a design decision but also the justification for it, the other alternatives considered, the tradeoffs evaluated, and the argumentation that led to the decision.*"

Ball et al. (2001) define design rationale to

> "*encompass the documentation of the active processes of reasoning and decision-making that led to the artefact design — including the justification for design decisions, records of design alternatives considered and trade-offs evaluated, and details of the argumentation and communication processes associated with design work.*"

Tang et al. (2006) state:

> "*Design rationale captures the knowledge and reasoning that justify the resulting design.*"

DeNucci (2012) uses the following definition:

> "*Design rationale captures the reasoning behind design decisions.*"

In this dissertation, design rationale is defined as follows:

> "*Design rationale explains and justifies the decisions leading to a concept design.*"

These definitions differ but have multiple common elements. In principle, design rationale is about the justification of design decisions (DeNucci, 2012). Therefore, design rationale can be used by the designer to understand past design decisions. This is important when reworking or validating the concept design. For this reason, ship design decisions and calculations were noted in a Book of Calculations in the past, which was signed off when approving a ship design, as Andrews (2021) describes. He adds that, nowadays, computer programs and spreadsheets are used by designers to make calculations and generate designs. Part of the design rationale, therefore, is integrated into these tools. However, both assumptions and sources of information should be noted (Andrews, 1986). Such rationale is to include the major design drivers, i.e. main design criteria with the highest size and cost impact (Duchateau, 2016).

In the current state of the ship design process, a pre-generated concept design is typically discussed and evaluated during a design session. Also, design changes might be proposed and discussed during the session. However, these changes are typically processed outside the session. This is in part because of the lack of sufficiently fast and flexible design tools (e.g. see Section 2.1). This is the reason Part II not only focuses on space-level decisions but also aims at supporting functional-level decisions. Note that WARGEAR might be applied during a design session (Part I), although design changes are likely to be proposed on the level of the functional arrangement during early-stage design. Therefore, a design session is often a design *review* session. Figure 4.1 shows how a design session relates to generative and documentation activities. The following principle design activities are discussed:

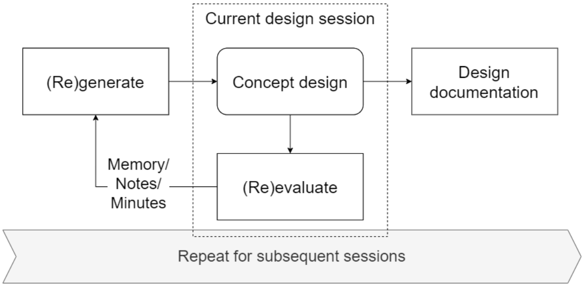


Figure 4.1: Representation of activities around current design (review) sessions.

1. **Generation**. Prior to design sessions, designers generate concept designs based on predefined or assumed requirements and preferences, to provide insight into the consequences of these requirements (Van Oers et al., 2018).

2. **Evaluation**. During the design session, two- or three-dimensional views and performance evaluations (e.g. speed, stability, etc.) of the concept design are evaluated and result in the expression of additional or revised requirements and preferences. Designers rely on experience and reasoning to estimate whether these preferences can actually be implemented and how that might be done. It's only when the designer returns to the drawing table (i.e. *Generation*) that the actual impact of the design decisions becomes known. Often multiple variations to solve a problem might be a way to overcome the "assumptions". For example, a stability problem could result in several options being investigated for the next step (e.g. a wider hull, lower centre of gravity (CoG), or a combination of both).

3. **Design documentation**. Typically, design documentation is done via reconstruction of the concept design, with the support of minutes of meetings, notes, and the designer's memory (Section 1.1.3). However, capturing design rationale separately from design tools leads to a (partial) lack of context of decisions, since these notes can not be directly associated (via a software link) with corresponding objects in the design (Pawling, 2007, p.130). Also, Pawling (2007) experienced that the success of a separate logbook is a function of the user's conscientiousness to use it consequently. This is an issue, both in case insight into the design process is needed and when rework needs to be performed (as elaborated in Section 1.1.1).

An explicit focus on design rationale in early-stage ship design is missing, partly because design rationale itself is not a direct deliverable in the ship design process (DeNucci, 2012). Indeed, the main focus lies on the concept design (in the form of a variety of drawings and calculations) and a consistent set of requirements (DeNucci, 2012; van Oers et al., 2018). To enhance this situation, Pawling et al. (2017) states that

> "*the provision of an automatically generated logbook [has been long seen as desirable in computer-aided ship design]. This would provide inexperienced designers with an invaluable tool when reviewing and reassessing previous design decisions and provide an audit trail when used in anger. Given the interrelation between so many functions, such audit trails would be particularly helpful in identifying sources of unexpected or undesired changes to the model and further assist in better design assurance in early design decision-making.*"

Such a tool is not only invaluable for inexperienced designers, but also experienced designers could benefit from such developments. Yet, such a logbook would only record the consequences of decisions - not their supporting rationales.

To address this issue, a logical proposal is to enable the capture of design rationale within design tools to capture the context of decisions. This would address the issue with the current design documentation described above. If concept designs are reviewed outside the design tools (e.g. via a presentation), the proposal could require designers to capture design rationale twice. Indeed, first, notes need to be taken during design sessions, and second, these notes need to be captured in the design tool. To address this consequential issue, a subsequent proposal is to consider a more integrated process of concept design generation, evaluation, and documentation.

The nature of early-stage ship design is dynamic and exploratory since it requires the elucidation of requirements via the generation and discussion of (creative) concept designs with stakeholders (Section 1.1.1). Dorst and Cross (2001) describes creative design to be the intertwined co-development of problem formulation and solution ideas, which reflects the nature of requirements elucidation in early-stage complex ship design (Andrews, 2018b). According to Fischer and Shipman (2011), "cultures of participation [allowing users to become active contributors to design], supported by socio-technical environments, have the potential to exploit the opportunities provided by the synergy of collective design rationale and social creativity." So, if design rationale can be integrated with collaborative design sessions, supported by suitable design tools, the collective problem-solving power might be capitalised on in ship design. For the sake of this dissertation, the focus is again on ship layout design.

Hence, to address the issues involving current design sessions and to take advantage of the benefits of collaborative design, a new design process is proposed. Figure 4.2 shows this proposed design process, which comprises the following activities:

1. **Generation**: Concept design generation will be partly preparatory to the design session. However, (semi-automated) concept generation tools might be used during the design session to generate concept designs in real-time to support the ongoing dialogue with up-to-date concept designs.

2. **Evaluation**: Like the traditional design session, designers evaluate the value of generated concept designs, and propose changes, express preferences, etc. The

in-session generation of concept designs enables real-time evaluation of the feasibility of these changes, see Sections 3.3 and 4.2.

3. **Design rationale**: During the *Generation* and *Evaluation*, *Design rationale* is captured and stored in a database for future access. This design rationale can be retrieved to evaluate past decisions, used in the evaluation (e.g. Measures of Performance (MoPs)), or in layout generation tools, as proposed in Section 4.2.

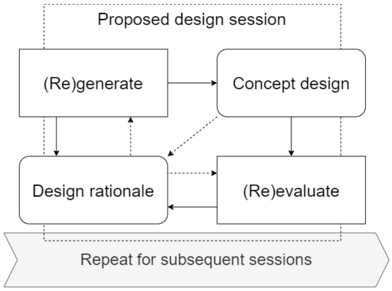


Figure 4.2: Representation of activities in the proposed design process, with a focus on design rationale being captured and reused during design sessions.

The proposed reuse of the captured design rationale might improve the benefits of using the design rationale method, such as improved communication en decision documentation. The proposal can be compared to shared display to support the generation of negotiated knowledge in a stakeholder dialogue (Section 1.1.1). However, on the one hand, it does not fully eliminate the need for manual input of rationales. On the other hand, an integration of design rationale and ship layout design tools would reduce manual work, since the “What” (i.e. the concept design and its underlying system properties) is typical data processed in design tools. Also, data coherency could be automated, again reducing manual work.

There are various use cases for design rationale. After design rationale is captured, it may be used in various ways:

1. Primary, captured design rationale can be used by designers downstream of the design process to check whether past decisions are still valid.

2. Using system interrelationships to identify design drivers prior to arranging systems in a layout (Gillespie, 2012; Pawling et al., 2015). Network centrality metrics were used to identify the systems with (potentially) the highest impact on the layout.

3. Using system interrelationships to generate concept designs (Gillespie et al., 2013). Predefined networks can be used as input for the actual arrangement process (Andrews, 1986; Esbati, 2018). These examples show that it is possible to (automatically) generate layouts that contain, for instance, a minimal number of unsatisfied required adjacency and separation interrelationships. A network approach is also used to design distributed ship service systems for combatants (Duchateau et al., 2018; Habben Jansen, 2020) and submarines (Mukti et al., 2022).

4. Using system interrelationships to evaluate generated concept designs (Pawling & Andrews, 2018; Roth, 2016; Sun, 2019). Similar to above, network centrality metrics were used to rank concept designs (Roth, 2016) and to identify key systems (Pawling & Andrews, 2018). Pawling and Andrews (2018) stress the importance of node compliance to the actual systems in the concept design for understandability of centrality values and proposes to use centrality metrics to sort design options for design support.

Since there are various use cases for design rationale, the key question is how these use cases can be enabled during collaborative design, as envisioned above. Indeed design rationale methods do not yet fully align. Therefore, the challenges of design rationale, discussed in more detail in Chapter 1, need to be tackled. In short, the four challenges discussed are:

1. Design rationale methods can be perceived to be not cost-effective.
2. Designers can be reluctant to take the time to document the decisions they did not take or took and then were rejected.
3. Such methods can be intrusive in the design process.
4. There can be a social barrier to documenting decisions.

To address the problem outlined above and enable the envisioned future state, a design rationale method enabling in-situ design rationale capture during ship layout design needs to comply with the following requirements[1]:

DRQ1. **The method must be applicable for early-stage collaborative design activities and promote feedback-driven conversations**, that is, it is to support the creation and capture of negotiated knowledge.

DRQ2. **The method must enable the capture and review of design decisions, the rationale behind these decisions, and temporal relationships between design decisions**, i.e. it must capture *what* is changed, *how*, *why*, and *when*. That is, the method should enable the capture of both the decisions made by designers and the context of these decisions. The latter is essential for retrieval purposes.

DRQ3. **The method must provide immediate rationale-based feedback to increase the benefits relative to the costs of capturing design rationale**, to enhance the designers' willingness and ability to spend effort in using the method. Without sufficient benefit for the designers using a design rationale method, the method will likely not be used.

DRQ4. **The method must be generic**, i.e. applicable for layout design of all ship types, to allow a wide and standardised application in ship design processes.

[1] DRQ: Design Rationale method Requirement.

DRQ5. **The method must be easy to use and integrated within design tools**, to enhance the ability to capture the context of design rationale (Section 1.1.3) and to reduce the intrusiveness and thus to improve the potential to be accepted by designers. This allows 'shared display' as discussed above.

Before continuing, the following two notes regarding the four design rationale-related challenges are made:

1. The challenge regarding reluctance towards capturing rejected decisions is not directly covered by the method requirements. However, such decisions could be captured if DRQ2 is satisfied. Indeed, for wicked problems such as complex ship design, it is impossible to determine a priori the importance and value of and the interactions between decisions (Teisman, 2000). Hence, decisions that were thought to be inferior can become design drivers, when the design problem changes.

2. The challenge regarding the social barrier to design rationale capturing is not covered by these method requirements. Instead, this issue should be addressed by organisations themselves. For instance, a safe and transparent organisational culture should be present to prevent designers from feeling this reluctance.

4.1.2. CLASSIFYING DESIGN RATIONALE METHODS

In this section, a range of design rationale generation methods (i.e. how design rationale might be captured) and design rationale representation schemes (i.e. how captured design rationale might be structured for storage) are briefly discussed. Subsequently, the compliance of these generation and representation methods with the requirements set in Section 4.1.1 will be investigated.

DESIGN RATIONALE GENERATION

Lee (1997) distinguishes five approaches in which design rationale can be produced. Design rationale production entails how the rationale residing in the designer's mind is expressed and captured. These five approaches are:

1. **Reconstruction**. In this approach, design rationale is produced after the design is completed. This can be done by, for instance, reverse-engineering the design artefact (e.g. by asking questions like 'Which decisions did we take, and why?'), or by conducting interviews with the designers involved. Disadvantages of this approach are the cost of reconstructing design rationales (i.e. it takes time and effort *after* finishing the design activity) and the potential of introducing biases by the reconstructing designer.

2. **Record-and-replay**. In record-and-replay, design rationales are captured while they unfold in the design process. This approach could disrupt the flow of design activities if designers often need to switch between designing and documenting design decisions.

3. **Methodological byproduct**. In this approach, design rationale 'emerges' from the design process. To achieve this, designers use a specific method, which steps pro-

duce both the product *and* corresponding design rationale. This approach addresses the disadvantages of the approaches described above but can be too limited in scope (i.e. it might only be applicable for a small range of problems).

4. **Apprentice**. In this approach, the design rationale computer system plays a role. The system follows the designer's actions, asks questions when it does not understand or disagrees with the designer, and learns from the designer's response. For example, if a designer arranges a product such that it does not satisfy regulations, the computer could request additional justifications. This approach allows design rationale to be captured while they unfold but requires a sufficiently rich initial knowledge base.

5. **Automatic generation**. In this approach, the computer system automatically attempts to construct the design rationale from the traced evolution of the concept design. This comes at less cost than the apprentice approach. However, reconstructing the actual, human-taken decisions and corresponding rationales might be challenging. McCall (2019) attempts to address this issue, as well as to reduce the effort for more human-centric design rationale capture approaches, by automatically formalising informal transcripts of design sessions. However, linking transcribed design rationale to specific design elements might still be challenging.

DESIGN RATIONALE REPRESENTATION

Design rationale can be captured in various forms. Lee (1997) distinguishes three forms, although he acknowledges that "formality is typically a continuum":

1. **Informal**. Informal representations allow for easy capturing of design rationale in an unstructured form (e.g. notebooks, video recordings). It allows designers to capture design rationale in a form that is most natural to them. A major drawback is the low interpretability of the captured rationale by computer systems.

2. **Formal**. Formal representations use formalised objects to represent design rationale aspects. An advantage is that computer systems can use this structured data. However, using a formal representation might feel unnatural and limiting to the designer.

3. **Semi-formal**. Semi-formal representations attempt to combine the benefits of both formal and informal representation schemes. For example, a formal representation is used to represent the identifier of the system a design rationale refers to and an informal representation is used to capture a textual explanation of the human justification.

What is captured within design rationale can also differ. Multiple design rationale representation schemes have been developed in the past to enable the capture of the essential aspects of design decision-making. What is considered essential might differ between the intended applications of rationale capturing. For example, if the goal is to construct a database of adjacency and separation constraints to apply Gillespie (2012)'s network partitioning approach, a formal representation is needed. If the goal is to understand the human intention behind these constraints, a semi-formal approach might be best (e.g. DeNucci, 2012).

4.1.3. CONCLUSION FROM DESIGN RATIONALE METHOD CLASSIFICATION

Based on the discussion of design rationale generation methods and design rationale representation schemes above, the following observations can be made:

1. Overall, there seems a sufficient research basis to follow Lee (1997), who concludes:

   > "*Issues such as how to represent design rationales will not pose many new problems for developers of future design rationale systems. However, a few critical issues have been neglected. For example, I believe providing for cost-effective use, domain-knowledge generation, and integration give rise to many open research questions.*"

   This aligns with the problem outlined in Section 4.1.1.

2. Upon comparing the proposed design process, method requirements and design rationale generation methods, the record-and-replay method seems to fit best. However, this method can be intrusive in the design process.

3. Based on the first observation and the discussion of design rationale representations, a semi-formal representation scheme fits the requirements best.

Hence, the discussion above gives rise to the following research gap: there is a need for an in-situ design rationale method for collaborative design decision-making during early-stage complex ship design. This gap is further specified into the following two-fold goal for this chapter:

1. To develop a design rationale method to aid designers in the continuous capturing and reuse of design rationale during the collaborative concept design process, and

2. To evaluate how the developed design rationale method benefits collaborative design decision-making such that it leads to better insight into design issues across design teams and better concept designs during a single design session[2].

4.2. PROOF OF CONCEPT METHOD

This section describes a new conceptual design rationale method, as well as its initial integration into a ship layout design tool to fulfil the first goal stated above. A key question for design rationale methods is: How is the captured design rationale reused? That is, which services should the design rationale method provide? This is a key question since the design of a design rationale system is mainly determined by the services it provides (Lee, 1997).

Therefore, an important aspect, and indeed a requirement for the method, is to provide sufficient benefits to outweigh the costs of capturing rationales. For a designer, these costs might be related to cognitive, capture, retrieval, and usage barriers (Horner & Atwood, 2006).

[2]The long-term benefits are evaluated in Chapter 5.

In general, the benefits of design rationale methods are twofold. First, benefits can be an implicit by-product of using a design rationale method. For instance, being forced to express design rationale can improve decision-making, even if the method focuses on documentation (Shipman & McCall, 1997). Second, benefits can be explicitly intended services (Lee, 1997). This dissertation focuses on the latter category and proposes the following intended design support services:

1. **Design rationale capture, storage, and retrieval**. Design rationale can be automatically generated from, for instance, design documentation (Lee, 1997), such as ship design rules and regulations. However, the proposed method is to capture evolving design considerations emerging from collaborative design sessions. Hence, designers need to be able to capture design rationales as these emerge (in situ).

   Further, captured design rationale needs to be accessible and retrievable, e.g. to review past design decisions or to recall the justification and context of these decisions.

2. **Design guidance**, that is, to generate design insight based on captured design rationale, to inform the collaborative design session on the impact of requirements and preferences. The following forms of design guidance are proposed:

   - Feedback on design rationale *satisfaction* and *consistency*. Satisfaction expresses whether a single design rationale is achieved in the current concept design. Consistency can relate to the syntactic and semantic content of a captured design rationale (DeNucci, 2012), but in this dissertation, it refers to the consistency of the set of rationales. That is, it expresses which rationales conflict directly or indirectly. A direct conflict is a conflict between two rationales (e.g. A<B, and B<A). An indirect conflict is a conflict within a larger set of rationales (e.g. A<B, B<C, and C<A), i.e. the set does not comply with the transitive property. Indeed, the first two relationships imply A<C, which conflicts with the last relationship.
   - Using design rationale-based **Measures of Performance (MoPs)**. Such MoPs are concept design evaluation functions that use captured design rationale. Such MoPs can provide high-level information on the quality of the concept design, and be used to visualise the impact of different trade-offs between conflicting system arrangements. See an example in the box below. Note that for complex ships it might be hard to make all design aspects measurable (e.g. style (Andrews, 2012a), aesthetics) as elaborated in Section 1.1.2.
   - Design rationale-based **automated, real-time, concept design generation**. Such tools support the collaborative decision-making process by providing a way to quickly explore viable solution alternatives. This would overcome the need to wait for another "round" (e.g. a design session) to further explore the most feasible path or solution. For instance, if the most viable solution to a stability issue can be identified during a design session, direction for a more detailed analysis can be provided. In addition, stakeholders are better

aligned, since they both are aware of the design problem and the potential solution.

Similarly, (semi-)automated layout design tools (such as WARGEAR, discussed in Part I) might be used to evaluate the impact of an evolving set of design rationale on the overall concept design, by generating alternative concept designs that satisfy captured design rationale. Integrating elucidation of requirements and design rationale with the evolvement of the concept design during collaborative design is expected to benefit mutual understanding (Le Poole et al., 2022b). How such tools are best used during collaborative design decision-making has been discussed in Section 3.3.

Specific example for MoPs (Bullet 2)
Examples of design rationales in ship layout design are the relative positions between systems (DeNucci, 2012). A subset of all systems could be preferred to be adjacent to improve logistical performance. In this case, a design rationale-based MoP could calculate the Manhattan distance between this subset of systems. Note that more elaborate MoPs can be developed (see for example (Le Poole, 2018)). Figure 4.3 shows three different layouts, in which arrows indicate preferred logistic adjacency relationships, and the MoP represents the Manhattan distance between systems (Table 4.1).

Since these MoPs might be dynamic (i.e. capturing new design rationales changes the input of MoPs), MoP values might not be directly comparable between designs. In the example in Figure 4.3, the MoP for Layout 2 is higher than Layout 1, but it also considers an additional relationship. Similarly, the MoP for Layout 3 is the lowest but considers a partially different set of systems. Therefore, MoPs might be best suited to provide design guidance from the current state of the concept design, i.e. designers might use MoPs to identify how to improve the current concept design. Additionally, previous concept designs might be re-evaluated using an up-to-date set of design rationales.

Besides MoPs that use specific design rationales to provide information on the quality of the concept design, the whole set of captured design rationales might be evaluated using a range of network metrics. Examples are network centrality metrics, which can be used to identify systems likely to drive or constrain a ship arrangement, as demonstrated by Gillespie (2012) and Pawling and Andrews (2018).

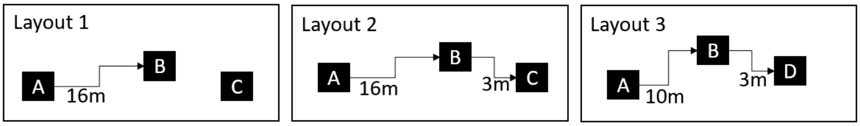


Figure 4.3: Dynamic MoP, calculating Manhattan distance using design rationale related to logistics for three layouts.

Table 4.1: Design rationale-based MoP. Manhattan distance for three layouts in Figure 4.3.

System relationship	Layout 1	Layout 2	Layout 3
d(A-B) [m]	16	16	10
d(B-C) [m]	-	3	-
d(B-D) [m]	-	-	3
MoP: Total Manhattan distance [m]	**16**	**19**	**13**

4

To enable these intended design support services, the method (and accompanying tool for the experiment) intends to aid designers in the continuous capturing and reuse of design rationale during the collaborative concept design process. Figure 4.4 shows the architecture of the developed method. The left three elements (1, 3, and 9) represent the intended continuous design rationale capturing during the concept design process. The right four elements (5-8) are the new design rationale-based functionalities that the design rationale method offers to the designer to support the design process. In the centre are two connecting elements (2 and 4), where the design rationale method can identify design changes in real-time and trigger the supportive functionalities to provide immediate feedback. The database serves as the long-term memory for the method. As such, the method stores design rationale and concept design changes and enables its retrieval.

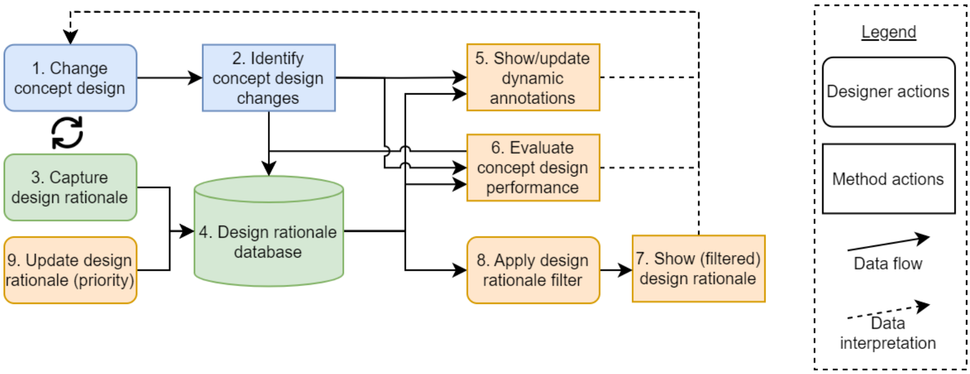


Figure 4.4: Schematic overview of the developed method for ship layout design. Blue: support and identification of manual design changes. Green: design rationale capturing and storage. Orange: design rationale retrieval and feedback.

In blue, the support of manual design changes and the identification of these design changes are shown (Section 4.2.1). Green elements are related to design rationale capturing and storage (Section 4.2.2). Aspects related to design rationale retrieval and feedback are indicated in orange (Section 4.2.3).

4.2.1. INTEGRATION BETWEEN DESIGN RATIONALE AND SHIP DESIGN TOOLS

The design rationale method is intended to be used *during* design work, such that it supports ongoing design decision-making. Hence, the designer needs to be able to continuously design *and* capture design rationales (i.e. on-the-fly or record-and-replay) . To reduce the intrusiveness of the design rationale method in the design process and enhance the ability for computer-based feedback to the designer, an integration between the design tool and the design rationale method is required.

While the focus of Part I was on gaining design insights beyond the functional arrangement, Section 4.1.1 explained the focus of Part II is both on space-level and on functional arrangement-level design decision-making. Indeed, the functional arrangement encompasses the major design decisions. Detailed arrangements as developed by WARGEAR and GAPs are largely based upon the functional arrangement. Therefore, the focus in Part II is also on manual design tools, in contrast with Part I. Since WARGEAR builds upon functional arrangements and is an automated tool, it is not considered for integration with the design rationale method.

One of the standard CAD design tools used in ship design is Rhinoceros (McNeel, n.d.), see for instance, (Kana et al., 2022; Kana & Rotteveel, 2018; van Oers et al., 2018; le Poole et al., 2022c; Takken, 2009). Rhinoceros offers users various possibilities to develop compatible custom extensions using the visual scripting language add-on Grasshopper or Python-based scripting, for instance. Rhinoceros is chosen as the design tool for this research because it's already applied to ship design and offers the possibility to develop custom extensions (e.g. custom Graphical User Interfaces (GUIs)).

The design rationale method was implemented in a custom-developed GUI, shown in Figure 4.5. Through this GUI, designers can concurrently perform layout design work in Rhinoceros' main interface and use the design rationale method to capture and retrieve design rationale. The numbers in Figure 4.5 correspond to the elements shown in Figure 4.4. Note that the current implementation in this proof-of-concept uses Rhinoceros as a pure CAD program and lacks functionalities that an actual ship design tool would offer (e.g. stability calculations). However, this is identical to the use of Rhinoceros in the GAP generation process discussed in Section 2.1. By using Rhinoceros, it enables future expansions, such as elaborated on in Chapter 5.

The integrated design rationale method allows the designer to change the concept design (Step 1), while the method is able to identify which design changes are made and when (Step 2):

1. *Change concept design* (designer). The design rationale method is to support designers during design activities. Hence the designer needs to be able to change the concept design.

2. *Identify concept design changes* (method). The method needs to identify design changes for two reasons. First, to support computer-based design feedback. Implemented examples of such feedback are the design rationale-based MoPs (Step 8) and the dynamic annotations feature of the design rationale feedback algorithm (Step 9), as elaborated in Section 4.2.3. Second, for research purposes, it is necessary to evaluate what has been changed to the concept design to relate these changes to the way the rationale method is used. However, such information could also be used to gain insight into actual design processes.

4.2.2. CAPTURING AND STORING DESIGN RATIONALE

Before design rationale can be used, it needs to be captured (Step 3) and stored (Step 4) (DeNucci, 2012). Such capture and storage is especially important when automated

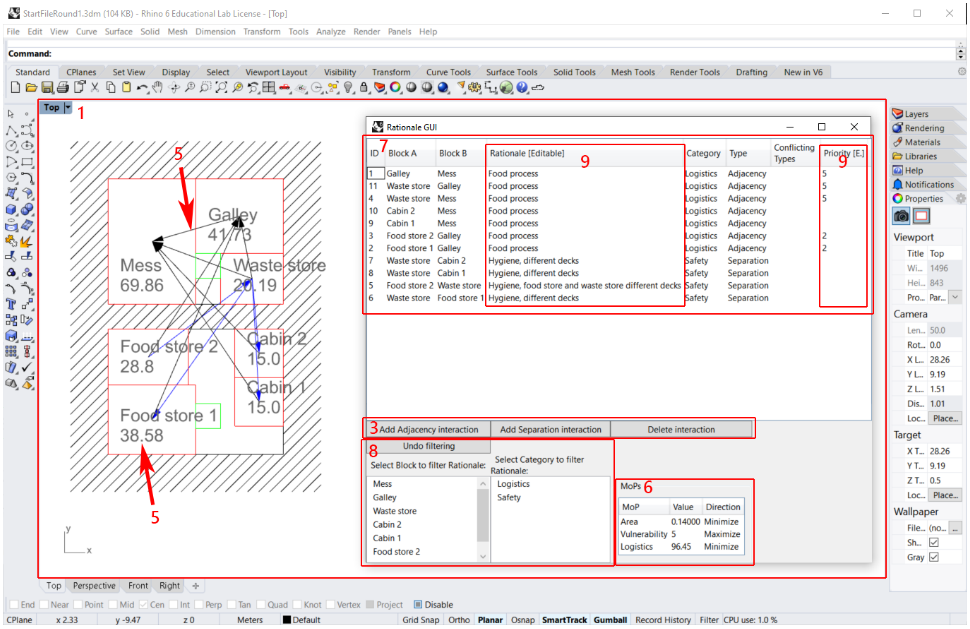


Figure 4.5: Screenshot of design rationale method integrated in a Rhinoceros GUI. Numbers indicate steps in the method. Steps 2 and 4 are executed in the program's background.

computer-based design rationale support is required. Indeed, computers will not be able to provide such support if design rationale is not explicitly captured and adequately stored. The design rationale capturing and storage steps are implemented as follows:

3. *Capture design rationale* (designer). Besides the integration of design tools and the design rationale method, the issue of intrusiveness is addressed in two ways:

    (a) A predefined design rationale structure based on DeNucci (2012)'s definition of interactions is used. An interaction is a preferred or required relation between two systems and its justification. An example of an interaction is: *the ammunition store should be adjacent to the gun* [relation] *to reduce the* dangerous transport of ammunition through the ship [justification] (Le Poole et al., 2022c). By using a predefined design rationale structure, designers need to think less about *how* to represent design rationale. This is especially important to ensure comprehensibility by both humans and computers (DeNucci, 2012). The predefined design rationale structure is semi-formal, as it comprises both formal (e.g. system names, interaction types, etc.) and informal elements (e.g. the rationale or justification). System Properties (e.g. preferred system position or sizing) have been hard-coded in the setup of the design problem and are used in Step 5.

    (b) Designers can select systems *in the layout* to define interactions. It was expected that this would be more intuitive than using drop-down menus, for

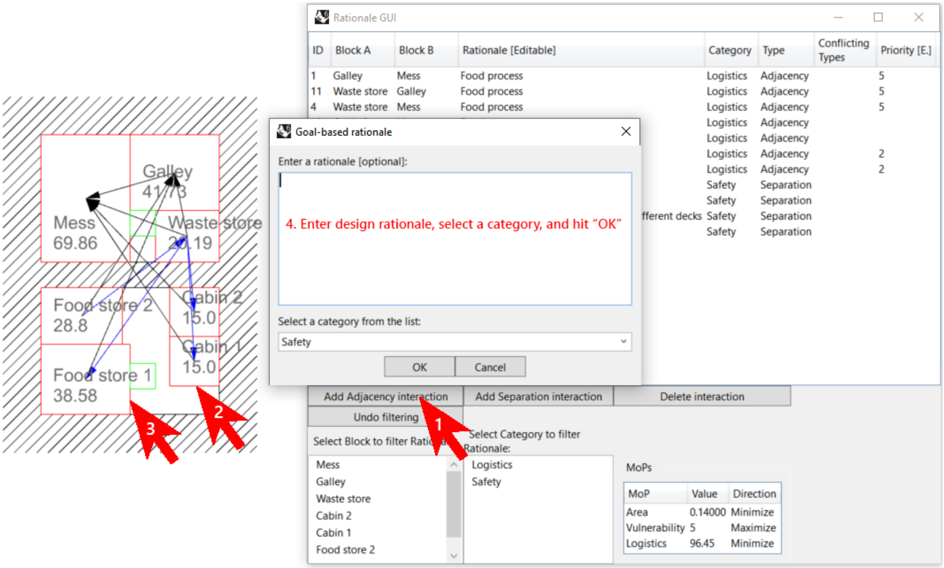


Figure 4.6: Screenshot of rationale capturing action to capture an adjacency interaction between 'Cabin 1' and 'Food store 1'. Step 1: select interaction type; Step 2 and 3: select corresponding systems in the layout; Step 4: complete design rationale. Subsequently, the captured rationale will be added to the database, and shown in the list in the main GUI.

instance, and thus reduce the effort required to capture design rationale. The designer action 'capture rationale' is illustrated in Figure 4.6.

4. *Design rationale database* (method). All rationales and design changes are captured in a database for future reference. On the one hand, this allows the analysis of the design process, such as elaborated in Section 4.4. On the other hand, this allows the designer to refer to past concept designs and supporting rationale, or to take a past concept design as the starting point for another design iteration. Besides storing design rationale, the database is also used to store design changes and the performance of concept designs.

It is important to note that the current version of the design rationale method is tailored to the experiment, which is elaborated in Section 4.3. Consequently, a fully operational design rationale method might need the implementation of additional or altered functionalities. For instance, currently, only a limited number of interaction types is included in the method, i.e. only adjacency and separation for multiple categories. In practice, one might want a more gradual distinction of the required relative distance between systems, e.g. should be separated; might be separated; might be adjacent; should be adjacent (see DeNucci, 2012).

4.2.3. DESIGN RATIONALE RETRIEVAL AND FEEDBACK

Capturing design rationale has limited benefit when the design rationale is not used. Therefore, the method uses captured design rationale to provide visual feedback (Step

5) and evaluates the performance of the concept design, based on the current status of the concept design and captured design rationale (Step 6). Additionally, the designer can retrieve (Step 7), filter (Step 8), and update (Step 9) the captured design rationale when required. Steps 5 to 9 are further detailed below:

5. *Show/update dynamic annotations* (method). The method provides visual support to the designer by showing dynamic annotations overlaying the concept design. Examples of such annotations are arrows representing interactions between systems and textual annotations showing current system sizing. The position and orientation of such annotations are dynamically updated when the method identifies design changes (Step 2). Further, colouring is used to distinguish between, for instance, interaction types (e.g. adjacency and separation). Such annotations could be extended via additional context menus that open when an annotation is selected, for example.

6. *Evaluate concept design performance* (method): The method uses the captured rationales (Step 3) in MoPs to inform designers of the quality of the concept design. For instance, design rationale related to logistics might be used in MoPs considering Manhattan distances between logistically connected systems (Le Poole, 2018; le Poole et al., 2022c). To allow for real-time feedback, MoPs that require high computational efforts should be avoided unless such information is considered essential to make the right decisions.

7. *Show (filtered) design rationale* (method). An overview of previously captured rationales is provided to enable designers to review the design based on captured rationales. Since the number of rationales might be high, the designer might filter the rationale to view applicable rationales only. Also, the designer is informed on directly conflicting, i.e. contradictory, design rationales. For instance, when two systems are related by both adjacency and separation constraints.

8. *Apply design rationale filter* (designer). As explained above, the designer can decide which rationale is shown. Possible rationale filters are 'system name', 'category', 'timestamp' (i.e. date/time rationales are captured), and 'systems in current view' (i.e. only show rationales based on the zoom level and position of the design tool). In the current implementation, the filtering only applies to the interactions shown in the GUI and is based on the names of systems related to interactions and interaction categories. List boxes are used. Additionally, the table with design rationale shown in the GUI can be sorted (i.e. by ascending or descending numerical or alphabetical value) by clicking the table headers.

9. *Change design rationale (priority)* (designer). In the GUI, each rationale can be given a priority indication. This can be used by design teams to capture initial trade-offs or varying preferences for interactions, for example. Also, the justification of each interaction can be altered to capture new design considerations. The updated design rationale is stored uniquely in the database.

4.3. CASE STUDY 5 - SHORT-TERM DESIGN RATIONALE APPLICATION

This section elaborates on the experimental setup to evaluate the developed design rationale method. As elaborated in Section 4.1, the goal is to assess how this method benefits collaborative design decision-making, such that it leads 1) to better insight into design issues across design teams and 2) to better concept designs during a single design session. Note that this experiment is aimed to provide insight for the evaluation of the design rationale method, as such it does not fully mimic a real-world design scenario, as discussed in Section 4.6.

4.3.1. DESIGN PROBLEM

In the experiment, design teams consisting of three participants were tasked with two small layout design problems, both containing two compartments and seven systems, see Figure 4.7. The compartments in Layout 1 are arranged vertically adjacent and connected via a 2x2m staircase. The compartments in Layout 2 are arranged horizontally adjacent and connected via a 2m-wide passageway.

4.3.2. EXPERIMENT SETUP

The task of each design team was to drag and scale all systems 'manually' into a sufficing layout. Each team member was assigned one of the following roles: 'Naval Architect', 'Logistics Specialist', or 'Safety Specialist'. Typically, the Naval Architect in the team operated the tool, similar to real ship design processes. Team members were given a role sheet with requirements related to their specific roles. These requirements could be discussed, but the role sheets could not be shared with other team members. The content of the role sheets is summarised in Table 4.2. These requirements comprise System Prop-

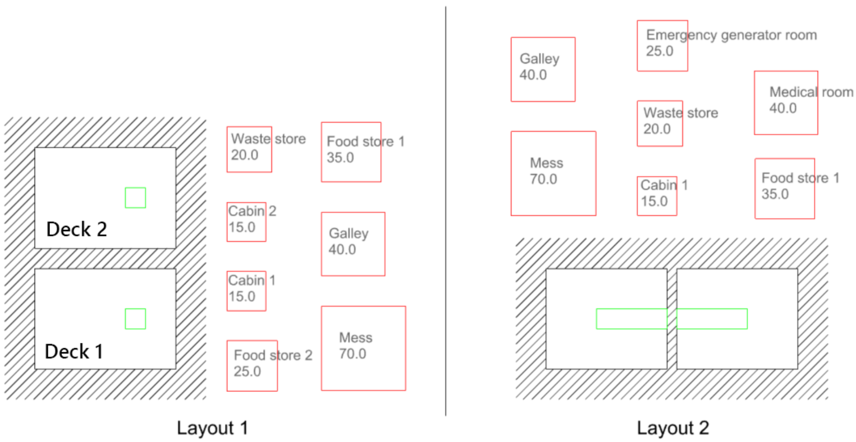


Figure 4.7: Visualisation of the two layouts (black) and corresponding staircase or passageway (green) and systems (red). The numerical value in each system is its required area $[m^2]$. The current area of systems shown equals the required area. Compartments are sized 14m by 10m (Layout 1) and 12m by 10m (Layout 2).

erties (i.e. required area) and Interactions (i.e. relative positions between systems).

Table 4.2: Role sheet information for each role in a design team. *: In addition, visually check realistic aspect ratios. **: *Adjacency*: systems are in the same compartment. *Separation*: systems are in different compartments. ***: Adjacency is measured in Manhattan distance between systems. If spaces are in different compartments, the path includes the passageway or staircase.

Naval architect		**Safety specialist**			
System Name	**Required area [m^2]***	**System Name A**	**System Name B**	**Interaction type****	**Rationale**
Mess	70	Galley	Mess	Adjacency	Hygiene
Galley	40	Food store(s)	Waste store	Separation	Hygiene
Food store 1	35	Cabin(s)	Waste store	Separation	Hygiene
Food store 2	25	Medical room	Waste store	Separation	Hygiene
Cabin 1	15	Medical room	Emergency generator room	Separation	Noise
Cabin 2	15	Emergency generator room	Cabin(s)	Separation	Noise
Waste store	20	Emergency generator room	Galley	Separation	Noise
Emergency generator room	25	Emergency generator room	Mess	Separation	Noise
Medical room	40				

Logistics specialist			
System Name A	**System Name B**	**Interaction type*****	**Rationale**
Mess	Galley	Adjacency	Food process
Galley	Food store(s)	Adjacency	Food process
Mess	Cabin(s)	Adjacency	Food process
Mess	Waste store	Adjacency	Food process
Galley	Waste store	Adjacency	Food process

The quality of each layout was captured via three MoPs based on System Properties and Interactions. The first MoP, Area Measure of Performance (AMoP), is given in Equation 4.1 and evaluates the sizing performance of the layout (see Section 2.4.8).

$$AMoP_i = \sum_j \max(0, (RA_j - AA_j)) \tag{4.1}$$

Where:

system $j \in$ systems in layout i.

RA_j is the Required Area for system *j*.

AA_j is the Achieved Area for system *j*. If AA_j is larger than RA_j, no penalty or reward to the overall score is given. If systems overlap, the overlapping area is subtracted from *AA* for these systems[3].

The second MoP, Logistic Measure of Performance (LMoP), is given in Equation 4.2 and assesses the layout from a logistical point of view. It is based on the Manhattan distance between all pairs of logistically connected systems.

$$LMoP_i = \sum_k MD(k) \tag{4.2}$$

[3]This stimulates the development of layouts with rectangular-shaped systems (contrary to e.g. L-shaped systems).

Where:

$k \in$ Interactions related to Logistics in layout i

MD is given by Equation 4.3 and is the Manhattan Distance between two systems s and t in Interaction k.

$$MD(s,t) = \begin{cases} |x_s - x_t| + |y_s - y_t| & \text{if } s \text{ and } t \text{ in same compartment } i \\ |x_s - x_{LSi}| + |y_s - y_{LSi}| + |x_{LSj} - x_t| & \text{if } s \text{ and } t \text{ in compartments } i \text{ and } j \\ \quad +|y_{LSj} - y_t| + |x_{LSi} - x_{LSj}| & \text{respectively} \\ \quad +|y_{LSi} - y_{LSj}| + |z_{LSi} - z_{LSj}| & \end{cases} \tag{4.3}$$

Where:

$(x_s,\ y_s)$ and (x_t,y_t) are the geometric centres of system s and t in compartment i respectively.

LS is a logistic system (i.e. a staircase or passageway) between compartments i and j.

$(x_{LSi},y_{LSi},z_{LSi})$ and $(x_{LSj},y_{LSj},z_{LSj})$ are the geometric centres of LS in compartment i and j respectively.

The third MoP, Safety Measure of Performance (SMoP), is given in Equation 4.4 and assesses the layout from a safety point of view. It captures how many safety-related constraints are satisfied and unsatisfied in the layout.

$$SMoP_i = n(\text{satisfied } SI_i) - n(\text{unsatisfied } SI_i) \tag{4.4}$$

Where:

SI_i is the set of Interactions related to Safety in layout i.

n is the number of elements in the subsets with satisfied and unsatisfied elements).

To test the experimental setup, a preliminary version of the design problem and tool was provided to 12 Master's and PhD students. A main lesson learned was that providing MoPs to participants distracted them from directly considering the layouts. Instead, their attention was drawn to understanding and optimising the MoPs, in order to optimise the design. That is, design choices were principally made because the rough MoPs indicated that the layout would become better. This became apparent when one of the testing participants explicitly asked how one of the MoPs was calculated, "so that we would be better able to optimise the MoPs". However, the MoPs are aimed to provide *guidance* in the design process and thus to support collaborative rational design decision-making on the design problems (Section 4.2). For example, LMoP provides a measure of logistic performance but does not consider walking routes within compartments. Based on this observation, the decision was made to hide the MoPs during the experiment for the participants. However, the MoPs were still calculated to allow for the analysis of the design processes.

Each experiment took two hours and was structured as follows:

1. Introduction to the research background and experiment by the researcher (20 minutes).
2. Familiarisation exercise in teams (10 minutes). This exercise was designed to familiarise participants with the problem, Rhinoceros, and the design rationale method. If the experiment took place online, 'break-out rooms' were used in this and the two subsequent items.
3. Experimental round 1 in teams (30 minutes).
4. Experimental round 2 in teams (30 minutes).
5. Questionnaire, comprising 17 closed and 9 open questions (20 minutes). This questionnaire can be found in the Appendix C.1 and was aimed to elicit participant satisfaction with their teams' design process and resulting layouts, and to receive feedback on the design rationale method.

4.3.3. EVALUATION

To evaluate the design rationale method, each team used a baseline method for one design problem and the design rationale method for the other design problem. The design rationale method is the method as presented in Section 4.2, with the exception of the hidden MoPs. The baseline method does not enable the capture and retrieval of design rationale, which also prevents the design rationale-based feedback. Hence, the baseline method forces teams to rely on verbal communication only. Using a baseline method besides the design rationale method allows for a comparison between the measured design quality and perceived satisfaction between the use of these two methods. Eventually, this comparison indicates the performance of the proof-of-concept design rationale method.

Two potential main learning effects had been identified. The first learning effect is that participants could learn the nature of the presented design problems and they could approach the second design problem in a similar manner to the first design problem if that approach was found successful. The second learning effect is related to the order in which the baseline and the design rationale methods are used. The support that the design rationale method provides could stimulate participants to approach the second design problem in a different way compared to the situation where this aid was not provided. For example, the visual support of the dynamic annotations might trigger participants to consider the design problem from a network perspective. Although both learning effects need to be analysed to evaluate the performance of the design rationale method, only one learning effect could be studied due to the low number of participants. The second learning effect was selected because is considered to be the most significant since it is more related to the performance of the design rationale method. To elucidate the selected learning effect across the use of these methods, approximately half of the participants (Group A) used the baseline method first while the others (Group B) commenced with the design rationale method, see Section 4.3.4.

4.3.4. PARTICIPANTS

Participants of the experiment comprised Delft University of Technology (TU Delft) Marine Engineering Master and PhD students (n=15) and experts (n=15) from the DMO, Netherlands Organisation for Applied Scientific Research (TNO), and DAMEN Naval, under informed consent. The experiment took place in five sessions between September 2021 and February 2022. The experiment protocol was approved by the TU Delft Human Research Ethics Committee.

Recruitment for student participation was done via a course taught by the researcher and (co-)promotors at TU Delft. This was done via online announcements in the digital student learning environment Brightspace and email, and in-class announcements. Furthermore, students were recruited from the research lab of the co-promotor. Recruitment for expert participation took place via the professional network of the researcher.

Participants were subdivided into teams of three persons. Each team comprised persons with the same affiliation. Each team was assigned to Group A or B. Teams in Group A used the baseline method in the first experimental round, while teams in Group B commenced with the design rationale method. Table 4.3 shows the distribution of participants over teams and groups.

4

Table 4.3: Summary of participants. [1]: The researcher (student) participated in one team to complete an expert team. (n): number of teams online. [m]: number of participants in test run

	Group A		**Group B**			
	Participants	**Teams**	**Participants**	**Teams**	**Total participants**	**Total teams**
Experts	5[1] [0]	2 (0)	9 [0]	3 (0)	15	5
Students	13 [1]	4 (1)	3 [3]	1 (1)	15	5
Total participants	18		12			
Total teams		6		4		

Due to COVID-19-related restrictions, two teams performed the experiment via an online environment. All teams participated either entirely in-person or entirely online. No mixed in-person/online teams took part. The effect of online versus offline participation was not studied.

Four participants from the test run in August 2021 did participate in an online session in January 2022. Hence, these participants were more familiar with the design rationale method and a general idea of the design problem provided during the experiment. One team was entirely composed of participants who had been involved in testing, although in different testing teams. This group was not excluded from the analysis, because: 1) the baseline method was not tested by these participants, 2) the design problem was changed substantially, and 3) the limited availability of other, non-biased participants.

4.4. CASE STUDY 5 - RESULTS

This section elaborates on the qualitative and quantitative results obtained from the experiment to evaluate how the developed design rationale method benefits collaborative design decision-making, such that it leads to better insight into design issues across design teams and to better concept designs during a single design session. The qualitative and quantitative data are retrieved from logged data by the methods and a post-experiment questionnaire. The data analysis is structured by the following questions:

1. Section 4.4.1 answers: 'How is the method used by design teams over time?' This will provide insight into the general use of the design rationale method during the design sessions.

2. Section 4.4.2 answers: 'How does the method support the negotiation process within design teams?' This section aims to gain insight into the impact of the design rationale method on the teams' decision-making process.

3. Section 4.4.3 answers: 'How does the use of the method impact the quality of concept designs?' This section will investigate if the design rationale method enhances the quality of concept designs, that is, if the use of the method leads to better concept designs.

4. Section 4.4.4 answers: 'How does the use of the method impact satisfaction with the concept design across design teams?' This section investigates the relationship between participant satisfaction and the use of the design rationale method.

5. Section 4.4.5 answers: 'What are the perceived benefits of the method?' This section investigates the participant's view on the method and how it might benefit ship design.

4.4.1. USE OF THE DESIGN RATIONALE METHOD OVER TIME

The design rationale method adds new activities to the design process (e.g. the explicit capturing of design rationale and setting of priorities). Furthermore, the visual feedback (i.e. arrows representing interactions and the overview of captured rationales in the GUI) is expected to enhance the participants' overview of the design problem. In contrast, the added functionalities take time and effort. In this section, the use of the design rationale method in the arrangement process is investigated based on traced designer actions.

Use of functionalities over time

First, the use of design rationale method functionalities over time is investigated to identify which functionalities are used when in the process. As the baseline method does not offer functionalities related to design rationale, only the use of the design rationale method was investigated. The following high-level actions could be performed using the design rationale method:

1. *Open*: The design rationale GUI was opened. This happens at the start of the design process or after the tool crashes. The latter occurred relatively often during the earlier experiments.

2. *Close*: The design rationale GUI was closed.

3. *Add rationale*: The team added an interaction using the design rationale method.

4. *Delete rationale*: An interaction was removed, e.g. because the team selected the wrong interaction type.

5. *Rationale edit*: The priority or justification of an interaction was changed.

6. *Filtering*: The team used one of the filtering options in the GUI to retrieve a selection of captured design rationale.

Unfortunately, not all actions were traced for all teams. For instance, traceability of the filtering action was only implemented after Team 7 performed the experiment. This significantly limits the analysis of the use of functionalities and only enables some rough observations based on the design processes of Teams 8-10. In the remainder of this chapter, the data of all teams is used in the analysis, with one exception. Team 1 needed to restart after the tool had crashed completely. Hence the quantitative data from Team 1 is not considered reliable enough for the analysis of the design process. Team 1's answers to the questionnaire results are used, however.

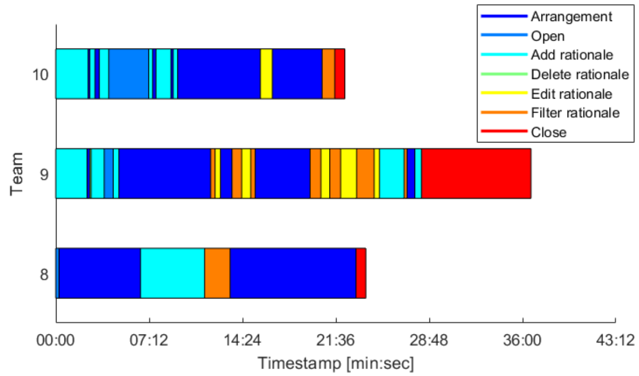


Figure 4.8: Use of the design rationale method over time by Teams 8-10.

Figure 4.8 shows which actions were used by Teams 8-10 in the design rationale method as well as when these teams arranged systems. Teams 8 and 10 worked on the first design problem, while Team 9 worked on the second one. In all cases, design rationale was captured in the first half of the experimental round. Team 9 also captured design rationale after completing the experiment. The large red 'Close' bar indicates that Team 9 waited for approximately 8 minutes to close the GUI after conducting the last action in the GUI. Table 4.4 provides the average use of each action across the three teams. The 'Close' action is not considered in the data to remove the excess waiting before closing the GUI. Approximately 55% of the experimental round was spent on the actual arrangement of systems, while 22% was used on design rationale capture. The possibility to filter design rationale in the GUI was used relatively often as well (10%). There was only minimal deletion of design rationale. Typically, design rationale was only deleted when an error was made upon rationale addition.

Impact of using the design rationale method

Second, the process of arranging systems was further investigated to evaluate whether the design rationale method triggers designers to approach design problems differently (e.g. to consider the whole design problem upfront contrary to considering large systems first), i.e. the second learning effect identified in Section 4.3.3. An initial analysis of the development of concept designs over time showed that teams differed with respect

Table 4.4: Average use of actions in the design rationale method for Teams 8-10, as a percentage of each team's experimental round duration.

	Arrangement	Open	Add rationale	Delete rationale	Edit rationale	Filter rationale
Team 8	69.1%	1.1%	21.3%	-	-	8.5%
Team 9	46.1%	2.5%	22.2%	0.4%	12.2%	16.6%
Team 10	51.6%	14.3%	25.2%	-	4.3%	4.5%
Mean	55.6%	5.9%	22.9%	0.4%	5.5%	9.9%
Standard deviation	12.0%	7.3%	2.0%	0.0%	5.6%	6.1%

to *when* they did *what* modifications to systems, e.g. resizing and moving. To further investigate this observation, the following six types of system modifications are defined:

1. *Resize outside*: a system is resized outside a compartment.
2. *Resize inside*: a system is resized inside a compartment.
3. *Move outside*: a system is repositioned outside a compartment.
4. *Move inside*: a system is repositioned inside a compartment.
5. *Move cross*: a system is repositioned across a compartment boundary, either from outside to inside the compartment or vice versa.
6. *Move cross-compartment*: a system is repositioned from one compartment to another.

A visual explanation of these six system modifications is provided in Figure 4.9.

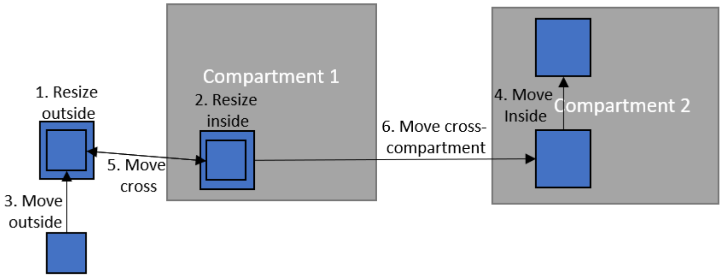


Figure 4.9: Visual explanation of the six types of modifications to systems.

Additionally, the design timeline is divided into five phases with equal duration. Since each round lasted for around 30 minutes, each phase corresponds to approximately six minutes. It is expected that different system modifications are applied in different phases. For example, towards the end of a round, most major decisions on system positioning have likely been made, and most actions are related to fitting all systems into the layout (i.e. modifications 'move inside' and 'resize inside' are expected to dominate).

All system modifications captured during the experiment are categorised to the type of modification and design phase. Also, a differentiation between experimental rounds 1 and 2 and between using the baseline or design rationale method is made. Subsequently, the contribution of each modification in each phase is calculated using Equation 4.5.

$$contribution_{i,j,k} = \frac{\sum_k \frac{\text{number of modifications}_{i,j}}{\text{total number of modifications for } k}}{n} \tag{4.5}$$

Where:

$i \in$ Phases.

$j \in$ types of modifications.

$k \in n$

n is the number of teams in the same experimental round and group.

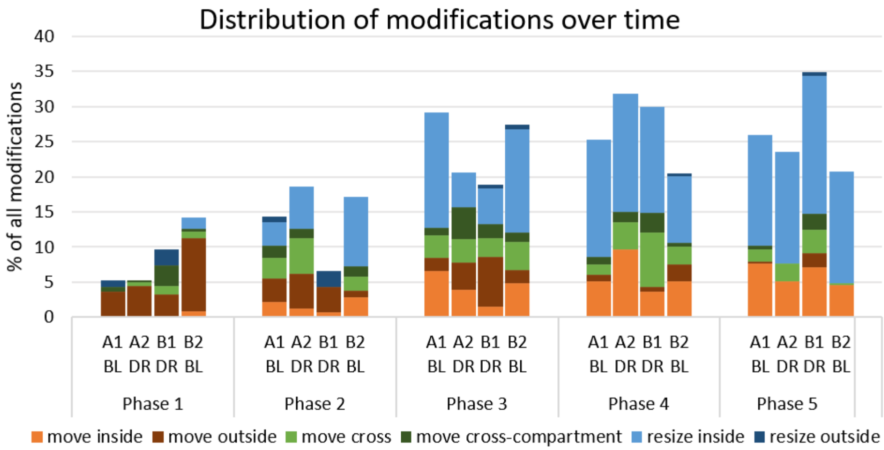


Figure 4.10: Distribution of modifications over time. A1: Group A, Layout 1. A2: Group A, Layout 2, etc. BL: baseline method. DR: design rationale method. Each category (e.g. all five A1 BL columns) adds to 100%.

The results are shown in Figure 4.10. The following observations can be made:

1. The modifications *'move inside'* and *'resize inside'* were dominant in later design phases. This holds for both different design problems and different methods. This corresponds with the expectation above.

2. The modification *'resize inside'* was used significantly more than other types of modifications. This might be explained by the observations that 1) the last design changes were primarily performed to make the layout fit, 2) moving a system in the correct position was easier than modifying its size into the proper shape, and 3) resizing was used to reposition systems, i.e. by extending the length of a system, it can be connected to a nearby adjacent system.

3. For Group B, the modification *'move outside'* was very dominant in Phase 1 when using the baseline method, as well as in Phases 1-3 when using the design rationale method. For Layout 1, Group B used the design rationale method and generally spent one or two phases on capturing design rationale. Subsequently, these teams used the systems and annotations to roughly figure out which layout was preferred, before commencing with the detailed arrangement of systems in compartments. This way of using the layout to perform initial major decision-making will be called *'network arrangement'*. An example is shown in Figure 4.11. For Layout 2, Group B used the baseline method. Hence, no time was required to capture design rationale. Consequently, network arrangement (although without interaction annotations) commenced already in Phase 1.

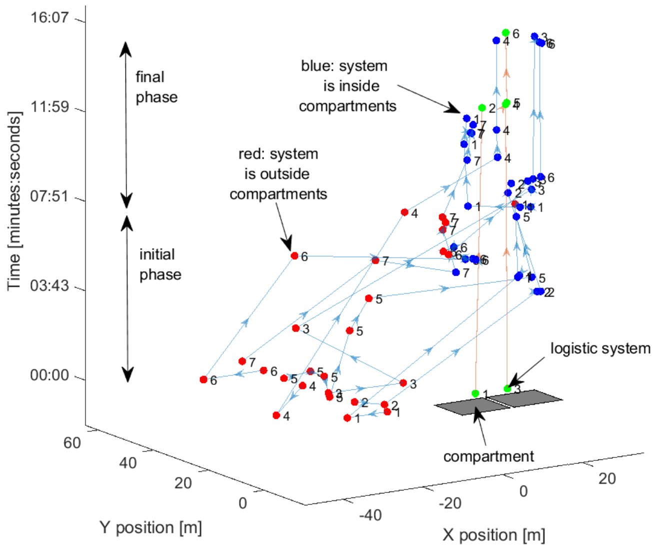


Figure 4.11: Network arrangement (initial phase) and detailed arrangement (final phase) demonstrated by Team 9, Layout 2. The graph shows the X,Y position of (logistic) systems over time. Each node represents a modification to the corresponding system. Note that the initial network arrangement was adapted during detailed arrangement: system 6 (Medical room) was moved to the right compartment, and systems 1 (Mess) and 7 (Emergency generator room) were moved to the left compartment.

A similar trend, although less clear, can be seen when comparing the *'move outside'* modification using the baseline and design rationale method for Group A. A slight increase of modification *'move outside'* can be observed from Layout 1 to Layout 2, i.e. from baseline to design rationale method, in the first three Phases.

Based on these observations, it is expected that the dynamic annotations provided by the design rationale method and the need to be explicit about all design rationale upfront can trigger teams to first arrange, or just group, systems roughly based on required interactions and area, and then arrange systems in detail. In other words, the teams seem to 'sketch' to support the negotiation process. Sketching is an important means of conveying design thinking but is hardly supported by today's ship design tools (Pawling & Andrews, 2011).

4. The modifications *'move cross'* and *'move cross-compartment'* were used relatively more when using the design rationale method, compared to the baseline method. Also, these modifications were mainly observed in later phases. This might indicate that the outcome of the initial decision-making using *'move outside'* modifications, as described in point 3, turned out to be infeasible when systems were actually arranged in the layout. Another explanation might be that different arrangements were investigated in later phases. Such investigation could be performed because, for instance, a specialist was not satisfied with the initial arrangement (which now had become more tangible than in the network representation) and wanted to improve the layout or because the team already identified multiple possible allocations of systems to compartments during the initial phases.

Despite the limited data logging, the results in this section indicate that all options in the design rationale method have been used. Furthermore, the results indicate that the design rationale method triggers teams to 'sketch' more often to support the negotiation and design process compared to the baseline method.

4.4.2. SUPPORT OF NEGOTIATION PROCESS

The design problems were deliberately created to contain conflicts, i.e. trade-offs were necessary. Hence, each team needed to negotiate to resolve these conflicts. This section focuses on the perceived support provided by the design rationale method in the design process.

To elucidate participants' perception of the supporting role, the post-experiment questionnaire contained six statements regarding this topic. The responses to these statements are presented in Figure 4.12. Generally, the design rationale method was perceived to support the decision-making process (80%) and was not distracting for most participants (54%) but was distracting for some participants (18%).

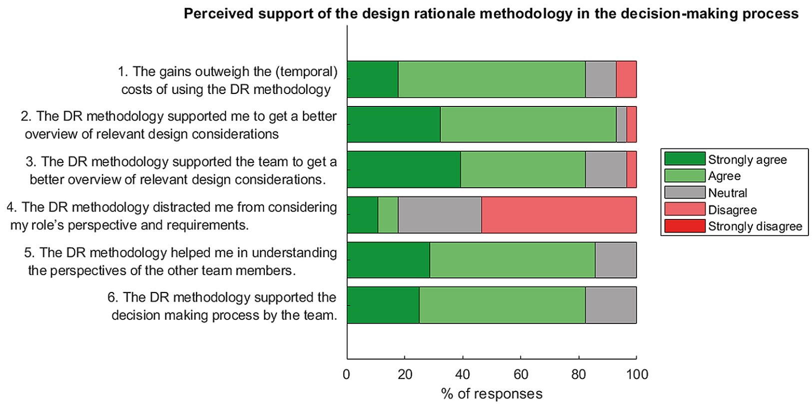


Figure 4.12: Perceived support of design rationale method in the decision-making process (n=28).

A key intended benefit of the design rationale method is providing an overview of relevant design considerations. Therefore, statements 2, 4, and 5 concern this aspect. Most participants indicated to (strongly) agree with these statements, respectively 86%, 82%, and 93%.

On average, 26% of the duration of design sessions using the design rationale method was used on rationale capturing, while the remainder was used to arrange the systems. The time spent on design rationale is a part of the effort required to apply design rationale during design. Still, 82% of the participants indicated that the gains outweigh the (temporal) costs of using the method.

Participants were also requested to describe how the design rationale method supported decision-making compared to the baseline method. The following statements are a representative selection of answers to this question:

> "*The DR method helped better to understand the interactions between the spaces.*"

> "*Better alignment of rationale and a more explicit discussion.*"

> "*It centralised the discussion.*"

> "*The baseline method resulted in 'chaos' and repetition in discussions.*"

> "*Explicit visualisation of each other's rationale helps [to] optimise together*[4]*. Even [the safety specialist] was looking at logistics and vice-versa...*"

> "*Forces a baseline of knowledge for [the] whole team.*"

These quotes and the responses to the closed questions indicate that the design rationale method supports the negotiation process by facilitating enrichment and negotiated knowledge (Section 1.1.1) and is perceived to provide a better understanding of the design problem within design teams.

4.4.3. QUALITY OF CONCEPT DESIGNS

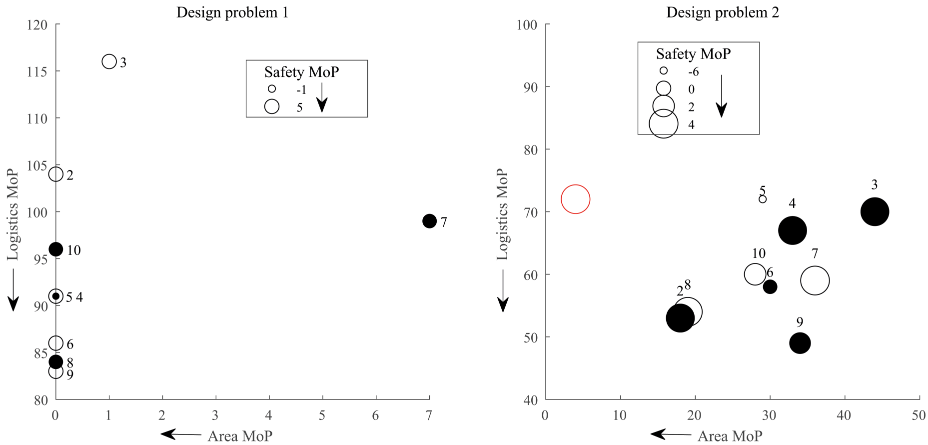
Since one part of the goal of the design rationale method is to improve the quality of design (see Section 4.1.3), the quality of the developed concept designs is investigated. The MoPs for each design discipline are used to measure the quality of each concept design.

QUALITY THROUGH MOPS

First, the quality of all generated final concept designs is compared. Figure 4.13 shows the three MoP scores for each final design of each team. For design problem 1, the final designs are concentrated along the LMoP axis. This means that these designs meet the required area requirements for all systems (i.e. $AMoP = 0$). Satisfying all area constraints is relatively easy in this design problem since the total available area is $280m^2$, while the total required area for placing all blocks (without considering logical placement) is only $220m^2$. Nonetheless, Teams 3 and 7 failed to meet the required area constraints. An investigation of the associated final layouts of these teams showed that the mismatch

[4]Optimisation here refers to making decisions based on perceived merit and objective numbers, such as system sizing.

between the required and achieved areas is resolvable. An explanation could be that, due to time limitations, teams did not put the 'finishing touch' to the layout. Most teams maximised SMoP, although Team 5 violated relatively many safety constraints (SMoP = -1). The results indicate an even spread in layout performance between teams using the baseline and the design rationale method.



(a) Design problem 1

(b) Design problem 2

Figure 4.13: MoP scores for the final designs of all teams for both design problems. Open nodes: baseline method; filled nodes: design rationale method; arrows point in favourable direction. Red: scores for Team 5 when accounted for leaving out the Emergency Generator Room (EGR). Teams 6 and 7 were online.

For design problem 2, the final designs are spread across all three MoPs. Four teams used the baseline method and six teams used the design rationale method for this design problem. Meeting all required area constraints is impossible for this design problem, since the total available area is $240m^2$, while the total required area is $245m^2$. With regard to safety constraints, one team satisfied four (SMoP = 0), two teams met five (SMoP = 2), and five teams met six (SMoP = 4) safety constraints. One team (Team 5) satisfied only two out of eight safety constraints (SMoP = -6). Although this is not apparent from Figure 4.13b, Team 5 decided to arrange the Emergency Generator Room (EGR) at another notional deck to solve the shortage of available area, see Figure 4.14d, thereby not adhering to the given constraints of the design problem. If this decision satisfies the required area of the EGR and interactions with other systems, this team scores AMoP = 4, SMoP = 2, while LMoP stays 55 and shifts to the Pareto front. This is shown as a red node in Figure 4.13b. Again, there seems to be an even spread in layout performance between teams using the baseline and the design rationale method.

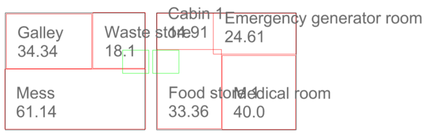
Teams 6 and 7 participated in an online session. All other teams completed the experiment in person. Scoring worst in the first round, Team 7 achieved an average score in the second round. Team 6 achieved a good performance in the first round and also achieved an average score in the second round. Therefore, online participation seems to give similar results compared to in-person participation in the experiment. As said above, the impact of online participation is not further studied.

VISUAL COMPARISON

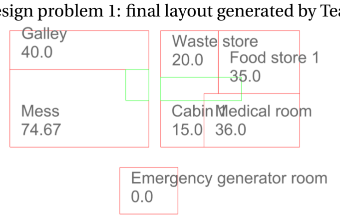
Second, a subset of the final concept designs is compared. For design problem 1, the final designs of Teams 4 and 5 are compared because these scored similar with respect to AMoP and LMoP but achieved a different SMoP. For design problem 2, the designs on the first two Pareto fronts (in the Logistic-Area plane) were compared.

(a) Design problem 1: final layout generated by Team 4.

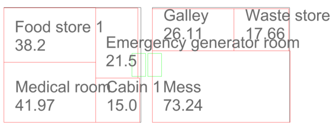
(b) Design problem 1: final layout generated by Team 5.



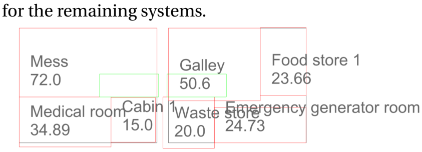
(c) Design problem 2: final layout generated by Team 2.



(d) Design problem 2: final layout generated by Team 5. The team decided to arrange the Emergency Generator Room at 'another deck' to have sufficient available area for the remaining systems.



(e) Design problem 2: final layout generated by Team 8.



(f) Design problem 2: final layout generated by Team 9.

Figure 4.14: Six of the twenty final layouts generated during the experiment.

Design problem 1: Teams 4 and 5

The final layouts of these teams are shown in Figures 4.14a and 4.14b, respectively. As said, these layouts scored the same score for AMoP and LMoP but had a different SMoP. For the Safety Specialist, the main consideration is in which relative compartment systems are arranged. From that perspective, the two layouts are very similar, despite being mirrored. The main difference is the location of the Waste Store and Food Store 2. Based on the prescribed interactions, Team 4 made a better trade-off from an SMoP perspective. However, Team 5 seems to have preferred reduced logistical movement in the food preparation process by locating a Food Store close to the Galley. It is noteworthy that Team 5 did use the design rationale method for this design problem.

Design problem 2: Teams 2, 8, and 9

From a safety perspective, Teams 2 and 8 created the same design. In contrast, Team 9 made a different trade-off in five of eight safety-related constraints. For instance, Team 9 decided to separate the Galley and Mess. Also, Team 9 differentiated from Teams 2 and 8 because it kept the default passageway size. As a result, less area was available to arrange systems, which is reflected in the higher AMoP (34, compared to 18 and 19 for Teams 2 and 8, respectively). Although the layouts are somewhat mirrored, the difference between the LMoP for Teams 2 and 8 is small (53 and 54, respectively). Team 9 achieved a better LMoP, scoring 49. The team also expressed five additional interactions after the arrangement was finished, which explain some of the design rationales behind the layout:

1. Galley and Food Store 1 are adjacent: *"Access to the Food Store via the Galley to enable the Galley to be larger."*

2. Mess and Galley are adjacent: *"Although in different compartments, connectivity is good."*

3. Emergency Generator Room and Galley must be separated: *"Subordinate to other noise-related separation constraints."*

4. Mess and Galley are separated: *"It's not possible to arrange both systems in the same compartment without introducing additional conflicts."*

5. Mess and Medical Room should be separated: *"Solve noise issues with insulation?"*

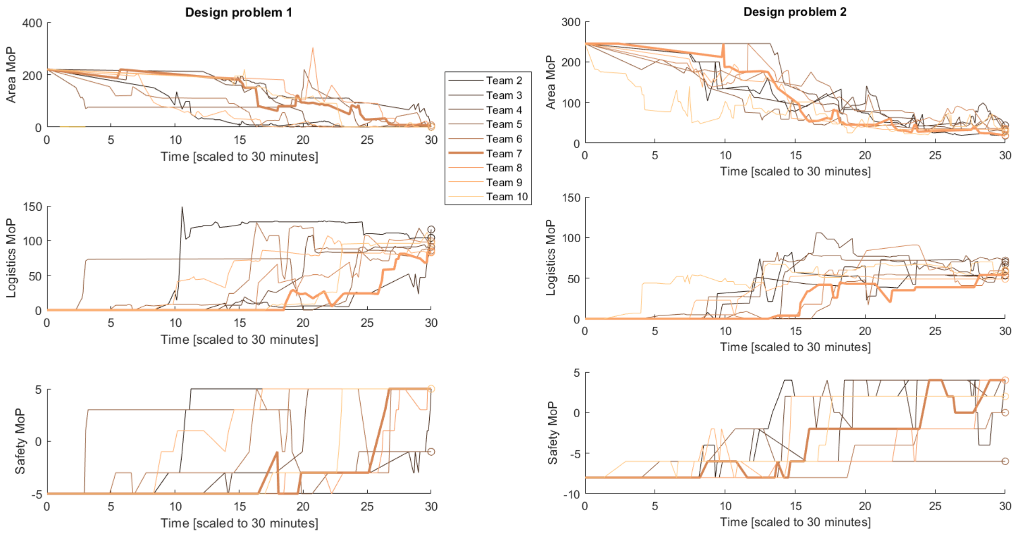
Using such additional rationales can help to better understand the team's intentions with the concept design, as well as indicate what future changes might be necessary to improve the layout.

TEMPORAL ASPECTS

Third, the development of concept designs over time is investigated. Figure 4.15 shows the development of MoPs over time for all teams (except Team 1) for the two design problems. The following observations can be made:

1. The MoPs show convergence over time and limited rework of the layout, i.e. many local adjustments were made. In Section 4.4.1, it was observed that in later phases teams seemed to focus on finalising the layout, such that all systems fit. Major decisions were taken during the early phases. This could explain, for instance, why the SMoP has many plateaus: once systems are positioned in preferred compartments, and teams keep to these decisions, the SMoP will not change.

2. MoPs can indicate when alternative arrangements are made later in the design process. For instance, see the highlighted Team 7 for design problem 1 and Teams 3 and 7 for design problem 2 in Figure 4.15. These teams moved systems across compartments relatively late in the design process, which can also be observed in the MoP traces. For example, Team 7 compromises LMoP for a significant improvement in SMoP at 25 minutes of the experimental round (design problem 1).



(a) (b)

Figure 4.15: Development of MoPs over time. Team 7 is highlighted.

3. The definition and implementation of MoPs can limit the amount of insight into the overall development of the concept design over time. For instance, both Logistics and Safety MoPs cannot consider interactions when systems are outside all compartments. Hence, the current LMoP and SMoP will not be able to provide absolute performance over the complete design process but can only be used to compare concept designs with the same systems arranged inside compartments. This problem could be partially addressed by adding a penalty to these MoPs when any system in an interaction is outside all compartments. However, useful MoP information may get obscured if penalty values are of the same order as the non-penalised version of the MoP.

4. In some cases, teams used a relatively long time to marginally increase the quality of the concept design. Both in post-experiment discussions and in the questionnaire, participants did indicate that the method could be more realistic if it could provide or be used to get insight into the costs and benefits (e.g. regarding material, time, and effort) of design changes.

CONCLUSION FROM QUALITY OF CONCEPT DESIGNS

Concluding, from a design quality point of view, the second design problem is more difficult. Specifically, this was due to the limited available area. Also, the results indicate that MoPs are valuable metrics to provide insight into the quality of the concept designs. However, a detailed manual evaluation of the concept designs is still required. Lastly, the results do not indicate that the design rationale method directly leads to qualitatively better concept designs compared to the baseline method. However, this could also be caused by the simple design problems used in this experiment. More complex design problems with multiple stakeholders might show more benefits.

4.4.4. SATISFACTION WITH GENERATED CONCEPT DESIGNS

The questionnaire was used to elucidate participant satisfaction with generated concept designs. For each round, participants were asked to respond to the following three statements:

1. *I'm satisfied with the layout from my role's perspective.*
2. *I'm satisfied with my input in the decision-making.*
3. *My input in the decision-making has been satisfactorily incorporated in the final design.*

Figure 4.16 presents the satisfaction of participants with the three statements presented above. Generally, participants were satisfied with the outcome of the design process. There is little difference between the expressed satisfaction across the use of the two methods since the balance between the responses Agree and Strongly Agree is almost equal for both methods.

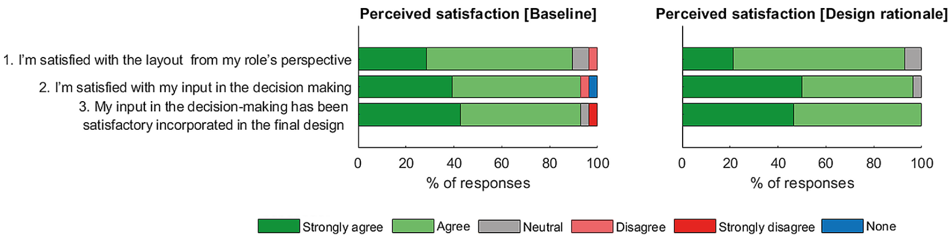


Figure 4.16: Satisfaction with quality of concept designs - comparison between baseline and design rationale method (n=28).

Only one person expressed dissatisfaction with the three statements when using the baseline method. This participant (the Logistics Specialist in Team 8) was unsatisfied with the outcome of design problem 2. From an MoP perspective, this is notable since Team 8 achieved a good performance from a logistics perspective (third best of all teams), see Section 4.4.3. This Logistics Specialist proposed to switch the positions of Food Store 1 and the EGR in the layout shown in Figure 4.14e. The Naval Architect (who operated the tool) objected without argumentation to implement this proposed change. Implementing this change would have resulted in an improvement of the LMoP by 3.9 for a reduction of 3.2 of the AMoP. However, this trade-off between Logistics and Area was not further discussed by the team. Hence, the question is whether the Logistics Specialist was dissatisfied with the layout, the team process, or both. Also, it would be interesting to know whether the satisfaction would be different if the actual MoP values were known to the team. Interestingly, this Logistics Specialist, when asked to describe how the design rationale method supported decision-making, replied:

> *"[It forces] a holistic approach to the design problem, instead of [allowing] for alpha behaviour to push your own interest."*

So, this participant still perceived the design rationale method to indeed support the collaborative design decision-making process.

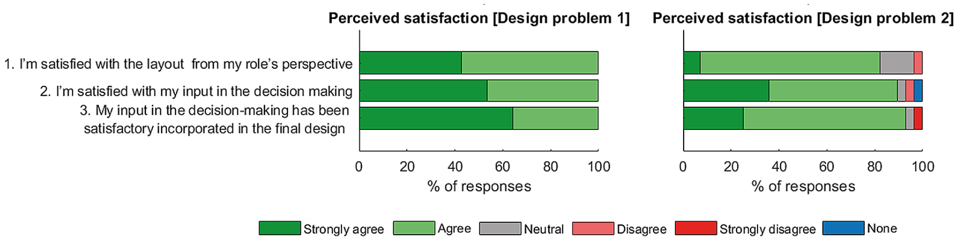


Figure 4.17: Satisfaction with quality of concept designs - comparison between two design problems (n=28).

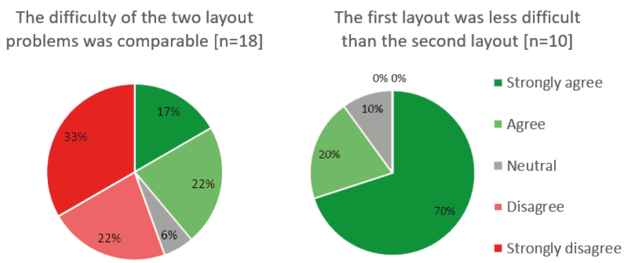


Figure 4.18: Perceived relative difficulty of design problems

Figure 4.17 presents the same satisfaction of participants, yet differentiated between the two design problems, instead of methods. For the first design problem, all participants expressed to be satisfied with the concept designs. 64% strongly agreed that the input was satisfactorily incorporated in the concept design. For the second design problem, most participants are satisfied with the outcome of the design process. Compared to the first design problem, fewer participants strongly agreed with the statements. A possible explanation is the difficulty of the design problem. As shown in Figure 4.18, most participants experienced dissimilar difficulties across the two design problems (left), of which the second design problem seemed to be more challenging (right)[5]. Indeed, in the second design problem, significant compromises were needed regarding system sizing, while the first design problem mainly contained concessions regarding relative positions (i.e. logistics and safety). This was already shown in Section 4.4.3.

Concluding, the analysis in this section indicates that participant satisfaction with the generated concept designs was generally good and is dependent on the difficulty of the design problem. Furthermore, the single case where a participant was not satisfied with the outcome supports the conclusion of Section 4.4.2.

[5]This question was changed during the execution of the experiments, hence the split in questions and *n*.

4.4.5. PERCEIVED BENEFITS

In this section, the participants' perception of the design rationale method is investigated. This investigation is done based on the following open questions from the questionnaire:

1. *Which functionalities of the design rationale method were most useful to the design case?*
2. *What additional functionalities of the design rationale method would be beneficial?*
3. *Would the design rationale method be beneficial for design (review) sessions, and why?*

The responses to these three questions are coded and visualised in Figures 4.19a to 4.19c, respectively.

4

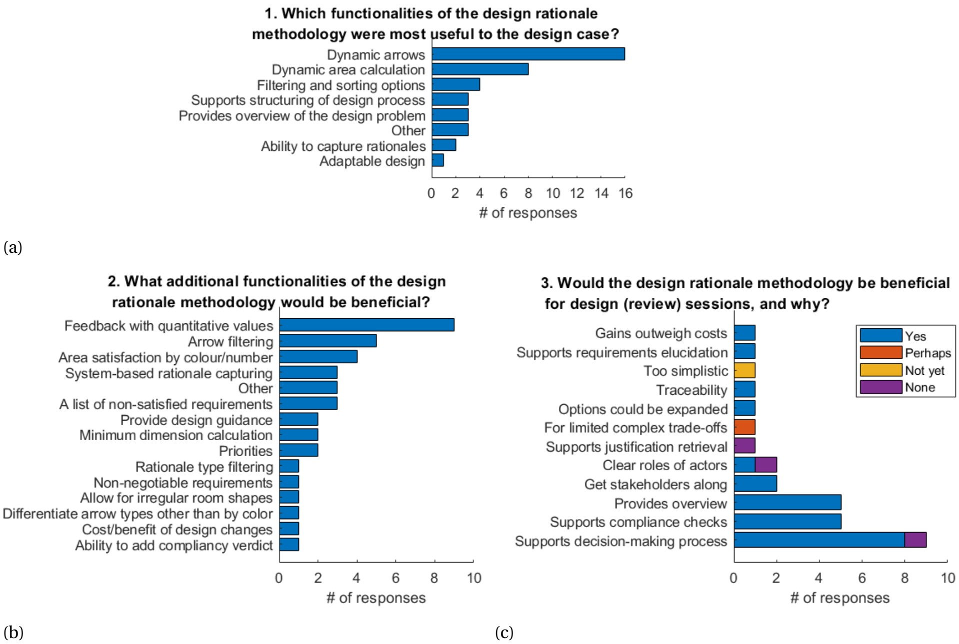


Figure 4.19: Perceived benefits and required additional functionalities of the design rationale method

First, participants were asked which functionalities of the design rationale method were most useful during the experiment. Based on Figure 4.19a, the dynamic annotations (i.e. arrows and area calculation) were mentioned 24 times in responses to this question. Four participants found the filtering and sorting of design rationale useful. Capturing design rationale itself was mentioned only 2 times. A possible explanation is that the action of design rationale capture takes much effort and only becomes useful when the captured design rationale is used, for instance, via design rationale retrieval and dynamic annotations to create an overview of the design problem (n=3).

Second, the questionnaire was used to elucidate any additional functionalities thought to be beneficial to the design rationale method (Figure 4.19b). Participants would like to receive quantitative feedback (n=9), visual and textual feedback on area satisfaction (n=4), and a list of non-satisfied requirements (n=3). Furthermore, five participants indicated that filtering the arrows would be beneficial. In a post-experiment discussion with one of the teams, participants explained that they would rather have annotations on demand (i.e. to show relevant parts of the interactions network), instead of the visualisation of the entire network as currently implemented. For the relatively small design cases in the experiment, the network was already quite extensive, see Figure 4.5. Three participants indicated that the ability to capture design rationale related to single systems is missing. For instance, Team 5 decided to exclude the EGR to solve the mismatch between available and required areas (Section 4.4.3) and commented that they were not able to capture this decision and justification by the design rationale method.

Third, while Question 1 asked for the benefits of the design rationale method to the *design case in the experiment*, Question 3 required participants to consider the use of the design rationale method in *actual design (review) sessions*. Participants were asked: "Would the design rationale method be beneficial for design (review) sessions, and why?" Figure 4.19c shows that most participants answered 'yes' but two participants considered this 'not yet' or 'perhaps' the case. Three participants only provided an open response (i.e. 'none'). The doubting participants participated in the same team and considered the design rationale method 'too simplistic' or 'maybe applicable for design problems with limited complexity'. This leads to the question of whether group experience is related to design tool acceptance. Most participants considered the design rationale method beneficial to design (review) sessions because it would support the decision-making process (n=9), support compliance checks (n=5), and provide an overview (n=5).

Concluding, participants were generally positive about the benefits of the design rationale method, both for the design case and actual design (review) sessions. Further, the participants expressed potential additional functionalities.

4.5. DISCUSSION

Although the experiment results show the benefits of the developed design rationale method, currently, the method is tailored to the design experiment. Therefore, the range of implemented design rationale types was limited. To make the design rationale method more suitable for actual ship design problems, attention must given to the representation and capturing of realistic ship layout design decisions, for instance, by expanding the interaction definition and inclusion of system properties and compromises (DeNucci, 2012; le Poole et al., 2022c). This will be addressed in Chapter 5.

The conducted experiment has some limitations as well, namely:

1. Although the design rationale method is implemented in Rhinoceros, an integration with actual ship design tools is currently missing. Therefore, the interplay between ship design tools and the design rationale method could not be investigated. In other words, the research assumes that the 'manual' design work is similar to using a human-centric ship layout design tool, such as DBB (Andrews & Dicks, 1997) or FIDES (Takken, 2009). Although such design tools provide more functionality,

the observed use of the design rationale method indicates that participants used it partly to 'sketch' during the negotiation process. Applying more computer-centric ship layout design tools, such as Packing (Van Oers, 2011b), will likely need a different implementation and process than presented here and is probably more focused on exploration and post-processing (DeNucci, 2012; Duchateau, 2016), see also Section 3.3. Also, the long-term (i.e. multi-session) effects of applying the design rationale method have not been evaluated. However, some of the beneficial functionalities of the design rationale method mentioned by participants also apply to the multi-session use of the method. For instance, it provides an overview, traceability, and justification retrieval, and it helps to involve stakeholders.

2. Design considerations were typically verbally discussed by design teams, but not always supported by, for instance, 'network arrangement'. To get insight into such discussions leading up to the capture of a rationale, audio or video recordings of design sessions could be a useful data source to get further insight into the role of design rationale in collaborative design decision-making.

3. Both groups of participants (experts and students) have their limitations concerning the experiment. On the one hand, experts are likely biased by their own experience with actual ship design processes and are likely to have reflected their thoughts in their responses (Andrews, 2022a). For instance, the participants doubting the usefulness of the design rationale method for actual design implicitly relate their response to their view of ship design. For example, one participant said the design rationale method is 'too simplistic'. Since ship design is much more complex in reality, this participant doubted if the method would stand in such a more complex environment.

   On the other hand, students are likely to lack ship design experience. Some of them will have studied ship design before, but some of the students might have joined the Maritime Engineering master program from a non-ship design background. Also, the demographic and corresponding cultural diversity between students is likely higher than in the expert group. This might contribute to significantly different group dynamics - and perhaps different outcomes of the design process. Due to the absence of audio recordings, this aspect could not be further investigated. The combination of students and experts is thought to give balance to the potential bias towards own experience and the perception of the design rationale method.

4. Unfortunately, the number of participants in the experiment was limited. It proved especially hard to recruit students. As a result, statistically significant results could not be obtained. Nevertheless, the quantitative and qualitative results obtained in the experiment give valuable indications for further development of the design rationale method.

4.6. CONCLUSION

The availability of design rationale, i.e. the justifications behind design decisions, in complex ship design is key as a knowledge base for the multiple participating actors and to performing informed iterative design. Therefore, the main research question for Part II set in Section 1.2 is: *"To what extent can design rationale methods support real-time design decisions during early-stage complex ship design?"* To answer this question, this chapter provided a gap analysis in Section 4.1. It was concluded that there is currently no design rationale method that is directly suitable to support design rationale capture and retrieval in collaborative complex ship layout design. Hence, the goal of this chapter was twofold, namely:

1. To develop a design rationale method to aid designers in the continuous capturing and reuse of design rationale during the collaborative concept design process, and

2. To evaluate how the developed design rationale method benefits collaborative design decision-making such that it leads to better insight into design issues across design teams and better concept designs during a single design session.

To fulfil the first goal, a new design rationale method was proposed in Section 4.2. The method was implemented as a proof-of-concept to demonstrate how design rationale might be captured and reused during the design of ship layouts. The method integrates a record-and-replay design rationale capturing approach with a layout design tool. A subset of DeNucci (2012)'s design rationale representation scheme is used. This method provides both short-term benefits (e.g. it enables the creation of a common knowledge base during design sessions) and long-term advantages (e.g. it allows the review of the context of past design decisions before changing a concept design).

To fulfil the second goal, a small-scale experiment was developed and executed, as described in Sections 4.3 and 4.4. Teams comprising three participants (i.e. students and experts from the industry) worked on two small-scale layout problems. The results of the design experiment indicate that using a design rationale method while designing a layout can have both measurable and perceived benefits. An example of the former is that the design rationale method motivates teams to use 'network arrangement', as indicated by the results (Section 4.4.1). Such network arrangement of systems visually supports the team in sketching the initial arrangement of systems. Participants generally perceived the design rationale method to facilitate enrichment and negotiated knowledge (Section 4.4.2), aspects aiding to provide a better understanding of the design problem within the entire design team. The results do not indicate that the design rationale method directly leads to qualitatively better concept designs compared to the baseline method. However, this could also be caused by the simple design problems used in this experiment (Section 4.4.3).

As such, this chapter shows that design rationale methods might be beneficial to support real-time design decision-making in small teams. Table 4.5 summarises the method's compliance with the requirements set in Section 4.1.1. All method requirements have been partially or completely fulfilled. However, to further enhance the proposed method, the following topics are deemed relevant for further investigation:

1. Further developments of the design rationale method, which include: a larger variation of design rationale types, including System Properties; filtering of annotations; and filtering based on keywords. Furthermore, the applicability of the design rationale method in multi-session design for larger design problems (e.g.

Table 4.5: Proof-of-concept (PoC): compliance with method requirements.

ID	Requirement	Compliance PoC
DRQ1	The method must be applicable for early-stage collaborative design activities and promote feedback-driven conversations	Partially fulfilled (Section 4.2) - An evaluation in a more realistic ship design example is currently lacking.
DRQ2	The method must enable the capture and review of design decisions, the rationale behind these decisions, and temporal relationships between design decisions	Partially fulfilled (Section 4.2.2) - The current set of design rationale types is too limited.
DRQ3	The method must provide immediate rationale-based feedback to increase the benefits relative to the costs of capturing design rationale	Fulfilled (Sections 4.2 and 4.2.3)
DRQ4	The method must be generic	Partially fulfilled (Sections 4.2-4.4) - Especially the method's scalability and usability in a more realistic ship design example is currently lacking.
DRQ5	The method must be easy to use and integrated within design tools	Partially fulfilled (Section 4.2.1) - An integration with an actual ship design tool, suitable for overall ship synthesis, is lacking.

full ship size), the reuse of design rationale between iterations at various levels of design (e.g. macro, major, micro), and the integration with actual ship design tools have to be investigated. Attention must be paid to the role of layouts in the overall ship synthesis process. For instance, how can the method help to capture rationale related to resistance, style, or identified design drivers? This item is addressed in Chapter 5.

2. The development of a process description to guide designers in exploiting the opportunities of the design rationale method. For instance, Team 9 captured additional design rationale after finishing the design to explain design choices, i.e. performed reflection on the final design (and design process). Prescribing such steps would guide all users in how to best use the design rationale method. This item is also addressed in Chapter 5.

3. An evaluation of the usefulness of MoPs during actual design work. How do designers use these MoPs in practice, and how to avoid the excessive focus on optimising MoPs, as emerged during the practice run of the experiment? Also, which design rationale-based MoPs are suitable for real ship design, and how to ensure these consider the right set of design rationale? This is left for future work, see Chapter 6.

These developments are expected to enhance the method. Yet, the method should be used judiciously as the naval architect is, ultimately, responsible for design choices. As such, any insight and information provided by the design rationale method, as with any other source of information, should be carefully considered.

5

The Ship Design Rationale Method

[A Computer Aided Ship Design system] must be interactive with all the decisions consciously made by the designer (...). By so ensuring that the designer is aware, each time, of the decisions he is making, the issues these aspects raise are brought out into the open. Thus the choices can be questioned and altered if they are felt to be significant for the particular design or have the potential to reveal a better design solution.

D.J. Andrews (1985)

Chapter 4 discussed a proof-of-concept of a new design rationale method for collaborative design decision-making during early-stage complex ship design. However, development and evaluation of this proof-of-concept method for use in realistic ship design scenarios were needed. Therefore, in this chapter, this design rationale method (now named the Ship Design Rationale Method (SDRM)) will be extended and evaluated to fulfil the following twofold goal, as extended from Chapter 4:

1. To develop a design rationale method to aid designers in the continuous capturing and reuse of design rationale during the collaborative concept design process.
2. To evaluate how the developed design rationale method benefits collaborative design decision-making such that it leads to better insight into design issues across design teams and better concept designs over time for realistic ship design problems.

This chapter is structured as follows. In Section 5.1 the initial extension of the SDRM is presented. Then, Section 5.2 demonstrates how the SDRM can be used in the design

Sections 5.1 and 5.2 are based on Le Poole et al. (2022c) of an Oceangoing Patrol Vessel (OPV). Subsequently, further enhancements and implementation of the SDRM are discussed in Sections 5.3 and 5.4. Finally, the long-term benefits of the SDRM are evaluated in another design experiment, described in Section 5.5.

5.1. DEVELOPING AN ONTOLOGY FOR THE SHIP DESIGN RATIONALE METHOD

In this section, the proposed design rationale method is further developed to fulfil the requirements set in Chapter 4 . Specifically, the focus is on the question: what design rationale needs to be captured and how should this design rationale be represented?

Lee (1997) expected that rationale representation would not provide large challenges, compared to, for instance, cost-effective implementation of design rationale systems. For this reason, the rationale representation for ship layout design developed by DeNucci (2012) has been adopted as a starting point for the SDRM's rationale representation (as in Section 4.2.2).

The following design rationale types are defined:

1. A **System Property** is a (required or actual) quality or characteristic of a system and its justification. Examples of System Properties are required sizing (e.g. volume, area, aspect ratio, alternative positions) and preferred global positions (e.g. on deck 3, as high as possible).

2. An **Interaction** is a preferred (spatial) relationship between two (or more) systems (DeNucci, 2012), or System Properties. Additionally, an interaction comprises its justification, see Section 4.2.2.

3. A **Compromise** is the preferred solution to a set of conflicting or competing interactions (DeNucci, 2012) or System Properties, and its justification. For example, consider the required connectivity between a helicopter deck and the medical room in a frigate, depicted in Figure 5.1. This situation comprises three interactions (1-3). The related compromise is the preferred set of interactions, where the naval architect is to choose between set A = {interaction 1} and set B = {interactions 2 and 3}. Choosing set A might be justified by reasons such as 'reduced spatial impact' and 'less time to medical room'.

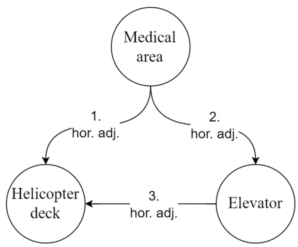


Figure 5.1: Required connectivity between a medical room and a helicopter deck, comprising a compromise between interaction 1 and interactions 2 and 3.

4. Designers can use **Notes** to explain design considerations not directly related to systems. An example is a decision to increase the hull width to increase transverse stability. Note that such decisions might be followed by system-related decisions, e.g. systems might be rearranged to make use of the new available space.

Figure 5.2 shows a schematic overview of the proposed design rationale types and interrelations in an ontology. Note that the network representation is only a possible form of presenting the set of rationales. The top level represents a (digital) model of the concept design, comprising several systems and subsystems. The System Properties define the concept design, mainly describing *what* is designed. Interactions and Compromises principally describe *why* the design is what it is. For each element, changes over time are traced (even if no rationale is expressed) to allow for a complete review of the design process.

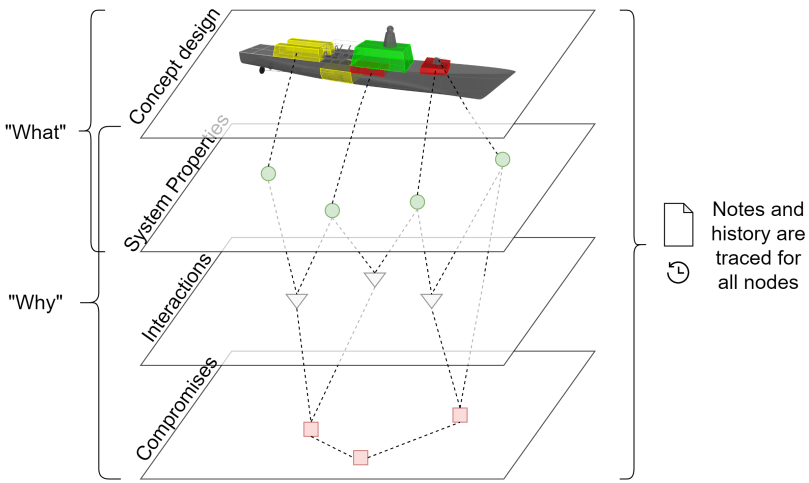


Figure 5.2: High-level visualisation of the developed ontology to show the interrelationships between ontology elements.

To each design rationale element, a *status*-indicator is added. This *status*-indicator can be used to express confidence or satisfaction for each object in a category, and its justification. The status can be set to:

1. *Agreed*, meaning: “At this moment, the current realisation of this object is satisfactory”.

2. *Non-agreed*, meaning: “At this moment, the current realisation of this object is not satisfactory, and to be improved”.

3. *Pending*, meaning: the status has not yet been expressed. When some property of a rationale is changed (e.g. the position of a system), the status might be automatically set to pending. In such cases, designers will need to express an explicit status for each element to ensure all design changes are eventually described.

Additionally, designers can express design rationale via a design impact indicator, and its justification. Such indication can be used, for instance, to review the priority of design decisions when design changes are required. At the moment the following three design impact indicators are included:

1. *Global* impact, meaning: This design aspect (i.e. System Property, Interaction or Compromise) impacts the overall ship arrangement. An example is an engine room, which might influence bulkhead positions, exhaust positions, and topside arrays such as radars.

2. *Regional* impact, meaning: This design aspect impacts significant portions of the ship arrangements, e.g. the arrangement of systems within a zone or compartment.

3. *Local* impact, meaning: This design aspect impacts the ship's arrangement only locally. For instance, the internal arrangement of the galley dictates how the galley is shaped, which impacts the directly adjacent systems only.

The combination of status and design impact indicators conveys the context-based priority of rationale elements. Such context is considered important to enhance the designers' ability to set coherent and meaningful priorities, in contrast to, for instance, numerical priorities. Such numerical values would need additional justification (Lehtola et al., 2004).

Besides the static description of *what* and *why*, the ontology needs to include temporal relationships (DRQ2). However, this is relatively easy as each design change and expressed design rationale can be given a timestamp when stored. No specific designer input is required here. Long-term storage of design rationale can be achieved via various types of databases.

5.2. CASE STUDY 6 - EVALUATING THE ONTOLOGY

To further evaluate the usefulness of the developed design rationale method, a preliminary version of the rationale ontology was used on-the-fly while designing a notional Oceangoing Patrol Vessel (OPV).

5.2.1. CASE STUDY SETUP

The OPV concept design is based on the design assignment provided in an earlier MSc-level design course lectured at TU Delft. This assignment was originally to be completed in groups of three students. Although system selection based on (overambitious) mission statements was part of the assignment, a predefined list of systems based on student reports was used as a starting point of the exercise. The total list with systems can be found in Appendix C.2. An arbitrary OPV hull size was taken as the starting point for the arrangement process.

Since the extended design rationale method was not yet implemented in code, a 'sheet of paper' exercise was carried out for this dissertation. Note that this limits the ability to evaluate the design guidance support service, presented in Section 4.2. The design rationale elements (e.g. System Properties, Interactions, and Compromises) are captured in Excel tables. An Excel and Rhinoceros-based design tool (Surface Ship Design Tool (SSDT)) was used to visualise the concept design, as well as to calculate system areas and volumes. The SSDT is loosely based on FIDES (Takken, 2009) and precedes the design tools presented by Kana and Rotteveel (2018) and Kana et al. (2022). Section 5.3 discusses the SSDT in more detail.

The case study was comprised of the following three steps:

1. **Capture prescribed rationales**. The design assignment prescribed System Properties, Interactions, as well as Compromises. System Properties comprise required sizing and global positions (e.g. *The floor of the cabins may not be situated more than 1 meter below the waterline*). An example of a prescribed Interaction is "*The Replenishment At Sea (RAS) stores need to be located in the vicinity of the RAS mast. Good logistics between the RAS mast and the stores is essential for smooth RAS operations with a reduced manning*". An example of a Compromise is the choice between "*medical area should be horizontally adjacent to the flight deck for medical reasons*", and "*medical area should be connected to the flight deck by an elevator*", see Figure 5.1.

2. **Concurrent arrangement of systems and capture of additional design rationale**. This step is comprised of two sub-steps:

   (a) Capturing design rationale *prior* to arranging systems. When a system was considered, the designer considered possible global and relative positions and captured these. When the designer was convinced about the considerations, the system was arranged subsequently. The expression of design rationale was found to support thinking about the design options, similar to sketching (Pawling & Andrews, 2011).

   (b) Capturing design rationale *after* arranging systems. This was done in cases where the preferred location of systems seemed constrained. Arranging systems first allowed for a quick check of whether the location was feasible.

3. **Analysis of developed design rationale network**. Networks are well suited to visualise the structure of objects and relations, such as design rationale. See for example DeNucci (2012) and Gillespie (2012) The design rationale network and concept design generated through the previous two steps were analysed to identify possible areas of attention for future research.

5.2.2. RESULTS

Using the rationale ontology was found to be rather straightforward (contributing to fulfilling DRQ5). It allowed the capturing of all rationales that were considered during the exercise. It also allowed easy reconsideration of earlier captured rationales (DRQ2). The main challenge using the method was the lack of linkages in the data, which required manual referencing between objects. Therefore, the endpoint of the exercise was considered to be the identification of the required descaling of the hull to reduce void space - as the additional required manual referencing was not sufficiently supported.

Table 5.1 summarises the growth of the network of captured System Properties, Interactions, and Compromises, as well as their relation to systems. The growth of the network indicates the importance of using useful ways to navigate and retrieve information. This is especially important to reduce information overload and to enable designers to show only the information that is important for the stakeholder dialogue.

Table 5.1: Design rationale network growth

Number of	Pre-arrangement	Post-arrangement
Systems	99	106
System Properties	11	43
Interactions	29	51
Compromises	3	9

Referring to DRQ5, the integration of a design rationale method and design tools is required to better capture the context of design decisions. Besides *capturing* design rationale inside the design tool, two concepts to visualise the captured design rationale in the context of the concept design were investigated:

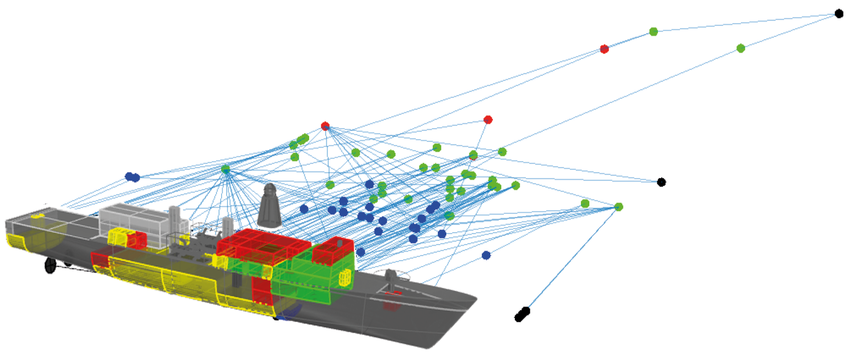
1. **Display the design rationale network, comprising System Properties, Interactions, and Compromises as layers over the concept design**, see Figure 5.3a. This implementation showed that the network might be hard to navigate. Although single-layer networks have been visualised over concept designs by others (e.g. Roth, 2016), the multi-layer ontology might be too complex for quick comprehension. Another constraint is the relatively limited information that can be expressed in node names without cluttering the network view.

2. **Visualise the design rationale network separately from the concept design, but with 'data on demand'**. This data encompasses more elaborate information behind nodes, which can be highlighted by navigating the network, as visualised in Figure 5.3b. This allows designers to see the network more effectively, compared to the visualisation above, and retrieve detailed information when required. In addition, it addresses the two constraints identified above.

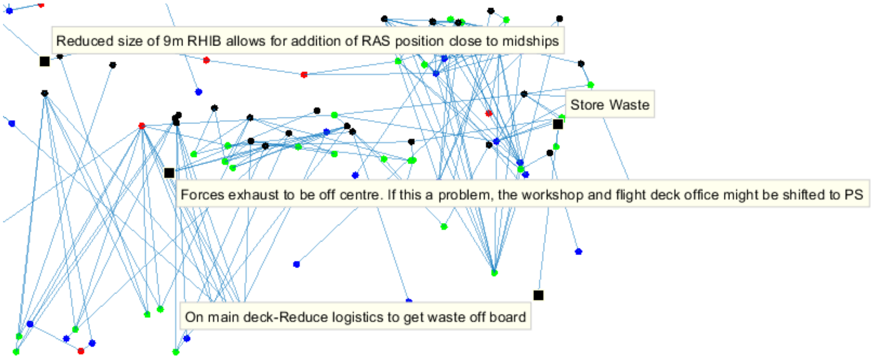
During the case study, the following three lessons learned were identified:

1. **Formalisation of interaction types** is required. This would improve the consistency of captured rationales, and support the development of rationale-based design guidance. Also, this supports designers by limiting the required effort to think about design rationale representation (Section 4.1.2).

2. The growth of the network and the high fidelity of the captured rationale require further investigation of **'data on demand'**. The designers should be able to change perspectives on the captured data. For instance, designers should be able to gain an overview of the whole network as well as have access to all details of a single rationale. Alternatively, the rationale could be presented as a timeline, which communicates which rationales have been considered and at what time step. In the current implementation, such a timeline is not well supported since design changes are not individually stored.



(a) Rationale network overlaying the OPV concept design. System nodes outside the concept design represent systems that have not yet been arranged.



(b) Data on demand, as demonstrated on a partial network. Node selection reveals detailed information on that node. Selected nodes are indicated by black squares.

Figure 5.3: Rationale network visualisation. Legend: Systems (black), system properties (blue), interactions (green), compromises (red).

3. **Trace design changes**, and automatically indicate whether these changes have been justified. This could support designers to add rationales at a late stage (e.g. after a collaborative design session) or allow for an evaluation of the design process.

Implementation of the full method is required to enable the application of the method in collaborative design sessions. Subsequently, the method can be fully evaluated. This will be the subject of the remainder of this chapter.

5.3. Enhancing the Surface Ship Design Tool

One of the requirements for the SDRM is to be integrated with a ship layout design tool (DRQ5). In Chapter 4 the choice between WARGEAR and a manual design tool was substantiated, see Section 4.2.1. The SDRM is developed mainly to support the design of the lower detail functional arrangement.

While the experiment described in Chapter 4 did not use a ship design tool, in the case study presented in Section 5.2 the SSDT was used. Since the SSDT enables designers to generate arrangements similar to the intended functional arrangement, the SSDT is seen as an acceptable representative ship layout design tool for this research.

To enable a full integration between a layout design tool and the SDRM, the decision was made to redevelop the SSDT from a coupled Excel-Rhinoceros code into a Rhinoceros-only application. This would allow for a similar Python-based implementation as the GUI developed in the proof-of-concept in Chapter 4. This could help achieve lower intrusiveness of the design tool-design rationale method combination and enable designers to focus more on the design task at hand. Also, this could allow for smoother integration between the design tool and SDRM.

In its core, the SSDT allows the designer to arrange planes (e.g. a helicopter deck), volumes (e.g. fuel tanks), and 'secondary objects' (e.g. detailed engine models) within a predefined, scalable, and selected hull form. These arrangeable objects, i.e. Volume Blocks, are arranged with respect to X-, Y-, and Z- reference planes. These reference planes can represent bulkhead and deck positions. The SSDT generates a 3D arrangement drawing and calculates a range of output data, including object area and volume, ship lightship and variable masses, and intact stability. Since the number of design parameters and options in the SSDT does not easily fit on one screen, the decision was made to distinguish between 'main sizing' parameters (e.g. hull selection and sizing, reference planes, block types), 'volume blocks' (i.e. the main systems in the concept design), and 'output' (e.g. details on weight and stability), see Figure 5.4.

To enhance the flexibility of the SSDT and to enable designers to capture design variations, the following two capabilities have been added:

1. **Design instances**: A design instance is a description of a particular state of the concept design. The only way to store design progress, design variants, and such, in the SSDT is by using design instances. Each design instance can be reloaded, changed, and subsequently restored as a new design instance. Also, the volume blocks of pairs of design instances can be merged. This allows designers to work on different aspects of the design separately. This could be useful in larger design settings (e.g. a naval architect working with various specialists at separate parts of the concept design at different moments in time).

   Thus, a design instance allows designers to capture the concept design as is, and work on that particular concept design at a later stage. All stored design instances in a project are shown in a collapsible tree grid in the SSDT and sorted by date and time, see Figure 5.4. An explaining description *must* be provided. This enables designers to differentiate various design instances, also at later stages. Such mandatory explanation is also intended to stimulate the expression of design rationale. Compared to the previous Excel/Rhinoceros implementation, this allows designers to store and explore more design variations as well as high-level justification of these variations. Furthermore, this allows for increased reproducibility of developed concept designs.

2. **Design sketching**: A new feature to the SSDT is 'design sketching'. Once an initial design is generated, the designer can scale non-secondary objects and drag

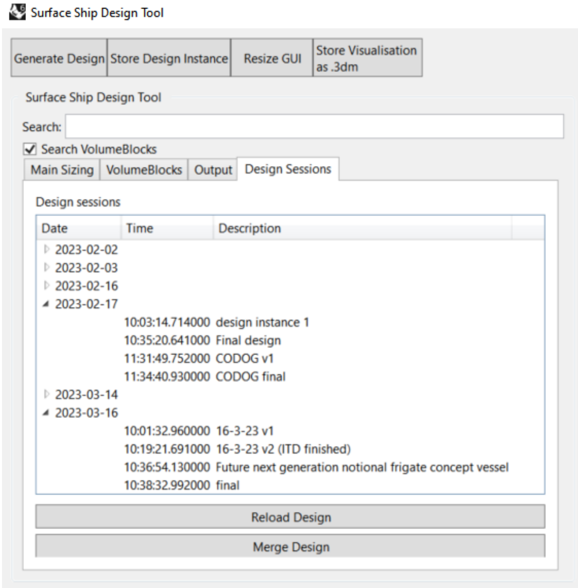


Figure 5.4: Example of captured design instances, sorted by date and time. Also, brief descriptions are shown.

Volume Blocks in the visualisation by using Rhinoceros' gumball functionality to quickly rearrange the layout, as illustrated in Figure 5.5. The new positions are automatically updated in the Volume Blocks table in the GUI, concerning the currently selected reference planes. This allows designers to change the concept design more interactively. To update the weight calculation as well as block trimming after sketching, designers should regenerate the design.

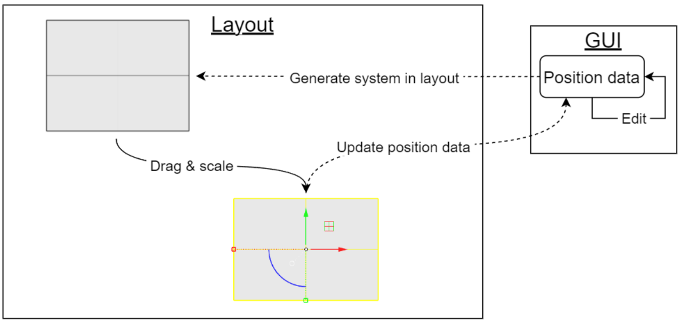


Figure 5.5: Visual explanation of design sketching. Dashed lines indicate automated actions in the SSDT. Solid lines indicate designer actions.

The developed SSDT allows designers to generate and discuss ship layouts (i.e. external cognitive objects, Section 1.1.1) to support both technical design aspects and the social interaction between designers and other actors.

5.4. FINAL SHIP DESIGN RATIONALE METHOD (SDRM)

In this section, the final developments of the SDRM are elaborated. As such, this section builds upon Sections 4.2 and 5.1 to fulfil the first goal for this chapter: *To develop a design rationale method to aid designers in the continuous capturing and reuse of design rationale during the collaborative concept design process.* First, Section 5.4.1 presents the developments of the rationale ontology and the SDRM. Then, Section 5.4.2 elaborates on the integration between the SSDT and SDRM-GUI. Finally, Section 5.4.3 presents the proposed use of the SDRM during a single design session, in response to the findings in Section 4.5.

5.4.1. UPDATES TO THE SDRM AND RATIONALE ONTOLOGY

Based on the findings in Chapter 4 and Section 5.2, the following elements have been changed, edited, or removed from the initial SDRM presented in Section 5.1:

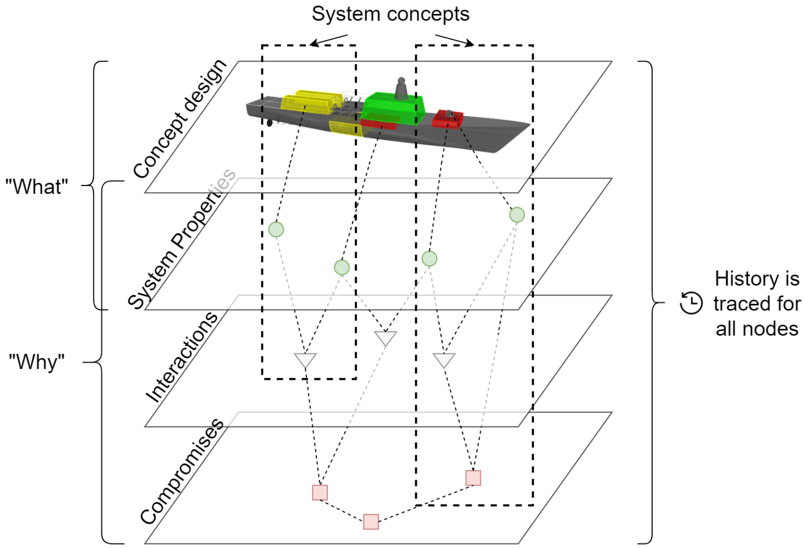


Figure 5.6: High-level visualisation of the extended ontology to show the interrelationships between ontology elements.

1. The **rationale ontology** was expanded, as shown in Figure 5.6. Across the ontology, *system concepts* can now be captured. A system concept is defined as is a set of systems with the rationale behind these systems. System concepts were added because it was recognised that there are many cases in which a decision is made regarding a set of systems. A designer may add design rationale to system concepts, such as system properties or interactions. Examples of system concepts are, a propulsion system concept comprising multiple engines, gearboxes, shafts, and propellers (e.g. Figure 5.7), or an accommodation system concept comprising a set of individual cabins and sanitary units (e.g. Figure 3.9).

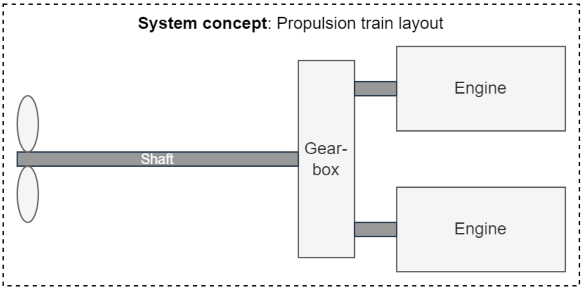


Figure 5.7: An example of a propulsion system concept.

2. The range of **rationale types** is expanded, as a formalisation of interaction types was identified as necessary in Section 5.2.2.

   (a) Besides the status and design impact indicators (Section 5.1), a numerical priority indication (ranging from 1-10) has been added, in line with Section 4.2.3. Although more abstract than the status and design impact indicators, the priority scale might be used as an intermediate means to capture differences in importance. The justification or notes of each design rationale element can be changed to provide additional insight into the reasoning behind the numerical priorities.

   (b) Table 5.2 provides an extensive overview of the various underlying items and corresponding allowed values for the four main design rationale types currently in the SDRM. See Table 5.3 for the meaning of abbreviations used in Table 5.2.

3. The '**Notes**' proposed in Section 5.1 are not implemented as individual elements in the SDRM. Indeed, notes are included in each design rationale element, as shown in Table 5.2. Besides these notes, designers may explain design considerations not directly related to systems (e.g. hull sizing) through *design instances* and *system properties*. For instance, a system property can now be linked to the ship's hull, which was not possible in Case Study 6 (Section 5.2). Note that 'Notes' have been removed from the original ontology too (Figure 5.6).

4. A **search bar** has been implemented, which allows for the filtering of concept design and design rationale elements in the GUI based on user queries. The network visualisation is responsive to search queries as well. The search bar supports both textual and numerical inputs as well as OR statements. An OR statement can be entered with a semicolon between keywords. For example, ' flight deck; hangar' will retrieve all objects with some attribute (e.g. name, justification) including the word 'flight deck' or 'hangar'. This search functionality supports 'data on demand' (Section 5.2.2), allowing designers to search for specific items in the searchable tables. The searchable tables can be set using check boxes, such that, for example, only system properties are filtered.

5

Table 5.2: Items and allowed values within design rationale elements in the SDRM. See Table 5.3 for the meaning of abbreviations. [1]: in line with Lamb (2003, p.5-24). [2]: comprising both physical and logical elements. [3]: Length Overall (LOA).

Item	System Property	Interaction	Compromise	System Concept
Element A	One or more BTs, VBs, SCs, XRs, YRs, ZRs, SH, XX	One or more SPs, BTs, VBs, SCs, XRs, YRs, ZRs, XX	One or more SPs, INs, COs, XX	One or more VBs, XRs, YRs, ZRs, XX
Element B	*N/A*	*idem to above*	*idem to above*	*N/A*
Preferred item	*N/A*	*N/A*	Element A, Element B	*N/A*
Category	*N/A*	Physical, Logical[1]	*N/A*	*N/A*
Constraint:				
Constraint type	Amount, Position, Area, Volume, Length, Width, Height	forward of, aft of, below, above, vertically adjacent, vertically separated, horizontally adjacent, horizontally separated, radially adjacent, radially separated, provide cooling, provide power, provide data, provide logistic capability[2]	*N/A*	*N/A*
Constraint objective	minimum, maximum, exact value, minmax, inv_minmax	*idem to left*	*N/A*	*N/A*
Lower boundary	*Number* (e.g. 1.54) or *Text* (e.g. XR1)	*idem to left*	*N/A*	*N/A*
Upper boundary	*Number* (e.g. 1.54) or *Text* (e.g. XR1)	*idem to left*	*N/A*	*N/A*
Unit	%LOA[3], m, m^2, m^3, #, deck, decks, zone, zones, compartment, compartments, reference plane, W, kW, MW	*idem to left*	*N/A*	*N/A*
Status	Pending, Agreed, Non-agreed		*idem to left*	
Impact	Global, Regional, Local		*idem to left*	
Priority	0-10		*idem to left*	
Justification	*Text*		*idem to left*	
Notes	*Text*		*idem to left*	

Table 5.3: Meaning of object abbreviations in the SSDT-SDRM

Object is related to	Abbreviation	Meaning
Design Rationale	CO	Compromise
	IN	Interaction
	SC	System Concept
	SP	System Property
Concept design	BT	Block Type
	HU	Hull
	SL	Shaft Line
	VB	Volume Block
	XR	X Reference Plane
	YR	Y Reference Plane
	ZR	Z Reference Plane
Unidentified	XX	Unknown object

A search bar was chosen over the list boxes used in Section 4.2. The main reason is that the number of list boxes can explode rapidly due to the growing number of tables (i.e. the four design rationale types) and corresponding columns in these tables. In contrast, a search bar can be implemented such that it covers multiple, complete tables. A drawback of keyword-based design rationale search is that designers should be explicit about their query, which might not always be possible (Zhang et al., 2013). However, solutions to retrieve semantically similar results exist (e.g. Zhang et al., 2013), but have not been implemented in the SDRM.

5. Besides the ability to reload design instances (Section 5.3), **design rationale can also be reloaded**. This can be done when a design instance is reloaded. Currently, design rationale can be reloaded in three ways:
   (a) All design rationales created since the selected design instance. This includes multiple versions of the same rationale item, for instance, because of changed justifications.
   (b) The most recent versions of all design rationales created since the selected design instance. This is the default setting.
   (c) The most recent versions of all design rationales created for the selected design instance.
6. **Design rationale network visualisation** has been implemented in the SSDT. The use of networks to get insight into the relationships between design rationale elements is common (e.g. DeNucci, 2012) and was also used in Section 5.2. However, Rhinoceros is currently very limited in its support of plotting and sophisticated Python libraries. As a result, plots of data need to be defined in terms of objects in Rhinoceros viewports, if one wants to visualise data on the spot without switching between computer programs. For example, in Section 5.2, MATLAB (Matterlab, n.d.) was used to create the network visualisations. Currently, two types of 2D network visualisations can be created, as visualised in Figure 5.8. The designer can decide to view one of these network visualisations or the 3D concept design.

5

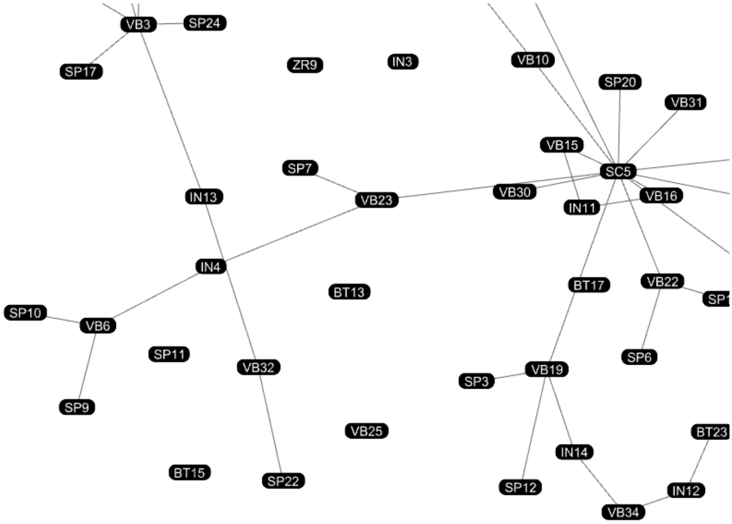
These updates to the SDRM are believed to result in a useful design rationale method for ship layout design, allowing for the capturing of ship-level to system-level design rationale as well as the retrieval of this design rationale.

5.4.2. IMPLEMENTATION AND INTEGRATION WITH THE SSDT

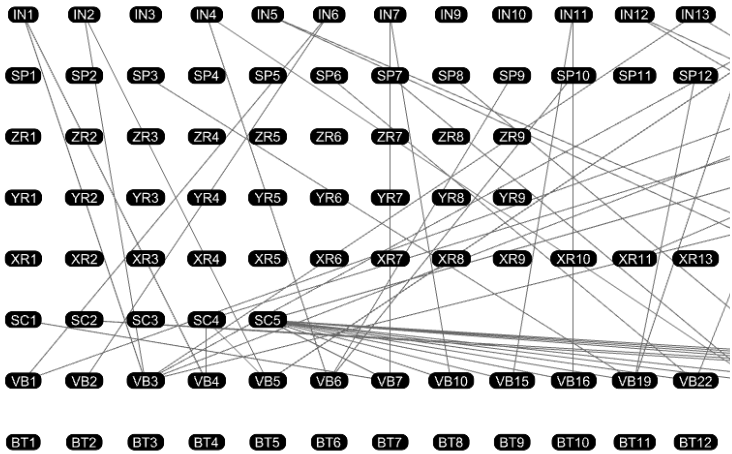
In line with the implementation of the SSDT (Section 5.3), the GUI for the SDRM is Python-based. This allows for a smoother integration between the SSDT and SDRM, to fulfill DRQ5. Such integration is expected to lower the intrusiveness of the design tool-design rationale method combination and enable designers to focus more on the design task at hand, and indeed a method requirement (DRQ5), see Section 4.1.1.

To implement the SDRM into the SSDT, the following items were considered:

1. **Separation of SSDT and SDRM within the GUI.** A key choice was whether the content of the SDRM should be visible at all times or if the content could be put into an additional tab page of the SSDT. Examples of the latter are the 'design instances' and 'volume blocks' tab pages. The choice was made to separate the SSDT and SDRM content because it allows designers to work on any aspect of the design (e.g. main sizing or volume blocks) while being able to capture and reuse design



(a) Force-directed graph.



(b) Sorted graph, based on node type (vertical placement) and node ID (horizontal placement).

Figure 5.8: Examples of partial design rationale network visualisations in Rhinoceros. Nodes represent concept design and design rationale elements. Links represent relationships between these elements.

rationale without changing tab pages. Hence, this solution improves the designer's ability and reduces the (mental) effort to cross-reference between concept design and design rationale elements. Therefore, the method's intrusiveness is reduced. A drawback of this choice is that the amount of information on the screen increases.

2. **Design for two-screen layout.** To maximise the designer's view on the concept design and on the GUI, the design of the GUI is intended for a two-screen layout. One screen shows the concept design and the other shows the GUI to the designer. The SSDT is placed on the left and SDRM right side in the GUI, see Figure 5.9. The

Rhinoceros viewport with the concept design under development can be seen on the left side. This is also the place where the design rationale network is visualised (Section 5.4.1). The right side shows the GUI. Figure 5.10 shows the integrated GUI in isolation.

This layout can also be used in, for instance, design sessions involving multiple stakeholders. The designer could show the layout to the stakeholders on a main screen while having access to the GUI on a computer or laptop. During the case study described below, both teams used a main screen to show the concept design and sat together behind a laptop to make changes to the design and capture design rationale. If the GUI size is changed by the user (e.g. to a one-screen layout), the 'resize' button redistributes the SSDT and SDRM content to the available space on the screen.

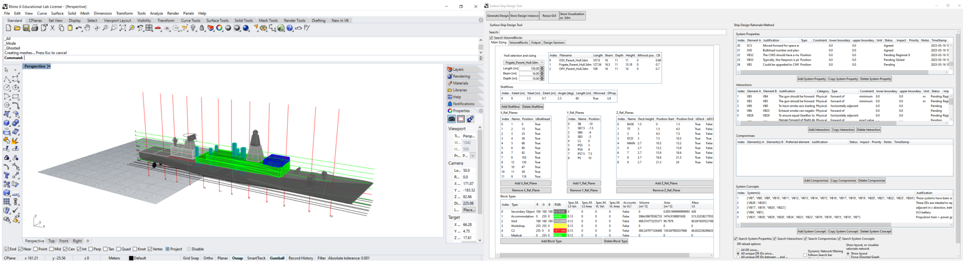


Figure 5.9: Overview of Rhinoceros viewport with concept design (left) and integrated SSDT-SDRM GUI (right).

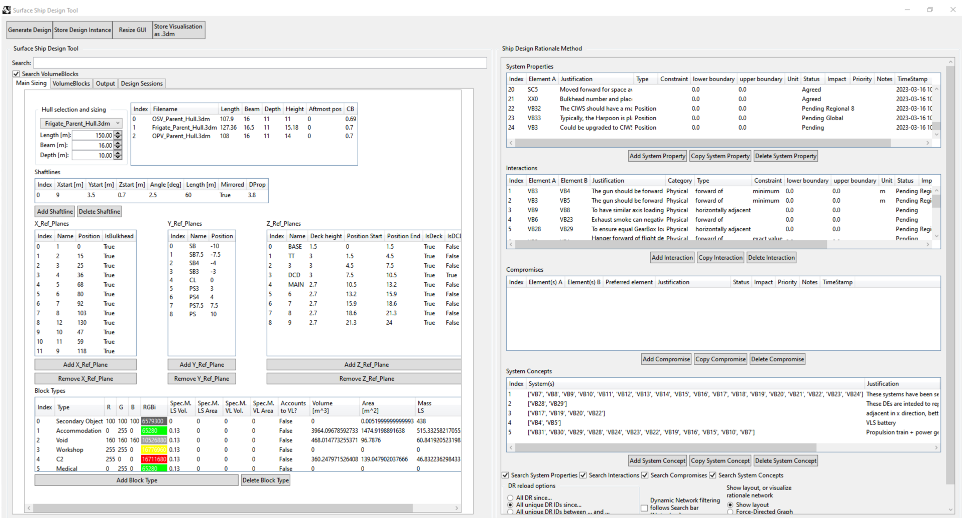


Figure 5.10: Integrated SSDT-SDRM GUI.

3. **Use of popup menus.** In line with Section 4.2.2, popup menus are used when design rationale or design instances are captured. This brings the designer's focus to

the capturing activity and reduces the number of input boxes etc. in the main GUI. Similarly, volume blocks can be selected in the layout when the corresponding design rationale is captured, as in Section 4.2.

The implementation of the SSDT has been discussed in Section 5.3. Regarding the SDRM, tables with captured design rationale are shown in the GUI, see Figure 5.10. Because of the different attributes of these elements, one table is given for each of the four main design rationale elements (i.e. System Properties, Interactions, Compromises, and System Concepts). Below each table, buttons are present to add a new element (triggering a popup menu and object selection in the layout), to copy or to delete a selected existing element.

At the bottom of the GUI, check boxes are present to allow the designer to search only specific tables. Furthermore, design rationale reload options, the option to use filtering of the design rationale network when the search bar is used, and design rationale visualisation options are presented.

5.4.3. USING THE SDRM DURING A DESIGN SESSION

In Chapter 4, the development of a process description to guide designers in exploiting the opportunities of the design rationale method was identified as beneficial. Indeed, prescribing such steps would guide all users in how to use the design rationale method in the envisioned manner (i.e. on the fly etc.). A process of using the SDRM during a single design session is therefore developed and shown in Figure 5.11. To ensure designers and other stakeholders involved in a design session get the most benefits from the SDRM, each design session is divided into three phases, comprising of one or more steps.

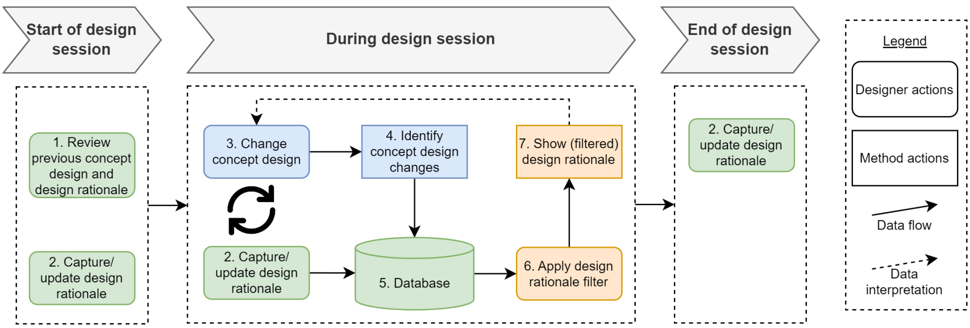


Figure 5.11: Process for using the SDRM during a design session.

- In the first phase, designers look back on the current status of the concept design as well as the rationale behind this concept design. This phase comprises two steps. For convenience, the steps are explained, although there is much overlap with Section 4.2.

    1. *Review previous concept design and design rationale* (designer). This step aims to get designers up-to-date with the status of the concept design.

2. *Capture/update design rationale* (designer). Depending on the outcome of Step 1, designers might update existing or capture additional design rationales. The method supports the capture of design rationale by providing a wide range of design rationale types and a rationale ontology (Section 5.4.1).

- In the main phase, the concept design is changed in response to emerging design issues, new insights, the need to complete the design, etc. While working on the design, design rationale is captured on the fly, which is in line with Sections 4.2, 5.1 and 5.4.2.

2. *Capture/update design rationale* (designer). Iteratively and concurrently with Step 3, the designers should capture and update design rationale on-the-fly. In line with Section 5.2, design rationale can be captured prior to and after corresponding systems are arranged.
3. *Change concept design* (method). The design rationale method is primarily intended to support designers during design activities. Hence the designer needs to be able to change the concept design. Design changes might be influenced by the design rationale that is presented to the designers (Step 7).
4. *Identify concept design changes*. The method needs to identify design changes for two reasons. First, to support computer-based design feedback. Section 4.2.3 showed the design rationale-based MoPs and the dynamic annotations, as examples. Second, for research purposes, it is necessary to evaluate what has been changed to the concept design to relate these changes to how the rationale method is used. However, such information could also be used to gain insight into the actual design process, such as: how the concept design changed over time, on which parts rework was performed, etc.
5. *Database* (method). All rationales and design instances are captured in a database for future reference. On the one hand, this allows the analysis of the design process, such as elaborated in Section 5.6. On the other hand, this allows the designer to refer to past concept designs and supporting rationale, or to take a past concept design as the starting point for another design iteration.
6. *Apply design rationale filter* (designer). Since the number of captured rationales might be high, the designer might filter the rationale to view applicable rationales only. As explained in Section 5.4.1, a search bar was implemented to provide a textual filtering capability. Furthermore, a visualisation of the captured design rationale in a network is provided to give visual insight into the structure of the captured design rationale.
7. *Show (filtered) design rationale* (method). An overview of previously captured rationales is provided to enable designers to review the design based on captured rationales. The method takes the input to the search bar into account to show the requested subset of the design rationale.

- The final phase comprises one step:

2. *Capture/update design rationale* (designer). In this step, designers look back on the work performed during the main phase to see if any important changes have not been documented yet. This step ensures all important decisions and supporting design rationale are captured for future retrieval.

This proposed process is focused on design rationale capture and retrieval while the designer works on the concept design. Over time, applying this process will help designers build *and* maintain a knowledge base of the decisions taken during the development of the concept design. Note, this focus on on-the-fly design rationale capture does not exclude other ways to work with design rationale, such as using relationships within design rationale networks as input to automated layout design tools (such as demonstrated in Section 3.3).

5.5. CASE STUDY 7 - LONG-TERM DESIGN RATIONALE APPLICATION

This section elaborates on the setup of a case study to evaluate the developed SDRM in accordance with the conclusions from Chapter 4 and the second goal for this chapter: *To evaluate how the developed design rationale method benefits collaborative design decision-making such that it leads to better insight into design issues across design teams and better concept designs over time for realistic ship design problems.* The specific goals for the case study are stated in Section 5.5.1. Subsequently, Section 5.5.2 elaborates on the design problem and experiment setup. In Section 5.5.3, the method of result evaluation is briefly discussed. Finally, Section 5.5.4 elaborates on the participants in the case study.

5.5.1. GOALS

Chapter 4 identified the need to investigate the applicability of the SDRM in more realistic design settings. Specifically, the following items required further demonstration:

1. The applicability of the design rationale method in multi-session design for larger design problems (e.g. full ship size). This is required to analyse the benefits of the SDRM for more complex design problems and to evaluate the SDRM's benefits over time.

2. The reuse of design rationale between iterations at various levels of design (e.g. macro, major, micro). This can help identify what is useful design rationale in practice (e.g. if details on system sizing are essential, or if rationale on macro, functional level is most important). In turn, such insights might be used for more detailed guidelines for the process discussed in Section 5.4.3.

3. The integration with actual ship design tools. This integration has already been described in Section 5.4.

Since item 3 is already demonstrated, the first two items will be evaluated through a demonstration and evaluation of the extended SDRM in a more realistic, multi-session design experiment.

5.5.2. DESIGN PROBLEM AND EXPERIMENT SETUP

The case study comprised a design problem to be solved in small teams. The design problem to be solved was a partial arrangement of a notional frigate. Specifically, the arrangement comprised the layout of topside systems (e.g. weapons and sensors) and propulsion systems (e.g. engines and gearboxes). This design problem was chosen because of the interactions between the topside and propulsion system design. For example, engine sizing impacts bulkhead placement, which in turn is needed to constructively support main sensors. Another example is the relative placement of exhausts to sensors due to exhaust gas plumes and corresponding sensor performance. The design problem is subject to changes in requirements over time.

Over the course of two days, three design sessions took place. In each of the design sessions, one variation of the notional frigate arrangement was to be completed. The variations are summarised in Table 5.4. Starting with a pre-generated baseline design with a Combined Diesel-electric and Gas (CODLAG) propulsion system and one main sensor (I-mast, i.e. Integrated Mast), two requirement changes result in increased complexity of the problem. First, a required speed change forces a different propulsion concept (Combined Diesel or Gas (CODOG), instead of CODLAG) to be implemented. Second, a second main sensor, Goalkeeper Close-In Weapon System (CWIS), and a Harpoon missile launcher are added to the previous design.

5

Table 5.4: List of systems for the three design variations in Case Study 7.

Design session	Description	Propulsion Systems	Sensors and Weapon Systems
1	CODLAG + 1 primary sensor	2x Electric Motor (3MW) 4x Diesel Generator set (3MW) 2x Diesel Generator set (1.8MW) 1x Gas Turbine (35MW) Gearbox Diesel Engine Exhausts Diesel Generator Exhausts Gas Turbine Intake Gas Turbine Exhaust	I-mast Main gun 76mm Vertical Launch System (2 Mk41 cells) Helicopter deck (NH90) Helicopter hangar (NH90)
2	CODOG + 1 primary sensor	2x Diesel Engine (9MW) 2x Diesel Generator set (1.8MW) 1x Gas Turbine (35MW) Gearbox Diesel Engine Exhausts Diesel Generator Exhausts Gas Turbine Intake Gas Turbine Exhaust	I-mast Main gun 76mm Vertical Launch System (2 Mk41 cells) Helicopter deck (NH90) Helicopter hangar (NH90)
3	CODOG + 2 primary sensors	2x Diesel Engine (9MW) 2x Diesel Generator set (1.8MW) 1x Gas Turbine (35MW) Gearbox Diesel Engine Exhausts Diesel Generator Exhausts Gas Turbine Intake Gas Turbine Exhaust	I-mast Main gun 76mm Vertical Launch System (2 Mk41 cells) Helicopter deck (NH90) Helicopter hangar (NH90) Smart-L radar Goalkeeper (CWIS) Harpoon missile launcher (1x)

To evaluate the effectiveness of the SDRM over time, the first two design sessions are completed back-to-back, while there is at least a two-week gap between the second and third design session, see Figure 5.12. Because of this gap, some details from the first two sessions are expected to be forgotten by participants. If such details were captured using the SDRM, the added benefit of the SDRM in multi-session design over longer periods of time can be demonstrated. The schedule of the experiment is provided in Table 5.5.

Table 5.5: Schedule of experiment

Timestamp	Duration	What	Who
Prior		Setup: Creation of Baseline Design	Researcher
1st day	15 min	Introduction to experiment	Researcher
	15 min	Familiarisation with SSDT (and SDRM)	Researcher/Participants
	60 min	Design session 1: Expand Baseline Design	Participants
	60 min	Design session 2: Design Variant 1	Participants
	15 min	Questionnaire + Informed Consent	Participants
2nd day	90 min	Design session 3: Design Variant 2	Participants
	30 min	Questionnaire	Participants

Figure 5.12: Timeline of design sessions. Design Sessions 1 and 2 take place on one day. Design Session 3 takes place at least two weeks later.

The participants were given a design brief to inform them about the design scenario and the participants' goals. Furthermore, participants working with the SDRM were given a summarising sheet, which provided a definition for design rationale, explained the high-level design rationale types in the ontology, and pictured the process explained in Section 5.4.3. This sheet is intended to help participants in their familiarisation with the SDRM and is provided in Appendix C.3.

Design Brief

Scenario

You are part of a two-person team tasked with the design of a notional frigate. At this point in the design process, the focus is on the topside design as well as on the main machinery arrangement. Major decisions on suitable systems have been made already. Furthermore, an initial high-level concept design is available. However, the concept design might be subject to major changes in requirements in the future.

As a team, you need to develop one or more feasible concept designs. Because of the foreseen future design changes, documentation of your design decisions is a key aspect of your work.

Main goal

- Maximise sensor height and field of view
- Maximise weapon effectiveness
- Arrange all given and otherwise necessary systems

Consider sensor and exhaust placement, construction, survivability etc.

Design constraints

- Minimise hull size

5.5.3. EVALUATION

To evaluate the SDRM, the documentation of design rationale in the SDRM will be compared to traditional means of design documentation (e.g. handwritten notes). The use of traditional design documentation as a baseline allows for a comparison between the design processes of using these two methods. Eventually, this comparison indicates the added value of the SDRM.

The means of traditional design documentation were not prescribed but could include hand-written notes, digital notes, drawings, hand calculations, etc. All notes taken during the experiment were collected for evaluation. Other sources of information are the data tracked in the design tools, observations made by the researcher during the case study, and a questionnaire handed out to participants.

5

5.5.4. PARTICIPANTS

Participants of the experiment comprised experts (n=4) from the Defence Materiel Organisation, under informed consent. The experiment took place in four sessions in February and March 2023 and the experiment protocol was approved by the TU Delft Human Research Ethics Committee. Recruitment for expert participation took place via the professional network of the researcher.

Participants were subdivided into teams of two persons, i.e. Team A and B. Team A used the modified SSDT in combination with traditional design documentation. Team B used the modified SSDT with the SDRM extension. Compared to Case Study 5, presented in Chapter 4, the number of participants is low. As a consequence, the results will be indicative, not decisive, for conclusions on the long-term effectiveness of the SDRM, see also Section 5.7. This was deemed acceptable for the following reasons: 1) the main goal is to *demonstrate* various items, allowing for a more limited number of participants[1], 2) recruiting students proved challenging in Chapter 4, and 3) the design problem requires, at least, a basic understanding of naval combatant design which students might lack.

5.6. CASE STUDY 7 - RESULTS

To evaluate the performance of the SDRM and to achieve the goals stated in Section 5.5.1, the data analysis is structured by the following questions:

1. Section 5.6.1 answers: ‘How is the SSDT(-SDRM) applied over time?’ The answer to this question will give insight into the actual use of the method over time. This includes the identification of functionalities not used by the teams as well as the development of concept designs over time.

2. Section 5.6.2 answers: ‘To what extent are design decisions documented?’ In this

[1] However, a larger number of participants is preferred to substantiate potential insights.

section, the principle differences between the decision documentation using traditional means of documentation and using the SDRM will be investigated. The aim is to get insight into the performance of the SDRM relative to that of traditional design documentation.

3. Section 5.6.3 answers: 'How is the SSDT(-SDRM) perceived?' This section investigates the participants' view on the SDRM and how it benefits ship design.

5.6.1. APPLICATION OF SSDT AND SDRM OVER TIME

In this section, the use of the SSDT by Team A in the traditional design documentation setting and the use of the SSDT-SDRM by Team B in the new design rationale setting are investigated. This is done by high-level evaluation of data captured in the SSDT-SDRM.

Changes to objects over time

Participants could change and add various objects in the SSDT (e.g. volume blocks, bulkhead positions) and SDRM (e.g. system properties, interactions). Because the SDRM is intended for concurrent use during a design session, as described in Section 5.4.3, it is worthwhile to investigate how the SSDT and SDRM have been applied over time. Figures 5.13a and 5.13b show for Team A and B respectively when objects have been changed over time.

5

The horizontal axis is a non-linear timeline, with various time stamps indicated to give an appreciation of the progress of time. Furthermore, the five stages of the experiment (Table 5.5) are indicated by an alternating grey-white background shading. For instance, the second white area represents the break between Design Sessions 2 and 3.

The vertical axis represents all relevant objects in the concept design. The codes comprise an abbreviation related to the object type as specified in Table 5.3 and a unique identification number for each object.

Black dots mean that activity at the corresponding time stamp was recorded for the given object. Such activity could be the addition, removal, repositioning, or resizing of the object, as well as a change to the object name or description.

Vertical red dashed lines indicate moments at which teams captured design instances. These design instances are indicated by Roman numbers and will be further reviewed in the subsequent sections. When a design instance is captured, all current objects are stored. This results in an activity to all current objects, which explains the coinciding of object-related activity and design instance lines. Also, this explains gaps in design instance lines, as corresponding objects did not exist at these time stamps, but were removed earlier or added later.

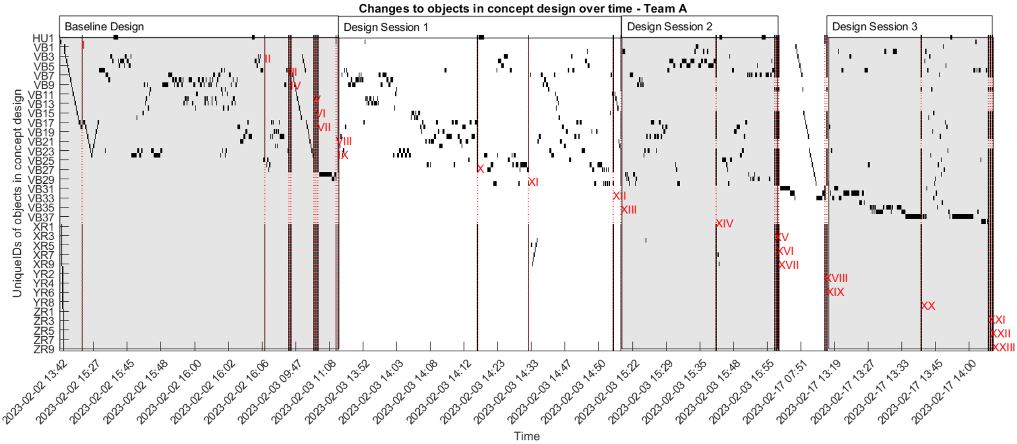
Figure 5.13a shows only objects related to the concept design, as Team A only used the SSDT. In Figure 5.13b both concept design and design rationale-related objects are shown because Team B also used the SDRM. The names of volume blocks (VBx) are provided in Appendix C.5.

Having explained the various elements of Figure 5.13, the following observations can be made:

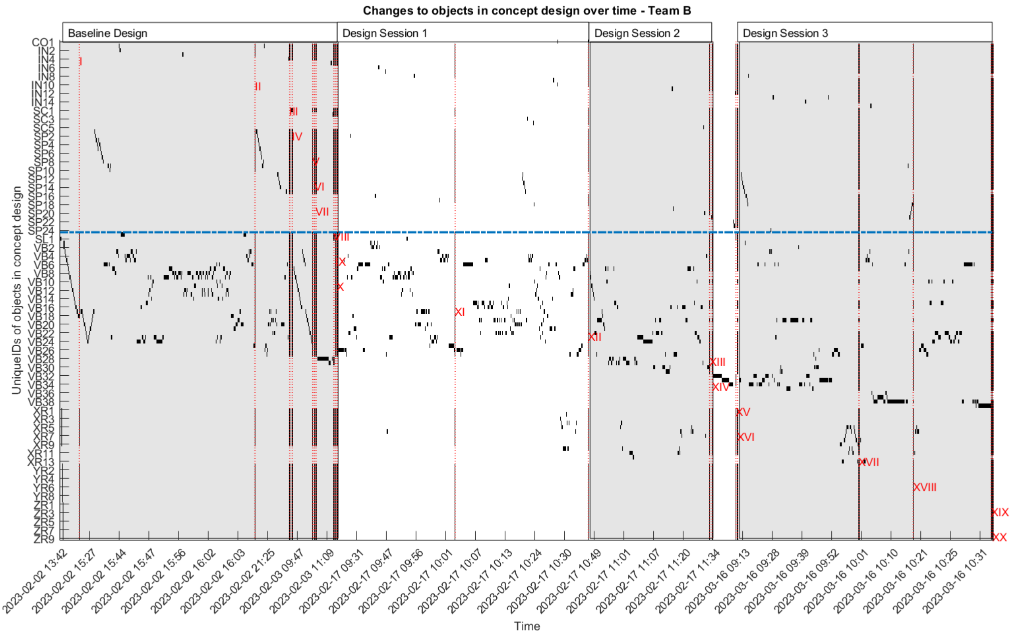
1. Team A gave the most attention to the arrangement of volume blocks. Other design-related objects are barely touched (e.g. X-reference planes in Design Sessions 1

and 2).

2. Team B also gave the most attention to the arrangement of volume blocks. However, there is more activity regarding X-reference planes compared to Team A. For example, Team B added X-reference planes to the design in Design Sessions 2 and 3.



(a) Team A.



(b) Team B. Blue line indicates split between design rationale elements (above the line) and concept design elements (below the line).

Figure 5.13: Changes to concept design elements and design rationale objects over time.

3. Both teams did not change Y- and Z-reference plane properties.

4. Compared to Team B, Team A seemed to focus more on the subsequent arrangement of small subsets of systems. This can be observed in Design Sessions 2 and 3. Team B arranged a wider range of systems throughout the three Design Sessions.

5. Team A used design instances prior to major changes to the concept design. This can be observed in Design Sessions 1 and 2, where the team captured design instances X, XI, and XIV prior to changing the hull and/or X-reference planes (i.e. bulkhead positions). This is less clear for Team B.

6. Design rationale was captured concurrently with design work by Team B in the three Design Sessions. While design rationale was captured throughout Design Session 1, in Design Session 2, design rationale was mainly captured towards the end of the session after focusing on design work first. The team commenced Design Session 3 with activities related to design rationale. Also, prior to saving design instance XVIII, the team captured design rationale. As will be shown in Section 5.6.2, the latter is prior to cleaning up the concept design. This is in line with the intended process described in Section 5.4.3.

Figures 5.14 and 5.15 show histograms of the total number of activities related to all objects for each team, excluding the activities related to storing design instances. Including the latter would only result in an upward shift of all objects. The magnitude of this shift is equal to the number of design instances an object is in.

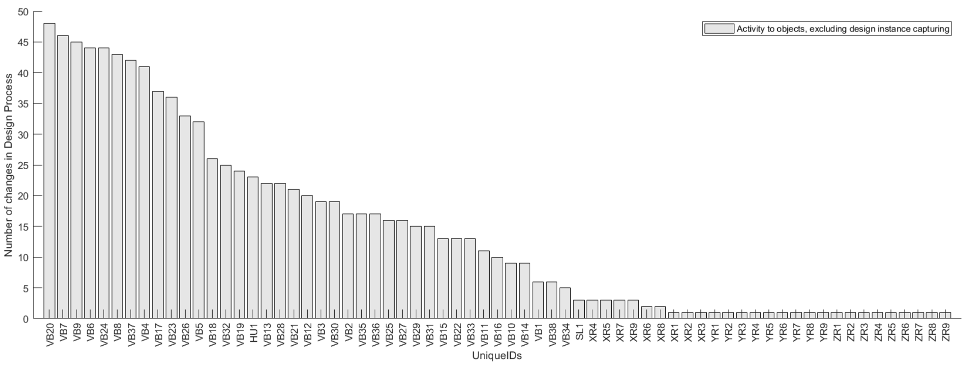


Figure 5.14: Team A - Traditional

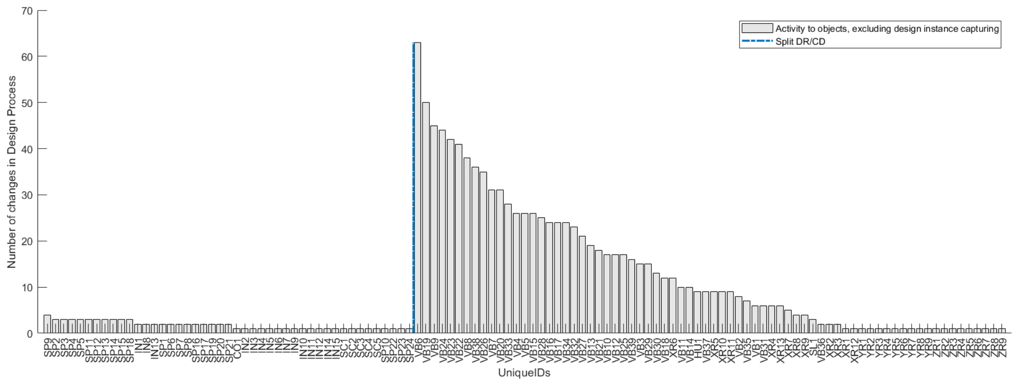


Figure 5.15: Team B - SDRM

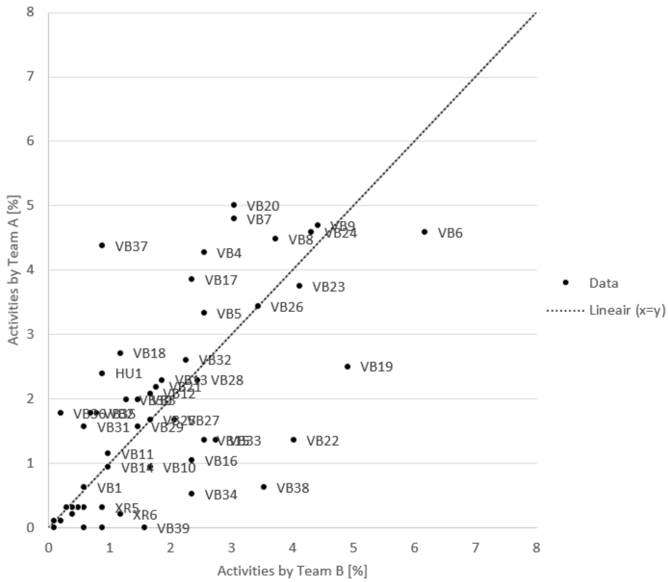


Figure 5.16: Comparison of activities per object between Teams A and B. Objects with $WS > 10$ are labelled.

Based on the histograms, the following observations can be made:

1. As shown above, the Y- and Z-reference planes only get initiated, but are not changed subsequently. Although the Y-reference planes are merely meant to support the transverse arrangement of systems, the Z-reference planes also define deck heights and have therefore a significant impact on the ship's stability.

2. Team B does create multiple design rationales but frequently these items are changed (up to 4 times per design rationale element).

3. In line with the observations above, the histograms show clearly that most activities are related to volume blocks. Figure 5.16 shows a comparison between the activities between Teams A and B (except for design rationale objects). Since the naming of blocks is mostly consistent between teams (Table C.2), the concept designs can be compared well. The figure shows a positive correlation between activities across teams. Table 5.6 shows the twenty most changed objects, based on WS_i, the weighted sum of activities of teams (Equation 5.1). For both teams, the position of the I-mast was changed the most. Overall, the twenty most changed objects are similar across both teams.

$$WS_i = 0.5\left(\frac{Act_{i,A}}{Act_{total,A}} + \frac{Act_{i,B}}{Act_{total,B}}\right) \cdot Act_{total,AB} \tag{5.1}$$

In which Act_i, x is activity i for Team x, with x is Team A or Team B, and $Act_{total,x}$ is the total number of activities for Team A or B.

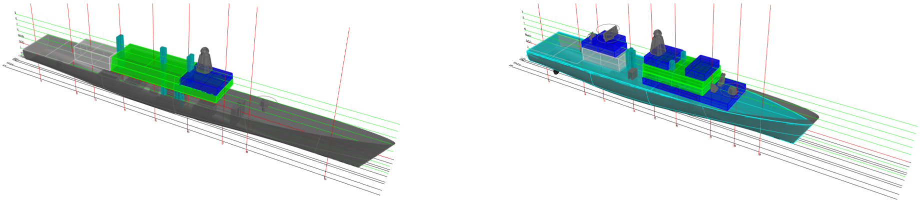
Table 5.6: Twenty most changed objects.

UniqueID	Object Name Team A	Object Name Team B
VB6	I-mast	I-mast
VB9	EM 3MW	EM 3MW
VB24	GT intake	GT intake
VB8	EM 3MW	EM 3MW
VB20	DG exhaust-aft	DG exhaust
VB23	GT exhaust	GT exhaust
VB7	Gear box	Gear box
VB19	DG exhaust	DG exhaust
VB26	Accommodation 1	Accommodation 1
VB4	Mk41 VLS	Mk41 VLS
VB17	DG exhaust-aft	DG exhaust
VB5	Mk41 VLS	Mk41 VLS
VB22	DG exhaust	DG exhaust
VB37	Accommodation 1	*No name*
VB32	Mk141 Harpoon	CIWS Goalkeeper
VB28	DE MAN20V2833D 9100kW	DE MAN20V2833D 9100kW
VB13	DG MAN20V175D 3000kWe	DG MAN20V175D 3000kWe
VB38	Bridge	Mast
VB33	SMART-L radar	Mk141 Harpoon
VB21	DG exhaust	DG exhaust

Evolution of concept designs

Subsequently, the evolution of the concept designs will be briefly reviewed. Figures 5.17a and 5.17b show two concept designs generated by Team A. The figures show, for example, a significant difference in hull size as well as design maturity. Appendix C.4 provides visualisations of all design instances captured by both teams. Although such figures provide some insight into the evolution of the design (e.g. major hull size changes), they provide limited insight into the many activities between design instances (as shown in Figure 5.13). Since most changes are related to volume blocks, the focus will be on volume blocks.

Specifically, the change of volume block position between design instance captures is considered, because the positioning of blocks can indicate both a sizing and a positioning change. Indeed, in the SDRM, volume block position and sizing in X, Y, and Z-



(a) X (b) XXII

Figure 5.17: Examples of concept designs generated by Team A (same scale).

directions are determined by two parameters per direction. These two parameters indicate the beginning and end points of each block. For example, a block defined between $x_1 = 5m$ and $x_2 = 7.5m$ is $2.5m$ long and its centre is located at the average of x_1 and x_2, thus $x = 6.25m$. Thus, a change in size and position can both change the average of the positioning parameters. Of interest are the interdependencies between systems, i.e. what impact does the arrangement of one system have on other systems? Based on combination theory, the number of potential interactions between multiple systems is, at least[2], $C(n,2)$. In this experiment, n equals 39, hence $C(39,2) = 741$ interrelations between systems are possible.

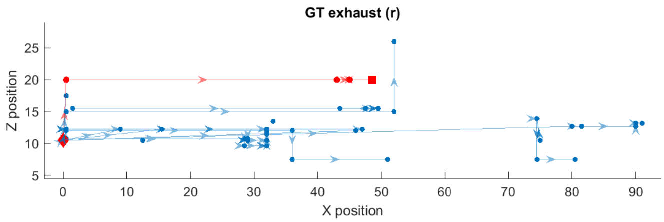


Figure 5.18: Example of a network of system positions (nodes) over time. Arrows indicate how systems are relocated. One system is highlighted (red) with its start (diamond) and end (square) positions indicated.

To show the evolution of the concept design over time, a graphical representation of the ship's system positions is created, as shown in Figure 5.18. The figure serves explanatory purposes and shows a side view (XZ-plane) of a ship with multiple systems. The nodes represent system positions. Arrows indicate how system positions change over time. One system path is highlighted in red, with its start and end positions indicated by a diamond and a square respectively. The figure shows that the highlighted system (Gas turbine exhaust, red) is first moved up by $10m$ and then moved forward by approximately $45m$. After some small changes, the system is moved forward to its final position at $x = 50m$.

For the sake of brevity, only the impact of the following key systems will be investigated, based on the principle changes between design sessions[3]:

[2] The total number of possible interactions could be higher due to, for instance, conflicting interactions between systems. This fact reduces the reported effectiveness of the RKC approach in DeNucci (2012, p.151).

[3] Note that this investigation can be repeated of other subsets of systems to gain a wider understanding of the impact of system arrangement on the overall design space.

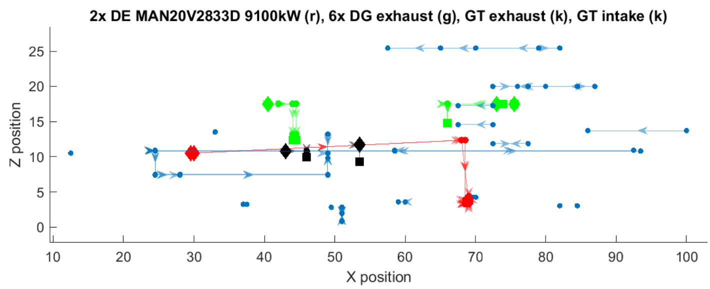
1. *The effect of the change of the propulsion concept on the layout of exhausts (Design Session 1 versus 2).*

   The main change from CODLAG to CODOG is the exchange of 2 Electric Motors (EMs), 4 Diesel Generator (DG) sets by 2 large Diesel Engines (DEs). The corresponding exhausts are also to be changed. This change is expected to impact the arrangement of the exhausts. Indeed, the arrangement of DG sets is more flexible than that of DEs, since the latter need to be directly coupled to the gearbox.

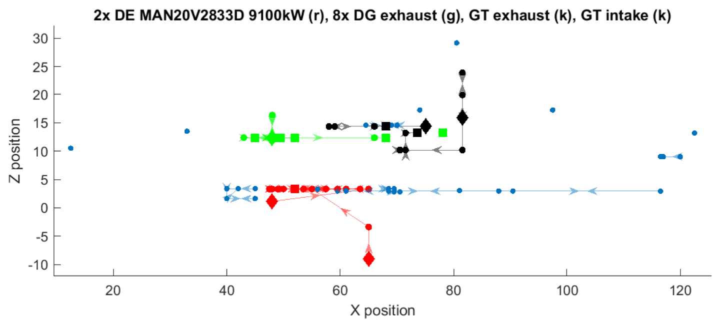
   For Teams A and B respectively, Figures 5.19a and 5.19b show the positions of the new DEs (red), exhausts (green), and Gas Turbine (GT) air intake and exhaust (black) during Design Session 2. The starting positions represent the final position of the systems in Design Session 1 (i.e. the CODLAG concept).

   For Team A, the change of propulsion concept resulted in a slightly more centralised arrangement of exhausts. The aft exhausts were placed slightly forward, and the forward exhausts were placed slightly aft. In contrast, Figure 5.19b indicates Team B achieved a more distributed arrangement of DG exhausts, while the GT inlet and exhausts were moved slightly aft. However, especially the results for Team B show that many changes were made to system positions. This raises the

5



(a) Team A



(b) Team B

Figure 5.19: Example of changes to positions of systems due to the addition of other systems: The effect of the change of the propulsion concept on the layout of exhausts (Design Session 1 versus 2). Color codes: r = red, g = green, k = black.

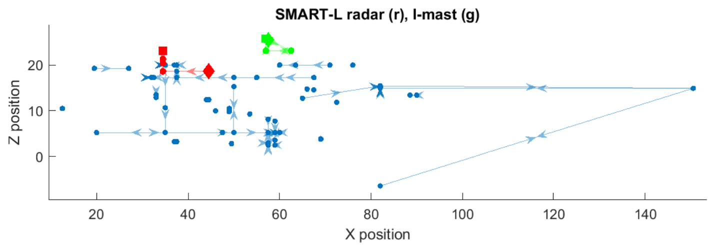
question of which other systems also impacted the locations of the exhausts. This will not be further investigated here but is mentioned to indicate that pinpointing interdependencies is not always straightforward.

2. *The effect of the addition of the SMART-L radar on the position of the original I-mast (Design Session 2 versus 3).*

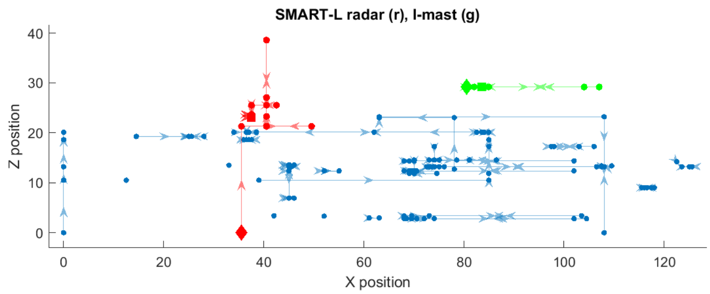
   Besides the addition of weapon systems, in Design Session 3 a large SMART-L long-range radar was added to the concept design. Therefore, the teams needed to decide on the position of both the I-mast (a shorter-range radar) and the new SMART-L radar. Aspects to be taken into account are, for instance, structural support, field of view, and exhaust gas contamination.

   For Teams A and B respectively, Figures 5.20a and 5.20b show the positions of the SMART-L (green) and I-mast (red) during Design Session 3. The starting position of the I-mast represents its final position in Design Session 2.

   Team A changed the position of the SMART-L, as imported to the design, only to a limited extent. The radar was moved back by 10m and up by 3m. Team B performed more exploration of possible locations for the same system. Both teams decided to place the SMART-L towards the aft (this is also the approximate position of the SMART-L at the Dutch Air Defence and Command Frigate (LCF) frigates). What stands out for both teams is the changes to the position of the I-mast. Al-



(a) Team A



(b) Team B.

Figure 5.20: Example of changes to positions of systems due to the addition of other systems: The effect of the addition of the SMART-L radar on the position of the original I-mast (Design Session 2 versus 3). Color codes: r = red, g = green.

though both teams changed the position of the I-mast during Design Session 3, its final position is close to its starting position. Hence, the effect of the addition of the SMART-L radar on the position of the I-mast seems neglectable. To be noted, however, is the multitude of other design changes that can be observed in Figure 5.20b. To gain a deeper understanding of interrelationships within the design, more system combinations can be investigated.

Observations by the researcher
Next, key observations made during the experiment will be discussed in support of the results elaborated on above.

1. Team A did perform some weight calculations to check the longitudinal trim as well as intact stability (see Section 5.6.2). This was not explicitly part of the design task but was undertaken at the team's initiative to check some basic feasibility of the concept designs.

2. Team A used design instances to capture the main decisions after smaller periods of design work. Via reconstruction (see Section 4.1), the team tried to recall and summarise the main decisions made since the previous design instance was stored. Because even in the short term remembering all details can be challenging (Todd & Marois, 2004), not all relevant design aspects might be stored this way. For example, the reasoning behind the midships arrangement of the Vertical Launch System (VLS) is not captured nor is the fact that this arrangement of the VLS affected the arrangement of accommodation and Command and Control (C2) volume blocks.

   Because design instances typically are captured on important milestones of the design process, the SDRM allows for more 'on-the-fly' design rationale capturing compared to design instance capturing.

3. When starting Design Session 3, Team A did not review past design decisions. When the latest concept design was opened, one of the participants just said: 'Yes, I do recognise this [concept design]'. Subsequently, the team commenced the new design task. Also, the concept design was barely changed to fit the additional topside systems. Actually, the propulsion system arrangement was kept as-is between design instances XIX and XX (see Figure 5.14). The placement of exhausts in relation to the other topside systems was considered, but eventually not changed.

4. Team B commenced with using the SDRM process in Design Session 1, as instructed by the summarising sheet (Appendix C.3). However, this was perceived as intrusive in the design process. Hence, the team made the decision to focus on designing in Design Session 2, prior to capturing the decisions made during this session. This is observed in Figure 5.15. Notably, during Design Session 3, the team took up the SDRM process on its own initiative. As will be seen in Section 5.6.3, using the SDRM was perceived as less intrusive than during the earlier sessions. This indicates a learning process, which is to be expected when using a new design tool (Reinertsen, 1997).

One of the participants reflected after Design Session 2: 'I had the feeling that designing was twice as smooth' compared to Design Session 1. However, it was observed that the design discussion was significantly less explicit about the 'why' behind design changes during Design Session 2 compared to Design Sessions 1 and 3. This was confirmed by the participants, when saying 'although the second design problem was easier.' Even if design considerations were not captured as design rationale elements, the use of the SDRM seems to trigger more explicit design reasoning. This is in line with the benefits of design rationale found in Chapter 4.

5. Both teams discussed *how* design rationale was to be captured. For example, Team B agreed upon a design change. However, when one of the participants added a justification to the design rationale element being captured, the other participant did not agree with the justification as it was perceived to insufficiently describe what was actually decided and arranged in the layout.

6. Incomplete design rationale can lead to confusion. For example, the baseline design comprised design rationale related to the GT exhaust and the radar (Interaction 5 in Table C.6). When reviewing the design rationale at the start of Design Session 1, the team questioned why this rationale only applied to the GT exhaust, or if the same rationale applies to all exhausts.

In conclusion, based on the investigation in this section, it can be stated that key systems (e.g. those that got the most attention) can be identified by tracing the evolvement of the concept design. Also, insight into interdependencies between systems can be gained by investigating the evolution of the concept design for subsets of systems. However, within the time intervals investigated, many other design changes might be performed, increasing the likelihood of other interdependencies playing a role too. Hence, the question is what the real intention of the designers (in this case, participating teams) was. Expressing this intention is clearly the terrain of design rationale and seems to be supported by the SDRM. Therefore, Section 5.6.2 will investigate the decision documentation during the experiment.

5.6.2. DESIGN DECISION DOCUMENTATION OVER TIME

To analyse the level of design decision documentation, the data captured in the SSDT and SDRM, as well as hand-written documentation by the participants will be investigated.

Review of captured design instances

Firstly, Figure 5.13 showed that throughout the design process of Team A, 23 design instances were captured, compared to 20 by Team B. The details of these design instances are presented in Tables 5.7 and 5.8 respectively. The design instances captured by the participants (underlined IDs) are differentiated from the design instances captured by the researcher (non-underlined IDs). The 'Commit Message' is the obligatory explaining description that needs to be provided when storing a design instance.

The principle observation made when comparing these commit messages between Teams A and B is that Team A captured significantly more details on the concept design

Table 5.7: Stored Design Instances for Team A.
ID indicates design instances stored by the team. *: translated from Dutch. **: VST: Voortstuwingstrein, i.e. propulsion train.

Traditional

ID	Timestamp	Phase	Commit Message
I	2023-02-02 14:03	Baseline	Import of initial systems for Baseline design. Exit to solve a coding error in BlockType addition.
II	2023-02-02 16:15	Baseline	Import of all objects, and initial arrangement of some of them.
III	2023-02-02 21:32	Baseline	Baseline design
IV	2023-02-02 22:43	Baseline	Update of Baseline concept
V	2023-02-03 10:21	Baseline	Update of object masses.
VI	2023-02-03 10:25	Baseline	Baseline concept
VII	2023-02-03 11:08	Baseline	Baseline with updated weights.
VIII	2023-02-03 11:15	Baseline	CODOG main DE systems
IX	2023-02-03 11:22	Baseline	Baseline wrong weight calculation
X	2023-02-03 14:18	Design Session 1	Propulsion train (VST**) design v1*
XI	2023-02-03 14:30	Design Session 1	Concept 1 - on the basis of VST config. 1*
XII	2023-02-03 14:52	Design Session 1	Concept 2 - different VST to achieve less casings and improved survivability than concept 1*
XIII	2023-02-03 14:55	Design Session 2	Baseline round 2
XIV	2023-02-03 15:39	Design Session 2	Round 2 - concept 2, compared to concept 1 improved COG and slightly smaller vessel*
XV	2023-02-03 16:06	Design Session 2	Round 2 - concept 2, tov concept 1 improved COG and slightly smaller vessel
XVI	2023-02-03 16:09	Design Session 2	End of session
XVII	2023-02-16 16:36	Break	Last design, but with faster version of the code.
XVIII	2023-02-17 08:02	Break	Additional topside systems.
XIX	2023-02-17 08:09	Break	Previous design, but with reloaded with new code.
XX	2023-02-17 13:44	Design Session 3	Concept v3 systems Round 3
XXI	2023-02-17 14:10	Design Session 3	Concept 3 Systems Round 3
XXII	2023-02-17 14:12	Design Session 3	Concept v3 Round 3: GM=2.0m, deplacement= 6560 tons. Addition of Smart-L, Goalkeeper, 2x Harpoon. No weight propeller and shaft -> compensates for position COG-X*
XXIII	2023-02-17 14:56	Design Session 3	End of session

Table 5.8: Stored Design Instances for Team B.
ID indicates design instances stored by the team.

SDRM

ID	Timestamp	Phase	Commit Message
I	2023-02-02 14:03	Baseline	Import of initial systems for Baseline design. Exit to solve a coding error in BlockType addition.
II	2023-02-02 16:15	Baseline	Import of all objects, and initial arrangement of some of them.
III	2023-02-02 21:32	Baseline	Baseline design
IV	2023-02-02 22:43	Baseline	Update of Baseline concept
V	2023-02-03 10:21	Baseline	Update of object masses.
VI	2023-02-03 10:25	Baseline	Baseline concept
VII	2023-02-03 11:08	Baseline	Baseline with updated weights.
VIII	2023-02-03 11:15	Baseline	CODOG main DE systems
IX	2023-02-03 11:22	Baseline	Baseline wrong weight calculation
X	2023-02-16 16:41	Baseline	Baseline, with new, faster code and corrected weight calculation.
XI	2023-02-17 10:03	Design Session 1	design instance 1
XII	2023-02-17 10:35	Design Session 1	Final design
XIII	2023-02-17 11:31	Design Session 2	CODOG v1
XIV	2023-02-17 11:34	Design Session 2	CODOG final
XV	2023-03-14 19:27	Break	Additional topside systems.
XVI	2023-03-14 19:28	Break	Previous design, but reloaded with new code.
XVII	2023-03-16 10:01	Design Session 3	16-3-23 v1
XVIII	2023-03-16 10:19	Design Session 3	16-3-23 v2 (ITD finished)
XIX	2023-03-16 10:36	Design Session 3	Future next generation notional frigate concept vessel
XX	2023-03-16 10:38	Design Session 3	final

than Team B. For Team A, the design instances were the only means provided by the SSDT to capture design rationale. Note that no guidance was given concerning what is to be captured in these commit messages to either team.

Team B captured very limited information with these commit messages, which might be hard to understand when returning to these designs after longer periods of time. For instance,

- Design instance XIII is described as ‘CODOG v1’, meaning it’s the first CODOG variant created in Design Session 2. However, it is also the final variant, as the researcher created a backup design instance XIV only three minutes later.

- Design instance XVIII is described as ‘16-2-23 v2 (ITD finished)’, signifying this concept has its ITD design finished. This design instance was cleaned up a bit, and subsequently stored as design instance XIX ‘Future next generation notional frigate concept vessel’.

In contrast, Team A provided insight into:

- *Relations between design instances.* For instance, design instance XII states that this concept design has fewer exhaust casings and improved survivability compared to design instance XI.

- *Performance characteristics of concept designs.* For instance, design instance XXII provides an intact stability performance (GM=2.0m) and estimated displacement (6560 tons).

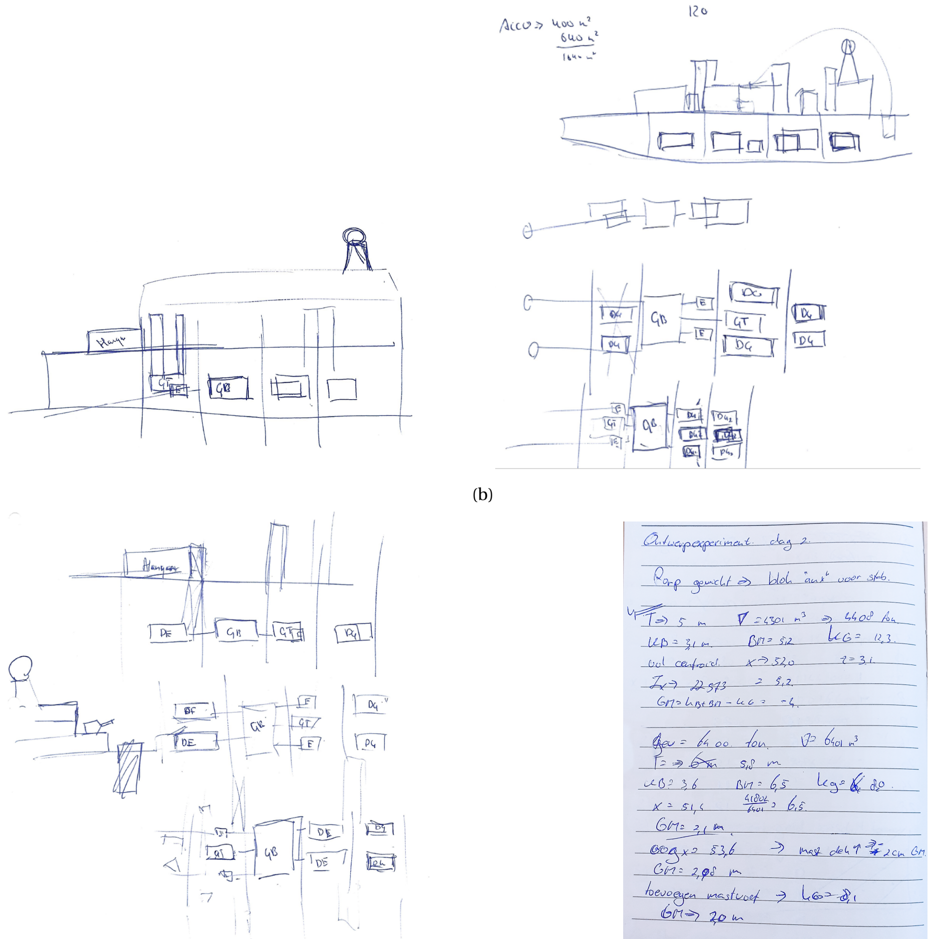
- *Considerations concerning the layout of concept designs.* For instance, design instance XIV mentions that one of the considerations is to place weapons in different sections.

Review of handwritten documentation
Secondly, the handwritten documentation will be evaluated. Figure 5.21 shows various examples of hand-written documentation made by Team A. Figures 5.21a-5.21c show various sketches of potential layouts for propulsion and topside systems.

The sketches often include exhaust stacks to indicate how conflicts between topside layouts and propulsion system layout could be resolved. For example, the top sketch in Figure 5.21c shows a side view of a propulsion system layout including bulkhead positions. The sketch indicates that the aft exhaust stack might be arranged towards the aft of the vertically adjacent helicopter hangar. The left part of Figure 5.21c shows an arrangement of the front main gun and a VLS. Note that this arrangement has not been included in any stored concept design, as Team A settled on a midships arrangement of the VLS from Design Instance XI (Figure C.1).

Figure 5.21d shows the calculation of the intact stability of various concepts (resulting in GM=-4m, GM=2.1m, and GM=2.0m). Also, a note states that the propeller weight is not considered in the stability calculation (‘Prop gewicht –> blok “aux” voor stab.’) Note that this information is also captured in the commit message of design instance XXII. To improve the stability calculation, the team added an aqua-coloured volume block to account for missing weight in the mostly empty hull (e.g. Figure 5.17b). This way, the team could make a more accurate weight estimation.

The role of the sketches was mainly to communicate ideas on potential arrangements of systems within the team. Typically, a sketch was made and discussed before the idea was executed using the SSDT. Also, sketches were used as ‘thinking sketches’ (van der Lugt, 2005). Team B did not sketch on paper to support design discussions but did use the SSDT to roughly arrange systems to support ideas.



(a) (b) (c) (d)

Figure 5.21: Overview of hand-written documentation from Team A. (a)-(c) Sketches of aspects of layout. (d) Stability calculation.

Review of captured design rationale

Thirdly, the design rationale in the SDRM is reviewed. Since design instances and hand-written documentation are the only sources for design documentation by Team A, the remainder of this section will focus on the design rationale captured in the SDRM by Team B. In total, 45 of the design rationale elements have been captured in the design process of Team B. Of these 45 elements, 25 have been captured by the researcher while setting up the baseline design and designs with additional systems for the two variant designs. The remaining 20 are captured by the design team during the experiment. The captured design rationale is provided in Appendix C.6. The following observations can be made:

1. Some rationale are duplicates (e.g. System Properties 1-6, which are related to different DG exhausts). The advantage of these duplicates is that for all systems, the relevant rationale is captured. A disadvantage is the possible overflow of information - again pointing towards the need to show 'data on demand'.

2. The Priority attribute has rarely been used, only for some of the interactions.

3. Notes have never been used.

4. To understand the rationale, the concept design is required. Indeed, the design rationale is related to (mostly) specific volume blocks, block types, and reference planes. However, it should be noted that design rationale not necessarily is captured at the moment corresponding design changes are made. This is especially the case when the designer focuses on designing before capturing design rationale (as Team B during Design Session 2) instead of capturing design rationale on-the-fly (i.e. as intended by the process discussed in Section 5.4.3).

   For example, System Property 20 relates to System Concept 5 (12 systems comprising the 'Propulsion train + power generation (including reference frames)') and states the System Concept is 'Moved forward for space availability'. From this justification, it's unknown, for example, how much the propulsion train was moved forward.

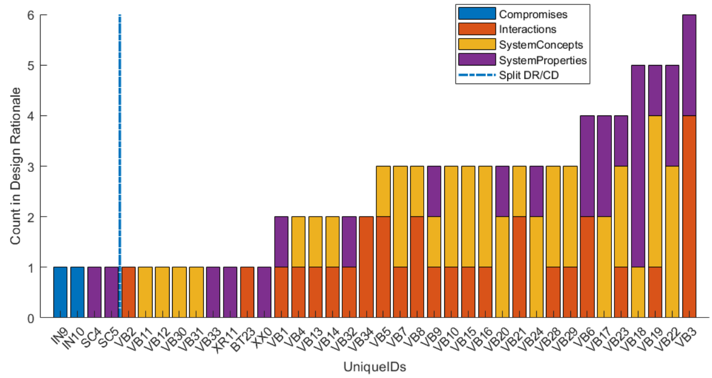
5. Besides the concept design, the justification is often necessary to understand the design rationale. For example, System Property 25 states that VB3 (the main 76mm gun) should be placed at a minimum of 25 [m]. In addition, the justification explains that the main gun should be arranged "preferably 25m from [the] bow to protect from green water."

   This is also the case when not all constraint data is entered. For example, System Concept 4 comprises the 'VLS battery' (VB4-5). SC4 is linked to System Property 18, with the justification "Aft of main gun, forward of bridge. Separated from accommodation for survivability" but without constraint data. Note, the justifications often are succinct and could benefit from linguistic checks. Users might reduce the number of words to reduce the required effort for capturing design rationale. The succinct justifications might be expanded later on (e.g. after the design session).

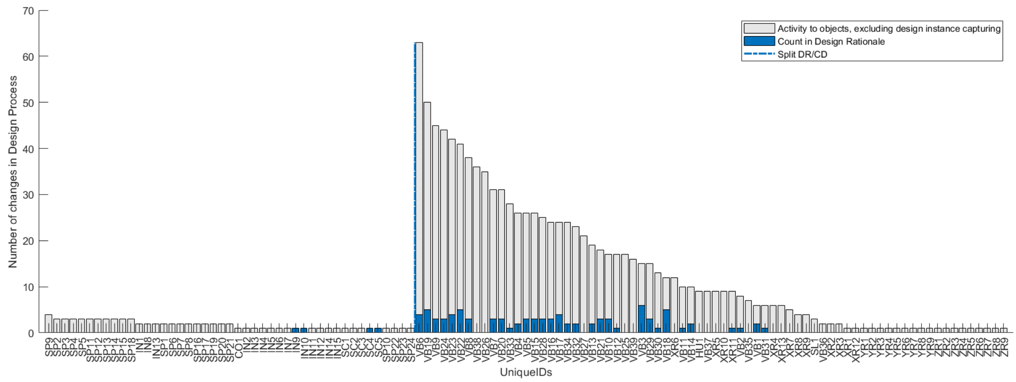
5

In Section 5.6.1 it was observed that design rationale was captured by both the researcher and participants concurrently with designing. Figure 5.22a shows for all unique objects the frequency of appearance in design rationale. Note that object XX0 is a yet undefined object within the object list. In this case, the team attempted to capture a System Property relative to bulkheads, with the justification "[The] bulkhead number and placement [is] not yet final; bulkheads [are] to be added at [the] fore end". At the time of capturing, adding System Properties to reference planes was not possible. Hence, the team diverted to using the general XX0 identifier. The following observations can be made:

1. 46% of design-related objects and 9% of design rationale-related objects is at least once referred to in the captured design rationale.



(a) Histogram of objects referred to in the four main design rationale types.



(b) Relation between actions to objects and total frequency of reference in design rationale.

Figure 5.22: Object referred to in captured design rationale by Team B

2. Most references in the captured design rationale are made to volume blocks. Furthermore, design rationale is captured related to a reference plane (i.e. XR11) and a block type (i.e. BT24). Section 5.6.1 already showed that most design activities were also related to volume blocks. Figure 5.22b shows an overlay of Figures 5.15 and 5.22a. From this figure, a slight positive correlation between the activities related to concept design objects and the frequency of occurrence in design rationale can be observed. This means that objects that are changed more are mentioned more often in the captured design rationale.

3. Only four design rationale elements, namely two interactions and two system concepts, were related to other rationales. The two interactions are referred to in Compromise 1. The two system concepts are related to System Properties 18 and 20.

Concluding, both the SSDT and SDRM provide means to capture design rationale while the concept design evolves. The SSDT enables designers to capture ‘design instances’ with a justification. Team A used design instances in a reflective manner to capture major design decisions and considerations. Team B used design instances to store concept design variants but captured less useful information in the descriptive commit messages. Instead, Team B mainly used the SDRM to capture design rationale. The captured design rationale provides insight into the status of the design (e.g. the status of bulkhead positions) and design considerations (e.g. the distance between the main gun and the ship's bow).

In Section 5.6.1, it was found that concept designs on themselves can only partially provide insight into the decision-making process. What's lacking in concept designs is the intention of the designers. The results in this section indicate that both the SSDT and SDRM provide useful means to capture this intention. Hence, combining designing and in-situ design rationale capturing seems to be a promising solution to the problem outlined in Chapters 1 and 4.

5.6.3. PERCEIVED BENEFITS OF THE SSDT-SDRM

In this section, the responses to the questionnaire will be investigated to get insight into the participants' perceived benefits of the SDRM.

Closed questions

First, the responses to the 22 closed questions in the questionnaire are investigated. These questions and responses are provided in Table 5.9. The participants were asked to respond on a five-point Likert scale. Only the lower and upper bound values were provided for each question, as shown in the table. Participants filled out the questionnaire at the end of Design Sessions 2 and 3, i.e. at the end of days 1 and 2 of the experiment. Some questions were only applicable to specific Design Sessions (i.e. questions 1-9). Questions 10-17 were asked both times while questions 18-22 were only asked to the participants using the SDRM. The evaluation of the responses to the closed questions is structured per the topic of the questions.

Available time and difficulty of the design problem

Regarding the difficulty of the design problems and the available time, the following observations can be made:

1. Participants P1 and P2 considered the time available for Design Session 1 mostly sufficient (4 on the five-point 'insufficient-sufficient' scale. These participants used the SSDT only. However, Participants P3 and P4, using the SDRM, scored only 2 and 3. All participants considered the time available for Design Sessions 2 and 3 sufficient (5).

2. Participants P1 and P2 mostly considered the difficulty of the design tasks in the different Design Sessions average (3) or easy (2), although P2 experienced the third design task to be hard (4).

   The participants using the SDRM considered the difficulty of the design tasks to reduce throughout the Design Sessions. The first design task was considered hard (4), the second average (3) while the difficulty of the third design task was rated 2 and 3.

These results might indicate the participants using the SDRM needed to get used to the method (see also item 4 in the list with observations in Section 5.6.1). Also, the

Table 5.9: Responses of the four participants (P1-4) to the closed questions in the questionnaire. The responses are split between days and between using traditional documentation and the SDRM.

					Responses							
					Traditional				**SDRM**			
			Five point Likert scale		**Day 1**		**Day 2**		**Day 1**		**Day 2**	
Topic	ID	Question	Lower bound	Upper bound	P1	P2	P1	P2	P3	P4	P3	P4
Available time and difficulty of the design problem	1	The time available for completing Design Session 1 was:	Insufficient	Sufficient	4	4			2	3		
	2	The time available for completing Design Session 2 was:	Insufficient	Sufficient	5	5			5	5		
	3	The time available for completing Design Session 3 was:	Insufficient	Sufficient			5	5			5	5
	4	The difficulty of the task in Design Session 1 was:	Too easy	Too hard	3	2			4	4		
	5	The difficulty of the task in Design Session 2 was:	Too easy	Too hard	3	2			3	3		
	6	The difficulty of the task in Design Session 3 was:	Too easy	Too hard			3	4			2	3
Quality of developed concept designs	7	I'm satisfied with the design in Design Session 1	Strongly disagree	Strongly agree	2	4			3	2		
	8	I'm satisfied with the design in Design Session 2	Strongly disagree	Strongly agree	4	4			4	4		
	9	I'm satisfied with the design in Design Session 3	Strongly disagree	Strongly agree			4	3			4	5
	10	I'm satisfied with my input in the decision-making	Strongly disagree	Strongly agree	4	4	5	4	3	4	4	5
	11	My input in the decision-making has been satisfactorily incorporated in the final designs.	Strongly disagree	Strongly agree	3	5	4	4	4	4	4	5
Design decision documentation	12	We documented all design decisions in the experiment	Strongly disagree	Strongly agree	2	4	2	3	4	2	4	4
	13	This level of decision documentation is comparable to real ship design (in situ) design documentation	Strongly disagree	Strongly agree	4	4	4	3	2	2	2	3
	14	The documented design decisions reflect the outcome of the design dialogue	Strongly disagree	Strongly agree	4	4	2	4	4	4	5	4
	15	The design documentation was important to support the design dialogue	Strongly disagree	Strongly agree	3	4	1	4	3	3	3	3
	16	Our design dialogues were structured	Strongly disagree	Strongly agree	2	3	3	3	4	2	5	4
	17	We were able to reuse previously documented design decisions	Strongly disagree	Strongly agree	1	4	2	4	2	5	5	5
Ship Design Rationale Method (SDRM)	18	The SDRM was easy to learn	Strongly disagree	Strongly agree					4	4	4	5
	19	The SDRM process enhanced the documentation of decisions	Strongly disagree	Strongly agree					4	4	4	5
	20	The integration between a layout design tool and design rationale improves ship design	Strongly disagree	Strongly agree					4	3	4	3
	21	The SDRM distracted me from designing	Strongly disagree	Strongly agree					5	4	3	1
	22	I would apply the SDRM during actual design work	Strongly disagree	Strongly agree					4	4	3	4

results might indicate that the participants needed to familiarise themselves with the design problem during Design Session 1, while the subsequent design problems were merely derivatives thereof.

Quality of developed concept designs

Regarding the perceived quality of the concept designs developed during the experiment, the following observations can be made:

1. Participants using the SDRM are equally or slightly more satisfied with the design across the three Design Sessions.
2. Three of the four participants (i.e. P1, P3, and P4) were slightly more satisfied with their input during the second day than during the first day.
3. Participants P1 and P4 considered that their input was slightly better implemented in the final design of Design Session 3 compared to the initial two Design Sessions,

while P2 experienced a slight reduction. All four participants considered their input to be satisfactorily implemented (scoring 3-5).

The results do not indicate significant differences between the perceived quality across both teams. In line with the findings in Section 4.4.3, the availability of a design rationale method does not seem to directly lead to better concept designs.

Design decision documentation
Regarding the documentation of design decisions, the following observations can be made:

1. Compared to the participants using the traditional design documentation, the participants using the SDRM believe to have documented more, and even most, design decisions (Question 12).

2. Participants P1 and P2 considered the limited design decision documentation to be in line with real ship design (in situ) design documentation, while participants P3 and P4 considered the more elaborate design documentation in the SDRM to be less in line with ship design practice (Question 13).

3. Except for P1 on day 2, all participants indicated that the documented design decisions reflected the outcome of the design dialogue (Question 14).

4. Participant P2 was positive about the support of the design documentation in the dialogue (4) while P1 was less positive (scoring 3 and 1 for days 1 and 2 respectively). Participants P3 and P4 considered the support of their design documentation not important but also not unimportant (3).

5. Participants using the SDRM considered their design dialogues to be more structured than participants using traditional design documentation (Question 16). This is in line with observations in Section 4.4.2.

6. Participant P1 did not agree with the statement ‘We were able to reuse previously documented design decisions’ (1 and 2) while P2 did agree (4). While P3 did not agree with the statement on day 1 (2), the participant did strongly agree on the second day (5). P4 did strongly agree with the statement on both days.

Although the perceived quality of concept designs might not differ regardless of using the SDRM, these results indicate that the documentation benefits of the SDRM are already visible over the course of the experiment, in the perception of the participants. It enables the capture of more, and even most, design decisions compared to using the SSDT only. Also, it enables the reuse of this rationale over time.

Ship Design Rationale Method
Finally, the following observations can be made regarding the responses to the questions on the SDRM:

1. The SDRM was perceived as ‘easy to learn’ by both P3 and P4 (Question 18).

2. The participants agreed that the SDRM process (see Section 5.4.3) enhanced the documentation of design decisions (Question 19).

3. Participant P3 was positive about the improvement the integration between the SDRM and a ship layout design tool brings to the design process (4), but P4 neither disagreed nor agreed (3).

4. Both participants P3 and P4 expressed a reduced distraction over the duration of the design process (Question 21).

5. Both P3 and P4 are positive about applying the SDRM during actual design work. However, on day 2, P3 neither disagreed nor agreed to apply the SDRM.

These results underline the findings above: the SDRM is perceived to enhance the documentation of design decisions at reducing costs (i.e. effort) over time. However, regarding the application of the SDRM, there might be some hesitation. Indeed, it needs to fit within the overall ship synthesis process (see Sections 1.1.2 and 4.6).

Open questions
Second, the responses to the closed questions are discussed. The questions and responses are provided in Table 5.10. Question 1 was provided to Team A only, while Questions 2, 3, and 6 were provided to Team B only. The other questions were provided to all participants. The following observations can be made:

1. The responses to Question 1 regarding the teams' decision documentation reflect the observations made in Section 5.6.2 regarding the documentation of design decisions by Team A.

2. Team B considers the ability to capture the various main design rationale types as most beneficial (Question 2).

3. Team B provides three options for additional functionalities of the SDRM (Question 3):

   (a) Tracking of changes of design rationale over time. This option was not used by the team but is available (see item 6 in Section 5.4.1).

   (b) Addition of bulkheads to system concepts. During Design Sessions 1 and 2, this was not possible yet. This functionality was implemented before Design Session 3 in response (see also Table 5.2).

   (c) The ability to capture decisions related to future updates. For example, the team considered using the 76mm main gun as a CWIS (Question 4).

4. P4 (Team B) mentions 'that all the rationale of which I thought that it would not be directly clear to a future reviewer' was captured. In contrast, P1 (Team A), using traditional design documentation, states that 'a lot of rationale per block or instance has not been stored'. After Day 2, this is further specified: 'rationale concerting "effective" placement is lost. "Supporting blocks" are now "red" blocks without context. The decision to enlarge the ship is undocumented'.

5. The main benefit of the SSDT for Team B is visual feedback on the feasibility of the concept design (Question 5). This is also mentioned by Team A. The latter also includes the ability to store high-level rationale (i.e. design instances) as a benefit of the SSDT.

6. Team A considers the transparency of past design decisions as a key benefit of capturing design rationale (Question 8). P3 mentions that a less structured method may also be used to discuss design rationale, but a structured method is beneficial for knowledge transfer. P4 considers the SDRM to support 'future decision-making', because 'it allows for capturing of rationale and traceability between sessions and designers'. Finally, P3 states design decision-making was supported by the SDRM, because 'the fact that it was to be used forced us into discussing design considerations more explicitly'.

7. Based on the responses to Question 7 no major differences in the structure of teams' negotiation processes can be noted.

8. Regarding takeaways from using the SSDT, P1 mentions that 'having means of storing information/designs/rationale creates flexibility and creative space' and 'verifying and validating (incl. explaining) the solution' takes more time than generating that solution in the first place[4] (Question 9).

9. Regarding takeaways from using the SSDT, the responses of both P3 and P4 are noteworthy. On Day 1, P3 considered that 'spending time on the tool itself distracts from actually designing, but could eventually "pay back"'. However, P2 reflected on Day 2 that 'the tool distracted me less from designing compared to last time.' On Day 1, P4 mentioned 'it is not always necessary to capture rationale for seemingly trivial decisions, but can be especially helpful on compromises and decisions regarding larger system concepts', but considered that 'it's helpful to be able to review decisions and rationale from previous design sessions when you continue with your work' nonetheless.

As discussed in Section 5.6.2, the combination of concept designs and design rationale is important to gain a more complete understanding of the decision-making process. To illustrate the importance of in-situ design rationale, consider the response of Participant 1 in item 4 above. The participant states "'Supporting blocks" [i.e. below radar systems] are now *"red"* blocks without context'. Even just after the design session, this color was remembered incorrectly: it should be blue (Figure 5.17b). Although a small example, it indicates that reconstructing the concept design and design process can be challenging.

Concluding, the responses to the open questions align well with the earlier findings in this section. The results indicate both the SSDT (for concept design development and evaluation) and SDRM (for design rationale capturing and making the dialogue more explicit) were beneficial in the experiment.

[4]This division is in line with the identified activities of layout design identified in Section 1.1.2.

Table 5.10: Responses of the four participants (P1-4) to the open questions in the questionnaire. The responses are split between days and between using traditional documentation and the SDRM.

		Responses							
		Traditional				**SDRM**			
		Day 1		Day 2		Day 1		Day 2	
ID	Question	P1	P2	P1	P2	P3	P4	P3	P4
1	Please describe how you documented your design decisions?	Only in names of blocks and very brief descriptions of save files [designs]	In the end, at saving, we gave the design the rationales. Also [we] saved another copy with changes	Only via a short description during a design save. Furthermore, [in] a notepad on paper -> hard to reuse	With the use of 'store design instance' and adding info to the file.				
2	Which functionalities of the design rationale methodology were most useful to the design case?					System properties (top right area of user interface)	Capturing rationales, interactions, and compromises of/ between system concepts.	Capturing interactions	Capturing system properties and interactions
3	What additional functionalities of the design rationale methodology would be beneficial?					Time tracing for how rationale changes over time, maybe it's already there, but we didn't use it yet.	Addition of bulkhead spacing to system concepts	Track history of decision (it may be already available, but we haven't used it. In real-life cases this will be useful).	Some 'provision for' function to capture decisions related to future updates
4	Where you able to capture all rationale that you wanted to capture? If not, which rationale weren't you able to capture?	No, a lot of rationale per block or instance has not been stored.	The placing of the accommodation and bridge location	No, rationale concerting 'effective' placement is lost. 'Supporting blocks' are now 'red' blocks without context. The decision to enlarge the ship is undocumented	We hadn't enough time to complete the whole vessel	Bulkheads as a whole -> watertight integrity	We captured all the rationale of which I thought that it would not be directly clear to a future reviewer	Properties of systems (operational/ functional)-> potentially use 76mm gun as CIWS in future	Future function of 76mm as CIWS
5	Please describe how the design tool supported design decision-making?	3D view is always good to have. Furthermore, being able to store previous concepts and sketch in 3D was very helpful	If used with more concepts and document it, you will not redo tasks or options that you already checked.	The ability to connect systems to reference points. Saving 'intermediate' designs with brief descriptions	It helps to see where you stopped last time and what the decisions [at that time] were.	Direct visual feedback on if/how components fit into compartments	I'm not sure the design tool supports decision-making directly, but it allows quick visualisation which in turn supports the design process.	Visual check of sanity of design	Easy visualisation of design decisions and their influence

Table 5.10: Responses of the four participants (P1-4) to the open questions in the questionnaire. The responses are split between days and between using traditional documentation and the SDRM - continued.

		Responses							
		Traditional				**SDRM**			
		Day 1		Day 2		Day 1		Day 2	
ID	Question	P1	P2	P1	P2	P3	P4	P3	P4
6	Please describe how the design rationale methodology supported design decision-making?					The fact that it was to be used forced us into discussing design considerations more explicitly.	I'm not sure the SDRM supports decision-making for easy problems solved in a limited time window, but it allows for capturing of rationale and traceability between sessions and designers. It thus improves future decision-making in larger problems.	Given that we had to use the tool/rationale, it forced us into (better) discussing and documenting decisions	Retracing earlier design rationale and constant checks on interactions.
7	How would you describe your team's negotiation process?	Very open, but rather uncontrolled	Good understanding and open discussion	Open and rather unstructured. After the placement of the systems, 'clear' goals were chosen on the spot	It was good because we understood each other. That helps in the decision-making	[P4] had the initiative, I responded with questions and feedback, together resulting in the final designs	Somewhat unstructured but constructive	Fast equal decision-making early on. The last 20 minutes was mostly 'polishing' the design -> no substantial new insights during that phase.	Fairly efficient with little conflict
8	Would the design rationale methodology be beneficial for design (review) sessions, and why?	Yes, storing previous insights helps quick decision-making. Meaning improvement of transparency	Yes, the open view and fast concept development help with review sessions	Yes, for starters the context of the design problem can help in understanding the final result. Having a clearer goal in the beginning will help steer the design process.	It would help because you can show the rationales in the design and helps with the discussion.	For design sessions the rationale can also be discussed without the methodology/tool, but for knowledge transer this structured approach would be beneficial.	Yes! Reviewers and designers will be able to retrieve more of the context of a decision.	Yes see response to question 6	Yes, because it shows thought processes and interactions
9	What takeaways do you have from the experiment and using the design rationale methodology?	Seeing a 3D model really helps, having a 'fallback' helps and a solid quick weight estimation helps	The documentation option is nice and helps	Having means of storing information/ designs/ rationale creates flexibility and creative space. Next to that, giving a design solution is quick, verifying and validating (incl. explaining) the solution takes time.	It helps to use previous designs, an options to check concepts of systems that can be saved and used for new designs (e.g. SEWACO combinations)	Spending time on the tool itself distracts from actually designing, but could eventually 'pay back'	It is not always necessary to capture rationale for seemingly trivial decisions, but can be especially helpful on compromises and decisions regarding larger system concepts.	The tool distracted me less from designing compared to last time.	It's helpful to be able to review decisions and rationale from previous design sessions when you continue with your work.

5.7. DISCUSSION

The results described in the previous section indicate the SDRM is a promising development to enable designers to capture design rationale on-the-fly during the ship layout design process. The case studies presented in Sections 5.2, 5.5, and 5.6 showed that the method can be used for more complex ship arrangement problems (i.e full-size ship in 3D), while Chapter 4 only investigated small-scale 2D layouts. In this section, the following items are briefly discussed:

5

1. *The use of the SDRM in the complete ship synthesis process.* The results described in this chapter show that, despite the specific design rationale representation, the SDRM can be used during overall ship synthesis (Section 1.1.2). The SDRM allows designers to capture System Properties related to the overall ship. Furthermore, the development of design instances in the SSDT allows for the documentation of overall ship design considerations, such as the 'major decisions' documented by Andrews (2022b, Table 2), and design drivers.

2. *Design knowledge inherently built into the SSDT.* Design knowledge can be inherently built into design tools. This can be the case, for instance, when tools are very complex (Reinertsen, 1997). Andrews (2012b) states that design tools should not be a black box. A question to ask is which knowledge is built into the SSDT and what could be the corresponding consequences.

   In principle, there is very limited design knowledge built into the SSDT. A principal example of such knowledge could be the weight and stability calculations. However, these calculations also help designers speed up otherwise time-consuming design activities.

   Furthermore, the baseline design could contain various unjustified items. For example:

   (a) The specific weights of block types have a profound effect on the ship's stability but are not justified. Note that a designer can add System Properties to block types to add such justifications.

   (b) The predefined hull forms are not justified but could be via System Properties.

   (c) In the experiment, the Y- and Z-reference planes were not changed by the teams. However, for example, the Z-reference planes define deck heights and thus vertical positions of systems. Hence, the initial reference planes can also have a significant effect on the ship's stability.

3. *The definition of the Likert scale values.* In Chapter 4, the values across the Likert scale questions in the questionnaire were all defined. In this chapter, only the meaning of the lower and upper boundaries were defined. In hindsight, this choice made the interpretation of responses more difficult because an interpretation of the intermediate values by the researcher was required.

4. *Unused functionalities in the SSDT-SDRM.* Since the search and network functionalities were not used, no feedback on their usefulness was received. A post-experiment suggestion by Team A was to construct "a tree [e.g. network] of rationale and design instances to provide insight into captured decisions". This is already partially supported because the SDRM can generate graphs of design rationale (e.g. Section 5.4.1).

5. *Retrieval of design rationale over time.* Although design rationale can be retrieved (see Section 5.4), the presentation of design rationale elements in the form of tables does limit the designers ability to investigate how design rationale is changed over time. The current implementation works best for the default option to reload 'the most recent versions of all design rationales created'. However, insights into changes over time might be required, for instance, to avoid past design pitfalls.

6. *Implications of number of participants.* Unfortunately, the number of participants in the experiment was limited. As a result, statistically significant results could not be obtained. Nevertheless, the quantitative and qualitative results obtained in the experiment give valuable indications to assess the long-term benefits of the SDRM as elaborated in Section 5.6.

7. *The challenge of science to address wicked problems.* In addition to item 5 above, one should consider to what extent scientific methods are suited to solve wicked problems. As discussed in Section 1.1.1, the key scientific prepositions of reductionism, repeatability, and refutation are challenged by the nature of wicked problems. As a consequence, the applicability of the experiments conducted in Part II might be limited for real-life ship design problems. However, the experiments might be sound from a scientific point of view and can provide useful insight into the interactions between actors and the use of supporting design methods.

Furthermore, the participating teams provided the following considerations in both the questionnaire and post-experiment dialogues:

1. To capture hand-written documentation (such as discussed in Section 5.6.2), the SDRM could be coupled the SDRM to a tablet with sketching capability. This way, for example, screenshots of the design could be easily annotated and the storage of these sketches can be improved.

2. In line with the discussion of the use of the SDRM in the overall ship design process, the suggestion was made that the developed tools could be used by (different) designers working on different subsystems (e.g. Sensors, Weapons, and Command (SEWACO), propulsion concepts, etc.). Subsequently, the designs of the subsystems could be merged into the overall design. This would require the supporting database to be accessible by these designers.

3. A challenge for design rationale systems is how to deal with tacit knowledge and implicit decisions. That is, the designer can make decisions without being aware of that fact. For instance, by focusing on the arrangement of the propulsion train, implicitly, the design space for the topside (or further downstream, accommodation

spaces) might be reduced. How could these decisions be captured? This dissertation, and also DeNucci (2012) for example, suggests that such design rationale can be captured by triggering designers to express 'why' they made certain decisions or to explain 'why the design is what it is'. By regularly capturing design rationale, design dialogues can be made more explicit (Section 5.6). Hence, using the proposed method can be seen as a synchronising and testimonial activity (Section 1.1.1).

4. Not only can design rationale methods be intrusive (Burge & Brown, 2000), also the design tool can be a bottleneck in the design process. For example, the SSDT uses the names of reference planes in the positioning of volume blocks. However, this limits the rapid changeability of, for instance, the order of the reference planes.

5. The approach might be less applicable for designing the complete concept design with a large design team (e.g. too many suggestions to handle by the designer using the tools). Still, the developed tools might be useful for design review in such larger settings. In that case, the focus shifts from concept design development to elucidating design rationale and stakeholder preferences. This activity, which is essential during early-stage ship design (Chapter 1), can be supported by the SDRM.

5.8. CONCLUSION

The main research question for this part of the dissertation was: *"To what extent can design rationale methods support real-time design decisions during early-stage complex ship design?"* This question was already partially answered in Chapter 4. However, development and evaluation of this proof-of-concept method for use in realistic ship design scenarios were needed. In this chapter, the two-fold goal was further specified:

1. To develop a design rationale method to aid designers in the continuous capturing and reuse of design rationale during the collaborative concept design process, and

2. To evaluate how the developed design rationale method benefits collaborative design decision-making such that it leads to better insight into design issues across design teams and better concept designs over time for realistic ship design problems.

To fulfil the first goal, the proof-of-concept method was further developed by proposing a design rationale ontology in Section 5.1. This ontology was tested on an Oceangoing Patrol Vessel (OPV) design problem to investigate the method's scalability in Section 5.2. The results indicated that the developments allow for on-the-fly capturing of meaningful design rationale and that the method should allow designers to retrieve 'data-on-demand'.

Subsequently, the Surface Ship Design Tool (SSDT) was selected as a design tool to be integrated with the design rationale method and further developed in Section 5.3. Then, the expanded proof-of-concept design rationale method (Chapter 4) was enhanced into the Ship Design Rationale Method (SDRM) in Section 5.4. The SDRM was integrated with the SSDT to allow designers to concurrently perform concept design (in this case applied to ship layout design) in small-scale collaborative settings and capture design rationale (i.e. the justification behind corresponding design decisions).

To fulfil the second goal, Sections 5.5 and 5.6 elaborate on a multi-phase design experiment with two-person teams to demonstrate the SDRM's long-term benefits on a frigate design problem with changing requirements. The results indicate that the SDRM can be used to capture design rationale on-the-fly, even for large-scale design problems. Also, it was observed that, in line with the findings in Chapter 4, the use of the SDRM seems to trigger more explicit design reasoning compared to an increased focus on designing. The results show limited reuse of captured design rationale, although participants acknowledge that the SDRM enables such reuse. The most noticeable reuse was observed at the start of the design session, to get up-to-date with the status of the design.

In Table 5.11, the method requirements set in Chapter 4 and the compliance of the SDRM to these requirements are summarised. All method requirements are fulfilled throughout this chapter. To be noted is the compliance of DRQ3, i.e. the reduced rationale-based feedback (e.g. MoPs) compared to Chapter 4. Since this requirement was already fulfilled by the proof-of-concept, further evaluation of such design guidance options was not further investigated in this chapter but is left for future research (Chapter 6).

Table 5.11: SDRM: compliance with method requirements. [1]: Proof-of-concept - for details see Chapter 4.

ID	Requirement	Compliance PoC[1]	Compliance SDRM
DRQ1	The method must be applicable for early-stage collaborative design activities and promote feedback-driven conversations	Partially fulfilled	Fulfilled (Section 5.4)
DRQ2	The method must enable the capture and review of design decisions, the rationale behind these decisions, and temporal relationships between design decisions	Partially fulfilled	Fulfilled (Section 5.4)
DRQ3	The method must provide immediate rationale-based feedback to increase the benefits relative to the costs of capturing design rationale	Fulfilled	Fulfilled (Section 5.4) - although less capabilities have been implemented in the SDRM compared to Chapter 4.
DRQ4	The method must be generic	Partially fulfilled	Fulfilled (Section 5.5)
DRQ5	The method must be easy to use and integrated within design tools	Partially fulfilled	Fulfilled (Section 5.4)

In conclusion, to answer the research question, the results described in this chapter indicate that the developed SDRM can be used to capture design rationale in real-time during early-stage complex ship design. This applies to both layout design and wider ship synthesis, as demonstrated in the case studies. Also, the results indicate that the SDRM triggers more explicit design conversations. This is beneficial during early-stage design, in which the stakeholder dialogue is especially important. Although there is potential value in the captured design rationale, only limited reuse was observed during the experiment. It was noticed that designers used the SDRM to re-familiarise themselves with the concept design and corresponding design rationale. The longer-term benefits and challenges (e.g. data management) of the SDRM (e.g. for the duration of a whole ship design process) need further investigation.

6
CONCLUSIONS

He ends, of course, by satisfying neither the Commander who is responsible for the men's living conditions nor the Gunnery Officer who is responsible for the guns, but that is the natural fate of the designers of ships – the speed enthusiasts, the gunnery experts and the advocates of armour protection, the men who have to keep the ships at sea and the men who have to handle them in action all combine to curse the designer.
Then comes the day of battle and the mass of compromises, which is a ship of war, encounters another ship of war, which is a mass of different compromises, and then, ten to one, the fighting men on the winning side will take all the credit to themselves and the losers – such of them that survive – will blame the designer all over again.

C.S. Forester "The Ship" (1942)

This chapter provides the conclusions of this dissertation. First, the early-stage ship design problem addressed in this dissertation is revisited. Subsequently, the conclusions that can be derived from the two-fold solution to this problem, described in Parts I and II are provided. Then, the main contributions of this research are summarised. Finally, some recommendations for further research are given.

6.1. REVISITING THE PROBLEM

In Chapter 1, the overall research problem was described. In summary, this problem can be described as follows:

1. During the early-stage design of complex vessels, such as naval vessels, a 'wicked' problem needs to be solved (Section 1.1.1). On the one hand, the design problem (i.e. requirements) needs to be formulated. On the other hand, via the stakeholder dialogue, this problem formulation is influenced by the generated solutions (e.g. concept designs). Such a problem can be described as a 'wicked' problem, which

lacks a consensus on the problem and solution across stakeholders. To inform the stakeholder dialogue, designers need to gain insight into the technical feasibility, costs, and risks of these requirements and potential design solutions. These aspects need to be addressed early on because of the lock-in of the concept design by, mostly, early decisions. In this situation, rework is considered a challenge because of the high cost of late design changes and subsequently might be reduced by providing designers with more accurate information on technical feasibility and risk. Yet, wicked problems cannot be solved by technological solutions only, as there is an interrelated social aspect to be considered as well, as illustrated by the quote by Forester at the beginning of this chapter.

2. Layout design is selected as a prime example of an important aspect of ship design in Section 1.1.2, for the following reasons. Firstly, the ship's layout represents the integration of all design aspects. Secondly, the layout is input to many design disciplines and is essential in the stakeholder dialogue. Thirdly, layouts are difficult to capture in requirements. Typically, layouts are developed with increasing fidelity (i.e. level of detail) throughout early-stage design. In complex ship layout design, a principal challenge is the effort required to obtain insights into potential detailed sizing and integration issues and risks that might be encountered later in the design process. Underestimating such risks can cause costly and time-consuming rework. Also overestimating such risks could result in oversized and too costly designs. Current design methods lack the speed or detail to provide sufficient insight into these risks.

3. Besides the identification of physical integration issues, designers require design rationale (i.e. the justification of design decisions) to make informed design decisions (e.g. when rework is required), as described in Section 1.1.3. The challenge is that current design tools do not support the designer to capture and reuse design rationale in a meaningful way. In practice, design rationale may be documented. In addition, existing research shows that design rationale can be captured and reused by individual ship designers. However, there is currently no suitable design rationale method that allows for integrated, in-situ documentation of design rationale during the complex ship layout design. This is however essential to capture both the decision and its context, the concept design. as described in Section 1.1.2. Hence, it is currently unknown how the potentially intrusive activity of design rationale capture can be effectively integrated into the complex ship layout design process.

The identification of these challenges led to the following overall research goal for this dissertation (Section 1.2):

To reduce the effort required to identify and solve detailed layout integration issues during social-technical early-stage complex ship design via automated layout generation and design rationale capturing.

The next section will elaborate on the research questions set to fulfill this research goal.

6.2. CONCLUSIONS

In Section 6.1, the research problem was revisited and the overall research goal was restated. This research goal was further specified into two main research questions. In this section, first, the answers to the research questions will be given. Then, a conclusion will be drawn if and how the overall research goal is satisfied. The main research questions were:

RQ1. *To what extent can automated layout generation methods support real-time design decisions during early-stage complex ship design?*

This research question has been answered in Part I of this dissertation.

First, from the nature of ship layout design, Section 2.1 established that such an automated layout generation method should be able to generate arrangement drawings of sufficient detail (i.e. space level) at sufficiently high speed (i.e. in the order of minutes). Also, it should be responsive to the designer and provide feasible and believable results. A set of six method requirements was proposed.

Second, a literature review of layout generation methods and tools revealed that no existing layout design method fulfils all requirements.

Third, the mathematical working principles of a new ship layout generation method called WARship GEneral ARrangement (WARGEAR), were described in Section 2.4. WARGEAR allows designers to rapidly generate and evaluate concept designs for ship layouts at a higher level of detail, based on a predefined functional arrangement comprising the ship's main building blocks.

(a) First, the designer provides the main input to WARGEAR.

(b) Subsequently, WARGEAR arranges passageways and staircases with a probabilistic staircase placement algorithm.

(c) Then, it uses a network-based approach combined with probabilistic selection for the allocation of spaces to compartments. The allocated spaces are arranged using cross-correlation to enable a very fast arrangement of large layouts.

(d) Finally, a 'carving'-based approach is applied to ensure connectivity throughout the ship. The method is steered by a bi-level particle swarm optimisation code.

These mathematical principles of WARGEAR make it capable of generating space-level arrangements in a matter of minutes, meeting the method requirements.

Chapter 3 described three case studies applying WARGEAR and one case study extending WARGEAR's algorithm. These case studies showed the following:

(a) Case Study 1 (Section 3.1) showed that WARGEAR can help at gaining insights into sizing and integration issues. This was demonstrated via a notional surface vessel. Also, these design insights can be used in subsequent design iterations - i.e. WARGEAR can be used in an interactive manner. After generating a set of design solutions using WARGEAR, a detailed layout was selected and analysed. Based on the analysis, the input to WARGEAR was adapted and an improved set of layouts could be generated in approximately 15 minutes.

(b) Case Study 2 (Section 3.2) showed that WARGEAR can generate design insights in a limited time (in the order of minutes) for a realistic design case. This exercise also showed that in two days, more design variations can be studied using WARGEAR compared to a human designer generating a General Arrangement Plan (GAP) using regular Computer-Aided Design (CAD) software (150 hours). Yet, the outcomes of the WARGEAR and CAD-based design studies were comparable.

(c) While the first two case studies focused on validating WARGEAR, Case Study 3 (Section 3.3) extended a part of the WARGEAR algorithm to investigate how automated design tools for ship layout design can be used in a real-time manner. The results show how insight can be gained into complex and incomprehensible interrelationships been systems and the overall ship layout in a relatively limited time. In addition, it was found that the calculation time can be minimal, even for complex problems, but most effort is required for input generation and post-processing of the results. Tailor-made and flexible visualisation tools can help designers to identify items of interest faster.

(d) Case Study 4 provided an additional application of WARGEAR. It was integrated with a queueing-based logistic performance assessment to support naval architects during the early-stage design of internal layout and process-driven ships. An application to a Landing Platform Dock (LPD) confirmed WARGEAR can be used to identify sizing and integration issues early on. In addition, the results of the queueing-based method could be combined with detailed layouts to provide insight into possible bottlenecks in the operational processes and the relation of these bottlenecks to the ship's layout.

The case study results indicate that the mathematical principles of WARGEAR make it capable of generating space-level arrangements in a matter of minutes. Furthermore, the results show that WARGEAR fulfils the six method requirements (Section 3.5). Based on these results, the research question can be answered. Automated layout generation methods can be used to support real-time design decisions during early-stage complex ship design, if the following aspects are satisfied:

- The method can generate layouts and corresponding data and information in the order of minutes. This is achievable with proper modelling, even for detailed layouts.
- The method allows for rapid (i.e. also in the order of minutes) evaluation of layouts and underlying data and information to gain insights into the feasibility and risks of these layouts. This is supported by data visualisation and metrics.
- The method is responsive to stakeholder needs (which includes the need for specific insights). This can be achieved by, among others, generating detailed layouts based on a predefined (potentially changing) lower fidelity model.

As such, the layouts generated by WARGEAR function as external cognitive objects (Section 1.1.1).

RQ2. *To what extent can design rationale methods support real-time design decisions during early-stage complex ship design?*

This research question has been answered in Part II of this dissertation. First, Section 4.1 investigated how an integration between design rationale capturing and design could benefit early-stage design, by, for instance, stimulating the generation of negotiated knowledge in the stakeholder dialogue. Subsequently, a set of five method requirements for such in-situ design rationale capture was proposed. Subsequently, it was concluded that no existing design rationale method is directly suitable to support design rationale capture and retrieval in collaborative complex ship layout design. Hence, the goal was:

(a) To develop a design rationale method to aid designers in the continuous capturing and reuse of design rationale during the collaborative concept design process, and,

(b) To evaluate how the developed design rationale method benefits collaborative design decision-making such that it leads to better insight into design issues across the design team and better concept designs during a single design session (for a small design problem in Chapter 4) and over time (for realistic ship design problems in Chapter 5).

To fulfil the first goal, a new design rationale method was proposed in Section 4.2. The method was implemented as a proof-of-concept to demonstrate how design rationale might be captured and reused during the collaborative design of ship layouts. The method integrates a record-and-replay design rationale capturing approach with a layout design tool. A subset of an existing design rationale representation scheme is used. This method provides both short-term benefits (e.g. it enables the creation of a common knowledge base during design sessions) and long-term advantages (e.g. it allows the review of the context of past design decisions before changing a concept design).

In Section 5.1, the proof-of-concept method was further developed by proposing a new design rationale ontology. This ontology was tested on an Oceangoing Patrol Vessel (OPV) design problem to investigate the method's scalability in Section 5.2. The results indicated that the developments allow for on-the-fly capturing of meaningful design rationale and that the method should allow designers to retrieve 'data-on-demand'.

Subsequently, the Surface Ship Design Tool (SSDT) was selected as a design tool to be integrated with the design rationale method and further developed in Section 5.3. Then, the expanded proof-of-concept design rationale method was enhanced into the Ship Design Rationale Method (SDRM) in Section 5.4. The SDRM was integrated with the SSDT to allow designers to concurrently perform concept design (in this case applied to ship layout design) in small-scale collaborative settings and capture design rationale (i.e. the justification behind corresponding design decisions) on the fly.

To fulfil the second goal, a small-scale experiment was developed and executed, as described in Sections 4.3 and 4.4. Teams comprising three participants (i.e. students and experts from the industry) worked on two small-scale layout problems. The results of the design experiment indicate that using a design rationale method while designing a layout can have both measurable and perceived benefits. An example of the former is that the design rationale method motivates teams to use 'network arrangement', as indicated by the results (Section 4.4.1). Such network arrangement of systems visually supports the team in sketching the initial arrangement of systems. Participants generally perceived the design rationale method to facilitate enrichment and negotiated knowledge (Section 4.4.2), aspects aiding to provide a better understanding of the design problem within the entire design team. In other words, the SDRM functions as a means for synchronising between actors (Section 1.1.1).

Additionally, Sections 5.5 and 5.6 elaborate on a multi-phase design experiment with two-person teams to demonstrate the SDRM's long-term benefits on a frigate design problem with changing requirements. The results indicate that the SDRM can be used to capture design rationale on-the-fly, even for large-scale design problems. Also, it was observed, in line with the findings in Chapter 4, that the use of the SDRM seems to trigger more explicit design reasoning compared to an increased focus on designing. The results show limited reuse of captured design rationale, although participants acknowledge that the SDRM enables such reuse. The most noticeable reuse was observed at the start of design sessions, to get up-to-date with the status of the design.

Thus, the development of the SDRM and the conducted experiments fulfil the twofold goal stated above. In addition, the SDRM fulfils the method requirements (Section 5.8). Based on these results, the research question can be answered. The results described above indicate that the developed SDRM can be used to capture design rationale in real-time during early-stage complex ship design. This applies to both layout design and wider ship synthesis. Also, the results indicate, the SDRM triggers more explicit design conversations. This is beneficial during early-stage ship design, in which the stakeholder dialogue is especially important. Although there is potential value in the captured design rationale, only limited reuse was observed during the experiment. It was noticed that designers used the SDRM to re-familiarise themselves with the concept design and corresponding design rationale. The longer-term benefits (e.g. for the duration of a whole ship design process) need further investigation.

Now that the two main research questions have been satisfactorily answered, the overall research goal, *To reduce the effort required to identify and solve detailed layout integration issues during social-technical early-stage complex ship design via automated layout generation and design rationale capturing*, is revisited. In addition to the detailed responses to the two main research questions above, the following additional remarks will be made to revisit the research goal:

- The effort required to identify and solve detailed layout integration issues in complex ship layout design is significantly reduced by WARGEAR, thus automated layout generation (Part I). Indeed, a 150 work hours manual GAP generation design task can now be completed, including a range of variation studies, within hours. Additional manual work might be required if the level of detail needs to be further increased (e.g. in support of further risk mitigation studies).

- The capture and subsequent reuse of design rationale takes effort but seems acceptable in a small-scale collaborative design setting, i.e. two to three persons per team as investigated in Part II. It allows the capture of layout design issues for further reconsideration, potentially reducing costly and time-consuming rework. In large-scale settings (e.g. over 10 persons), the effort to capture design rationale in the proposed way might be too intrusive, but this needs further investigation.
- The automated layout generation and design rationale capture efforts described in this research are not directly coupled (e.g. to the extent that it was coupled in past research in ship design). This is due to a different focus regarding the fidelity of layouts in both parts. While WARGEAR aims at generating detailed layouts based on functional layouts, the SDRM focuses on overall ship synthesis on functional and spatial levels. Therefore, the coupling is present in the design process, since both methods are related to functional layouts (e.g. compare Figure 2.2b).

In conclusion, the research presented in this dissertation contributes to solving wicked early-stage complex ship design problems (specifically for ship layout design) by allowing designers, first, to identify potential sizing and integration issues in complex ship layouts, with less effort than required in current ship design practice. Second, the methods enable designers to capture their design reasoning in the context of the progressing concept design, which supports both current and future decision-making. However, due to the nature of early-stage complex ship design, the methods will not be *decisive* but are nonetheless *supportive* to the human designer.

6.3. SCIENTIFIC CONTRIBUTIONS

The research presented in this dissertation led to an enhanced way of supporting design decision-making during early-stage complex ship layout design. In this section, the following scientific contributions of the research will be discussed:

1. *The development of a layout design method to provide real-time insight into layout sizing and integration issues during early-stage complex ship design.*
   Specifically, this method, WARGEAR, enables designers to rapidly generate and evaluate concept designs for ship layouts at a higher level of detail. This reduces potential risks later in the design process and, thus, costly rework. The generation of detailed layouts within minutes, based on a predefined functional arrangement, is enabled through the first use of cross-correlation to enable a very fast arrangement of large layouts on a space level. In addition, a probabilistic staircase placement algorithm, a network-based approach combined with probabilistic selection for allocation of spaces to compartments, and a 'carving'-based approach to ensure connectivity, have been used to create realistic detailed arrangement plans. WARGEAR is evaluated in both theoretical and practical applications, with positive results, described in this dissertation. Just as some of these developments

were inspired by previous research in other research fields, the modelling techniques presented in Part I can also be applied to non-ship layout generation (e.g. architecture).

2. *The development of a design rationale method to enable design decision traceability during early-stage complex ship layout design.*
   A new design rationale method, the SDRM, was developed to enable designers to concurrently work on the concept design, and capture and retrieve design rationale. Therefore, the SDRM is integrated with an early-stage ship layout design tool (the SSDT). The SSDT was re-implemented and extended to support the research, as well as to support educational purposes at Delft University of Technology. Besides the long-term documentation advantages of design rationale capturing, this research found that an important effect of such an integrated approach to design rationale for ship layout design is that it triggers designers to be more explicit about their design reasoning and decisions. This in itself already improves collaborative design decision-making.

3. *The experimental evaluation of short and long-term benefits of design rationale capture and reuse for complex ship layout design.*
   Two case studies were executed to evaluate the SDRM. In the first case study, a small design problem is used in three-person teams to investigate the role of design rationale within design sessions. In the second case, a partial frigate arrangement problem is used in two successive design sessions (with an interval of multiple weeks) to investigate the value of design rationale over longer periods of time.

This dissertation was financially supported by the Dutch Ministry of Defence, and many of the examples and case studies throughout the study are related to naval vessels. Yet, the ethical aspects and justification of warfare are not addressed in this dissertation. However, these aspects need careful consideration. For instance, in politics, careful consideration should be made regarding the use of military power to achieve political goals. Furthermore, designers of naval vessels should also be aware of the ethical aspects of naval warfare and should consider these, where applicable, in, for instance, their advice in the stakeholder dialogue.

6.4. RECOMMENDATIONS

Based on the research presented in this dissertation, the following recommendations for future research are given:

1. *Integration of WARGEAR with manual design tools (such as Functional Integrated Design Exploration of Ships (FIDES) and the SSDT).* Above, the coupling of design rationale and automated layout generation via the design process was highlighted. Further integration of the tooling would reduce the designer's workload and speed up design work, resulting in more rapid design insights. The development of interactive data exploration and decision tracing might be a promising and essential research direction to gain such insights. Also, WARGEAR might be extended to

allow the arrangement of less flexible arrangeable parts of the ship, such as machinery rooms. This enables the designer to use the available tooling to support the stakeholder dialogue as required.

2. *An evaluation of the usefulness of Measures of Performance (MoPs), and other design rationale-based design feedback, during actual design work.* This is remaining work from Case Study 5 in Chapter 4 (see also Section 5.8). How do designers use these MoPs in practice, and how to avoid the excessive focus on optimising MoPs? Also, which design rationale-based MoPs are suitable for real ship design, and how to ensure these consider the right set of design rationale? Carefully generated MoPs can provide useful design insights (e.g. Sections 2.4.8 and 4.3.2).

3. *Enable the storage and retrieval of hand-written documentation in the context of the concept design.* The coupling of digital sketching applications (e.g. tablets) to ship design tools to store hand-written documentation in the context of the design. This would allow designers to work on subsequent thinking, talking, and storing sketches, that could contain important design rationale.

4. *Evaluation of long-term benefits and challenges of the SDRM.* The benefits of design rationale for the duration of a whole ship design process have not been investigated in this dissertation. Also, ways to support designers with 'data-on-demand' to filter through the potentially large database with design rationale need attention.

5. *Prepare for solving wicked problems.* This dissertation focused on reducing design rework. However, since early-stage complex ship design is about solving a wicked problem, it is, on the one hand, worthwhile to investigate what the impact of design changes is on the overall design progress, cost, etc. On the other hand, the impact of social aspects on similar metrics should be investigated too. This would allow for a balanced view on what future designers (i.e. today's and future students) and today's employees should be trained for. As indicated in this dissertation, this will likely have both a technical and a social focus.

6

6.5. SUPPLEMENTARY DATA AVAILABILITY

The table below provides, where possible, references to data sets underlying the results presented throughout this dissertation. Access to the source code developed for this dissertation may be granted for research and educational purposes. This is subject to written approval from Delft University of Technology, the Netherlands Materiel and IT Command and the author of this dissertation.

Internal reference	DOI/Remark
Case Study 1 (Section 3.1)	https://doi.org/10.4121/19106903.v1
Case Study 2 (Section 3.2)	No further data is provided due to confidentiality of the case study details
Case Study 3 (Section 3.3)	https://doi.org/10.4121/20141636.v2
Case Study 4 (Section 3.4)	https://doi.org/10.4121/e3eb2bab-8e28-4477-a34f-ba7c94d0d80b
Case Study 5 (Section 4.4)	https://doi.org/10.4121/21502338.v1
Case Study 6 (Section 5.2)	https://doi.org/10.4121/19430396.v1
Case Study 7 (Section 5.6)	https://doi.org/10.4121/e3eb2bab-8e28-4477-a34f-ba7c94d0d80b

A

EVALUATING WARGEAR ELEMENTS

This appendix contains supplementary material to Part I of this dissertation.

A.1. EVALUATING THE ALLOCATION METHOD

This section contains supplementary material to Section 2.4.5.

In an attempt to falsify the allocation method outlined in Section 2.4.5, the following variations to the method have been tested.

1. The fraction between the degrees of spaces and compartments is removed from $P_{allocation,ij}$.
2. The fraction between the areas of spaces and compartments is removed from $P_{allocation,ij}$.
3. The order in which spaces are allocated is determined via sorting by ascending degree only.
4. The order in which spaces are allocated is determined via sorting by descending area only.

Each variation is separately tested on two sets of input. Each variation is tested 10,000 times with randomly generated roulette wheel positions x. Besides the four variations the baseline allocation method without alternations, i.e. variation 0, is tested on the test case input. The two input sets contain 100 and 75 spaces, to be allocated to 9 and 14 compartments respectively. The two input sets differ mainly in the number of interactions between compartments. The former has fewer interactions, i.e. it's less likely that a non-optimal allocation of some spaces will lead to problems for other spaces. Also, the compartments in the latter set are relatively small, which could make allocation difficult. However, the total available area in comparison to the total required area is larger in the

A

latter set (30% more area available than required) than in the former set (16%), which could compensate for the challenge posed by the size of the compartments. Details on space and compartment sizes for the two cases can be found in Tables B.1, B.2, and B.3.

Table A.1: Results of 10,000 allocation attempts for five variations to the allocation method. [1]: Out of 100 spaces times 10,000 attempts. [2]: Out of 75 spaces times 10,000 attempts

	Test case 1 (100 spaces)		Test case 2 (75 spaces)	
	Number of non-allocated spaces[1]	% of arranged spaces	Number of non-allocated spaces[2]	% of arranged spaces
0	2	0.0002	2249	0.2999
1	729	0.0729	28381	3.7841
2	0	0.0000	1627	0.2169
3	1	0.0001	3510	0.4680
4	11089	1.1089	0	0.0000

The results of the two tests are summarised in Table A.1. The results show that for the first test variation 0, 2 and 3 outperform variation 1 and 4. However, for the second test, variation 4 outperforms all other variations. At the same time, variations 0, 2 and 3 perform comparable and significantly outperform variation 1. The difference in the performance of variation 4 for the two tests might be an indication that the variations are sensitive to the space list and the allocation of spaces to functional blocks. The results indicate that taking the required area of spaces and available area in compartments into account is important. The negative result of variation 1 (and 4) shows that it is important to take the degree of compartments and spaces into account.

Further, the area utilisation of compartments is investigated, i.e. to which extent do the variations use the available area in compartments? For the sake of brevity only the three overall best-performing variations, i.e. variations 0, 2 and 3, are discussed here. Figure A.1 provides the mean, median, minimum, and maximum area utilisation over 10,000 allocation attempts for test case 2. The following main observations can be made:

1. Variation 0 and 3 perform similarly, with an exception for compartment 3, where the median utilisation of variation 0 is lower than that of variation 3.

2. The mean area utilisation of variation 2 differs from variation 0 and 3 in almost half of the compartments. In some cases the utilisation is higher and in others lower. A high compartment utilisation at this point in the layout generation process could be an indication of design integration issues later on when spaces are actually arranged (see Section 2.4.6) and connected (see Section 2.4.7).

3. The minimum area utilisation of variation 2 is frequently more than ten percent point lower than variations 0 and 3, indicating that for these allocations either not all spaces could be allocated, or the other compartments have a (too) high utilisation as discussed above.

In conclusion, the baseline method (variation 0) outperforms variation 3 with regards to the number of allocated spaces and provides a slightly more balanced allocation from a compartment utilisation perspective than variation 2. Therefore the initial proposed allocation method remains best and is used further in the model.

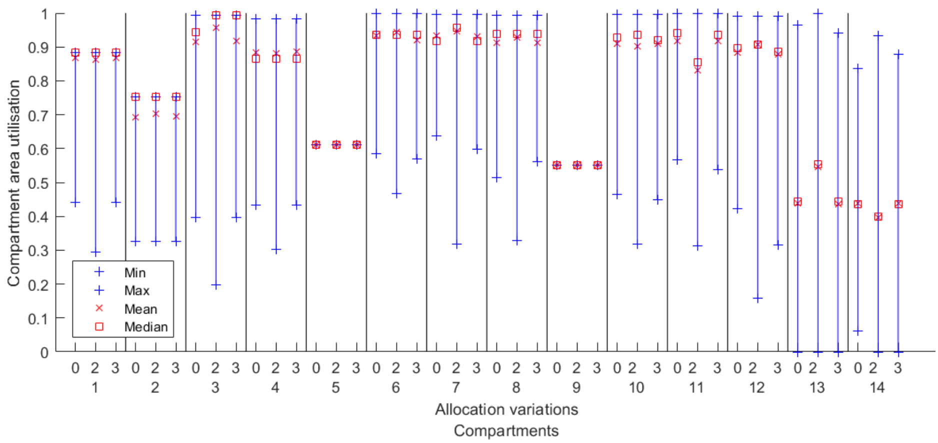


Figure A.1: Compartment area utilisation (fraction of total available area) for each compartment for variations 0, 2 and 3 in test case 2.

A.2. EVALUATING THE SPACE ARRANGEMENT METHOD

This section contains the supplementary material to Section 2.4.6.

This section elaborates on the performance of the new cross-correlation-based space arrangement method, presented in Section 2.4.6. To do so, the performance of the cross-correlation arrangement method will be compared to the performance of the seed and growth algorithm previously used in WARGEAR (le Poole et al., 2019). Both algorithms will be used to arrange different sets of spaces in various positioning matrices. Also, the performance of the various space order, position selection, and initial space orientation will be investigated.

Space variations

The performance of the space arrangement method is tested with various sets of spaces, which are presented in Table A.2. The variations include sets of spaces with equal size (variation 1, 2, 3, and 9), sets with spaces with small differences in size (variation 4, 7, and 8), and with a large difference in size (variation 5, 6, and 10). Variation 8 will likely result in spaces with areas larger than RA, as the sizes 23, 34, and 43, when divided by an integer number, result in a modulus.

Positioning matrix variations

Six positioning matrix variations are used in this case study, comprising a square, rectangle, and L-shape positioning matrix, as presented in Table A.3. Each of these shapes is sized in relation to the space variations, in two ways:

1. The area is equal to the sum of the area required by the spaces.
2. The area exceeds the sum of the area required by the spaces by 10%.

Arrangement problems where no spare area is available are considered to be harder than the problems where there is void space (Peng et al., 2014). The former variation is mainly

Table A.2: Ten sets of spaces with varying required area used in the space arrangement test case.

Space	**Space variations**									
nr.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	20	20	20	40	100	100	30	43	100	100
2	20	20	20	40	50	100	30	43	100	100
3	20	20	20	20	50	50	30	34	100	100
4	20	20	20	20	50	50	30	34	100	100
5	20	20	20	20	50	20	30	27		20
6		20	20		20	20	25	27		20
7		20	20		20	10	25	27		20
8		20	20		20	10	25	27		20
9		20	20		20	10	25	23		
10		20	20		20	10	25	23		
11			20				20			
12			20				20			
13			20				20			
14			20				20			
15			20				20			

used to test the performance of WARGEAR, while the latter better represents actual ship design layout problems. Indeed in ship layouts the available area for space arrangement is typically larger than the area strictly required for spaces, as additional area for staircases and passageways is required. Typically margins are used to account for staircases, passageways, and for space arrangement considerations. The grid size in this test case is 1x1 meter.

Arrangement variations

The optimisation algorithm used in this test case is a Particle Swarm Optimisation (PSO). The choice for the PSO is further elaborated on in Section 2.4.8. In Table A.5 the arrangement options are provided. The number of required variables for each option is also given. An arrangement variation is a combination of one option from each of the three categories. For instance, one variation is 3-A-I, in which the optimisation algorithm is used to determine the arrangement order, using n_{space} variables, the first available position is selected for each space, and the first available orientation is used to determine the size of matrix B. For *Position selection* a fifth option is also considered, in which the optimisation algorithm can select one, preferably the best, position selection option from A to D. This option has been included to investigate whether space and positioning matrix-based selection of the position selection method yield better results compared to a fixed position selection method. In total 3 (1-3)x5 (A-E) x2 (I-II) = 30 arrangement variations need to be studied to determine the performance of the cross-correlation space arrangement method. Additionally, the seed and growth space arrangement method is also applied for each space and positioning matrix variation.

Two stopping criteria have been used in the test case. First, the optimisation is stopped when an optimal design is found, i.e. all spaces have met their required area and the theoretically maximum objective value has been reached. The objective function is given in Equation A.1 and its maximum value is 100. The second stopping criteria is the maximum number of iterations of the optimisation algorithm. The settings of the optimisation algorithm are given in Table A.4 and have been established based on experience using WARGEAR. For each variation, the time till a stopping criteria is reached, $TimeTillStop$, is stored as well as the best arrangement, and the objective value

Table A.3: Six different positioning matrices used for space arrangement in the test case. Since the grid size is 1x1 m, for a given a real number x, $\lceil x \rceil$ denotes ceil(x), which returns the least greater integer than or equal to x. For example, $\lceil 1.2 \rceil = 2$.

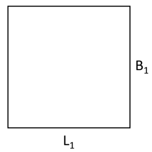
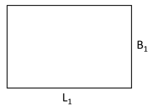
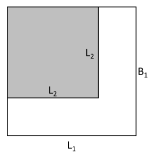
Shape	**Area** $[m^2]$	**Dimensions [m]**	**Visualisation**
Square 1	$A = \sum_{i=1}^{nspaces} RA_i$	$L_1 = B_1 = \lceil \sqrt{A} \rceil$	
Square 2	$A = 1.1 \cdot \sum_{i=1}^{nspaces} RA_i$	$L_1 = B_1 = \lceil \sqrt{A} \rceil$	As Square 1
Rectangle 1	$A = \sum_{i=1}^{nspaces} RA_i$	$L_1 = \lceil 1\frac{1}{2} \cdot \sqrt{A} \rceil$ $B_1 = \lceil \frac{2}{3} L_1 \rceil$	
Rectangle 2	$A = 1.1 \cdot \sum_{i=1}^{nspaces} RA_i$	$L_1 = \lceil 1\frac{1}{2} \cdot \sqrt{A} \rceil$ $B_1 = \lceil \frac{2}{3} L_1 \rceil$	As Rectangle 1
L-shape 1	$A = \sum_{i=1}^{nspaces} RA_i$	$L_1 = B_1 = \lceil 1\frac{1}{2} \cdot \sqrt{A + L_2^2} \rceil$ $L_2 = 10$	
L-shape 2	$A = 1.1 \cdot \sum_{i=1}^{nspaces} RA_i$	$L_1 = B_1 = \lceil 1\frac{1}{2} \cdot \sqrt{A + L_2^2} \rceil$ $L_2 = 10$	As L-shape 1

Table A.4: PSO settings.

	PSO	**Explanation**
NumIt	20	Number of iterations
PopSize	10	Population size
w	0.5	Inertia weight
$\mathbf{w_{damp}}$	0.9	Inertia Weight Damping Ratio
c1	0.5	Personal Learning Coefficient
c2	2.5	Global Learning Coefficient

$BestObjectiveValue$ of that arrangement. To reduce the sensitivity of results to the randomised starting point of optimisation calculations, five runs are completed for each variation.

$$ObjectiveValue = \frac{\sum_{i=1}^{n_{space}} \max(0, RA_i - AA_i)}{n_{space}} \tag{A.1}$$

The various arrangement methods are assessed based on the $ObjectiveValue$, the number of successful arrangements, as well as on the $TimeTillStop$. Indeed, reduced calculation time is important to enable near-real-time feedback to naval architects (Duchateau, 2016; le Poole et al., 2020).

Results

Each arrangement variation has been used to arrange all space variations in all position-

A

Table A.5: Arrangement method options. Each arrangement method variation consists of a space order option, a position selection option, and an initial space orientation option.

	Space order	**Required variables**
1.	Large to small	0
2.	Small to large	0
3.	Optimiser selected order	n_{space}
	Position selection	
A.	First available position	0
B.	Positions closest to CL	0
C.	Positions as far from the compartment's centre	0
D.	Optimiser selected positions	n_{space}
E.	Optimiser selects from A.-D	n_{space}+1
	Initial space orientation	
I.	First orientation in $B_{range,dimensions}$	0
II.	Optimisation algorithm selects longitudinal or transverse direction from $B_{range,dimensions}$, when choice is possible	n_{space}
S&G	Seed and growth	n_{space}

ing matrix variations. This test has been conducted five times to reduce the likelihood that the optimisation algorithm stopped in local optima, resulting in 10x6x5 = 300 layouts for each arrangement variation. Figure A.2 shows a histogram of the quality of resulting layouts for each arrangement variation. The following observations can be made:

1. The optimiser selected space order (3) is more effective than the fixed order methods (1 and 2). This is both the case for layouts that meet their required area (RA), and for layouts with achieved area (AA) = 95 till 99% of RA. However, arranging spaces large to small proves to yield better results than arranging small spaces prior to large ones.

2. Only three arrangement methods are able to fully arrange more than 50% of the layouts, namely 3AII, 3CII, and 3EII. Since the option E actually selects one of the options A till D, a further investigation into this selection was made. Figure A.3 shows a histogram with the four position selection options. Again, the options A and C yield better results than the options B and D.

3. The variable initial space orientation (II) consistently outperforms the fixed initial space orientation (I), regardless of the space order and position selection method.

4. The seed and growth arrangement method is outperformed by 28 of the 30 cross-correlation methods based on layouts at 100% RA, and by all cross-correlation methods if the 95-100% interval is considered. Therefore the change to cross-correlation based arrangement can easily be justified.

Besides quality concerns, the seed and growth algorithm was found to be time consuming. Therefore the computation time required by the arrangement variations will be considered next, see Figure A.4. The main observation to be made is that cross-correlation based methods require up to 1.125 seconds to complete their arrangement attempt, while the seed and growth algorithm requires up to 2.971 seconds. So, based on the lowest speeds of the methods, cross-correlation methods are almost three times faster than the seed and growth algorithm. However, the speed difference between cross-correlation and the seed and growth algorithm is even larger if the average speed is con-

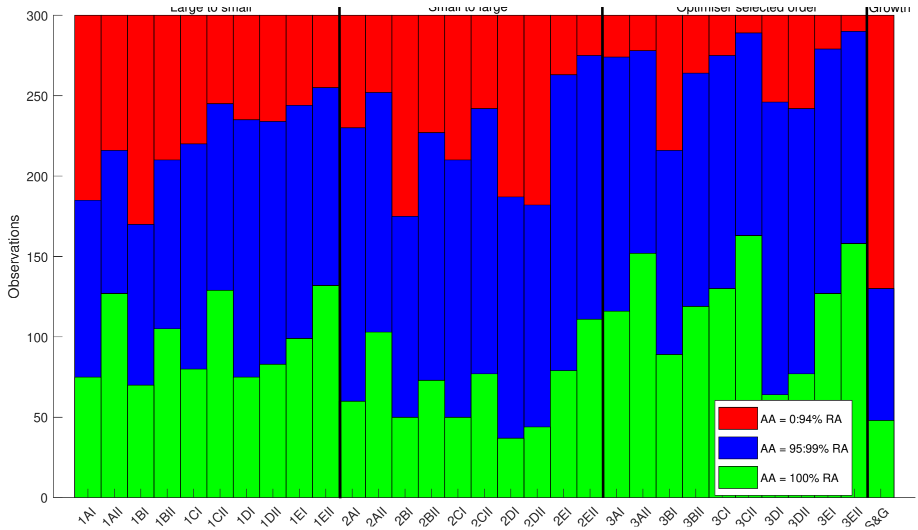


Figure A.2: Histogram with quality of layouts for each arrangement variation. See Table A.5 for definitions of arrangement variations.

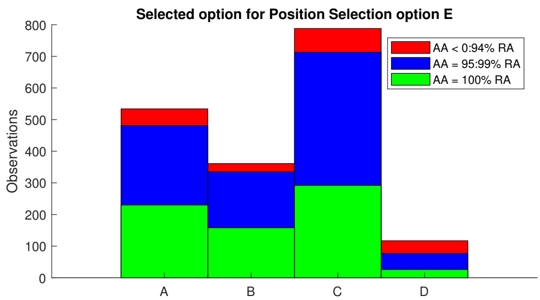


Figure A.3: Histogram with success of selected options by position selection option E.

sidered. On average cross-correlation based methods require 0.022 seconds to find layouts that meet the required area. In contrast, the seed and growth method requires 0.436 seconds to get the same results. Thus, on average, cross-correlation is 20 times faster than seed and growth.

The test case above shows that cross-correlation methods outperform the seed and growth algorithm on the two performance criteria, i.e. quality and calculation time. However, in order to select the preferred arrangement method, the performance of the overall method should be considered. More elaborate test cases with the WARGEAR method showed that there is a tight relationship between space arrangement and local connectivity, see Section 2.4.7. Indeed, the arrangement of the spaces, and, implicitly, the unused area left between spaces determines how much area needs to be 'carved' from spaces to ensure connectivity. This results in a reduction of the size of arranged

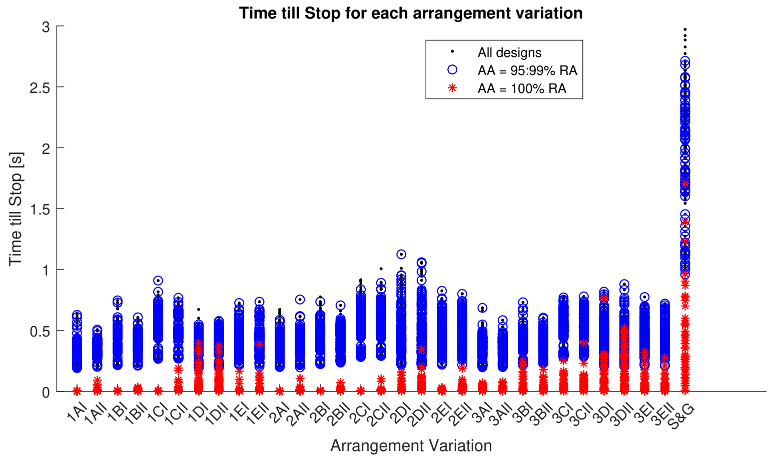


Figure A.4: Scatter plot with time till stopping criteria was reached for each arrangement variation.

spaces, and thus spaces that initially met their required area might fail to meet this criterion after connectivity has been established. This issue is further elaborated on in Section 2.4.7. The best-performing arrangement variations 3AII, 3CII, and 3EII will be further tested in Section A.3 as well, where the selection options for 3EII are limited to 3AII and 3CII.

A.3. EVALUATING THE SPACE ARRANGEMENT AND PASSAGEWAY CARVING METHODS

This section comprises the supplementary material to Section 3.1.

The initial case in the notional surface vessel case study presented in Section 3.1 is one of three tests used to investigate the combination of space arrangement approaches and the passageway carving approach. Indeed, in Section 2.4.7 the need for a test with the integrated method was expressed. In Appendix A.2, three arrangement variations were found to be performing well (3AII, 3CII, and 3EII). The three tests elaborated on here are respectively numbered 1a-1c. Based on the results of these three tests the final arrangement variation will be chosen. For details on the case study setup, see Section 3.1.

The results of the three tests are summarised in Table A.6. The results show that arrangement variation 3CII and 3EII (tests 1b and 1c) outperform arrangement variation 3AII (test 1a). This can be observed as follows:

1. The final obtained objective score F is lower for tests 1b and 1c than for test 1a. This means that the difference between required and achieved area in test 1a is larger than in tests 1b and 1c.

2. The number of non-allocated spaces in tests 1b and 1c is lower than in test 1a. This is not directly a result of the arrangement variation, since the allocation is steered

by the outer optimisation loop (PSO1). However, the behaviour of the arrangement variation does influence the behaviour of PSO1.

3. The objective score F before local connectivity is higher for test 1a than for the final iterations of tests 1b and 1c. This is both caused by a less efficient arrangement of spaces and the non-allocated spaces.

Similar observations can be used to show that test 1c outperforms test 1b. Therefore arrangement variation 3EII will be used in the case study as described in Section 3.1, and is implemented in the overall WARGEAR method.

Table A.6: Summary of results of the three initial cases

	A	B			
Test number	**Mimimum F obtained [m^2]**	**Minimum number of spaces not-allocated**	**#B**	**A|B [m2]**	**Run time [s]**
1a	19.68	0	2	27.72	832.59
1b	15.84	0	175	15.84	764.17
1c	12.60	0	163	12.60	737.05

B

SUPPLEMENTARY MATERIALS FOR WARGEAR CASE STUDIES

B.1. INPUT TO ALLOCATION TESTS

This section contains the input for the allocation tests presented in Appendix A.1.

Table B.1: Compartment area for allocation test case 1 and 2.

Test case 1	Compartment	1	2	3	4	5	6	7	8	9
	Area [m^2]	288.0	288.0	287.3	288.0	251.3	288.0	288.0	288.0	288.0
Test case 2	Compartment	1	2	3	4	5	6	7		
	Area [m^2]	135.9	153.0	100.6	92.6	163.6	128.2	125.4		
	Compartment	8	9	10	11	12	13	14		
	Area [m^2]	106.7	90.7	129.4	176.4	94.8	119.1	164.9		

Table B.2: Space area and allowed allocation of 100 spaces to nine compartments for allocation test 1.

Space number	Area [m^2]	Compartments									Space number	Area [m^2]	Compartments								
		1	2	3	4	5	6	7	8	9			1	2	3	4	5	6	7	8	9
1	115						1				51	20	1	1	1				1	1	
2	125						1				52	20	1	1	1				1	1	
3	130					1					53	20	1	1	1				1	1	
4	50					1					54	20	1	1	1				1	1	
5	50				1						55	20	1	1	1				1	1	
6	50				1						56	20	1	1	1				1	1	
7	40				1						57	20	1	1	1				1	1	
8	30				1						58	20	1	1	1				1	1	
9	30				1						59	20	1	1	1				1	1	
10	25				1						60	20	1	1	1				1	1	
11	20									1	61	20	1	1	1				1	1	
12	12									1	62	20	1	1	1				1	1	
13	12									1	63	20	1	1	1				1	1	
14	12									1	64	20	1	1	1				1	1	
15	12									1	65	20	1	1	1				1	1	
16	12									1	66	20	1	1	1				1	1	
17	12									1	67	20	1	1	1				1	1	
18	12									1	68	20	1	1	1				1	1	
19	12									1	69	20	1	1	1				1	1	
20	10							1	1	1	70	20	1	1	1				1	1	
21	10							1	1	1	71	20	1	1	1				1	1	
22	10							1	1	1	72	20	1	1	1				1	1	
23	10							1	1	1	73	20	1	1	1				1	1	
24	10							1	1	1	74	20	1	1	1				1	1	
25	10							1	1	1	75	20	1	1	1				1	1	
26	10							1	1	1	76	20	1	1	1				1	1	
27	10							1	1	1	77	20	1	1	1				1	1	
28	10							1	1	1	78	20	1	1	1				1	1	
29	10							1	1	1	79	20	1	1	1				1	1	
30	10							1	1	1	80	20	1	1	1				1	1	
31	15							1	1	1	81	20	1	1	1				1	1	
32	15							1	1	1	82	20	1	1	1				1	1	
33	15							1	1	1	83	20	1	1	1				1	1	
34	15							1	1	1	84	20	1	1	1				1	1	
35	15							1	1	1	85	20	1	1	1				1	1	
36	15							1	1	1	86	20	1	1	1				1	1	
37	15							1	1	1	87	20	1	1	1				1	1	
38	15							1	1		88	20	1	1	1				1	1	
39	15							1	1		89	20	1	1	1				1	1	
40	15							1	1		90	20	1	1	1				1	1	
41	15							1	1		91	20	1	1	1				1	1	
42	15							1	1		92	20	1	1	1				1	1	
43	15							1	1		93	20	1	1	1				1	1	
44	15							1	1		94	20	1	1	1				1	1	
45	15							1	1		95	20	1	1	1				1	1	
46	20	1	1	1				1	1		96	20	1	1	1				1	1	
47	20	1	1	1				1	1		97	20	1	1	1				1	1	
48	20	1	1	1				1	1		98	20	1	1	1				1	1	
49	20	1	1	1				1	1		99	20	1	1	1				1	1	
50	20	1	1	1				1	1		100	20	1	1	1				1	1	

Table B.3: Space area and allowed allocation of 75 spaces to fourteen compartments for allocation test 2.

Space nr.	Area [m^2]	Compartments														Space nr.	Area [m^2]	Compartments													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14			1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	10		1	1												39	15				1		1	1	1		1	1	1		
2	35		1	1												40	15				1		1	1	1		1	1	1		
3	20		1	1												41	15				1		1	1	1		1	1	1		
4	10													1	1	42	15				1		1	1	1		1	1	1		
5	10													1	1	43	15				1		1	1	1		1	1	1		
6	10										1	1	1	1	1	44	15				1		1	1	1		1	1	1		
7	10										1	1	1	1	1	45	20	1		1	1		1	1	1						
8	10										1	1	1	1	1	46	20	1		1	1		1	1	1						
9	10										1	1	1	1	1	47	20	1		1	1		1	1	1						
10	10										1	1	1	1	1	48	20	1		1	1		1	1	1						
11	10										1	1	1	1	1	49	20	1		1	1		1	1	1						
12	10										1	1	1	1	1	50	20	1		1	1		1	1	1						
13	13										1	1	1	1	1	51	20	1		1	1		1	1	1						
14	13										1	1	1	1	1	52	20	1		1	1		1	1	1						
15	13										1	1	1	1	1	53	20	1		1	1		1	1	1						
16	13										1	1	1	1	1	54	20	1		1	1		1	1	1						
17	13										1	1	1	1	1	55	20	1		1	1		1	1	1						
18	13										1	1	1	1	1	56	20	1		1	1		1	1	1						
19	13										1	1	1	1	1	57	20	1		1	1		1	1	1						
20	13										1	1	1	1	1	58	20	1		1	1		1	1	1						
21	13										1	1	1	1	1	59	20	1		1	1		1	1	1						
22	13				1		1	1	1		1	1	1			60	20	1		1	1		1	1	1						
23	13				1		1	1	1		1	1	1			61	20	1		1	1		1	1	1						
24	15				1		1	1	1		1	1	1			62	20	1		1	1		1	1	1						
25	15				1		1	1	1		1	1	1			63	20	1		1	1		1	1	1						
26	15				1		1	1	1		1	1	1			64	20	1		1	1		1	1	1						
27	15				1		1	1	1		1	1	1			65	20	1		1	1		1	1	1						
28	15				1		1	1	1		1	1	1			66	20	1		1	1		1	1	1						
29	15				1		1	1	1		1	1	1			67	20	1		1	1		1	1	1						
30	15				1		1	1	1		1	1	1			68	20	1		1	1		1	1	1						
31	15				1		1	1	1		1	1	1			69	20	1		1	1		1	1	1						
32	15				1		1	1	1		1	1	1			70	20	1		1	1		1	1	1						
33	15				1		1	1	1		1	1	1			71	20	1		1	1		1	1	1						
34	15				1		1	1	1		1	1	1			72	20	1		1	1		1	1	1						
35	15				1		1	1	1		1	1	1			73	50									1					
36	15				1		1	1	1		1	1	1			74	100					1									
37	15				1		1	1	1		1	1	1			75	50		1												
38	15				1		1	1	1		1	1	1																		

B.2. INPUT TO CASE STUDY 1

This Appendix contains the input for Case Study 1 presented in Section 3.1.

Table B.4: Space list and space characteristics for Case Study 1

ID	Name	Area m^2	AR low	AR high	FBB name	FBB numbers
1	Store 1	15	0.5	1	Operational rooms and offices	4
2	Store 2	15	0.5	1	Operational rooms and offices	4
3	Store 3	20	0.5	1	Operational rooms and offices	4
4	Mess	35	0.5	1	Accommodation cabins	15
5	Galley	15	0.5	1	Accommodation cabins	15
6	Cabin 1	10	0.5	1	Accommodation cabins	13
7	Cabin 1	10	0.5	1	Accommodation cabins	13
8	Cabin 1	10	0.5	1	Accommodation cabins	13
9	Cabin 1	10	0.5	1	Accommodation cabins	13
10	Cabin 1	10	0.5	1	Accommodation cabins	13
11	Cabin 1	10	0.5	1	Accommodation cabins	13
12	Cabin 1	10	0.5	1	Accommodation cabins	13
13	Cabin 1	10	0.5	1	Accommodation cabins	13
14	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
15	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
16	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
17	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
18	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
19	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
20	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
21	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
22	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
23	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
24	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
25	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
26	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
27	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
28	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
29	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
30	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
31	Cabin 2	15	0.5	1	Accommodation cabins	14 16 17 18 19 20
	Total:	**450**				

Table B.5: List with functional building blocks and available area in compartments. [1] YES: these FBBs are not available for space arrangement for Case Study 1

FBB number	FBB name	Area m^2	Blocked[1]	Deck	Compartment	Available area m^2
1	Passage ways and staircases	58.5	Yes	1	3	97.92
2	Passage ways and staircases	24	Yes	2	2	51.84
3	Helicopter hangar	120	Yes	2	3	110.16
4	Operational rooms and offices	54.1	No	2	4	110.16
5	Operational rooms and offices	54.1	Yes	3	3	55.08
6	Operational rooms and offices	54.1	Yes	3	4	110.16
7	Void	54.1	Yes		**Total:**	**535.32**
8	Void	54.1	Yes			
9	Void	95.9	Yes			
10	Void	30.3	Yes			
11	Void	24.9	Yes			
12	Void	24.9	Yes			
13	Accommodation cabins	100.1	No			
14	Accommodation cabins	54.1	No			
15	Accommodation cabins	54.1	No			
16	Accommodation cabins	52.8	No			
17	Accommodation cabins	52.8	No			
18	Accommodation cabins	54.1	No			
19	Accommodation cabins	53.6	No			
20	Accommodation cabins	53.6	No			
21	Propulsion room	93.3	Yes			
22	Generator room	100.1	Yes			

B

B.3. INPUT TO CASE STUDY 3

This Appendix contains the input for Case Study 3 presented in Section 3.3.

Table B.6: Case Study 3a: Compartments

ID	Area m^2	Global Position		X1 [m]	Y1 [m]	Z1 [m]	X2 [m]	Y2 [m]	Z2 [m]
1	38.4	Aft	Bottom	0	0	0	8	0	3
2	48	Forward	Bottom	8	0	0	8	0	3
3	51.2	Aft	Top	0	0	3	8	0	3
4	64	Forward	Top	8	0	3	8	0	3

Table B.7: Case Study 3a: Systems and System Properties

ID	Name	Area m^2	Global Position
1	A	30	Top
2	B	30	
3	C	20	
4	D	20	Bottom
5	E	30	
6	F	30	Forward
7	G	10	
8	H	15	

Table B.8: Case Study 3a: Interactions

System A	System B	Interaction type
1	2	-1
1	5	-1
2	1	1
2	6	1
4	3	-3
4	6	2
5	2	-2
6	5	-1

Table B.9: Case Study 3a: Interaction Types

ID	Description	Value
1	In same compartment	
-1	In different compartments	
2	Maximum Manhattan distance [m]	10
-2	Minimum Manhattan distance [m]	5
-3	Minimum radial distance (compartments)	1

Table B.10: Case Study 3b: Interaction Types

ID	Description	Value
1	In same compartment	
-1	In different compartments	
2	Maximum Manhattan distance [m]	35
-2	Minimum Manhattan distance [m]	5
-3	Minimum radial distance (compartments)	1

Table B.11: Case Study 3b: Compartments

ID	Area $[m^2]$	Global Position		X1 [m]	Y1 [m]	Z1 [m]	X2 [m]	Y2 [m]	Z2 [m]	ID	Area $[m^2]$	Global Position		X1 [m]	Y1 [m]	Z1 [m]	X2 [m]	Y2 [m]	Z2 [m]
1	30	Aft	Bottom	0	0	0	12	0	3	19	96	Center	Mid	48	0	6	12	0	3
2	48	Aft	Bottom	12	0	0	12	0	3	20	96	Forward	Mid	60	0	6	12	0	3
3	48	Center	Bottom	24	0	0	12	0	3	21	60	Forward	Mid	72	0	6	12	0	3
4	48	Center	Bottom	36	0	0	12	0	3	22	60	Aft	Top	0	0	9	12	0	3
5	48	Center	Bottom	48	0	0	12	0	3	23	96	Aft	Top	12	0	9	12	0	3
6	48	Forward	Bottom	60	0	0	12	0	3	24	96	Center	Top	24	0	9	12	0	3
7	30	Forward	Bottom	72	0	0	12	0	3	25	96	Center	Top	36	0	9	12	0	3
8	45	Aft	Mid	0	0	3	12	0	3	26	96	Center	Top	48	0	9	12	0	3
9	72	Aft	Mid	12	0	3	12	0	3	27	96	Forward	Top	60	0	9	12	0	3
10	72	Center	Mid	24	0	3	12	0	3	28	60	Forward	Top	72	0	9	12	0	3
11	72	Center	Mid	36	0	3	12	0	3	29	60	Aft	Top	0	0	12	12	0	3
12	72	Center	Mid	48	0	3	12	0	3	30	96	Aft	Top	12	0	12	12	0	3
13	72	Forward	Mid	60	0	3	12	0	3	31	96	Center	Top	24	0	12	12	0	3
14	45	Forward	Mid	72	0	3	12	0	3	32	96	Center	Top	36	0	12	12	0	3
15	60	Aft	Mid	0	0	6	12	0	3	33	96	Center	Top	48	0	12	12	0	3
16	96	Aft	Mid	12	0	6	12	0	3	34	96	Forward	Top	60	0	12	12	0	3
17	96	Center	Mid	24	0	6	12	0	3	35	60	Forward	Top	72	0	12	12	0	3
18	96	Center	Mid	36	0	6	12	0	3										

Table B.12: Case Study 3b: Systems and System Properties. * & = both items should be satisfied, and | = one of the items needs to be satisfied

ID	Name	Area m^2	Global Position*	ID	Name	Area m^2	Global Position*
1	Commanders Cabin	55	Top & Forward	46	Rating cabins (4 persons per cabin)	12.312	Mid | Bottom
2	Officers cabins (1 person per cabin)	15.39	Top	47	Rating cabins (4 persons per cabin)	12.312	Mid | Bottom
3	Officers cabins (1 person per cabin)	15.39	Top	48	Sanitary & showers (1 cabin per 12 rating)	12.312	Top
4	Officers cabins (1 person per cabin)	15.39	Top	49	Sanitary & showers (1 cabin per 12 rating)	12.312	Mid
5	Officers cabins (1 person per cabin)	15.39	Top	50	Sanitary & showers (1 cabin per 12 rating)	12.312	Bottom
6	Officers cabins (2 person per cabin)	13.851	Top	51	Baggage storage	14.4	
7	Officers cabins (2 person per cabin)	13.851	Top	52	Dayroom officers	12.55	Top
8	Officers cabins (2 person per cabin)	13.851	Top	53	Dayroom petty officers	57.7	Mid
9	Officers cabins (2 person per cabin)	13.851	Top	54	Dayroom rating	20.4	Bottom
10	Officers cabins (2 person per cabin)	13.851	Top	55	Laundry	29.2	
11	Officers cabins (2 person per cabin)	13.851	Top	56	Fitness room	35	
12	Petty officers cabins (1 person per cabin)	15.39	Mid	57	Meeting room	48.4	Top
13	Petty officers cabins (1 person per cabin)	15.39	Mid	58	Bakery	8.64	Bottom
14	Petty officers cabins (1 person per cabin)	15.39	Mid	59	Galley, scullery & servery	55.72	Bottom
15	Petty officers cabins (1 person per cabin)	15.39	Mid	60	Mess	67.2	Mid
16	Petty officers cabins (2 persons per cabin)	13.851	Mid	61	Medical area	39.68	
17	Petty officers cabins (2 persons per cabin)	13.851	Mid	62	Dry stores	72	
18	Petty officers cabins (2 persons per cabin)	13.851	Mid	63	Bridge	60	Top & Forward
19	Petty officers cabins (2 persons per cabin)	13.851	Mid	64	Command central	96	
20	Petty officers cabins (2 persons per cabin)	13.851	Mid	65	Radio central	30	
21	Petty officers cabins (2 persons per cabin)	13.851	Mid	66	Intel room	12	
22	Petty officers cabins (2 persons per cabin)	13.851	Mid	67	Engineering office	48	Aft
23	Petty officers cabins (2 persons per cabin)	13.851	Mid	68	Briefing room	24	Top
24	Petty officers cabins (2 persons per cabin)	13.851	Mid	69	Compass room	12	Forward
25	Petty officers cabins (2 persons per cabin)	13.851	Mid	70	Compass room	12	Aft
26	Petty officers cabins (2 persons per cabin)	13.851	Mid	71	Computer room	66	Forward
27	Petty officers cabins (2 persons per cabin)	13.851	Mid	72	Computer room	66	Aft
28	Petty officers cabins (2 persons per cabin)	13.851	Mid	73	Workshop electrical	20	
29	Petty officers cabins (2 persons per cabin)	13.851	Mid	74	Workshop mechanical and welding	40	
30	Petty officers cabins (2 persons per cabin)	13.851	Mid	75	HVAC rooms (1 per zone)	22	Forward
31	Petty officers cabins (2 persons per cabin)	13.851	Mid	76	HVAC rooms (1 per zone)	22	Aft
32	Petty officers cabins (2 persons per cabin)	13.851	Mid	77	NBCD filter rooms (1 per zone)	5	Forward
33	Petty officers cabins (2 persons per cabin)	13.851	Mid	78	NBCD filter rooms (1 per zone)	5	Aft
34	Petty officers cabins (2 persons per cabin)	13.851	Mid	79	Fire fighting room	5	
35	Petty officers cabins (2 persons per cabin)	13.851	Mid	80	Chiller unit forward	20	Forward
36	Petty officers cabins (2 persons per cabin)	13.851	Mid	81	Chiller unit aft	20	Aft
37	Petty officers cabins (2 persons per cabin)	13.851	Mid	82	Fresh water maker 1	10	Forward
38	Petty officers cabins (2 persons per cabin)	13.851	Mid	83	Fresh water maker 2	10	Aft
39	Petty officers cabins (2 persons per cabin)	13.851	Mid	84	Hydrophore (fresh water pump equipment)	10	
40	Petty officers cabins (2 persons per cabin)	13.851	Mid	85	Main switchboard	30	Forward
41	Rating cabins (4 persons per cabin)	12.312	Mid | Bottom	86	Main switchboard	30	Aft
42	Rating cabins (4 persons per cabin)	12.312	Mid | Bottom	87	Emergency switchboard	8	
43	Rating cabins (4 persons per cabin)	12.312	Mid | Bottom	88	Waste store	20.4	
44	Rating cabins (4 persons per cabin)	12.312	Mid | Bottom	89	Engine room	48	Center & Bottom
45	Rating cabins (4 persons per cabin)	12.312	Mid | Bottom				

Table B.13: Case Study 3b: Interactions

A	B	Int. type	A	B	Int. type	A	B	Int. type	A	B	Int. type	A	B	Int. type	A	B	Int. type	A	B	Int. type
1	2	2	13	20	2	16	35	2	20	39	2	26	30	2	35	39	2	89	37	-3
1	3	2	13	21	2	16	36	2	20	40	2	26	31	2	35	40	2	89	38	-3
1	4	2	13	22	2	16	37	2	21	22	2	26	32	2	36	37	2	89	39	-3
1	5	2	13	23	2	16	38	2	21	23	2	26	33	2	36	38	2	89	40	-3
1	6	2	13	24	2	16	39	2	21	24	2	26	34	2	36	39	2	89	41	-3
1	7	2	13	25	2	16	40	2	21	25	2	26	35	2	36	40	2	89	42	-3
1	8	2	13	26	2	17	18	2	21	26	2	26	36	2	37	38	2	89	43	-3
1	9	2	13	27	2	17	19	2	21	27	2	26	37	2	37	39	2	89	44	-3
1	10	2	13	28	2	17	20	2	21	28	2	26	38	2	37	40	2	89	45	-3
1	11	2	13	29	2	17	21	2	21	29	2	26	39	2	38	39	2	89	46	-3
1	60	2	13	30	2	17	22	2	21	30	2	26	40	2	38	40	2	89	47	-3
2	3	2	13	31	2	17	23	2	21	31	2	27	28	2	39	40	2	89	48	-3
2	4	2	13	32	2	17	24	2	21	32	2	27	29	2	41	42	2	89	49	-3
2	5	2	13	33	2	17	25	2	21	33	2	27	30	2	41	43	2	89	50	-3
2	6	2	13	34	2	17	26	2	21	34	2	27	31	2	41	44	2	89	52	-3
2	7	2	13	35	2	17	27	2	21	35	2	27	32	2	41	45	2	89	53	-3
2	8	2	13	36	2	17	28	2	21	36	2	27	33	2	41	46	2	89	54	-3
2	9	2	13	37	2	17	29	2	21	37	2	27	34	2	41	47	2	89	56	-3
2	10	2	13	38	2	17	30	2	21	38	2	27	35	2	42	43	2	89	57	-3
2	11	2	13	39	2	17	31	2	21	39	2	27	36	2	42	44	2	89	58	-3
3	4	2	13	40	2	17	32	2	21	40	2	27	37	2	42	45	2	89	59	-3
3	5	2	14	15	2	17	33	2	22	23	2	27	38	2	42	46	2	89	60	-3
3	6	2	14	16	2	17	34	2	22	24	2	27	39	2	42	47	2	89	61	-3
3	7	2	14	17	2	17	35	2	22	25	2	27	40	2	43	44	2	89	63	-3
3	8	2	14	18	2	17	36	2	22	26	2	28	29	2	43	45	2	89	64	-3
3	9	2	14	19	2	17	37	2	22	27	2	28	30	2	43	46	2	89	65	-3
3	10	2	14	20	2	17	38	2	22	28	2	28	31	2	43	47	2	89	66	-3
3	11	2	14	21	2	17	39	2	22	29	2	28	32	2	44	45	2	89	67	-3
4	5	2	14	22	2	17	40	2	22	30	2	28	33	2	44	46	2	89	68	-3
4	6	2	14	23	2	18	19	2	22	31	2	28	34	2	44	47	2	89	69	-3
4	7	2	14	24	2	18	20	2	22	32	2	28	35	2	45	46	2	89	70	-3
4	8	2	14	25	2	18	21	2	22	33	2	28	36	2	45	47	2	89	71	-3
4	9	2	14	26	2	18	22	2	22	34	2	28	37	2	46	47	2	89	72	-3
4	10	2	14	27	2	18	23	2	22	35	2	28	38	2	47	2	2	89	73	-3
4	11	2	14	28	2	18	24	2	22	36	2	28	39	2	48	13	2	89	74	-3
5	6	2	14	29	2	18	25	2	22	37	2	28	40	2	49	42	2			
5	7	2	14	30	2	18	26	2	22	38	2	29	30	2	51	2	2			
5	8	2	14	31	2	18	27	2	22	39	2	29	31	2	52	13	2			
5	9	2	14	32	2	18	28	2	22	40	2	29	32	2	52	60	2			
5	10	2	14	33	2	18	29	2	23	24	2	29	33	2	53	42	2			
5	11	2	14	34	2	18	30	2	23	25	2	29	34	2	53	60	2			
6	7	2	14	35	2	18	31	2	23	26	2	29	35	2	54	60	2			
6	8	2	14	36	2	18	32	2	23	27	2	29	36	2	58	60	2			
6	9	2	14	37	2	18	33	2	23	28	2	29	37	2	58	62	2			
6	10	2	14	38	2	18	34	2	23	29	2	29	38	2	59	60	2			
6	11	2	14	39	2	18	35	2	23	30	2	29	39	2	59	62	2			
7	8	2	14	40	2	18	36	2	23	31	2	29	40	2	63	1	1			
7	9	2	15	16	2	18	37	2	23	32	2	30	31	2	75	76	-1			
7	10	2	15	17	2	18	38	2	23	33	2	30	32	2	77	78	-1			
7	11	2	15	18	2	18	39	2	23	34	2	30	33	2	80	81	-1			
8	9	2	15	19	2	18	40	2	23	35	2	30	34	2	85	86	-1			
8	10	2	15	20	2	19	20	2	23	36	2	30	35	2	88	58	1			
8	11	2	15	21	2	19	21	2	23	37	2	30	36	2	88	59	1			
9	10	2	15	22	2	19	22	2	23	38	2	30	37	2	88	60	1			
9	11	2	15	23	2	19	23	2	23	39	2	30	38	2	89	1	-3			
10	11	2	15	24	2	19	24	2	23	40	2	30	39	2	89	2	-3			
12	13	2	15	25	2	19	25	2	24	25	2	30	40	2	89	3	-3			
12	14	2	15	26	2	19	26	2	24	26	2	31	32	2	89	4	-3			
12	15	2	15	27	2	19	27	2	24	27	2	31	33	2	89	5	-3			
12	16	2	15	28	2	19	28	2	24	28	2	31	34	2	89	6	-3			
12	17	2	15	29	2	19	29	2	24	29	2	31	35	2	89	7	-3			
12	18	2	15	30	2	19	30	2	24	30	2	31	36	2	89	8	-3			
12	19	2	15	31	2	19	31	2	24	31	2	31	37	2	89	9	-3			
12	20	2	15	32	2	19	32	2	24	32	2	31	38	2	89	10	-3			
12	21	2	15	33	2	19	33	2	24	33	2	31	39	2	89	11	-3			
12	22	2	15	34	2	19	34	2	24	34	2	31	40	2	89	12	-3			
12	23	2	15	35	2	19	35	2	24	35	2	32	33	2	89	13	-3			
12	24	2	15	36	2	19	36	2	24	36	2	32	34	2	89	14	-3			
12	25	2	15	37	2	19	37	2	24	37	2	32	35	2	89	15	-3			
12	26	2	15	38	2	19	38	2	24	38	2	32	36	2	89	16	-3			
12	27	2	15	39	2	19	39	2	24	39	2	32	37	2	89	17	-3			
12	28	2	15	40	2	19	40	2	24	40	2	32	38	2	89	18	-3			
12	29	2	16	17	2	20	21	2	25	26	2	32	39	2	89	19	-3			
12	30	2	16	18	2	20	22	2	25	27	2	32	40	2	89	20	-3			
12	31	2	16	19	2	20	23	2	25	28	2	33	34	2	89	21	-3			
12	32	2	16	20	2	20	24	2	25	29	2	33	35	2	89	22	-3			
12	33	2	16	21	2	20	25	2	25	30	2	33	36	2	89	23	-3			
12	34	2	16	22	2	20	26	2	25	31	2	33	37	2	89	24	-3			
12	35	2	16	23	2	20	27	2	25	32	2	33	38	2	89	25	-3			
12	36	2	16	24	2	20	28	2	25	33	2	33	39	2	89	26	-3			
12	37	2	16	25	2	20	29	2	25	34	2	33	40	2	89	27	-3			
12	38	2	16	26	2	20	30	2	25	35	2	34	35	2	89	28	-3			
12	39	2	16	27	2	20	31	2	25	36	2	34	36	2	89	29	-3			
12	40	2	16	28	2	20	32	2	25	37	2	34	37	2	89	30	-3			
13	14	2	16	29	2	20	33	2	25	38	2	34	38	2	89	31	-3			
13	15	2	16	30	2	20	34	2	25	39	2	34	39	2	89	32	-3			
13	16	2	16	31	2	20	35	2	25	40	2	34	40	2	89	33	-3			
13	17	2	16	32	2	20	36	2	26	27	2	35	36	2	89	34	-3			
13	18	2	16	33	2	20	37	2	26	28	2	35	37	2	89	35	-3			
13	19	2	16	34	2	20	38	2	26	29	2	35	38	2	89	36	-3			

C

Supplementary materials for Ship Design Rationale Method

This appendix contains supplementary materials for Part II.

C

C.1. QUESTIONNAIRE PROOF OF CONCEPT

Questionnaire

My role: ☐ Naval architect - ☐ Logistics specialist - ☐ Safety specialist

My group: ☐ A - ☐ B - ☐ C - ☐ D

Closed Questions	**Strongly disagree**	**Disagree**	**Neutral**	**Agree**	**Strongly agree**
<u>Satisfaction</u>					
I'm satisfied with the layout resulting from the **first round** from my role's perspective	☐	☐	☐	☐	☐
I'm satisfied with my input in the decision-making in the **first round**	☐	☐	☐	☐	☐
My input in the decision-making has been satisfactory incorporated in the final design in the **first round**	☐	☐	☐	☐	☐
I'm satisfied with the layout resulting from the **second round** from my role's perspective	☐	☐	☐	☐	☐
I'm satisfied with my input in the decision-making in the **second round**	☐	☐	☐	☐	☐
My input in the decision-making has been satisfactory incorporated in the final design in the **second round**	☐	☐	☐	☐	☐
<u>Experiment Setup</u>					
The time available for completing the **first round** was sufficient	☐	☐	☐	☐	☐
The time available for completing the **second round** was sufficient	☐	☐	☐	☐	☐
The first layout was less difficult than the second layout.	☐	☐	☐	☐	☐
The familiarization round was helpful	☐	☐	☐	☐	☐
<u>Design rationale methodology (DR methodology)</u>					
The DR methodology was easy to learn	☐	☐	☐	☐	☐
The gains outweigh the (temporal) costs of using the DR methodology	☐	☐	☐	☐	☐
The DR methodology supported **me** to get a better overview of relevant design considerations.	☐	☐	☐	☐	☐
The DR methodology supported **the team** to get a better overview of relevant design considerations.	☐	☐	☐	☐	☐
The DR methodology distracted me from considering my role's perspective and requirements.	☐	☐	☐	☐	☐
The DR methodology helped me in understanding the perspectives of the other team members.	☐	☐	☐	☐	☐
The DR methodology supported the decision-making process by the team.	☐	☐	☐	☐	☐

Please turn over for other questions

Open questions

1. Which functionalities of the design rationale methodology were most useful to the design case?

2. What additional functionalities of the design rationale methodology would be beneficial?

3. Where you able to capture all rationale that you wanted to capture? If not, which rationale weren't you able to capture?

4. Please describe how the design rationale methodology supported decision-making, compared to the baseline methodology?

5. Are there any bugs or problems you want to report?

6. How would you describe you team's negotiation process?

7. Would the design rationale methodology be beneficial for design (review) sessions, and why?

8. What method do you use in practice to record your design rationale?

9. What take-aways do you have from using the design rationale methodology?

Thank you for your participation!

Please turn over for other questions

C.2. INPUT TO CASE STUDY 6

This appendix comprises the list of systems in the OPV case study presented in Section 5.2.

Table C.1: List of systems in Case Study 6

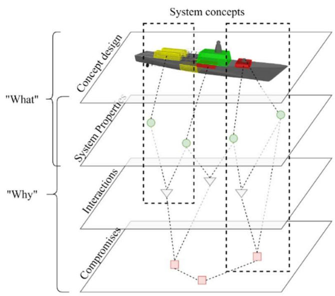
ID	Name	Area or volume	Type
1	Flight deck	area	Helicopter Deck
2	Hangar	volume	Helicopter Deck
3	Flight deck office	volume	Helicopter Deck
4	Foam room	volume	Workshop
5	Workshop helicopter	volume	Workshop
6	Heli spare parts store	volume	Store Dry Store
7	Heli ammo store	volume	Store Ammunition
8	F44 pump room	volume	Auxiliary
9	Accommodation cabin officers	volume	Accommodation
10	Accommodation cabin petty officers	volume	Accommodation
11	Accommodation cabin ratings	volume	Accommodation
12	Dayroom officers	volume	Accommodation
13	Dayroom petty officers	volume	Accommodation
14	Dayroom ratings	volume	Accommodation
15	Mess	volume	Accommodation
16	Galley, scullery, servery, bakery	volume	Accommodation
17	Meeting room	volume	Accommodation
18	Fitness room	volume	Accommodation
19	Laundry	volume	Accommodation
20	Shop	volume	Accommodation
21	Waste handling	volume	Store Waste
22	Baggage storage	volume	Accommodation
23	Medical area	volume	Medical
24	Bridge	volume	C2
25	Command central	volume	C2
26	Radio central	volume	C2
27	Intel room	volume	C2
28	Engineering office	volume	C2
29	Briefing room	volume	C2
30	Offices	volume	C2
31	Compass room	volume	C2
32	Compass room	volume	C2
33	Computer room	volume	C2
34	Computer room	volume	C2
35	Workshop electrical	volume	Workshop
36	Workship divers	volume	Workshop
37	Workshop mechanical and welding	volume	Workshop
38	Dry stores	volume	Store Dry Store
39	Rope and buson store forward	volume	Store Dry Store
40	Rope and buson store aft	volume	Store Dry Store
41	RAS stores	volume	Store Dry Store
42	Hazardous materials stores	volume	Store Ammunition
43	Spare stores	volume	Store Dry Store
44	F44 Helicopter fuel	volume	Tank F44
45	F76 fuel (MDO)	volume	Tank F76
46	Lube oil	volume	Tank Lube Oil
47	Fresh water	volume	Tank Fresh Water
48	Bilge water	volume	Tank Bilge Water
49	Grey black	volume	Tank Grey Black
50	Ballast	volume	Tank Ballast Water
51	ER fwd	volume	Propulsion Room
52	ER aft	volume	Propulsion Room
53	Aux ER fwd	volume	Propulsion Room
54	Genset	area	Void
55	Genset	area	Void
56	Genset	area	Void
57	Genset	area	Void
58	MAN 20V28/33D	area	Void
59	MAN 20V28/33D	area	Void
60	MTU 12V 2000 M94	area	Void
61	MTU 12V 2000 M94	area	Void
62	1 x E Genset	area	Void
63	Gearbox	area	Void
64	Gearbox	area	Void
65	HVAC	volume	Auxiliary
66	HVAC	volume	Auxiliary
67	NBCD	volume	Auxiliary
68	NBCD	volume	Auxiliary
69	Fire fighting room	volume	Auxiliary
70	Chiller unit forward	volume	Auxiliary
71	Chiller unit aft	volume	Auxiliary
72	Fresh water maker 1	volume	Auxiliary
73	Fresh water maker 2	volume	Auxiliary
74	Hydrophore (fresh water pump equipment)	volume	Auxiliary
75	Main switchboard forward	volume	Auxiliary
76	Main switchboard aft	volume	Auxiliary
77	Emergency switchboard	volume	Auxiliary
78	Steering machinery (portside & starboard)	volume	Auxiliary
79	Manifold	volume	Auxiliary
80	Boilers	volume	Auxiliary
81	Separator	volume	Auxiliary
82	Air compressor	volume	Auxiliary
83	Sewage treatment	volume	Auxiliary
84	Watermist installation	volume	Auxiliary
85	Other pumps (ballast/lubrication oil/bilge/ grey and black water)	volume	Auxiliary
86	RAS mast	volume	Workshop
87	RHIB_9m_davits_SB	area	Secondary Object
88	S-band navigation radar	area	Void
89	X-band navigation radar	area	Void
90	sensor_mirador	area	Secondary Object
91	Vigili 100 RESM	area	Void
92	CIWS	area	Void
93	gun_76mm	area	Secondary Object
94	ROMG .50	area	Void
95	ROMG .50	area	Void
96	CIWS ammo store	volume	Store Ammunition
97	Oto Melara 76mm ammo store	volume	Store Ammunition
98	ROMG .50 ammo store	volume	Store Ammunition
99	ROMG .50 ammo store	volume	Store Ammunition
100	Exhaust Fwd	volume	Void
101	Exhaust Aft	volume	Void
102	Food elevator	volume	Acces
103	Mooring deck fwd	volume	Void
104	Mooring deck aft	volume	Void
105	Elevator	volume	Acces
106	radar_imast	area	Secondary Object

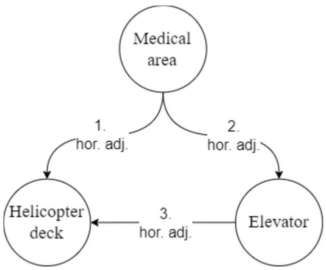
C.3. SUMMARISING SHEET SDRM

Ship Design Rationale Documentation

Design rationale

Design rationale "[encompasses] the **documentation** of the **active processes of reasoning and decision-making** that led to the **artefact design** — including the **justification** for design decisions, records of **design alternatives** considered and **trade-offs** evaluated, and details of the **argumentation** and **communication processes** associated with design work." (Ball et al., 2001)





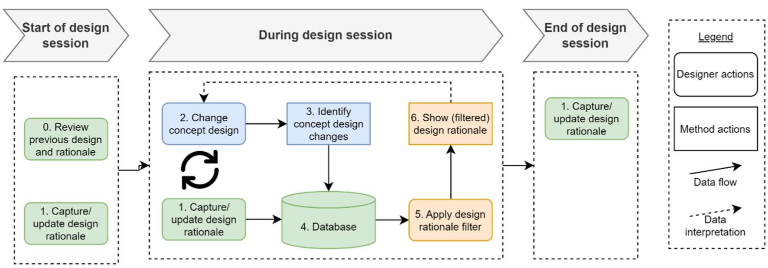
System Concept: contains a (sub)set of the layout (or it's systems) and the design rationale behind this (sub)set + rationale behind the concept and can be related to other design rationale, e.g. a compromise between two propulsion concepts.

System Property: a (required or actual) quality or characteristic of a system and its justification. Examples of System Properties are required sizing (e.g. volume, area, aspect ratio, alternative positions) and preferred global positions (e.g. on deck 3, as high as possible).

Interaction: a preferred (spatial) relationship between two (or more) systems (DeNucci, 2012), or System Properties. Additionally, an interaction comprises its justification. An example of an interaction is: the ammunition store should be adjacent to the gun [relation], to reduce dangerous transport of ammunition through the ship [justification].

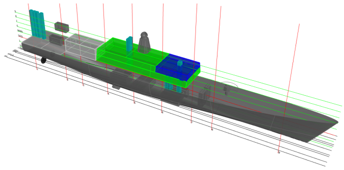
Compromise: is the preferred solution to a set of conflicting or competing interactions or System Properties and its justification. For example, consider the required connectivity between a helicopter deck and the medical room in a frigate, depicted above. This situation comprises three interactions (1.-3.). The related compromise is the preferred set of interactions, where the naval architect is to choose between set A = {interaction 1.} and set B = {interactions 2. and 3.}. Choosing set A might be justified by reasons as 'reduced spatial impact' and 'less time to medical room'.

Ship Design Rationale Process

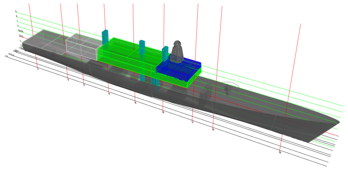


C.4. VISUALISATION OF DEVELOPED CONCEPT DESIGNS

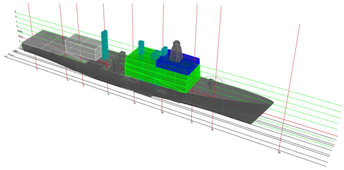
This section contains visualisations of all captured design instances by Team A and B.



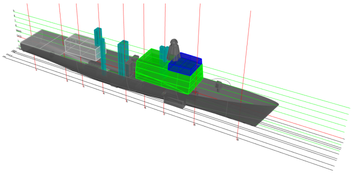
(a) Baseline



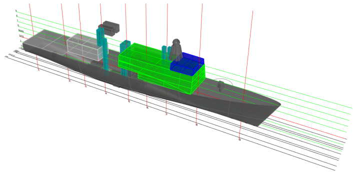
(b) X



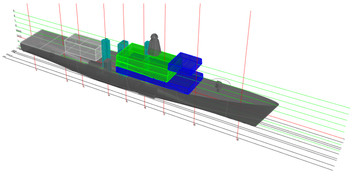
(c) XI



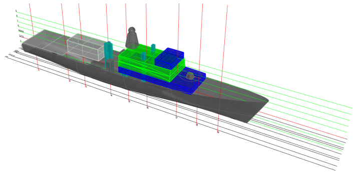
(d) XII



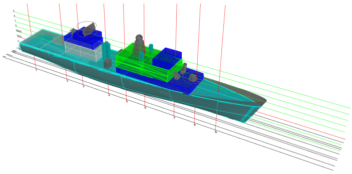
(e) XIII



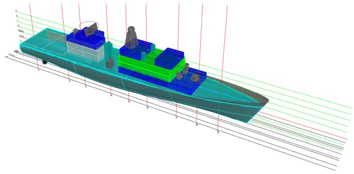
(f) XIV



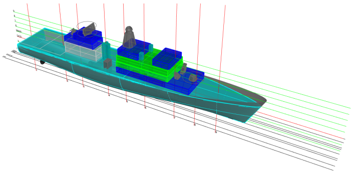
(g) XV



(h) XX

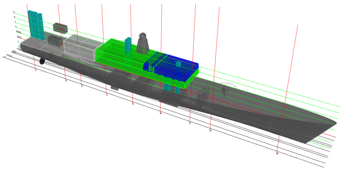


(i) XXI

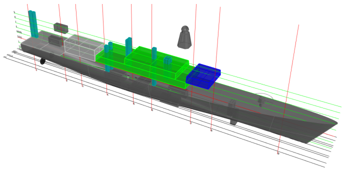


(j) XXII

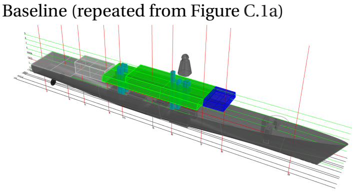
Figure C.1: Stored design instances by Team A.



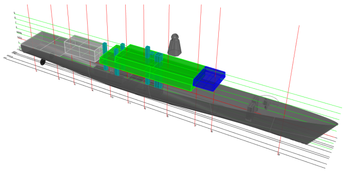
(a) Baseline (repeated from Figure C.1a)



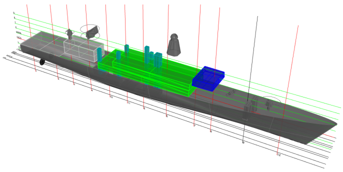
(b) XI



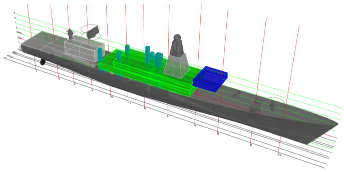
(c) XII



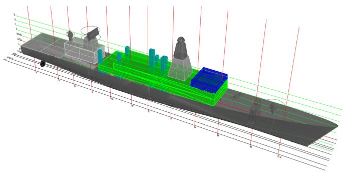
(d) XIII



(e) XVII



(f) XVIII



(g) XIX

Figure C.2: Stored design instances by Team B.

C.5. OBJECT NAMES OF UNIQUEIDS IN CASE STUDY 7

Table C.2: UniqueID versus object name for Teams A and B

	Object Name	
UniqueID	**Team A**	**Team B**
VB1	Flight deck	Flight deck
VB2	Hangar	Hangar
VB3	gun_76mm	gun_76mm
VB4	missile_mk41_vls	missile_mk41_vls
VB5	missile_mk41_vls	missile_mk41_vls
VB6	radar_imast	radar_imast
VB7	GearBoxFrigate	GearBoxFrigate
VB8	EM_3MW	EM_3MW
VB9	EM_3MW	EM_3MW
VB10	GT_LM2500_35MW	GT_LM2500_35MW
VB11	DG_MAN20V175D_3000kWe	DG_MAN20V175D_3000kWe
VB12	DG_MAN20V175D_3000kWe	DG_MAN20V175D_3000kWe
VB13	DG_MAN20V175D_3000kWe	DG_MAN20V175D_3000kWe
VB14	DG_MAN20V175D_3000kWe	DG_MAN20V175D_3000kWe
VB15	DG_MAN12V175D_1800kWe	DG_MAN12V175D_1800kWe
VB16	DG_MAN12V175D_1800kWe	DG_MAN12V175D_1800kWe
VB17	DG exhaust -aft	DG exhaust
VB18	DG exhaust	DG exhaust
VB19	DG exhaust	DG exhaust
VB20	DG exhaust- aft	DG exhaust
VB21	DG exhaust	DG exhaust
VB22	DG exhaust	DG exhaust
VB23	GT exhaust	GT exhaust
VB24	GT intake	GT intake
VB25	Bridge	Bridge
VB26	Accommodation 1	Accommodation 1
VB27	Accommodation 2	Accommodation 2
VB28	DE_MAN20V2833D_9100kW	DE_MAN20V2833D_9100kW
VB29	DE_MAN20V2833D_9100kW	DE_MAN20V2833D_9100kW
VB30	C2	DE exhaust
VB31	ciws_goalkeeper	DE exhaust
VB32	missile_mk141_harpoon	ciws_goalkeeper
VB33	radar_smart_l	missile_mk141_harpoon
VB34	missile_mk141_harpoon	radar_smart_l
VB35	Accommodation 1	DG exhaust
VB36	Accommodation 1	*None*
VB37	Accommodation 1	*None*
VB38	Bridge	Mast
VB39	*N/A*	Smart-L mast

C.6. DESIGN RATIONALE BY TEAM B

Table C.3: Compromises

ID	Elements A UniqueID	Elements B UniqueID	Preferred Item	Justification	Status	Impact	Priority	Notes	UniqueID	Active Status	Timestamp
1	[IN9]	[IN10]	Element B	With this propulsion layout and mast position it is difficult to seperate all the exhaust from the mast	Pending				CO1	True	2023-02-17 10:29:14.589000

Table C.4: System Concepts

ID	Elements A UniqueID	Justification	Status	Impact	Priority	Notes	UniqueID	Active Status	Timestamp
1	[VB7, VB8, VB9, VB10, VB11, VB12, VB13, VB14, VB15, VB16, VB17, VB18, VB19, VB20, VB21, VB22, VB23, VB24]	These systems have been selected to fulfil the current (low) speed requirements. The arrangement is yet to be determined.	Agreed	Global			SC1	True	2023-02-02 22:43:09.355000
2	[VB28, VB29]	These DEs are inteded to replace the four main DG sets when a CODOG propulsion layout is selected.	Pending				SC2	True	2023-02-03 11:14:11.240000
3	[VB17, VB19, VB20, VB22]	adjacent in x direction, better use of space and cleaner topside. Seperated in Y direction to conserve logistical routes on main decks	Pending				SC3	True	2023-02-17 10:20:37.959000
4	[VB4, VB5]	VLS battery	Pending				SC4	True	2023-02-17 10:22:02.511000
5	[VB31, VB30, VB29, VB28, VB24, VB23, VB22, VB19, VB16, VB15, VB10, VB7]	Propulsion train + power generation (including reference frames)	Pending				SC5	True	2023-02-17 11:25:48.133000

Table C.5: System Properties

ID	System Property type	Element A UniqueID	Constraint	Lower boundary	Upper boundary	Unit	Justification	Status	Impact	Priority	Notes	UniqueID	ActiveStatus	Timestamp
1	Position	VB17	minimum	14.0	0.0	m	The length of the DG exhaust is minimum 10 meters to accommodate silencers, filters etc.	Agreed	Regional	10		SP1	True	2023-02-02 15:30:24.392000
2	Position	VB18	minimum	14.0	0.0	m	The length of the DG exhaust is minimum 10 meters to accommodate silencers, filters etc.	Agreed	Regional	10		SP2	True	2023-02-02 15:30:32.850000
3	Position	VB19	minimum	14.0	0.0	m	The length of the DG exhaust is minimum 10 meters to accommodate silencers, filters etc.	Agreed	Regional	10		SP3	True	2023-02-02 15:30:37.307000
4	Position	VB20	minimum	14.0	0.0	m	The length of the DG exhaust is minimum 10 meters to accommodate silencers, filters etc.	Agreed	Regional	10		SP4	True	2023-02-02 15:30:40.356000
5	Position	VB21	minimum	14.0	0.0	m	The length of the DG exhaust is minimum 10 meters to accommodate silencers, filters etc.	Agreed	Regional	10		SP5	True	2023-02-02 15:30:43.367000
6	Position	VB22	minimum	14.0	0.0	m	The length of the DG exhaust is minimum 10 meters to accommodate silencers, filters etc.	Agreed	Regional	10		SP6	True	2023-02-02 15:31:05.616000
7	Position	VB23	minimum	19	0.0	m	The length of the GT exhaust is minimum 19 meters to accommodate silencers, filters etc.	Agreed	Regional	10		SP7	True	2023-02-02 15:31:40.451000
8	Position	VB24	minimum	19	0.0	m	The length of the GT intake is minimum 10 meters	Agreed	Regional	10		SP8	True	2023-02-02 15:36:22.418000
9	Position	VB6	minmax	X4	X6		The Imast should be located around midships to minimize movement, i.e. increase effectiveness.	Non-agreed	Global			SP9	True	2023-02-02 15:39:49.142000
10	Position	VB6	minimum	Deck 6			The Imast should be as high up as possible to increase effectiveness. Take care of motions, though.	Pending	Global			SP10	True	2023-02-02 15:41:01.781000
11	Position	VB17		0.0	0.0		These exhausts still need to be arranged	Pending				SP11	True	2023-02-02 21:29:32.238000
12	Position	VB19		0.0	0.0		These exhausts still need to be arranged	Pending				SP12	True	2023-02-02 21:29:32.779000
13	Position	VB20		0.0	0.0		These exhausts still need to be arranged	Pending				SP13	True	2023-02-02 21:29:32.826000
14	Position	VB22		0.0	0.0		These exhausts still need to be arranged	Pending				SP14	True	2023-02-02 21:29:32.889000
15	Position	VB9		0.0	0.0		This systems penetrates the hull...	Pending				SP15	True	2023-02-02 21:31:24.180000
16	Position	VB1	exact value	0.0	0.0		Flight deck at aft end of ship on main deck for safe helicopter operations	Pending				SP16	True	2023-02-17 09:39:35.567000
17	Position	VB3	minimum	25.0	0.0		preferably 25m from bow to protect from green water	Pending				SP17	True	2023-02-17 10:01:05.600000
18		SC4		0.0	0.0		Aft of main gun, forward of bridge. seperated from accomodation for survivability	Pending				SP18	True	2023-02-17 10:23:13.345000
19	Amount	XR11		0.0	0.0		Added for vulnerability reduction; compartimentation; damage length	Pending				SP19	True	2023-02-17 11:15:06.056000
20		SC5		0.0	0.0		Moved forward for space availability	Pending				SP20	True	2023-02-17 11:27:13.207000
21		XX0		0.0	0.0		Bulkhead number and placement not yet final; bulkheads to be added at fore end	Pending				SP21	True	2023-02-17 11:33:39.385000
22	Position	VB32		0.0	0.0		The CIWS should have a maximum field of view to reduce the need for two CIWS systems.	Pending	Regional	8		SP22	True	2023-03-14 19:24:13.860000
23	Position	VB33		0.0	0.0		Typically, the Harpoon is placed around the centre of the ship. Would a forward arrangement also be feasible, so save space midships?	Pending	Global			SP23	True	2023-03-14 19:25:29.917000
24	Position	VB3		0.0	0.0		Could be upgraded to CIWS functionality by the adition of PHAROS radar and Dart munition in the future.	Pending				SP24	True	2023-03-16 09:28:01.092000

Table C.6: Interactions

*: Exhaust

ID	Element A UniqueID	Element B UniqueID	Interaction category	Interaction type	Constraint	Lower boundary	Upper boundary	Unit	Justification	Status	Impact	Priority	Notes	UniqueID	Active Status	Timestamp
1	VB3	VB4	Physical	forward of	minimum	0.0	0.0	m	The gun should be forward of the VLS because of the limited available space in the bow area, and to increase the effective arc of the gun. The distance between gun and VLS is TBD.	Pending	Regional			IN1	True	2023-02-02 15:44:08.307000
2	VB3	VB5	Physical	forward of	minimum	0.0	0.0	m	The gun should be forward of the VLS because of the limited available space in the bow area, and to increase the effective arc of the gun. The distance between gun and VLS is TBD.	Pending	Regional			IN2	True	2023-02-02 15:44:08.510000
3	VB9	VB8	Physical	horizontally adjacent		0.0	0.0		To have similar axis loading, its best to have a mirrored arrangement of the propulsion train on the left and right side.	Pending				IN3	True	2023-02-02 15:59:01.071000
4	VB6	VB23	Physical	forward of		0.0	0.0		Exhaust smoke can negatively impact radar performance.	Pending				IN4	True	2023-02-02 21:32:27.075000
5	VB28	VB29	Physical	horizontally adjacent		0.0	0.0		To ensure equal GearBox loading, the main DE can be arranged side by side	Pending	Regional			IN5	True	2023-02-03 11:13:06.945000
6	VB2	VB1	Physical	forward of	exact value	0.0	0.0		Hanger forward of flight deck for helo ops	Pending				IN6	True	2023-02-17 09:41:39.257000
7	VB10	VB7	Physical	horizontally adjacent	minimum	0.0	0.0		As close to eachother a possible to reduce vulnerability	Pending				IN7	True	2023-02-17 09:47:34.403000
8	VB13	VB8	Physical	Provide Power		0.0	0.0		Power user and selected close to eachother to reduce system vulnerability	Pending				IN8	True	2023-02-17 09:56:15.080000
9	VB21	VB14	Physical	vertically adjacent		0.0	0.0		exhaust needs to be connected to genset	Pending				IN9	True	2023-02-17 10:27:16.718000
10	VB21	VB6	Physical	horizontally separated		0.0	0.0		Exhaust needs to be well aft of mast for DME	Pending				IN10	True	2023-02-17 10:28:37.511000
11	VB16	VB15	Physical	horizontally separated		0.0	0.0		Separation for vulnerability reduction	Pending				IN11	True	2023-02-17 11:12:01.989000
12	VB34	BT23*	Physical	forward of		0.0	0.0		Exhaust smoke can negatively impact radar performance.	Pending	Global			IN12	True	2023-03-14 19:26:59.945000
13	VB3	VB32	Physical	radially separated	minimum	180.0	180.0	MW	CIWS field of view should be maximised by having different systems in opposite orientations	Pending				IN13	True	2023-03-16 09:30:37.688000
14	VB19	VB34	Physical	horizontally separated	minimum	6.0	15.0	m	Exhaust smoke can negatively impact radar performance	Agreed	Regional			IN14	True	2023-03-16 09:41:53.592000
15	VB5	VB3	Physical	horizontally separated	minimum	1.0	0.0	compartment	Ammo stores for 76mm and VLS in different compartments to prevent damage propagation	Agreed				IN15	True	2023-03-16 10:06:21.135000

ACRONYMS

AA Achieved Area.

AMoP Area Measure of Performance.

AR Aspect Ratio.

C2 Command and Control.

CAD Computer-Aided Design.

CL Centre line.

CODLAG Combined Diesel-electric and Gas.

CODOG Combined Diesel or Gas.

CoG centre of gravity.

COMMIT Materiel and IT Command.

CWIS Close-In Weapon System.

DA Detailed Arrangement.

DBB Design Building Block approach.

DCD Damage Control Deck.

DE Diesel Engine.

DG Diesel Generator.

DMO Defense Materiel Organisation.

DOORS Dynamic Object-Oriented Requirements System.

EGR Emergency Generator Room.

EM Electric Motor.

FBB Functional Building Block.

FIDES Functional Integrated Design Exploration of Ships.

GA Genetic Algorithm.

GAP General Arrangement Plan.

GT Gas Turbine.

GUI Graphical User Interface.

IL Improved Layout.

ISA Intelligent Ship Arrangement.

ITD Integrated Topside Design.

JSS Joint Support Ship.

LCF Air Defence and Command Frigate.

LMoP Logistic Measure of Performance.

LOA Length Overall.

LPD Landing Platform Dock.

MBSE Model-Based Systems Engineering.

MoP Measure of Performance.

NCO Non-Commissioned Officer.

OPV Oceangoing Patrol Vessel.

PS Port side.

PSO Particle Swarm Optimisation.

R&D Research and Development.

RA Required Area.

RAS Replenishment At Sea.

RKC Reactive Knowledge Capture.

RWK Rework.

SB Starboard side.

SDRM Ship Design Rationale Method.

SEWACO Sensors, Weapons, and Command.

SMoP Safety Measure of Performance.

SSDT Surface Ship Design Tool.

TNO Netherlands Organisation for Applied Scientific Research.

TU Delft Delft University of Technology.

UR Undiscovered Rework.

VLS Vertical Launch System.

WARGEAR WARship GEneral ARrangement.

Bibliography

Aladib, L., & Lee, S. (2019). Pattern detection and design rationale traceability: An integrated approach to software design quality. *IET Software*, *13*(4), 249–259.

Allseas. (n.d.). Pioneering Spirit [Accessed: 2023-06-29]. https://allseas.com/equipment/pioneering-spirit-offshore-construction-vessel/

Andrews, D. (1981). Creative ship design. *Trans. RINA*, *123*, 447–471.

Andrews, D. (1986). An integrated approach to ship synthesis. *Trans. RINA*, *128*, 73–102.

Andrews, D. (2003). A creative approach to ship architecture. *The International Journal of Maritime Engineering*, *145*. https://doi.org/10.3940/rina.ijme.2003.a3.9031

Andrews, D. (2012a). Art and science in the design of physically large and complex systems. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *468*(2139), 891–912.

Andrews, D. (2012b). Marine requirements elucidation and the nature of preliminary ship design. *International Journal of Maritime Engineering*, *154*(A4).

Andrews, D. (2018a). Choosing the Style of a New Design - The Key Ship Design Decision. *International Journal of Maritime Engineering*, *Vol 160*(A1). https://doi.org/10.3940/RINA.IJME.2018.A1.457

Andrews, D. (2018b). The Sophistication of Early Stage Design for Complex Vessels. *International Journal of Maritime Engineering*, *Vol 160*(SE 18). https://doi.org/10.3940/RINA.IJME.2018.SE.472

Andrews, D. (2021). Design errors in ship design. *Journal of Marine Science and Engineering*, *9*(1), 34.

Andrews, D. (2022a). 100 things (or so) a ship designer needs to know. *SNAME 14th International Marine Design Conference (IMDC)*.

Andrews, D. (2022b). Babies, bathwater and balance–the fuzzy half of ship design and recognising its importance. *SNAME Maritime Convention*.

Andrews, D., & Dicks, C. (1997). The building block design methodology applied to advanced naval ship design. *Proceedings of the International Marine Design Conference, Newcastle, United Kingdom*, 3–19.

Andrews, D., & Pawling, R. (2007). Concept studies for a joint support ship. *Proceedings of RINA Warship 2007: The Affordable Warship, Bath, UK*.

Andrews, D., & Pawling, R. (2008). A case study in preliminary ship design. *Transactions of the Royal Institution of Naval Architects, RINA, Volume 150, Part A3, 2008, International Journal of Maritime Engineering*.

Aurisicchio, M., Bracewell, R., & Hooey, B. (2016). Rationale mapping and functional modelling enhanced root cause analysis. *Safety science*, *85*, 241–257.

Baker, R. (1956). Habitability in Ships of the Royal Canadian Navy. *Trans. SNAME 1956*.

Ball, L., Lambell, N., Ormerod, T., Slavin, S., & Mariani, J. (2001). Representing design rationale to support innovative design reuse: A minimalist approach. *Automation in Construction*, *10*(6), 663–674.

Bandecchi, M., Melton, B., Gardini, B., & Ongaro, F. (2000). The esa/estec concurrent design facility. *Proceedings of EUSEC*, *9*, 2000.

Baušys, R., & Pankrašovaite, I. (2005). Optimization of architectural layout by the improved genetic algorithm. *Journal of Civil Engineering and Management*, *11*(1), 13–21.

BBC. (2018). Spain's new submarine 'too big for its dock' [Accessed: 2023-06-19]. https://www.bbc.com/news/world-europe-44871788

van der Bles, R. (2019). *A Design Tool for Machinery Space Arrangement* (Master's thesis). Delft University of Technology.

Bourke, P. (1996). Cross correlation. *Auto Correlation—2D Pattern Identification*.

Bracewell, R., Wallace, K., Moss, M., & Knott, D. (2009). Capturing design rationale. *Computer-Aided Design*, *41*(3), 173–186.

Bratthall, L., Johansson, E., & Regnell, B. (2000). Is a design rationale vital when predicting change impact?–a controlled experiment on software architecture evolution. *Product Focused Software Process Improvement: Second International Conference, PROFES 2000, Oulu, Finland, June 20-22, 2000. Proceedings 2*, 126–139.

Brefort, D., Shields, C., Habben Jansen, A., Duchateau, E., Pawling, R., Droste, K., Jasper, T., Sypniewski, M., Goodrum, C., Parsons, M., Kara, M., Roth, M., Singer, D., Andrews, D., Hopman, H., Brown, A., & Kana, A. (2018). An architectural framework for distributed naval ship systems. *Ocean Engineering*, *147*, 375–385. https://doi.org/10.1016/J.OCEANENG.2017.10.028

Brown, D. (1986). Defining a warship. *Naval Engineers Journal*, *98*(2), 31–40.

Brown, D. (1987). The architecture of frigates. *Proc. Warship 1987: Int. Symp. On ASW, London*.

de Bruijn, J., & ten Heuvelhof, E. (2008). *Management in networks: On multi-actor decision making*. Routledge.

van Bruinessen, T. (2016). *Towards controlled innovation of complex objects. a sociotechnical approach to describing ship design* (Doctoral dissertation). Delft University of Technology.

Burge, J., & Brown, D. (2000). Reasoning with design rationale. In *Artificial intelligence in design'00* (pp. 611–629). Springer.

Burrus, C. S., & Parks, T. (1985). Convolution algorithms. *Citeseer: New York, NY, USA*, *6*.

Camozzato, D. (2015). *A method for growth-based procedural floor plan generation* (Doctoral dissertation). Pontifícia Universidade Católica do Rio Grande do Sul.

Carlson, C., & Fireman, H. (1987). General arrangement design computer system and methodology. *Naval engineers journal*, *99*(3), 261–273.

Cederlund, C. (2006). *Vasa I: The Archaeology of a Swedish warship of 1628*.

Charisi, N. D., Hopman, H., & Kana, A. (2022). Early-stage design of novel vessels: How can we take a step forward? *SNAME 14th International Marine Design Conference*.

Coello, C., Pulido, G., & Lechuga, M. (2004). Handling multiple objectives with particle swarm optimization. *IEEE Transactions on evolutionary computation*, *8*(3), 256–279.

Conklin, E., & Yakemovic, K. (1991). A process-oriented approach to design rationale. *Human–Computer Interaction*, *6*(3-4), 357–391.

Conklin, J. (2006). *Wicked problems & social complexity* (Vol. 11). CogNexus Institute Napa, USA.

Cooper, K. G. (1993). The rework cycle: Benchmarks for the project manager.

Cort, A., & Hills, W. (1987). Space layout design using computer assisted methods. *Naval Engineers Journal*, *99*(3), 249–260.

Cui, H., & Turan, O. (2010). Application of a new multi-agent hybrid co-evolution based particle swarm optimisation methodology in ship design. *Computer-Aided Design*, *42*(11), 1013–1027.

Daniels, A., & Singer, D. (2012). Intelligent Ship Arrangement (ISA) Space Projection Methodology. *Proceedings of the 12th International Conference on Computer Applications and Information Technology in the Maritime Industries, Liege, Belgium.*

Daniels, A., Tahmasi, F., & Singer, D. (2009). Intelligent Ship Arrangement (ISA) methodology improvements and capability enhancements. *Proceedings of International Marine Design Conference, Trondheim, Norway.*

DeNucci, T. (2012). *Capturing design: Improving conceptual ship design through the capture of design rationale* (Doctoral dissertation). Delft University of Technology.

Derbanne, Q. (2022). Brief history of rule loads and longitudinal strength of ships. *15th International Symposium on Practical Design of Ships and Other Floating Structures (PRADS) 2022.*

Dicks, C. (2000). *Preliminary design of conventional and unconventional surface ships using a building block approach* (Doctoral dissertation). University College London.

Dijkstra, E. (1959). A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, *1*, 269–271.

Do, Q., Cook, S., & Lay, M. (2014). An investigation of MBSE practices across the contractual boundary. *Procedia Computer Science*, *28*, 692–701.

Dorst, K. (1997). *Describing design: A comparison of paradigms* (Doctoral dissertation). Delft University of Technology.

Dorst, K., & Cross, N. (2001). Creativity in the design process: Co-evolution of problem–solution. *Design studies*, *22*(5), 425–437.

Drazin, R., Glynn, M., & Kazanjian, R. (1999). Multilevel theorizing about creativity in organizations: A sensemaking perspective. *Academy of management review*, *24*(2), 286–307.

Droste, K., & Hage, M. (2022). Support Functionality in the System Modelling: The Chicken or the Egg. *14th International Marine Design Conference (IMDC).*

Droste, K., Hopman, J., Kana, A., & van Oers, B. (2020). Introducing operational information into early stage ship design using queueing networks. *International Journal of Maritime Engineering*, *162*(A2).

Droste, K., Kana, A., & Hopman, J. (2018). Process-based analysis of arrangement aspects for configuration-driven ships. *Marine Design XIII, Volume 1: Proceedings of the 13th International Marine Design Conference (IMDC 2018), June 10-14, 2018, Helsinki, Finland*, 327.

Droste, K., & le Poole, J. (2020). Integrating detailed layout generation with logistic performance assessment to improve layout insights in early stage warship design.

15th International Naval Engineering Conference, 5-9 October, 2020. https://doi.org/10.24868/issn.2515-818X.2020.010

Duchateau, E. (2016). *Interactive evolutionary concept exploration in preliminary ship design* (Doctoral dissertation). Delft University of Technology.

Duchateau, E., de Vos, P., & van Leeuwen, S. (2018). Early stage routing of distributed ship service systems for vulnerability reduction. In *Marine Design XIII, Volume 2: Proceedings of the 13th International Marine Design Conference (IMDC 2018), June 10-14, 2018, Helsinki, Finland* (pp. 1083–1096). CRC Press.

Duffy, A. (1997). The "what" and "how" of learning in design. *IEEE Expert*, *12*(3), 71–76.

Ebrahimi, A., Brett, P. O., Erikstad, S. O., & Asbjørnslett, B. E. (2021). Influence of ship design complexity on ship design competitiveness. *Journal of Ship Production and Design*, *37*(03), 181–195.

Ebrahimi, A., Brett, P. O., Erikstad, S. O., Asbjørnslett, B. E., & Garcia Agis, J. J. (2020). Ship design complexity, sources, drivers, and factors: A literature review. *International Shipbuilding Progress*, *67*(2-4), 221–252.

Esbati, S. (2018). *Design for support in the initial design of naval combatants* (Doctoral dissertation). UCL (University College London).

Evans, J. (1959). Basic design concepts. *Journal of the American Society for Naval Engineers*, *71*(4), 671–678.

Falessi, D., Briand, L. C., Cantone, G., Capilla, R., & Kruchten, P. (2013). The value of design rationale information. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, *22*(3), 1–32.

Ferreiro, L., & Stonehouse, M. (1991). A comparative study of US and UK frigate design.

Fischer, G., Lemke, A., McCall, R., & Morch, A. (1991). Making argumentation serve design. *Human–Computer Interaction*, *6*(3-4), 393–419.

Fischer, G., & Shipman, F. (2011). Collaborative design rational and social creativity in cultures of participation. *Human Technology: An Interdisciplinary Journal on Humans in ICT Environments*.

Forester, C. (1942). *The Ship*.

Garrett, H. E. (1958). *Statistics in psychology and education*. Longmans, Green, Co.

Gaspar, H. M., Brett, P. O., Ebrahim, A., & Keane, A. (2014). Data-driven documents (d3) applied to conceptual ship design knowledge. *Proceedings COMPIT 2014*.

Gillespie, J. (2012). *A Network Science Approach to Understanding and Generating Ship Arrangements in Early-Stage Design* (Doctoral dissertation). University of Michigan.

Gillespie, J., Daniels, A., & Singer, D. (2013). Generating functional complex-based ship arrangements using network partitioning and community preferences. *Ocean Engineering*, *72*, 107–115.

Guo, Z., & Li, B. (2017). Evolutionary approach for spatial architecture layout design enhanced by an agent-based topology finding system. *Frontiers of Architectural Research*, *6*(1), 53–62.

Habben Jansen, A. (2020). *A markov-based vulnerability assessment of distributed ship systems in the early design stage* (Doctoral dissertation). Delft University of Technology.

Head, B. W., & Alford, J. (2015). Wicked problems: Implications for public policy and management. *Administration & society*, *47*(6), 711–739.

Horner, J., & Atwood, M. E. (2006). Effective design rationale: Understanding the barriers. In *Rationale management in software engineering* (pp. 73–90). Springer.

Hoving, A., & Wildeman, D. (2012). *Nicolaes Witsen and shipbuilding in the Dutch golden age*. Texas A&M University Press.

IBM. (n.d.). IBM Engineering Requirements Management DOORS Family [Accessed: 2023-12-21]. https://www.ibm.com/products/requirements-management

Iijima, J. (2022). What Should Be Added to Science for Solving Wicked Problems? In *Systems Research II: Essays in Honor of Yasuhiko Takahara on Systems Management Theory and Practice* (pp. 149–168). Springer.

IMO. (n.d.). Regulation 38 - Alternative design and arrangements [Accessed: 2023-06-19]. https://www.imorules.com/GUID-0398F641-291E-4BD8-BBB7-C8CA725836CD.html

Inoue, M., & Takagi, H. (2008). Layout algorithm for an EC-based room layout planning support system. *2008 IEEE Conference on Soft Computing in Industrial Applications*, 165–170.

Jarczyk, A., Löffler, P., & Shipman, F. (1992). Design rationale for software engineering: A survey. *Proceedings of the Hawaii International Conference on System Sciences*, *25*, 577–577.

Jiang, Z. (2021). Installation of offshore wind turbines: A technical review. *Renewable and Sustainable Energy Reviews*, *139*, 110576.

Kana, A., Brans, S., Bronkhorst, P., Charisi, N., Lupoae, L., van Lynden, C., Kao, I.-T., le Poole, J., & Zwaginga, J. (2022). Development and Lessons Learned of New Modular Ship Design Activities for Graduate Education During COVID. *SNAME 14th International Marine Design Conference (IMDC)*. https://doi.org/10.5957/IMDC-2022-225

Kana, A., & Rotteveel, E. (2018). Development and lessons learned of a block-based conceptual submarine design tool for graduate education. In *Marine Design XIII, Proceedings of the 13th International Marine Design Conference (IMDC 2018), June 10-14, 2018, Helsinki, Finland* (pp. 103–110). CRC Press.

Kana, A., Shields, C., & Singer, D. (2016). Why is naval design decision-making so difficult? *Warship 2016: Advanced Technologies in Naval Design, Construction, & Operation, 15-16 June 2016, Bath, UK*.

Keane, R., & Tibbits, B. (2018). The Goldilocks effect in naval ship design: Too little, too much, just right. *SNAME Maritime Convention*, D033S009R005.

Keane, R., & Tibbitts, B. (1996). A revolution in warship design: Navy-industry integrated product teams. *Journal of ship production*, *12*(04), 254–268.

Kennedy, J. (2010). Particle swarm optimization. *Encyclopedia of machine learning*, 760–766.

Keuning, J., & Pinkster, J. (1995). Optimisation of the seakeeping behaviour of a fast monohull. *Fast'95 conference*.

Klein, M. (1993). Capturing design rationale in concurrent engineering teams. *Computer*, *26*(1), 39–47.

Kooij, E. (2022). *Systems modeling in the naval domain* (Master's thesis). Delft University of Technology.

Kossiakoff, A., Sweet, W., Seymour, S., & Biemer, S. (2003). *Systems engineering: Principles and practices*. Wiley Online Library.

Kuofie, E. (2010). *Radex: A rationale-based ontology for aerospace design explanation* (Master's thesis). University of Twente.

Lamb, T. (Ed.). (2003). *Ship Design and Construction, Volumes 1-2*. SNAME.

Lee, J. (1997). Design rationale systems: Understanding the issues. *IEEE expert*, *12*(3), 78–85.

Lee, K., Roh, M., & Jeong, H. (2005). An improved genetic algorithm for multi-floor facility layout problems having inner structure walls and passages. *Computers & Operations Research*, *32*(4), 879–899.

Lehtola, L., Kauppinen, M., & Kujala, S. (2004). Requirements prioritization challenges in practice. *International Conference on Product Focused Software Process Improvement*, 497–508.

Li, G., Tang, H., Sun, Y., Kong, J., Jiang, G., Jiang, D., Tao, B., Xu, S., & Liu, H. (2019). Hand gesture recognition based on convolution neural network. *Cluster Computing*, *22*(2), 2719–2729.

Lipowski, A., & Lipowska, D. (2012). Roulette-wheel selection via stochastic acceptance. *Physica A: Statistical Mechanics and its Applications*, *391*(6), 2193–2196.

Lo, S., Lou, S., Lin, J., Freedman, M., Chien, M., & Mun, S. (1995). Artificial convolution neural network techniques and applications for lung nodule detection. *IEEE transactions on medical imaging*, *14*(4), 711–718.

Lopes, R., Tutenel, T., Smelik, R., De Kraker, K., & Bidarra, R. (2010). A constrained growth method for procedural floor plan generation. *Proc. 11th Int. Conf. Intell. Games Simul*, 13–20.

van der Lugt, R. (2005). How sketching can affect the idea generation process in design group meetings. *Design studies*, *26*(2), 101–122.

van Lynden, C., van Winsen, I., Westland, C., & Kana, A. (2022). Offshore wind installation vessels: Generating insight about the driving factors behind the future design. *International Journal of Maritime Engineering*, *164*(A2), 221–236.

MacLean, A., Young, R., Bellotti, V., & Moran, T. (1991). Questions, options, and criteria: Elements of design space analysis. *Human–computer interaction*, *6*(3-4), 201–250.

MacLean, A., Young, R., & Moran, T. (1989). Design rationale: The argument behind the artifact. *ACM SIGCHI Bulletin*, *20*(SI), 247–252.

Marson, F., & Musse, S. (2010). Automatic real-time generation of floor plans based on squarified treemaps algorithm. *International Journal of Computer Games Technology*, *2010*, 7.

Matterlab. (n.d.). Unitize by Matterlab [Accessed: 2020-11-05]. https://www.matterlab.co/unitize

Mavris, D., & DeLaurentis, D. (2000). Methodology for examining the simultaneous impact of requirements, vehicle characteristics, and technologies on military aircraft design.

McCall, R. (2010). Critical conversations: Feedback as a stimulus to creativity in software design. *Human Technology*, *6*(1), 11–37.

McCall, R. (2019). Using argumentative, semantic grammar for capture of design rationale. *Design Computing and Cognition'18*, 519–535.

McNeel, R. (n.d.). Rhinoceros [Accessed: 2022-09-22]. https://www.rhino3d.com/

Medjdoub, B., & Yannou, B. (2000). Separating topology and geometry in space planning. *Computer-aided design*, *32*(1), 39–61.

Merrell, P., Schkufza, E., & Koltun, V. (2010). Computer-generated residential building layouts. In *ACM SIGGRAPH Asia 2010 papers* (pp. 1–12).

Michaels, S. (2009). Matching knowledge brokering strategies to environmental policy problems and settings. *Environmental Science & Policy*, *12*(7), 994–1011.

Michalek, J., Choudhary, R., & Papalambros, P. (2002). Architectural layout design optimization. *Engineering optimization*, *34*(5), 461–484.

Mistree, F., Smith, W., & Bras, B. (1993). A decision-based approach to concurrent design. In *Concurrent engineering: Contemporary issues and modern design tools* (pp. 127–158). Springer.

la Monaca, U., Bertagna, S., Marinò, A., & Bucci, V. (2020). Integrated ship design: An innovative methodological approach enabled by new generation computer tools. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, *14*(1), 59–76.

Mukti, M., Pawling, R., & Andrews, D. (2022). The Network Block Approach Applied to the Initial Design of Submarine Distributed Ship Service Systems. *SNAME 14th International Marine Design Conference*.

Najafi, M., Faraji, S., Bazargan, K., & Lilja, D. (2020). Energy-efficient pulse-based convolution for near-sensor processing. *IEEE Intern. Symp. on Circuits and Systems (ISCAS)*.

Nick, E. (2008). *Fuzzy optimal allocation and arrangement of spaces in naval surface ship design.* (Doctoral dissertation). University of Michigan.

Oduguwa, V., & Roy, R. (2002). Bi-level optimisation using genetic algorithm. *Proceedings 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002)*, 322–327.

van Oers, B., Takken, E., Duchateau, E., Zandstra, R., Cieraad, S., van den Broek-de Bruijn, W., & Janssen, M. (2018). Warship Concept Exploration and Definition at The Netherlands Defence Materiel Organisation. *Naval Engineers Journal*, *130*(2), 63–84.

van Oers, B. (2007). Development and implementation of an optimisation-based space allocation routine for the generation of feasible concept designs. *6th international conference on computer and IT applications in the maritime industries, COMPIT'07, Cortona, 23-25 april 2007; D. Stapersma*.

van Oers, B. (2011a). *Designing the process and tools to design affordable warships* (Technical report). NATO-RTO-MP AVT-173 Workshop on virtual prototyping of affordable military vehicles, Sofia, Bulgaria.

van Oers, B. (2011b). *A packing approach for the early stage design of service vessels* (Doctoral dissertation). Delft University of Technology.

van Oers, B., Stapersma, D., & Hopman, H. (2008). Issues when selecting naval ship configurations from a pareto-optimal set. *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. https://doi.org/10.2514/6.2008-5886

Page, S. (2006). Path dependence. *Quarterly Journal of Political Science*, *1*(1), 87–115.

Partidario, M., & Sheate, W. (2013). Knowledge brokerage-potential for increased capacities and shared power in impact assessment. *Environmental Impact Assessment Review*, *39*, 26–36.

Pawling, R., Percival, V., & Andrews, D. (2017). A study into the validity of the ship design spiral in early stage ship design. *Journal of Ship Production and Design*, *33*(2), 81–100.

Pawling, R. (2007). *The application of the design building block approach to innovative ship design* (Doctoral dissertation). UCL (University College London).

Pawling, R., & Andrews, D. (2011). Design sketching for computer aided preliminary ship design. *Ship Technology Research*, *58*(3), 182–194.

Pawling, R., & Andrews, D. (2018). Seeing arrangements as connections: The use of networks in analysing existing and historical ship designs. In *Marine Design XIII, Proceedings of the 13th International Marine Design Conference (IMDC 2018), June 10-14, 2018, Helsinki, Finland* (pp. 307–326). CRC Press.

Pawling, R., Piperakis, A., & Andrews, D. (2015). Developing architecturally oriented concept ship design tools for research and education. *12th International Marine Design Conference (IMDC)*.

Peng, C., Yang, Y., & Wonka, P. (2014). Computing layouts with deformable templates. *ACM Transactions on Graphics (TOG)*, *33*(4), 1–11.

Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. *Swarm intelligence*, *1*(1), 33–57.

le Poole, J. (2018). *Integration of aboard logistic processes in the design of logistic driven ships during concept exploration: Applied to a landing platform dock design case* (Master's thesis). Delft University of Technology.

le Poole, J., Charisi, N., Droste, K., Habben Jansen, A., & Kana, A. (2022a). The design knowledge management square - a framework for early stage complex ship design. *The 15th International Symposium on Practical Design Of Ships And Other Floating Structures (PRADS)*.

le Poole, J., Duchateau, E., Hopman, J., & Kana, A. (2022b). Interactive multi-constrained system-to-compartment allocation to support real-time collaborative complex ship layout design decision-making. *16th International Naval Engineering Conference and Exhibition (INEC)*. https://doi.org/10.24868/10658

le Poole, J., Duchateau, E., Hopman, J., & Kana, A. (2022c). On-the-fly design rationale to support real-time collaborative naval ship layout design. *14th International Marine Design Conference (IMDC)*. https://doi.org/10.5957/IMDC-2022-233

le Poole, J., Duchateau, E., Hopman, J., & Kana, A. (2023). Development and experimental testing of a collaborative design rationale method for early-stage ship layout design. *International Journal of Naval Architecture and Ocean Engineering*, *15*, 100532. https://doi.org/10.1016/j.ijnaoe.2023.100532

le Poole, J., Duchateau, E., van Oers, B., Hopman, J., & Kana, A. (2019). Semi-automated approach for detailed layout generation during early stage surface warship design. *Proceedings of the International Conference on Computer Applications in Shipbuilding, Issue 3, Rotterdam, The Netherlands*, 37–48.

le Poole, J., Duchateau, E., van Oers, B., Hopman, J., & Kana, A. (2022d). WARGEAR: 'real time' generation of detailed layout plans of surface warships during early stage design. *Ocean Engineering, 250*, 110815. https://doi.org/10.1016/j.oceaneng.2022.110815

le Poole, J., Duchateau, E., van Oers, B., & Kana, A. (2020). A case study into an automated detailed layout generation approach in early stage naval ship design. *15th International Naval Engineering Conference, 5-9 October, 2020*. https://doi.org/10.24868/issn.2515-818X.2020.011

Poorkiany, M., Johansson, J., & Elgh, F. (2016). Capturing, structuring and accessing design rationale in integrated product design and manufacturing processes. *Advanced Engineering Informatics*, *30*(3), 522–536.

Poulis, I. (2022). *Application of Model Based System Engineering (MBSE) with Ship Design Arrangement Tool of advanced zero emissions Power, Propulsion and Energy Systems in Maritime Technology* (Master's thesis). Delft University of Technology.

Pruinelli, L., Stai, B., Ma, S., Pruett, R., & Simon, G. (2019). A likelihood-based convolution approach to estimate event occurrences in large longitudinal incomplete clinical data. *2019 IEEE International Conference on Healthcare Informatics (ICHI)*, 1–8.

Reinertsen, D. (1997). *Managing the design factory*. Simon; Schuster.

Rittel, H., & Webber, M. (1973). Dilemmas in a general theory of planning. *Policy sciences*, *4*(2), 155–169.

Roberts, N. (2000). Wicked problems and network approaches to resolution. *International public management review*, *1*(1), 1–19.

Roth, M. (2016). *Analysis of General Arrangements created by the TU Delft Packing Approach making use of Network Theory* (Master's thesis). Delft University of Technology.

Seaboats.net. (n.d.). NEW BUILD - 60m Offshore Support Vessel [Accessed: 2023-03-20]. https://www.seaboats.net/new-build-60m-offshore-support-vessel.-678445

Shi, Y., et al. (2001). Particle swarm optimization: Developments, applications and resources. *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546)*, *1*, 81–86.

Shields, C., Knight, J., & Singer, D. (2016). The design process as a complex system. *ASNE Day*.

Shields, C., & Singer, D. (2017). Naval design, knowledge-based complexity, and emergent design failures. *Naval Engineers Journal*, *129*(4), 75–86.

Shipbuilder. (n.d.). Shipbuilder [Accessed: 2023-12-21]. https://shipbuilder.nl/

Shipman, F. M., & McCall, R. J. (1997). Integrating different perspectives on design rationale: Supporting the emergence of design rationale from design communication. *AI EDAM*, *11*(2), 141–154.

Simon, H. (1996). *The Sciences of the Artificial, third edition*. MIT press.

Singer, D., Gillespie, J., Hopman, H., Duchateau, E., Andrews, D., & Pawling, R. (2012). Preliminary Ship Design General Arrangements Naval International Cooperative Opportunities in Science & Technology Program (NICOP) project.

Smulders, F. E., & Bakker, H. J. (2012). Modelling the inter-subjective level of innovation. *International Journal of Technology Management*, *60*(3/4), 221–241.

van Staalduinen, I. (2019). *The requirements analysis for a shallow water pipelay barge: The replacement of Tog Mor* (Master's thesis). Delft University of Technology.

Stevens, A. P. (2016). *Naval ship preliminary arrangements for operability and reduced vulnerability* (Master's thesis). Virginia Tech.

Sun, C. (2019). *Layout analysis of polar expedition cruise ship in early stage design: By accounting for subjective preference and fuzzy logic theory* (Master's thesis). Delft University of Technology.

Takken, E. (2009). *Concept Design by using Functional Volume Blocks with variable Resolution* (Master's thesis). Delft University of Technology.

Tang, A., Babar, M., Gorton, I., & Han, J. (2006). A survey of architecture design rationale. *Journal of systems and software*, *79*(12), 1792–1804.

Teisman, G. R. (2000). Models for research into decision-making processes: On phases, streams and decision-making rounds. *Public administration*, *78*(4), 937–956.

Tepper, N. (2010). *Exploring the use of Model-based Systems Engineering (MBSE) to develop systems architectures in naval ship design* (tech. rep.). Massachusetts Institute of Technology.

Todd, J., & Marois, R. (2004). Capacity limit of visual short-term memory in human posterior parietal cortex. *Nature*, *428*(6984), 751–754.

Tupper, E., & Rawson, K. (2001). *Basic ship theory volume 2* (Vol. 2). Elsevier.

Watson, D. (1998). *Practical ship design* (Vol. 1). Elsevier.

van der Weg, H. (2020). *Traceability in ship design: Connecting need, requirements and design in one information model* (Master's thesis). Delft University of Technology.

Wolff, P. (2000). *Conceptual design of warships* (Doctoral dissertation). Universiteit Twente.

Zandstra, R., Duchateau, E., & van Oers, B. (2016). Concept exploration for a mine-countermeasures vessel using a packing-based ship synthesis model. *SNAME Maritime Convention and 5th World Maritime Technology Conference*.

Zech, A., Stetter, R., Rudolph, S., & Till, M. (2022). Capturing the design rationale in model-based systems engineering of geo-stations. *Proceedings of the Design Society*, *2*, 2015–2024.

Zhang, Y., Luo, X., Li, J., & Buis, J. J. (2013). A semantic representation model for design rationale of products. *Advanced Engineering Informatics*, *27*(1), 13–26.

Zwaginga, J., Stroo, K., & Kana, A. (2021). Exploring market uncertainty in early ship design. *International Journal of Naval Architecture and Ocean Engineering*, *13*, 352–366.

ACKNOWLEDGEMENTS

Upon completing this dissertation, it's more than appropriate to thank God in the first place for providing everything that was needed, regardless of what I deserve. As Paul says in 1 Corinthians 4:7: 'For who maketh thee to differ from another? and what hast thou that thou didst not receive?' The Lord is worthy to be served.

There are also many people to thank for the help provided in the past years. First of all, my promotor, Prof. ir. Hopman. Dear Hans, thank you for investing your scarce time in my research when you were head of the department and your free time after your retirement. Your suggestions throughout the process have resulted in an increasingly stronger overall story-line.

Secondly, my co-promotor, Dr. Kana. Dear Austin, the countless meetings we had have contributed to a better understanding of the research problem, improvements of the solutions discussed in this dissertation, and interpretation of the results. Also, I could always ask for, often short-term, reviews of manuscripts and chapters. Thank you.

Then, I would like to thank the other members of the reviewing committee for the suggestions and, in advance, for the discussions during the defence ceremony. The work presented would not have been possible without the financial and in-kind contribution of the Netherlands Materiel and IT Command (COMMIT). There are many colleagues to thank for their contributions. I want to especially thank Etienne for the many discussions on ship layout design we had over the past years. You have provided me with the very necessary practical insights into the problems that I was working on in an academic setting. Also, your critical reviews have increased the quality of our academic output.

Thanks as well to the whole team of Life Cycle Modelling for providing me with a third office (besides Delft and home) occasionally. Furthermore, Bart and Siebe, thanks for offering me a job – after spending years thinking about ship design, it's a pleasure to now work collaboratively on actual ship design within your team.

The support of COMMIT, the Netherlands Organisation for Applied Scientific Research (TNO), and DAMEN Naval regarding the experiments described in Part II is much appreciated. Especially my thanks go to the participants of these organisations, as well as from Delft University of Technology (TU Delft), in the experiments. Without you, much of this research could not have been conducted.

I would like to thank all my colleagues from the department for the many lunches and talks we had (certainly before the pandemic). I want to thank my office mates (Koen, Agnieta, Berend, Nicole, and Alessia) for the many discussions, help, and social contacts over the years.

Thanks, Egbert, for being the first and only master's student I supervised during my time at TU Delft. Your topic was a challenging one, and you tackled it. I believe the process was full of learning opportunities, both for you and me.

Then, thanks to the THRIVE Institute team, especially Han, for providing the opportunity to work in a truly multidisciplinary team on a societal problem. Estel, Michelle, Lisa, Judith en Rebecca, thanks for our time in Rotterdam, and hopefully we contributed to improving the link between the ‘unreachable’ youth and the municipality.

Sander, thanks for sharing a passion. There are few people with whom I shared as much time on the road as you. Thanks for being one of my paranymphs. Thanks to all my other friends for sharing time and showing interest during my research.

Ik wil mijn familie bedanken. Mijn ouders, voor het steunen en aanmoedigen om vooral door te leren. Mijn broertjes en zusjes, voor de gezelligheid. Corniek, thanks for having joined me to Canada and for being my other paranymph. Similar paths, different thoughts – I hope you find the only Rest.

Finally, there are no words to thank my wife. Jacoline, you are amazing and I love you. This work could not have been completed without your support and firm wish for me to join dinner over the past years. I hope you understand what I did by the time this dissertation is defended. Jop, ik hoop dat mama ook leert om veerboten te bouwen van Duplo. Bas, ik kijk er naar uit dat we je, evenals Jop, de wonderen van de schepping mogen laten ontdekken. Lieverds, jullie zijn mijn thuis.

Soli Deo Gloria!

Opheusden, May 2024

Curriculum Vitae

Joan le Poole was born on October 20, 1994 in Kesteren, the Netherlands. In 2016, he completed pre-university education at the Van Lodenstein College in Kesteren (2007-2010) and Amersfoort (2010-2013). Subsequently, he achieved a BSc degree in Maritime Engineering at Delft University of Technology in 2016, with a thesis on autonomous navigation of unmanned vessels in a port environment. Also, a minor in Offshore Wind Energy was completed, in which Joan took courses in project management, asset management, and economics.

Subsequently, Joan obtained an MSc degree in Marine Engineering at Delft University of Technology with a specialisation in Ship Design, Ship Production, and Shipping Management. He conducted his Master's thesis (entitled *"Integration of aboard logistic processes in the design of logistic driven ships during concept exploration - Applied to a Landing Platform Dock design case"*) on the topic of the incorporation of logistic processes in PACKING to improve concept exploration for logistic-driven ships (e.g. amphibious vessels) at the Netherlands Defence Materiel Organisation (DMO).

After completing his studies, Joan accepted a position as a Young Researcher at Delft University of Technology. As a Young Researcher, he worked on the WARGEAR (WARship GEneral ARrangement) project in collaboration with the DMO. Joan developed a new method to rapidly provide insight into potential sizing and integration issues for naval ship layout design during the concept definition design phase.

The WARGEAR project was extended into a PhD at Delft University of Technology in 2019, which was again supported by the DMO. In this PhD project, the WARGEAR method was further developed. Also, the capture and reuse of design rationale during collaborative design decision-making were improved through the development of a new Ship Design Rationale Method (SDRM).

Besides research, Joan was involved in various educational tasks. He supported and taught in the Master's course *Design of Complex Vessels.* Furthermore, he supervised one student during his Master's thesis research.

In 2023, Joan started as a Cost Engineer in the Maritime Systems Division at the DMO, which is now renamed to Materiel and IT Command (COMMIT).

LIST OF PUBLICATIONS

Journal publications

1. **le Poole, J.**, Duchateau, E., van Oers, B., Hopman, H., and Kana, A.A. (2022). *WARGEAR: 'Real time' generation of detailed layout plans of surface warships during early stage design*, Ocean Engineering, **250**, 110815, https://doi.org/10.1016/j.oceaneng.2022.110815.
2. **le Poole, J.**, Duchateau, E., Hopman, H., and Kana, A.A. (2023). *Development and experimental testing of a collaborative design rationale method for early-stage ship layout design*, International Journal of Naval Architecture and Ocean Engineering, **15**, 100532, https://doi.org/10.1016/j.ijnaoe.2023.100532.
3. **le Poole, J.**, Duchateau, E., Hopman, H., and Kana, A.A. *Ship Design Rationale Method: On-the-fly design rationale capture during early-stage complex ship design*, plan to submit to Naval Engineers Journal.

Conference publications

1. **J.J. le Poole**, E.A.E. Duchateau, B.J. van Oers, J.J. Hopman, and A.A. Kana. *Semi-automated approach for detailed layout generation during early stage surface warship design.* In Proceedings of the 19th International Conference on Computer Applications in Shipbuilding, ICCAS 2019, Rotterdam, the Netherlands
2. **J.J. le Poole**, E.A.E. Duchateau, B.J. van Oers, and A.A. Kana. *A case study into an automated detailed layout generation approach in early stage naval ship design.* In Proceedings of the 15th International Naval Engineering Conference, 5-9 October, 2020, https://doi.org/10.24868/issn.2515-818X.2020.011.
3. K. Droste and **J.J. le Poole**. *Integrating detailed layout generation with logistic performance assessment to improve layout insights in early stage warship design.* In Proceedings of the 15th International Naval Engineering Conference, 5-9 October, 2020, https://doi.org/10.24868/issn.2515-818X.2020.010.
4. **J.J. le Poole**, E.A.E. Duchateau, J.J. Hopman, and A.A. Kana. *On-the-fly design rationale to support real-time collaborative naval ship layout design.* In Proceedings of the 14th International Marine Design Conference (IMDC 2022), 26-30 June, 2022, Vancouver, Canada, https://doi.org/10.5957/IMDC-2022-233.
5. **Joan le Poole**, Nicole Charisi, Koen Droste, Agnieta Habben Jansen, Austin A. Kana. *The Design Knowledge Management Square - a Framework for Early Stage Complex Ship Design.* In Proceedings of the 15th International Symposium on PRACTICAL DESIGN OF SHIPS AND OTHER FLOATING STRUCTURES (PRADS), October 9 - 13, 2022, Dubrovnik, Croatia.
6. **J.J. le Poole**, E.A.E. Duchateau, J.J. Hopman, and A.A. Kana. *Interactive Multi-Constrained System-to-Compartment Allocation to Support Real-Time Collaborative Complex Ship Layout Design Decision-Making.* In Proceedings of the 16th International Naval Engineering Conference, 8-10 November, 2022, Delft, the Netherlands, https://doi.org/10.24868/10658.

Other scientific publications

1. Maurits van den Boogaard, Andreas Feys, Mike Overbeek, **Joan le Poole**, and Robert Hekkenberg (2016) *Control concepts for navigation of autonomous ships in ports* In Proceedings of the 10th symposium on high-performance marine vehicles - HIPER'16, October 2016, Cortona, Italy.

2. Austin A. Kana, Sophia Brans, Philip Bronkhorst, Nicole Charisi, I-Ting Kao, Laurentiu Lupoae, Casper van Lynden, **Joan le Poole**, and Jesper Zwaginga. *Development and Lessons Learned of New Modular Ship Design Activities for Graduate Education During COVID.* In Proceedings of the 14th International Marine Design Conference (IMDC 2022), 26-30 June, 2022, Vancouver, Canada. https://doi.org/10.5957/IMDC-2022-225

3. P. Doornebos, M. Francis, **J. le Poole**, and A.A. Kana (2023). *Design and feasibility of a 30-to 40-knot emission-free ferry.* International Shipbuilding Progress, 70(2), 81-114.