A Ballast Optimisation Method for \textit{Pieter Schelte} Lifting Operations

Venetia Lazaratou

February 2015

Graduation Committee

Prof. ir. J. J. Hopman \hspace{1cm} Delft University of Technology
Ir. P. de Vos \hspace{1cm} Delft University of Technology
Dr. ir. S. A. Miedema \hspace{1cm} Delft University of Technology
Ir. J. van Lammeren \hspace{1cm} Allseas Engineering B.V.

Document Number: SDPO.15.004.m

Delft University of Technology
Ship Design, Production & Operations

Allseas Engineering B.V.
Heavy Lift Engineering
Thesis for the degree of MSc in Marine Technology in the specialization of Ship Design

A Ballast Optimisation Method for Pieter Schelte Lifting Operations

By

Venetia Lazaratou

Performed at

Allseas Engineering B.V.

This thesis (SDPO.15.004.m) is classified as confidential in accordance with the general conditions for projects performed by the TU Delft.

11 February 2015

Company Supervisors
Responsible supervisor: Ir. R. Bosland
E-mail: RBI@allseas.com
Daily Supervisor(s): Ir. J. van Lammeren
E-mail: JvLa@allseas.com

Thesis Exam Committee
Chair/Responsible Professor: Prof. ir. J. J. Hopman
Staff Member: Ir. P. de Vos
Staff Member: Dr. ir. S. A. Miedema
Company Member: Ir. J. van Lammeren

Author Details
Student number: 4277805
Author contact e-mail: venilaz@gmail.com
Acknowledgements

I would like to express my appreciation for the support of the many people who helped me in the course of this thesis. I would first like to thank my university supervisor, Professor Hopman, for his perceptive suggestions and guidance. My warmest thanks to my daily supervisor, Jasper van Lammeren, at Allseas for his unwavering enthusiasm and optimism, and the compelling discussions.

I would like to thank the other committee members, Professor Miedema and Peter de Vos, for their constructive criticism and interest. I would also like to express my gratitude to Etienne Duchateau and Chris Rose, of the Ship Design Laboratory of TU Delft, both for their instrumental help with the genetic algorithm optimisation and for their kind permission to use their NSGA-II implementation.

I am thankful to Allseas Engineering, and particularly to Rutger Bosland, for affording me the opportunity to work on such an interesting and challenging graduation project. Special thanks to Gouke, Patricia and Martin at Allseas for sharing their extensive knowledge of the PS ballasting system.

For their constant support, I would like to thank my family and friends, and especially Solon, for the thought-provoking discussions, inspiration and encouragement.

V. Lazaratou
Delft, 2014
Summary

When completed early next year, Allseas’ Pieter Schelte (PS) will be one of the largest vessels in the world and uniquely capable of single-lifting very heavy offshore platforms. The U-shaped slot formed by the two bows of the vessel allows it to sail-in around an offshore platform. With the aid of a specialised topsides lifting system (TLS) installed around the slot, the vessel can lift an entire topsides in a single lift and transport the structure to shore for decommissioning.

During such lifting operations, the PS is required to move through a wide range of draughts, not only to clear the bottom of the topsides while sailing-in around it, but also to connect the TLS and complete the remainder of the lift. The vessel has been designed with an enormous ballasting capacity in order to achieve the necessary range of draughts (12-25 m) without exceeding hydrostatic and safety requirements. The ballasting system is comprised of 87 ballast tanks, spanning the entire length and breadth of the enormous hull, and 6 pump rooms. The distribution of ballast tanks among the pump rooms is uneven, both with respect to the total volume of water ballast and to the total number of tanks per zone.

Based on preliminary analyses, the ballasting time required for lifting operations has been identified as considerable and as driving for their duration. A keen interest exists in finding a way to optimise ballasting configurations so as to minimise this time. However, the preparation of ballast configurations by hand is a complicated and time-intensive task due to the multitude of objectives and constraints associated with the analysis. Additionally, neither the in-house tools nor the currently available commercial software for loading condition preparation are equipped to generate ballast tank fillings for optimal time, or to consider multiple loading conditions simultaneously.

The importance of ballasting to lifting operations, as well as the complexity of preparing loading conditions for the PS, inspired the current master thesis, with the main objective:

To develop a ballast optimisation method, to be implemented in a tool, which will facilitate the preparation of loading conditions for Pieter Schelte lifting operations, with the primary goal of minimising total ballasting time for an operation.

The ballasting problem was modelled mathematically so as to consider the requirements of multiple loading conditions at once, with the ultimate goal of minimising the total ballasting time for the lifting operation. Hydrostatic and structural requirements for each loading condition were implemented as constraints in the model. To ensure that a sufficient pool of feasible solutions were retained per iteration of the optimisation, the particularly stringent target draught and trim constraints were modelled as objectives to be minimised. In addition to the main body of the optimisation, which was based on an existing implementation of the non-dominated
sorting genetic algorithm (NSGA-II), auxiliary functions were developed to update ship characteristics for changing tank fillings, evaluate structural constraints, and estimate the pumping time required for each transition between loading conditions.

The optimisation was implemented in MATLAB as a multi-objective genetic algorithm, and evaluated for a test case of three loading conditions. The optimisation method succeeded in meeting the thesis objective, as the ballast configurations produced by the optimisation for the test case resulted in reduced overall ballasting time, compared to the hand-prepared configurations. It was also possible to reduce the analysis time needed for the preparation of ballast configurations for the PS from weeks to hours, or at most days. Further improvements to the optimisation are suggested primarily with respect to the assumed pumping capacity and the feasibility of intermediate filling configurations between the loading conditions investigated.
## Contents

**Summary** 

List of Figures vii

List of Tables viii

Abbreviations ix

Symbols x

1 Introduction 1

1.1 Background 1

1.1.1 Allseas 1

1.1.2 Pieter Schelte 2

1.2 Topsides Lifting Operations with the PS 4

1.2.1 Importance of Ballasting for Lifting Operations 8

1.3 Preparing Ballast Tank Configurations for the PS 8

1.3.1 Brief Description of the Ballasting System 9

1.3.2 Preparing Ballast Configurations by Hand 10

1.3.3 Estimating Ballasting Time for Operations 11

1.4 Motivation for Development of Ballast Optimisation Method 12

2 Problem Definition 15

2.1 Main Objective 15

2.1.1 Additional Limitations to Scope 16

2.2 Supporting Research Questions 18

2.3 Existing Literature and Tools 18

2.3.1 Ballast Optimisation Methods 18

2.3.2 Ship Loading Software 20

2.3.3 Conventional ‘Loadmasters’ vs the Intended Ballast Optimisation Method 21

2.4 Approach 21

3 Ballasting System of the PS 23

3.1 Description of the Ballasting System 23

3.1.1 Ballast Tanks 23

3.1.1.1 Quick-Drop Tanks 24

3.1.2 Pumping System 25
3.1.2.1 Stripping System ........................................ 26
3.1.3 Zones .................................................................. 27
3.2 Ballasting Operations .............................................. 29
  3.2.1 Ballasting .......................................................... 29
  3.2.2 De-ballasting ....................................................... 31
3.3 Calculation of Pumping Time ..................................... 32

4 Method ...................................................................... 37
  4.1 Modelling ............................................................... 37
    4.1.1 The Structure of an Optimisation .......................... 37
    4.1.2 Modelling the Ballasting Problem as an Optimisation ........................................................................... 40
      4.1.2.1 Design Variables ............................................ 40
      4.1.2.2 Objectives ...................................................... 41
      4.1.2.3 Constraints ..................................................... 45
      4.1.2.4 Search Algorithm .......................................... 47
  4.2 Programming Implementation .................................... 55
    4.2.1 Numerical Computing Environment ..................... 55
    4.2.2 Non-linear Constrained GA Optimisation with MATLAB’s Global Optimisation Toolbox .................. 56
    4.2.3 NSGA-II ............................................................ 58
    4.2.4 Ballast Method Implementation ........................... 61
  4.3 Influence of GA Parameters on Optimisation ................. 66
    4.3.1 Design Variable Discretisation ............................. 67
    4.3.2 Population Size .................................................. 68
    4.3.3 Mutation Percentage .......................................... 69
    4.3.4 Effect of Constraints on Population Feasibility ........ 70
  4.4 Switching GAs: From MATLAB’s GA to NSGA-II .......... 70

5 Results ...................................................................... 73
  5.1 Description of Test Case ........................................... 73
    5.1.1 Threshold for Minimum Pumping Time ................ 74
  5.2 Hand-Calculated Loading Conditions for Test Case .......... 75
  5.3 Optimisation-Produced Loading Conditions for Test Case ... 78
    5.3.1 Validation of Results of Optimisation .................... 82
  5.4 Comparison of Hand-Calculated and Optimisation-Produced Results ......................................................... 83
  5.5 Discussion ............................................................. 86

6 Conclusions & Recommendations ................................ 88
  6.1 Conclusions ........................................................... 88
  6.2 Recommendations .................................................. 91
    6.2.1 Further Development of Ballast Optimisation Method ................................................................. 91
    6.2.2 Further Improvement of Ballasting Time for Lifting Operations ..................................................... 92

A Coordinate Systems .................................................. 94
B Ship Hydrostatics ...................................................... 95
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Artist’s impression of the <em>Pieter Schelte</em></td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Topsides positioned in slot of <em>Pieter Schelte</em></td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>(a) Jacket being lifted by JLS and (b) loaded on to aft section of <em>Pieter Schelte</em></td>
<td>3</td>
</tr>
<tr>
<td>1.4</td>
<td>Modular decommissioning of topsides in multiple lifts</td>
<td>4</td>
</tr>
<tr>
<td>1.5</td>
<td>(a) PS ballasting to deeper draught and (b) sailing-in around the platform</td>
<td>6</td>
</tr>
<tr>
<td>1.6</td>
<td>(a) Closer view of slot of PS as the vessel is de-ballasting to connecting draught and (b) with lifting beams extended and in contact with underside of platform</td>
<td>7</td>
</tr>
<tr>
<td>1.7</td>
<td>(a) Pre-tensioning and (b) fast-lift phases of topsides removal operation</td>
<td>7</td>
</tr>
<tr>
<td>1.8</td>
<td>(a) Move-out phase of operation and (b) offloading of the topsides onto a cargo barge</td>
<td>7</td>
</tr>
<tr>
<td>1.9</td>
<td>PS ballast system layout, including overview of zones and pump rooms</td>
<td>9</td>
</tr>
<tr>
<td>1.10</td>
<td>Hand-calculated ballast configurations for topsides removal operation</td>
<td>11</td>
</tr>
<tr>
<td>1.11</td>
<td>Procedure for preparing loading conditions by hand (adapted from PS Ballast Guideline, 2014). Phases highlighted in blue may be acted upon by an optimisation.</td>
<td>13</td>
</tr>
<tr>
<td>2.1</td>
<td>Overview of approach taken to realise thesis’ main objective</td>
<td>22</td>
</tr>
<tr>
<td>3.1</td>
<td>Ballast tank centre of gravity at 100% filling</td>
<td>24</td>
</tr>
<tr>
<td>3.2</td>
<td>Schematic representation of PS ballast tanks, including numbering, from Allseas’ LCT</td>
<td>25</td>
</tr>
<tr>
<td>3.3</td>
<td>Ballast tank capacities in metric tons for 100% filling levels</td>
<td>25</td>
</tr>
<tr>
<td>3.4</td>
<td>Layout of aft starboard pump room of PS</td>
<td>26</td>
</tr>
<tr>
<td>3.5</td>
<td>Screen print from Tribon ship model of PS, showing nearly full extent of the ballast system, from (PS WB System FMEA, 2010)</td>
<td>27</td>
</tr>
<tr>
<td>3.6</td>
<td>Minimum filling percentages of ballast tanks (corresponding to 1 m remaining water height in tank)</td>
<td>27</td>
</tr>
<tr>
<td>3.7</td>
<td>Simplified representation of ballast system, including approximate locations of ring main, pump rooms and zone isolation valves</td>
<td>28</td>
</tr>
<tr>
<td>3.8</td>
<td>Direction of flow and valve settings in aft starboard pump room during ballasting (PS Ballast Guideline, 2014)</td>
<td>30</td>
</tr>
<tr>
<td>3.9</td>
<td>Feasible combinations between ballast zones</td>
<td>30</td>
</tr>
<tr>
<td>3.10</td>
<td>Direction of flow and valve settings in aft starboard pump room during ballasting (PS Ballast Guideline, 2014)</td>
<td>31</td>
</tr>
<tr>
<td>3.11</td>
<td>Numbering of subgroups of tanks for de-ballasting, with crosses indicating tanks which may only be emptied by own zone’s pumps</td>
<td>32</td>
</tr>
<tr>
<td>3.12</td>
<td>Example of de-ballasting time calculation, starting with assignment of pump rooms to subgroups and steps for calculation</td>
<td>35</td>
</tr>
<tr>
<td>4.1</td>
<td>Structure of optimisation (adapted from Van Oers, 2011)</td>
<td>38</td>
</tr>
</tbody>
</table>
List of Figures

4.2 Algorithm for simple GA (adapted from Andersson, 2011) .................................. 54
4.3 Non-dominated solution fronts, from Van Oers (2011) ................................. 58
4.4 Crowding distance calculation for NSGA-II, from Van Oers (2011) ............... 59
4.5 Overview of NSGA-II procedure (from Deb et al, 2002) ............................ 60
4.6 Outline of NSGA-II search algorithm, from Van Oers (2011) ................. 62
4.7 Graphical overview (flowchart) of programming implementation ............. 66
4.8 Number of alternatives possible as a function of discretisation and number of loading conditions ................................................................. 68

5.1 Hand-calculated ballast configurations for loading conditions 1-3 ............... 76
5.2 Longitudinal structural loading distributions for hand-calculated LC 1-3 ....... 77
5.3 Optimisation results for pumping time, draught difference and trim objectives, with the solution selected for further analysis indicated by an arrow ............ 78
5.4 Optimisation-produced ballast configurations for loading conditions 1-3 .... 80
5.5 Longitudinal structural loading distributions for optimisation-produced LC 1-3 . 81

A.1 Ship coordinate system .............................................................................. 94

B.1 Hydrostatics of Pieter Schelte for lifting draught range, 12-25 m .......... 95

C.1 Pump and system curves for ballast system analysis and pump capacity calculation (PS WB Head Calculation, 2014) ........................................... 97

E.1 Maximum dimensions of ballast tanks .................................................... 101
## List of Tables

1.1 Main dimensions of *Pieter Schelte* .................................................. 3
1.2 Succession of loading conditions required for topsides removal operation .... 5
1.3 Hand-calculated ballast times for topsides removal operation, with single and combined zone use .................................................. 13

3.1 Distribution of ballast tanks among PS zones ....................................... 28
3.2 Comparison of ballasting capacity by pumps and by gravity (PS Ballast Guideline, 2014) ................................................................. 29
3.3 Feasible pump room combinations per tank subgroup for ballasting ........... 31
3.4 Feasible pump room combinations per tank subgroup for de-ballasting ....... 33

5.1 Loading conditions used as test case to validate and evaluate performance of ballast optimisation .................................................. 74
5.2 Estimation of lower threshold for pumping time per transition ................. 75
5.3 Actual values calculated for requirements for hand-prepared loading conditions .................................................. 76
5.4 Pumping times found for hand-calculated ballast configurations for test case, both by hand and using pumping scripts ................................. 77
5.5 Actual values calculated for requirements for optimisation-produced loading conditions .................................................. 79
5.6 Pumping times found for hand-calculated and optimisation produced ballast configurations for test case .................................................. 80
5.7 Ballasting pump room assignments per tank group for optimisation-produced LC 1-2 ............................................................................. 81
5.8 De-ballasting pump room assignments per tank group for optimisation-produced LC 2-3 ............................................................................. 82
5.9 GA parameters used for test case .................................................. 83
5.10 Comparison of hand-calculated and optimisation-produced ballast configurations .................................................. 84
5.11 Pumping times found for hand-calculated and optimisation produced ballast configurations for test case .................................................. 85
5.12 Time penalties for constraint violations of LC 1 of the optimisation-produced ballast configurations .................................................. 86

D.1 Critical objectives per operational phase .................................................. 100
D.2 Potential conflicts between objectives .................................................. 100
### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMC</td>
<td>active motion compensation</td>
</tr>
<tr>
<td>BM</td>
<td>bending moment</td>
</tr>
<tr>
<td>DP</td>
<td>dynamic positioning</td>
</tr>
<tr>
<td>FEM</td>
<td>finite element method</td>
</tr>
<tr>
<td>GA</td>
<td>genetic algorithm</td>
</tr>
<tr>
<td>JLS</td>
<td>jacket lift system</td>
</tr>
<tr>
<td>LC</td>
<td>loading condition</td>
</tr>
<tr>
<td>LCB</td>
<td>longitudinal centre of buoyancy</td>
</tr>
<tr>
<td>LCG</td>
<td>longitudinal centre of gravity</td>
</tr>
<tr>
<td>LCT</td>
<td>Loading Condition Tool</td>
</tr>
<tr>
<td>Lpp</td>
<td>length between perpendiculars</td>
</tr>
<tr>
<td>m</td>
<td>metres</td>
</tr>
<tr>
<td>MOEA</td>
<td>multi-objective evolutionary algorithm</td>
</tr>
<tr>
<td>NSGA-II</td>
<td>non-dominated sorting genetic algorithm, version II</td>
</tr>
<tr>
<td>PS</td>
<td><em>Pieter Schelte</em></td>
</tr>
<tr>
<td>Q/D</td>
<td>quick drop (tanks)</td>
</tr>
<tr>
<td>SA</td>
<td>simulated annealing</td>
</tr>
<tr>
<td>SF</td>
<td>shear force</td>
</tr>
<tr>
<td>t</td>
<td>metric tons</td>
</tr>
<tr>
<td>TCB</td>
<td>transverse centre of buoyancy</td>
</tr>
<tr>
<td>TCG</td>
<td>transverse centre of gravity</td>
</tr>
<tr>
<td>TLS</td>
<td>topsides lift system</td>
</tr>
<tr>
<td>VCG</td>
<td>vertical centre of gravity</td>
</tr>
<tr>
<td>WB</td>
<td>water ballast</td>
</tr>
<tr>
<td>1-D</td>
<td>one-dimensional</td>
</tr>
</tbody>
</table>
## Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c$</td>
<td>correction factor for pseudo de-ballasting time calculation</td>
<td>[-]</td>
</tr>
<tr>
<td>$i$</td>
<td>index referring to tank number</td>
<td>[-]</td>
</tr>
<tr>
<td>$j$</td>
<td>index referring to loading condition number</td>
<td>[-]</td>
</tr>
<tr>
<td>$lc$</td>
<td>number of loading conditions</td>
<td>[-]</td>
</tr>
<tr>
<td>$nvar$</td>
<td>number of design variables</td>
<td>[-]</td>
</tr>
<tr>
<td>$tnks$</td>
<td>number of ballast tanks</td>
<td>[-]</td>
</tr>
<tr>
<td>$x_{ij}$</td>
<td>design variable representing ballast tank filling levels</td>
<td>[-]</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

When completed early this year, Allseas’ *Pieter Schelte* (PS) will be one of the largest vessels in the world and uniquely capable of single-lifting very heavy offshore platforms. The PS is designed with an enormous ballasting capacity that is vital to lifting operations, ensuring that the vessel can move through a wide range of draughts without exceeding hydrostatic and safety requirements. However, based on preliminary analyses, the ballasting time required for lifting operations has been identified as considerable and as driving for their duration. A keen interest exists in finding a way to optimise ballasting configurations so as to minimise this time.

In this chapter, the ballasting problem is introduced. Initially, in section 1.1, background information is given on Allseas and the PS. In section 1.2, one of the key lifting operations to be performed with the vessel is discussed. Then, in section 1.3, the ballast system is briefly described, and the process of preparing loading conditions for the PS and estimating ballasting time for a lifting operation is outlined. Finally, in section 1.4, the motivation for developing a ballast optimisation method for the PS is summarised.

1.1 Background

1.1.1 Allseas

Founded in 1985, the Swiss-based Allseas group has established itself as a forerunner in offshore pipeline installation and subsea construction. Since pioneering the concept of pipelaying on dynamic positioning (DP) with the *Lorelay* in 1986, the company has continued to grow its
personnel and fleet, which currently includes over 2,500 people worldwide and five vessels, as well as its reputation as an industry innovator.

The *Pieter Schelte* (PS) is the most recent addition to the Allseas fleet, marking the entry of the company into the offshore heavy-lift market. The vessel has been in development for nearly 30 years, first envisioned in the 1980’s as a novel design solution to meet burgeoning North Sea decommissioning needs. It has been designed as a one-of-a-kind platform installation and removal vessel, with the additional capability of large-diameter pipelaying.

The PS was constructed at Korean shipyard Daewoo over a period of nearly three years. The vessel has recently arrived in Rotterdam and is expected to be ready for offshore operations in early 2015.

### 1.1.2 *Pieter Schelte*

At the front, the giant vessel consists of two bows, which form a U-shaped slot. A wide monohull comprises the aft end of the vessel. Figure 1.1 gives an artist’s impression of the PS. For reference purposes, the main dimensions of the vessel are listed in Table 1.1.

![Image](image_url)

**Figure 1.1:** Artist’s impression of the *Pieter Schelte*

The unique U-shape of the bows allows the vessel to sail-in around an offshore platform (figure 1.2). With the aid of a specialised topsides lifting system (TLS) installed around the slot, the vessel can lift an entire topsides in a one go and transport the structure to shore for decommissioning. The PS will be capable of the reverse operation as well, namely, the installation of a topsides in one lift. Additionally, with the aid of a jacket lift system (JLS) at the aft end of the vessel, the PS has been designed to remove and install jacket structures (figure 1.3).
<table>
<thead>
<tr>
<th>Dimension</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length Overall, L</td>
<td>383</td>
<td>m</td>
</tr>
<tr>
<td>Length Between Perpendiculars, (L_{pp})</td>
<td>370</td>
<td>m</td>
</tr>
<tr>
<td>Breadth (moulded), B</td>
<td>124</td>
<td>m</td>
</tr>
<tr>
<td>Draught (transit), T</td>
<td>12</td>
<td>m</td>
</tr>
<tr>
<td>Draught (scantling), (T_{sc})</td>
<td>27</td>
<td>m</td>
</tr>
<tr>
<td>Depth (min. moulded at side), D</td>
<td>29</td>
<td>m</td>
</tr>
<tr>
<td>Slot Length, l</td>
<td>122</td>
<td>m</td>
</tr>
<tr>
<td>Slot Breadth, b</td>
<td>59</td>
<td>m</td>
</tr>
<tr>
<td>Displacement at (T=12) m, (\Delta)</td>
<td>363,000</td>
<td>t</td>
</tr>
<tr>
<td>Displacement at (T=27) m, (\Delta_{\text{max}})</td>
<td>931,000</td>
<td>t</td>
</tr>
</tbody>
</table>

Table 1.1: Main dimensions of Pieter Schelte

Figure 1.2: Topsides positioned in slot of Pieter Schelte

Figure 1.3: (a) Jacket being lifted by JLS and (b) loaded on to aft section of Pieter Schelte
When completed, the giant vessel will be capable of single-lifting heavy topsides of up to 48,000 t and jacket structures of up to 25,000 t. The PS’s enormous lifting capability and specialised systems set it apart from other heavy-lift vessels in the same market since it may complete a topsides or jacket decommissioning in one, rather than multiple, modular (figure 1.4), lifts. This capability is beneficial because it means less time and offshore work-hours are required to decommission or install a platform. Structures may be removed as a whole, making them reusable at low refurbishing cost.

The main dimensions of the PS were selected based on the range of offshore platforms targeted for lifting. The envelope of platform dimensions determined the length and breadth chosen for the forward slot, as well as the hull dimensions and target draught range.

For instance, the height of the air gap between the water’s surface and the lowest part of the topsides can vary widely. Consequently, in order to attach the TLS beams to the platform at a suitable height, the PS was designed to be able to move through draughts from approximately 12 to 25 m. This requirement was also critical in the choice of a large depth (29 m) and considerable ballasting capacity.

1.2 Topsides Lifting Operations with the PS

One of the key operations to be performed by the PS is the removal of a topsides from an offshore platform, as part of decommissioning. In the following paragraphs, the topsides removal procedure is briefly described, so as to give an understanding of the operational requirements on the vessel. The vital role of ballasting to lifting operations is evident. Table 1.2 lists the
succession of loading conditions required for a given topsides removal operation and a step-by-step description of the operation follows.

<table>
<thead>
<tr>
<th>Loading Condition No.</th>
<th>Phase Description</th>
<th>% Topsides Load Transferred</th>
<th>Required Mean Draught [m]</th>
<th>Maximum Shear Force [ton]</th>
<th>Maximum Bending Moment [ton · m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>L.C. 1</td>
<td>Transit to Platform</td>
<td>0</td>
<td>12.00</td>
<td>25009</td>
<td>1800615</td>
</tr>
<tr>
<td>L.C. 2</td>
<td>Positioning</td>
<td>0</td>
<td>22.00</td>
<td>67023</td>
<td>3801298</td>
</tr>
<tr>
<td>L.C. 3</td>
<td>Attain Connecting Draught</td>
<td>0</td>
<td>19.08</td>
<td>67023</td>
<td>3801298</td>
</tr>
<tr>
<td>L.C. 4</td>
<td>Pre-tensioning</td>
<td>80</td>
<td>19.08</td>
<td>67023</td>
<td>3801298</td>
</tr>
<tr>
<td>L.C. 5</td>
<td>Pre-tensioning</td>
<td>80</td>
<td>17.18</td>
<td>67023</td>
<td>3801298</td>
</tr>
<tr>
<td>L.C. 6</td>
<td>Fast Lift &amp; Emptying of QD tanks</td>
<td>100</td>
<td>16.91</td>
<td>67023</td>
<td>3801298</td>
</tr>
<tr>
<td>L.C. 7</td>
<td>Moving Away from Platform</td>
<td>100</td>
<td>11.80</td>
<td>67023</td>
<td>3801298</td>
</tr>
<tr>
<td>L.C. 8</td>
<td>Transit to Shore</td>
<td>100</td>
<td>15.00</td>
<td>40014</td>
<td>2600888</td>
</tr>
<tr>
<td>L.C. 9</td>
<td>Prepare for Off-loading to Cargo Barge</td>
<td>100</td>
<td>24.00</td>
<td>67023</td>
<td>3801298</td>
</tr>
<tr>
<td>L.C. 10</td>
<td>Transfer of Topsides to Barge</td>
<td>0</td>
<td>24.00</td>
<td>67023</td>
<td>3801298</td>
</tr>
<tr>
<td>L.C. 11</td>
<td>Transit</td>
<td>0</td>
<td>12.00</td>
<td>25009</td>
<td>1800615</td>
</tr>
</tbody>
</table>

Table 1.2: Succession of loading conditions required for topsides removal operation

The lifting operation begins with the transit of the PS to the location of the offshore platform to be removed. Prior to the arrival of the PS, the platform will have been prepared for the lift. Necessary preparations include platform cleaning, removal of secondary steel below the

\[1\] In general, loading conditions are to be prepared with approximately zero trim and heel. An exception is made for the fast-lift, where no trim or heel limitations are set. Also, during transit of the vessel to shore, while carrying the topsides, a maximum static trim of \(\pm 0.2\) deg and a maximum static heel of \(\pm 0.3\) deg is prescribed (PS Basis of Design, 2010).
platform, sea-fastening of modules and other loose items for transport, reinforcement of lift points, and separation of the topsides from the supporting substructure (steel jacket or gravity-based structure).

Upon arrival at the platform, the PS must be sufficiently ballasted from its transit draught to a deeper draught so that it will remain clear of the underside of the topsides while manoeuvring into position around the platform (figure 1.5(a)). Then, on DP, the vessel will move-in around the platform foundation, positioning its bows on either side of the supporting structure of the platform, whether this is a steel jacket or gravity-based structure (figure 1.5(b)).

Once the vessel is in position around the platform, it must be appropriately de-ballasted in order to achieve the required “connecting” draught, whenever the latter differs from the draught used to clear the underside of the topsides. The connecting draught is the draught at which the TLS will be attached to the platform. Figure 1.6(a) gives a closer view of the slot of the PS as the vessel is de-ballasting to achieve the connecting draught.

When the connecting draught is attained, the motion compensation systems for surge, sway and heave are activated for the TLS beams. These systems are intended to allow for minimal impact during first contact between the topsides and the beams. Once vessel motions have been neutralised relative to the platform, the beams make contact with the underside (figure 1.6(b)). The combined use of motion compensation and DP may render this part of the operation power-critical.

Once the TLS beams are in contact with the topsides, the pre-tensioning phase of the lift operation begins (figure 1.7(a)). During pre-tensioning, the weight of the topsides is gradually transferred to the TLS. The main cylinders of the beams are pressurised, and the PS is de-ballasted to maintain the same average position during the load transfer. Pre-tensioning is complete when approximately 80% of the topsides load has been transferred to the vessel.
Chapter 1. Introduction

Figure 1.6: (a) Closer view of slot of PS as the vessel is de-ballasting to connecting draught and (b) with lifting beams extended and in contact with underside of platform

Figure 1.7: (a) Pre-tensioning and (b) fast-lift phases of topsides removal operation

Upon completion of pre-tensioning, the PS is ready to perform the fast-lift of the topsides. This is the point of no return for the lift operation. During the fast-lift, the topsides is quickly lifted clear of the substructure. The lift height is sufficient to compensate for vessel sinkage and wave-induced motions, preventing impact with the substructure. The sinkage, or draught increase due to the additional load, will be counterbalanced by emptying quick-drop ballast tanks, located in the forepeak of the PS, by gravity.

Figure 1.8: (a) Move-out phase of operation and (b) offloading of the topsides onto a cargo barge

With the topsides in the highest position, it is possible for the PS to move-out from around the
substructure of the platform (figure 1.8(a)). Once the vessel is at a safe distance, the topsides is secured for transport to shore (sea-fastening), and the vessel is ballasted to a transport draught. Finally, upon arrival at a sheltered destination near shore, the PS offloads the topsides to a cargo barge or dedicated “finger” pier (figure 1.8(b)).

1.2.1 Importance of Ballasting for Lifting Operations

During the topsides lifting operation, it is imperative that the PS progresses through the required sequence of draughts while remaining within an acceptable range of hydrostatic (trim, heel) and structural constraints. The allowable range of hydrostatic constraints is determined by the dimensions of the offshore platform and by the operational requirements of the vessel’s lifting systems. The structural constraints are derived from the ship’s strength for different operational phases. The ballasting system performs a key function in ensuring that the vessel adheres to these functional requirements.

There is also a direct relation between ballasting and the overall duration of the operation. In fact, preliminary analyses have indicated that ballasting time is driving for the operation. Thus, the faster the PS can ballast, or de-ballast, from one draught to the next, the smaller the total time which is required for the lifting operation. In harsh environments, like the North Sea, weather may limit the periods when a lifting operation can go forward. The less time such an operation requires, the more likely it is that the necessary fair-weather window will be found easily.

Furthermore, it is important to minimise the duration of the most critical phases of the lifting operation. For instance, an important transition occurs right after the PS has lifted the topsides and is on DP around the platform. The PS must de-ballast to a shallower draught to be able to safely sail out from around the platform (L.C. 6-7). To reduce the risk of an unwanted occurrence while the PS is precariously sailing out, it is imperative that this transition should take as little time as possible.

1.3 Preparing Ballast Tank Configurations for the PS

Although the ballasting system will be described in depth in chapter 3, a brief description is included in section 1.3.1 in order to motivate the need for a ballast optimisation method. In
section 1.3.2, the procedure for preparing ballast tank configurations by hand is explained, and in section 1.3.3, the current approach for estimating ballasting time is outlined.

1.3.1 Brief Description of the Ballasting System

The ballast system of the PS is comprised of 87 ballast tanks, which vary in size and cover nearly the entire length and breadth of the enormous hull. The system has been designed on a ring main. Isolation valves have been installed so as to be able to separate the system into six subsystems, or zones. Each zone is capable of operating independently of the others.

The total pumping capacity is provided by six pump rooms, with one pump room per zone. Each pump room is equipped with two centrifugal pumps operating in parallel. A drawing of the PS ballast system, including labelling of zones and pump rooms, is included as figure 1.9.

![PS ballast system layout, including overview of zones and pump rooms](image)

Figure 1.9: PS ballast system layout, including overview of zones and pump rooms

The distribution of ballast tanks among the zones is uneven, both with respect to the total volume of water ballast and to the total number of tanks per zone.

A limitation of the system is that a single pump room may only be operating in either ballasting or de-ballasting mode. This means that between two closed valves in the system, all tanks must either be filling or emptying. However, the system still allows for the combination of the whole or a portion of the different zones, and preliminary analyses indicate that such combinations may be used to mitigate the required ballasting time for a given configuration.
1.3.2 Preparing Ballast Configurations by Hand

The procedure of preparing loading conditions is initiated when the PS undertakes a new lifting project. Given the particulars of the offshore platform involved, a sequence of loading conditions is defined for the operation. Each loading condition is characterised by a required draught, an allowable range for trim, heel and structural loading, as well as by the external load exerted on the platform by the weight of the platform. Then, the ballast tank filling levels must be determined for each condition. The loading conditions and filling levels are currently prepared with the aid of Allseas’ Loading Condition Tool (LCT), which has been custom-developed for the PS in Excel.

The LCT is used to prepare and check a single loading condition at a time. Fixed data for the PS, which is the same for each loading condition, pre-exists in the tool. This data includes ship dimensions, tank boundaries and capacities, lightship definition, and lifting system dimensions and weights. Within the tool it is possible to set the position of lifting system components, like TLS beams, and input the topsides or jacket weight and position.

For each loading condition, the naval architect must select a filling level for all ballast tanks. The LCT can then calculate the corresponding hydrostatic, weight distribution, and structural loading data for that condition, among other functionalities, and report on whether the condition meets all constraints. Such constraints include a specified draught, nearly zero trim and heel, as well as sufficient structural integrity, among others. It is therefore possible to prepare a loading condition by trying-out ballast tank filling levels until the desired characteristics are achieved.

As a general rule, ballast tanks are chosen to meet the draught requirement first. They are selected in such a way as to avoid extreme structural loading of the hull and limit trim and heel. The process of finding a satisfactory configuration is an iterative one, enlightened by the naval architect’s intuition and knowledge of hydrostatics and ship strength. Figure 1.10 gives an example of a set of hand-calculated ballast plans prepared for a topsides removal operation.

It can be concluded that the preparation of loading conditions is a complicated and time-intensive task due to the multitude of objectives and constraints associated with the analysis. Characteristically, for the topsides removal operation described in section 1.2, the preparation of suitable loading conditions required approximately six weeks to complete.
1.3.3 Estimating Ballasting Time for Operations

Once a suitable sequence of loading conditions has been defined, the next step is to calculate the corresponding pumping time for each transition between conditions. The naval architects perform additional calculations in Excel to approximate this time, as it is not a functionality included in the LCT.

First, the total water volume transferred between each pair of loading conditions is found based on the difference in tank filling levels. The tanks ballasting, or filling with water, in each transition are considered separately from the tanks de-ballasting, or emptying water. Ballasting and de-ballasting are analysed individually because the pumping system capabilities are slightly different for each phase.

To determine the ballasting time, each tank that is filling must be assigned to a pump room, or zone. Then, the total water volume to be handled by each pump room is known, and the ballasting time for that pump room is calculated given the capacity of its pumps. The same approach is used for the tanks that are emptying to determine the total de-ballasting time.
For the transition from loading condition $n$ to $n+1$, the total pumping time required for the tanks in a zone is the sum of the times required for filling and emptying. The total time needed for the vessel to transition from loading condition $n$ to $n+1$ is the maximum of the zone times required for the transition. Finally, the total time for a given operation is found to be the sum of all transition times.

Table 1.3 gives the ballast times which were calculated by hand for the topsides lifting operation described in section 1.2. The third column of the table gives the pumping time for each transition between loading conditions, based on the individual use of zones (single zone use). The fourth column of the table gives transition times for the same operation when zones are used in combination with one another. The fifth column gives the specific combination of zones for which the ballast time in column four is estimated. When zones are used in combination, the number of pumps being used in parallel increases, and this can, depending on the distribution of tanks being filled, decrease the time required for filling.

Much like preparing a suitable ballast tank configuration, finding the best use of pump rooms to minimise the total ballasting/de-ballasting time for an operation is a prolonged process. All possible zone combinations should be considered to find the lowest time, something which can become especially complicated for de-ballasting. When de-ballasting, whole zones of tanks cannot be emptied in combination. Fewer tanks may be considered at one time, and the number of possible zone combinations increases drastically.

The ballast times estimated in this manner for the topsides removal operation have been deemed unsatisfactory, and this has generated interest in finding a better solution to the ballasting problem.

In summary, figure 1.11 gives a graphical representation of the procedure for preparing loading conditions by hand. Steps 3 and 4 of the figure are highlighted in blue in order to emphasize the phases of the procedure which might be acted upon by an optimisation.

### 1.4 Motivation for Development of Ballast Optimisation Method

Based on the information in the previous sections, it is possible to provide justification for the development of a ballast optimisation method for the PS. It has been established that there is a multitude of constraints for the sequence of loading conditions required for each lifting
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>L.C. 1</td>
<td>12</td>
<td>12.19</td>
<td>11.3</td>
<td>2+4, 5+6</td>
</tr>
<tr>
<td>L.C. 2</td>
<td>22</td>
<td>4.32</td>
<td>3.29</td>
<td>1+3, 2+4, 5+6</td>
</tr>
<tr>
<td>L.C. 3</td>
<td>19.08</td>
<td>3.68</td>
<td>1.66</td>
<td>2+4+6, 1+3</td>
</tr>
<tr>
<td>L.C. 4</td>
<td>19.08</td>
<td>3.35</td>
<td>2.23</td>
<td>1+3, 2+4, 5+6</td>
</tr>
<tr>
<td>L.C. 5</td>
<td>17.18</td>
<td>0.04</td>
<td>0.04</td>
<td>-</td>
</tr>
<tr>
<td>L.C. 6</td>
<td>16.91</td>
<td>10.83</td>
<td>6.41</td>
<td>1+3, 2+4, 5+6</td>
</tr>
<tr>
<td>L.C. 7</td>
<td>11.8</td>
<td>6.35</td>
<td>3.89</td>
<td>1+3, 2+4, 5+6</td>
</tr>
<tr>
<td>L.C. 8</td>
<td>15</td>
<td>13.91</td>
<td>9.61</td>
<td>1+3, 2+4, 5+6</td>
</tr>
<tr>
<td>L.C. 9</td>
<td>24</td>
<td>2.89</td>
<td>1.44</td>
<td>1+3, 2+4, 5+6</td>
</tr>
<tr>
<td>L.C. 10</td>
<td>24</td>
<td>17.82</td>
<td>13.95</td>
<td>1+3, 2+4, 5+6</td>
</tr>
<tr>
<td>L.C. 11</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>75.39</strong></td>
<td><strong>53.82</strong></td>
<td></td>
</tr>
</tbody>
</table>

Table 1.3: Hand-calculated ballast times for topsides removal operation, with single and combined zone use.

Figure 1.11: Procedure for preparing loading conditions by hand (adapted from PS Ballast Guideline, 2014). Phases highlighted in blue may be acted upon by an optimisation.
operation. It is time-consuming to prepare each loading condition by hand, and there will be a continued and increasing need for the procedure in the future, as the PS takes on additional heavy-lift projects.

Furthermore, it is clear that Allseas’ existing loading condition tool is not capable of optimising a sequence of loading conditions for minimum time. The LCT can only consider a single loading condition at a time and perform a go/no-go check of a hand-calculated ballast tank configuration. Even for a single loading condition, the LCT cannot generate a set of ballast tank fillings on its own, but relies on the expertise of the naval architect to iteratively set filling levels for each tank while ensuring compliance with constraints.

The calculation of pumping time is also not a functionality which is included in the LCT. In fact, there is not currently an automated approach for finding the optimal use of pump rooms for minimum total ballasting time. As the pumping system is also a quite complex, with many possible configurations for ballasting and de-ballasting respectively, it is difficult and time-consuming to identify a successful use of pump rooms, and it is not possible to ensure that the pump room usage found is the optimal one.

In the next chapter, the graduation project which was inspired by this problem is presented.
Chapter 2

Problem Definition

Based on the background information provided, it is now possible to define the problem central to this graduation project. In section 2.1, a concise expression of the project’s objective is given and limitations to the scope are described. Supporting questions, of interest in this thesis in combination with the main objective, are listed in section 2.2. Then, in order to situate this project within the existing research, a brief review of relevant literature and tools is included in section 2.3. In conclusion, the approach chosen to realise the project objective is outlined in section 2.4.

2.1 Main Objective

The importance of ballasting to lifting operations, as well as the complexity of preparing loading conditions for the PS, has been expounded on in the previous chapter. A summary of the motivation for the development of a ballast optimisation method was included in section 1.4. Given the general problem identified, it is now possible to more succinctly express the search for its solution. Thus, the objective of this graduation project is:

To develop a ballast optimisation method, to be implemented in a tool, which will facilitate the preparation of ballast configurations for Pieter Schelte lifting operations, with the primary goal of minimising total ballasting time for an operation.

The method is intended as a systematic and automated procedure for determining the ballast tank filling levels which are best suited to achieve target draught, trim and heel and remain
within allowable structural loading envelopes, while minimising total ballasting time. The concept of optimisation is included in the objective since the nature of the problem, as described in sections 1.3 and 1.4, lends itself to such an approach. Namely, ballasting time should be reduced as much as possible, within problem constraints, certainly so as to improve on the results of the hand-calculated loading conditions, if not to find a globally minimal solution.

The objective stated clarifies that the ballast method is intended to facilitate the preparation of ballast configurations. This means that the method is intended to reduce the time and effort required to prepare loading conditions for PS lifting operations. It is emphasised that the goal of this thesis is not to develop a real-time ballast automation tool, but rather an aid for naval architects to use in the planning stage. Since the method is intended to reduce the time required for preparation of loading conditions, an implicit limit is imposed on the computational time required by the optimisation. Given that current preparation time is in the order of weeks, the ballast method, including any solution time for the optimisation, should constitute an improvement on this.

The primary optimisation objective is also stated as minimising total ballasting time for an operation. The optimisation method is meant to consider a sequence of loading conditions simultaneously, such as comprise a lifting operation. The intent is to find such a sequence of ballast tank fillings as will require the least possible pumping time between one another, given the capabilities of the pumping system.

### 2.1.1 Additional Limitations to Scope

In order to fully clarify the aims of the graduation project, it is necessary to define a few additional limitations to the project scope.

First, the objective of the project is to determine the optimal use of the existing ballast system, as it currently installed in the vessel. Although more drastic modifications to the system design, such as increased pumping capacity, might be more effective in reducing the ballasting time, it is considered that such changes are prohibited within the scope of this thesis. Thus, the intent is to find an optimisation method which will improve the time required as much as possible for the existing design.

Also, the optimisation method is intended to produce ballast configurations which require the smallest possible transitional time between one another. The time for the transition is dependent
on the required water volume to be transferred and the effective use of pump rooms to achieve the water transfer. Within the scope of the thesis, operational practices which might be adopted to minimise ballasting time are not investigated, such as for instance simultaneous ballasting and navigation while the vessel is approaching the platform.

Additionally, topsides lifting will be the only operation considered for the current thesis. This operation type is the most ballasting-intensive, due to the multitude of draughts and loading conditions which the vessel must achieve. Furthermore, topsides lifting is the first capability which will be marketed, as the design and construction of the jacket lift system are still being finalised. Consequently, it may be said to be sufficiently representative of the operational requirements on ballasting.

Ship loading characteristics are calculated and updated based on the methodology set out in Allseas’ LCT, including any simplifying assumptions made, such as that forces on the vessel are assumed to be linearly distributed. In general, numerical integration is used for calculations rather than a “first-principles” approach (three-dimensional geometrical model), based on the prohibitive computational cost of including the latter in an optimisation with many function evaluations.

It is noted that the optimisation method is intended to provide a global solution for the main loading conditions of the lifting operation. It will not address the intermediate filling steps between loading conditions, or specify the order in which tanks should be ballasted to ensure that constraints are continuously met. A particular use of pumps may be suggested to minimise the total pumping time for a transition, but the total time estimation does not take intermediate structural integrity or hydrostatics into account. The only consideration during the time estimation is the concurrent or sequential use of pump rooms. Thus, while the suggested use of pumps may be the best possible in terms of time, no guarantee can be made, within the scope of this thesis, as to the feasibility of the transitional conditions.

Finally, the method developed does not include a simulation of the water flow within the ballast system piping for each transition. Rather, the nominal pumping capacity of the pumps is assumed for any time calculations, based on relevant analyses ensuring that such a capacity is, in general, feasible (Fisher, 2007; PS WB Head Calculation, 2014), and on the current practice within Allseas. Further discussion of the pumping capacity is included in appendix C.
Details of specific constraints which affect the problem modelling are described in depth in section 4.1.2.3.

2.2 Supporting Research Questions

A number of research questions are considered in addition to the main objective. These questions serve both to breakdown the thought process into clearer steps and to highlight elements of the problem which are of interest in a general sense, not only in narrow pursuit of the stated objective. These supplementary questions are as follows:

1. Which optimisation approach is best-suited to the ballasting problem?

2. How should functional requirements and design specifications of the existing system be modelled as boundary conditions or constraints?

3. How should the problem be modelled in order to find a set of N acceptable (not necessarily optimal) ballast configurations for N different loading conditions, which combined give optimal ballasting duration for an operation?

4. How can the ballast optimisation method be developed in such a way as to allow for general use or parametrisation?

2.3 Existing Literature and Tools

In this section, a brief overview is given of existing literature and software related to the ballast optimisation problem. Initially, in section 2.3.1, published papers relevant to the current project are reviewed. The ship loading software, or loadmasters, available on the market are described in section 2.3.2. Finally, in section 2.3.3, the key differences between such software and the ballast optimisation method under development are discussed.

2.3.1 Ballast Optimisation Methods

Though optimisation methods have been extensively applied in ship design and construction, little has been done thus far in the field of ballast plan optimisation. Here, the term ballast plan
Optimisation refers to research seeking a method of finding the best ballast tank configurations to optimise some aspect of ship characteristics or performance (i.e. hydrostatics, structural loading, etc.). Based on a survey of related literature, a few pertinent applications are summarized in this section.

Much of the ballast optimisation literature reviewed was related to barges utilised for load-out operations. This is because the type of ballasting problem addressed in such literature is similar to a simplified version of PS ballasting requirements for lifting operations. Namely, the barge moves sequentially through a series of loading conditions, while necessarily maintaining a required floating condition and strength. In a few cases, maximising the efficiency of ballasting operations is also an objective of the analysis, much as it is in the case of the PS.

Kurniawan and Ma (2008) developed an optimisation model for a ballast plan of a barge during jacket load-out. Their goal was to find a more accurate ballast plan for jacket load-out by taking the flexibility of the barge into account. Kurniawan and Ma used a multi-objective evolutionary algorithm (MOEA) to find the optimum ballast arrangement at every load-out stage, with the objectives of minimising deflection and curvature of the load-out configuration (modelled as a beam on elastic foundation) and maximising ballast transfer efficiency between load-out stages. Their modelling of ballast transfer efficiency, in terms of the water volume taken on, removed or moving within the ballast tank system for successive load-out stages, was of particular interest for this graduation project. Kurniawan and Ma concluded that the proposed method constituted an improvement on the conventional rigid barge method in terms of minimizing the deflection and curvature and maximizing ballast transfer efficiency.

For ocean-going vessels, ballast optimisation is often pursued as a form of automation, which will reduce human effort in loading condition preparation, or as a method of improving operational performance. With a view to the former, Chen et al (2009) developed a method for ballast plan preparation for ocean-going cargo ships using multi-objective optimisation with genetic algorithms (GAs). Their objectives were to minimise overall longitudinal still water bending moments and shear forces, while maximising intact stability performance. Chen et al built a mathematical model for automatic ballast plan preparation and applied a novel heuristic dynamic adjusting rule to their GA optimisation of the model. The loading conditions produced by the model were shown to have better longitudinal strength and intact stability performances than those prepared by the naval architect in the same time frame.
In terms of improving operational performance, ship loading software companies are increasingly investigating the area of trim optimisation through ballasting. The goal of trim optimisation is to sail at such a trim as will result in minimal hull resistance for a given draught and, consequently, in increased fuel savings for the vessel. The optimal trim required at a given draught is determined by towing tests or computer simulation. Then, the optimal ballast configuration required to achieve the desired trim, while meeting safety and stability requirements, is identified. Well-established software with this capability is that of Herbert-ABS and Interschalt (Fathom Shipping, 2014).

2.3.2 Ship Loading Software

In order to get an understanding of the capabilities of existing ship loading tools, a review of commercially-available loading software or “loadmasters” was conducted. DNV Rules define a loading computer as a “computer-based system for calculation and control of loading conditions for compliance with the applicable stability requirements and longitudinal and local strength requirements.” Loading computers are considered supplementary to the vessel’s loading manual and stability booklet.

An essential capability of loadmasters is to check a loading condition for adherence to hydrostatic, stability and structural loading requirements. The software packages differ with respect to the way such calculations are carried out, either by being approximated from stored tables of ship data (Totem Plus, Segula), or by a “first principles” approach using a three-dimensional model of the vessel (Autoship, SARC, Kongsberg). The loading computers are usually intended for use by crew on board the vessel, often prior to or during ship loading, and are sometimes able to check the condition of the vessel in real-time assuming appropriate sensors are installed (Interschalt, Herbert-ABS).

In terms of optimisation capabilities, several software packages are moving in the direction of trim and draught optimisation for improved fuel efficiency (Interschalt, Herbert-ABS, Kongsberg). In other words, for a given loading condition the software identifies the best ballasting arrangement in order to sail at a desired trim. The ideal trim, which minimises hull resistance for a given loading, is typically determined beforehand, either through towing tests or simulations. For offshore rigs and semisubmersibles, some software packages also output ballast tank fillings for a target draught, trim, and heel, by modifying a set of ballast tanks pre-selected by the user (Autoship).
2.3.3 Conventional ‘Loadmasters’ vs the Intended Ballast Optimisation Method

Based on the survey of commercially available ship loading software, it was concluded that the majority of these applications essentially provide a check of a single loading condition for trim, heel and longitudinal strength. They assist in concluding whether or not a given tank filling configuration is acceptable, but most are unable to generate optimal configurations based on target draught and structural limitations. A few loading software packages do provide some automated ballast optimisation capability, usually with the aim of reducing hull resistance, or for a small subset of ballast tanks which are selected by the user.

However, the main difference between the intended ballast optimisation method and conventional loading software centres on the inclusion of the duration element. Traditional loadmasters are usually not concerned with minimising the ballasting time, since conventional merchant vessels do not expect such drastic changes in loading condition (external loading, draught) during operations. The PS’s ballasting requirements are unique, both due to the vessel’s size and its operational profile. The time element requires the interpretation of pumping system parameters into the optimisation as constraints, contributing an additional layer of complexity to the development.

Furthermore, the consideration of an entire operation also sets the required scheme apart from conventional loading software. Even the few loading computers which include some manner of ballast optimisation, only do so for a single loading condition.

2.4 Approach

It is now possible to outline the approach taken towards realising the stated objective. The first step was to gather relevant information and data about the ballasting problem, regarding for instance the operational requirements for ballasting and the capabilities of the ballast pumping system.

Then, the modelling of the problem was considered, both in terms of mathematical representation and in terms of defining the structure of the optimisation. Key elements of the problem, such as design variables, constraints, and objectives, were quantitatively defined. Based on a literature review of possible choices, genetic algorithms were chosen to conduct the search.
With the basic structure and modelling of the problem in place, a computer-based procedure was developed to produce ballast configurations. The optimisation was implemented in MATLAB, using a non-dominated sorting genetic algorithm (NSGA-II).

Once the implementation was completed, a test case consisting of three loading conditions was selected to evaluate the efficacy of the modelling and gauge the method’s success in meeting the project’s main objective.

The approach described is depicted visually in figure 2.1.

The definition of the problem of interest and relevant data is covered in chapters 1-3. The mathematical modelling and programming implementation are discussed in chapter 4. The results produced by the model are presented and evaluated in chapter 5. Finally, the conclusions and recommendations of the research are included in chapter 6.
Chapter 3

Ballasting System of the PS

The ballasting system of the PS is central to this graduation project. Consequently, the current chapter is devoted to a more in-depth description of the system’s components (section 3.1), ballasting and de-ballasting operations (section 3.2), and the calculation of pumping time (section 3.3).

3.1 Description of the Ballasting System

As mentioned briefly in section 1.3.1, the many ballast tanks of the PS span the vessel’s length and breadth and are supplied with seawater by a dedicated pumping system. The system may be segregated into zones, which are capable of operating independently. A more detailed description of the system’s key components follows.

3.1.1 Ballast Tanks

The ballast system is comprised of 87 ballast tanks with a total capacity of approximately 759 274 t (740 755 m$^3$). The tanks account for the majority of the volume of the hull, extending from the bottom shell to to the main deck (PS WB System FMEA, 2010). The boundaries of the tanks are generally formed by the ship’s subdivision (10-12 frames) and main longitudinal bulkheads (PS WB System FMEA, 2010). Local variations in the length and height of ballast tanks depend on the boundaries of adjacent spaces like pump rooms or fuel tanks (PS WB
System FMEA, 2010). To give an idea of their relative sizes, the maximum dimensions of the PS’s ballast tanks are plotted in appendix E.

In figure 3.1, the longitudinal, transverse and vertical centres of gravity are plotted for each ballast tank at 100% filling. The figure is intended to give an overview of the spatial orientation of the vessel’s ballast tanks.

![Figure 3.1: Ballast tank centre of gravity at 100% filling](image)

Within Allseas documentation, the ballast tanks of the PS are depicted in a simplified way. Figure 3.2 gives a schematic representation of the ballast tanks taken from Allseas’ LCT, including tank numbering. All tank elevations are shown in one view. Figure 3.3 shows the same schematic, with a colour-scale representing the relative capacity of each tank at 100% filling level. All tanks are connected to the main line through a pipe with an isolation valve.

### 3.1.1.1 Quick-Drop Tanks

During the fast-lift phase of the lifting operation, when the last 20% of the topsides load is rapidly lifted by the TLS, the ballast system cannot react quickly enough to the extra weight placed on the fore part of the ship. Instead, four of the 87 ballast tanks are specially-designed as “quick-drop” (Q/D) tanks (tanks 013, 014, 015, 016). Two such tanks are located in the
forepeak of each bow of the PS and are intended to be emptied by gravity during the fast-lift, for rapid de-ballasting. The emptying of the Q/D tanks is expected to compensate for the sinkage of the fore part of the vessel due to the added load.

The Q/D tanks are filled by the ballasting system at the start of the lifting operation. After that, the tanks are isolated from the rest of the system until they are emptied by gravity during the fast-lift.

### 3.1.2 Pumping System

The total pumping capacity for the ballast system is provided by six pump rooms. All pump rooms are equipped with two centrifugal pumps, connected in parallel. Each pump has a nominal capacity of $3100 \frac{m^3}{h}$ and a power consumption of 490 kW.

Water enters and leaves the vessel from high and low sea inlets, through a seawater crossover and an overboard discharge below the waterline. The pump rooms are connected to the ring main through suction/discharge connections. The two ballast pumps in each pump room operate in
parallel across these connections. The ballast water treatment plant will be situated at their discharge.

Each pump has isolation valves placed at its suction and discharge, and a non-return valve at the discharge. Figure 3.4 shows the layout of the aft starboard pump room.

![Pump Room Layout](image)

**Figure 3.4:** Layout of aft starboard pump room of PS

Pump rooms are generally located in the lower part of the hull, as is piping for the water ballast system (PS WB System FMEA, 2010). Figure 3.5 is a screen print from the Tribon ship model, showing nearly the full extent of the ballast system (PS WB System FMEA, 2010).

### 3.1.2.1 Stripping System

A ballast stripping system is also installed to supplement the main ballast system. The stripping system removes the residual water from tanks that is beyond the suction capabilities of the main system, namely, the last 1 m of water in the tank. The vessel’s six bilge pumps are used for stripping, with two pumps able to act on each ballast tank. Each bilge pump has a nominal capacity of $300 \text{ m}^3/\text{h}$.

The stripping system has a much lower capacity than the main system, meaning the time to remove the last bit of water in each tank can be significant. For the purposes of the ballast optimisation, it is assumed that once a tank has been filled above 1 m of water, it will retain this water in all successive loading conditions.

Figure 3.6 gives the minimum filling percentages corresponding to the 1 m of residual water that must be removed by the ballast stripping system.
3.1.3 Zones

The water ballast system is capable of being subdivided into six, independent zones. The zones are interconnected by the ring main and may be separated from one another through segregation valves on the main line. When the segregation valves are closed, the tanks in a zone are only operated on by their respective pump room.

A simplified representation of the ballast system is included as Figure 3.7. In this representation,
all ballast tanks are coloured to indicate the zone they belong to, approximate pump room locations are shown (numbers in boxes), and the approximate ring main location is indicated with a bold black line. Yellow valve symbols cutting the main line show the positions of isolation valves used to separate the zones from one another.

The distribution of ballast tanks among the zones is uneven, both with respect to the total volume of water ballast and to the total number of tanks per zone. Table 3.1 gives an overview of ballast tank distribution among zones.

<table>
<thead>
<tr>
<th>Zone</th>
<th>No. of Tanks</th>
<th>Total Volume of Tanks [m$^3$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14</td>
<td>115675</td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>115658</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>121527</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td>86576</td>
</tr>
<tr>
<td>5</td>
<td>14</td>
<td>115591</td>
</tr>
<tr>
<td>6</td>
<td>23</td>
<td>185706</td>
</tr>
</tbody>
</table>

Table 3.1: Distribution of ballast tanks among PS zones

The assignment of a large number of tanks to zone 6 is due to the location of the PS’s main engine room above the tanks of the forward part of the zone (approximately above tanks 702, 704, 802, 804). In the vicinity of the engine room, the ballast tanks are smaller than the surrounding tanks as they are located beneath the hull volume of the engine room.
3.2 Ballasting Operations

3.2.1 Ballasting

During ballasting operations, tank branch valves are opened for all tanks needing to be filled. Segregation valves are also opened in the main line, depending on whether pumping is done using single zones or a combination of zones. Pumps run at their nominal capacity, and tank branch valves are closed per tank when the tank has filled to the required level. Figure 3.8(a) shows the direction of flow and valve settings in the aft starboard pump room during a typical ballasting operation.

In some instances, when tank levels would be below sea level, it was hypothesised that it might be possible to ballast tanks by gravity, bypassing the pumps entirely. Figure 3.8(b) illustrates the direction of flow and valve settings in the aft starboard pump room when ballasting by gravity.

Ballasting by gravity was attempted during the PS sea trials in Korea, and it was found to result in higher flow rates, meaning faster ballasting. Table 3.2 compares nominal ballasting capacity using pumps to ballasting capacity achieved using gravity during sea trials. The ballasting capacity using gravity was estimated based on the measured draught change for this operation during sea trials.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Ballasting Capacity $[\text{m}^3/\text{h}]$</th>
<th>Draught Change Capacity $[\text{m}/\text{h}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ballasting by pumps</td>
<td>37.200</td>
<td>0.65</td>
</tr>
<tr>
<td>Ballasting by gravity</td>
<td>79.400</td>
<td>1.67</td>
</tr>
</tbody>
</table>

Table 3.2: Comparison of ballasting capacity by pumps and by gravity (PS Ballast Guideline, 2014)

Despite the increased capacity possible using gravity, this type of operation has also been deemed more difficult to control, making it inherently riskier. Therefore, for lifting operations and calculations, ballasting by pumps is recommended.

When ballasting with the PS, it was originally assumed within Allseas that it would be possible to combine up to two whole zones at a time. The combination of zones is particularly worthwhile when the tanks to be filled are not evenly distributed among the various zones. For instance, a
zone with numerous tanks filling could be helped by a neighbouring zone with very few tanks filling.

The feasible two-zone combinations are dependent on the proximity of the zones and the location of segregation valves in the main line. In figure 3.9, arrows show which zones may be combined with each other, giving an overview of the feasible two-zone combinations for ballasting. Table 3.3 summarises the information from the figure.

Analyses of the pipework of the ballast system, conducted within Allseas and by third-parties, have determined that it is reasonable to assume nominal pump capacities for preliminary calculations when using a combination of zones (Fisher, 2007; PS WB Head Calculation, 2014). Friction losses in pipework do not significantly reduce the effective pumping capacity for the feasible zone combinations in figure 3.9.
### Table 3.3: Feasible pump room combinations per tank subgroup for ballasting

<table>
<thead>
<tr>
<th>Zone</th>
<th>Feasible Pump Room Combinations for Ballasting</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1, 1&amp;3, 1&amp;4</td>
</tr>
<tr>
<td>2</td>
<td>2, 2&amp;4, 2&amp;6</td>
</tr>
<tr>
<td>3</td>
<td>3, 1&amp;3, 3&amp;4, 3&amp;5</td>
</tr>
<tr>
<td>4</td>
<td>4, 1&amp;4, 2&amp;4, 3&amp;4</td>
</tr>
<tr>
<td>5</td>
<td>5, 3&amp;5, 5&amp;6</td>
</tr>
<tr>
<td>6</td>
<td>6, 2&amp;6, 5&amp;6</td>
</tr>
</tbody>
</table>

3.2.2 De-ballasting

During de-ballasting, the pumps empty ballast water from tanks and discharge the water overboard. Figure 3.10 shows the direction of flow and valve settings in the aft starboard pump room during a typical de-ballasting operation.

![Figure 3.10: Direction of flow and valve settings in aft starboard pump room during ballasting](PS Ballast Guideline, 2014)

De-ballasting operations differ from ballasting, in that it is generally not possible to combine zones with one another in their entirety. Restrictions are placed on which tanks a given pump room can de-ballast given the tanks’ proximity and the location of segregation valves in the
main line. It is possible to divide the tanks into sub-groups, or sub-zones, based on the pump rooms which may be used to de-ballast them.

Figure 3.11 shows the subdivision of ballast tanks into subgroups for de-ballasting. The tanks marked with a cross may only be de-ballasted using their own zone’s pumps. Table 3.4 summarises the pump rooms which may be used to de-ballast each subgroup.

For the feasible pump room combinations for de-ballasting, analyses by Allseas have shown that frictional losses in pipeworks will not significantly affect the pumping capacity. Thus, as was the case in ballasting, the nominal pumping capacity is used in calculations.

### 3.3 Calculation of Pumping Time

Given the capabilities for ballasting and de-ballasting with the six pump rooms of the PS, it is now possible to describe the way pumping time is calculated for each transition between loading conditions, and, ultimately, for the entire lifting operation.

Between two loading conditions, the ballasting and de-ballasting phases are considered sequentially. First, all of the tanks filling are identified. For each tank filling, the volume of water to be added to the tank is calculated from the difference in percent filling levels between the two conditions and the total capacity of the tank.

Then, given a total water volume to be added to each tank, a decision must be made as to which pump room or combination of pump rooms to use to fill it. Assuming that the ballast zones of the PS are used independently, meaning all zone segregation valves are closed in the main line, the total water volume to be added to all tanks is summed per zone. The total water
<table>
<thead>
<tr>
<th>Subgroup Number</th>
<th>Feasible Pump Room Combinations for De-ballasting</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>1</td>
</tr>
<tr>
<td>1b</td>
<td>1, 3, 1&amp;3</td>
</tr>
<tr>
<td>1c</td>
<td>1, 4, 1&amp;4</td>
</tr>
<tr>
<td>2a</td>
<td>2</td>
</tr>
<tr>
<td>2b</td>
<td>2, 4, 2&amp;4</td>
</tr>
<tr>
<td>2c</td>
<td>2, 6, 2&amp;6</td>
</tr>
<tr>
<td>3a</td>
<td>3, 5, 3&amp;5</td>
</tr>
<tr>
<td>3b</td>
<td>3, 1, 4, 1&amp;3, 3&amp;4</td>
</tr>
<tr>
<td>4a</td>
<td>4, 1, 3, 1&amp;4, 3&amp;4</td>
</tr>
<tr>
<td>4b</td>
<td>4</td>
</tr>
<tr>
<td>4c</td>
<td>4, 2, 2&amp;4</td>
</tr>
<tr>
<td>5a</td>
<td>5, 6, 5&amp;6</td>
</tr>
<tr>
<td>5b</td>
<td>5, 3, 3&amp;5</td>
</tr>
<tr>
<td>6a</td>
<td>6, 2, 2&amp;6</td>
</tr>
<tr>
<td>6b</td>
<td>6</td>
</tr>
<tr>
<td>6c</td>
<td>5, 6, 5&amp;6</td>
</tr>
</tbody>
</table>

Table 3.4: Feasible pump room combinations per tank subgroup for de-ballasting

volume per zone is divided by the total pumping capacity of the zone to give a pumping time, as in equation 3.1. For \( k = 1, 2, \ldots \) tnks, \( j = 1, 2, \ldots \) lc – 1, and \( z = 1, 2, \ldots 6 \):

\[
\text{ballast time}_z = \frac{\sum_{k | x_{kj+1} \geq x_{kj}} (x_{kj+1} - x_{kj}) \cdot v_k}{\text{pump capacity}_z} \tag{3.1}
\]

where \( x_{ij} \) is the ballast tank filling ratio for tank \( i \) and loading condition \( j \), tnks and lc are the total number of tanks and loading conditions, respectively, \( v_i \) is the volume of ballast tank \( i \) in \( m^3 \), and \( \text{pump capacity}_z \) is the pumping capacity of zone \( z \) in \( \frac{m^3}{h} \).

All six zones are pumping simultaneously. Thus, the total ballasting time for the given transition (from \( lc_j \) to \( lc_{j+1} \)) will be the maximum pumping time required by any of the zones, expressed
in equation 3.2. For \( j = 1, 2, \ldots, lc - 1 \) and \( z = 1, 2, \ldots, 6 \):

\[
ballast\ time_{lc,j,j+1} = \max_z[ballast\ time_z]
\]  

(3.2)

The de-ballasting time for the transition between loading conditions is calculated in the same manner. The total pumping time for the transition is the sum of the time required for ballasting and de-ballasting (equation 3.3). For \( j = 1, 2, \ldots, lc - 1 \):

\[
pump\ time_{lc,j,j+1} = ballast\ time_{lc,j,j+1} + deballast\ time_{lc,j,j+1}
\]  

(3.3)

Finally, the total pumping time for the operation is found as the sum of the pumping times for all transitions between loading conditions (equation 3.4). For \( j = 1, 2, \ldots, lc - 1 \):

\[
ballast\ time_{oper} = \sum_j pump\ time_{lc,j,j+1}
\]  

(3.4)

As explained in section 3.2, pump rooms are not limited to acting on the tanks within their zone. When they are used to act on tanks of a neighbouring zone, or in combination with another pump room, the calculation of pumping time is slightly different. Again, ballasting and de-ballasting are considered sequentially for each transition between loading conditions. However, instead of grouping all of the tanks belonging to a zone, the tanks being acted on by the same pump room, or combination of pump rooms, are considered together.

For example, assuming that the tanks in zones 1 and 3 are grouped for ballasting, then the pumping time will be calculated for the total volume of water filling in the group and for twice the pumping capacity of a single pump room. The total time for ballasting will then be considered for zones 1&3, 2, 4, 5, and 6, with the maximum group/zone time being leading for the transition.

Figure 3.12 gives an example of the de-ballasting calculation. First, a decision is made as to which pump room(s) to use to de-ballast each subgroup. Based on the assignment, the pumping time for each subgroup is calculated. When two pump rooms are used in combination to de-ballast a subgroup, the pumping time calculation considers twice the single pump room capacity.

Then, given the pumping times and pump room assignments of each subgroup, the total pumping time for each pump room is calculated. The total time that a given pump room is pumping is
Chapter 3. Ballasting System of the PS

Figure 3.12: Example of de-ballasting time calculation, starting with assignment of pump rooms to subgroups and steps for calculation

the sum of the subgroup pumping times acted on by it. For instance, pump room 1 must empty subgroups 1a, 1b, 1c and 3b. An essential concept for the pumping time calculation is that the following calculations are equivalent:

- Find pumping time for pump room 1 given total volume of subgroups 1a, 1b, 1c, 3b taken together and pumping capacity of pump room 1
- Find pumping time for pump room 1 given respective pumping times of subgroups 1a, 1b, 1c, and 3b considered separately, calculated by dividing each subgroup’s volume by the pumping capacity of pump room 1

When pump rooms are used in combination, the pumping time of the subgroups emptied must be added to the total pumping time of each pump room in the combination. For instance, in figure 3.12, subgroup 3a is emptied by pump rooms 3 and 5 simultaneously. The pumping time of subgroup 3a is added to the total de-ballasting time of both pump room 3 and 5. The reason for this is that both pump rooms are occupied while this subgroup is de-ballasting.

Finally, the maximum total de-ballasting time over all the pump rooms is leading for the transition. In the example given in figure 3.12, pump room 5 is leading for the transition.

The pumping time must be calculated in this fashion for each transition between loading conditions. The current approach within Allseas is to assign the ballast tanks to a pump room by
hand and complete any necessary time calculations in Excel spreadsheets. This is understand-
ably a time-consuming process, and finding the correct assignment of the many subgroups for
minimum time is non-intuitive. Each time loading conditions are changed, even slightly, the
calculation must be re-evaluated.

As part of this thesis, MATLAB scripts were developed by the author to automate the search
for an optimal pumping configuration for minimum time. The scripts are described in further
detail in section 4.2.4.

Since the start-up time for the ballast system and the time required for valve changes is consid-
erably smaller in magnitude than the time required for pumping (PS Ballast Guideline, 2014),
it was assumed negligible in the scope of the larger time optimisation and not considered in this
thesis.
Chapter 4

Method

In this chapter, the method chosen for implementing the ballasting problem as an optimisation is described. Section 4.1 gives an overview of the mathematical modelling of the problem, beginning with a generic description of the structure of an optimisation, and continuing with a discussion of ballast-specific optimisation elements. The choice of design variables, modelling of problem constraints and objectives, and selection of search algorithm are some of the essential elements covered. Then, in section 4.2 the implementation of the algorithm in the MATLAB numerical computing environment is outlined. A discussion of the influence of GA parameters on the performance of the algorithms is included in section 4.3. Finally, in section 4.4, justification is given for the decision made in the course of this research to switch from MATLAB’s GA to NSGA-II.

4.1 Modelling

4.1.1 The Structure of an Optimisation

In section 1.3, the determination of ballast configurations for lifting operations was established as a complex, iterative process with many constraints. The characteristics of this process rendered it ideally suited to an optimisation, wherein the best compromise between a number of potentially conflicting objectives would be sought. The implementation of the optimisation in a numerical computing environment was also proposed to improve the number of iterations performed and the multitude of constraints and objectives simultaneously considered, compared to what would be possible for the naval architect in the same period of time (Van Oers, 2011).
Figure 4.1 gives an overview of the structure of an optimisation, indicating critical components. These components are briefly described in this section and, more extensively with respect to the ballasting problem, in section 4.1.2.

**Objectives** An objective is a quantity that should be improved during the course of the optimisation, either by being minimised or maximised using the search algorithm. Van Oers (2011) describes an objective as lacking a threshold, since the optimisation aims to refine the objective until no further improvement is possible. Objectives are usually used to model soft constraints or negotiable requirements, which it would be good, but not essential, to satisfy.

Depending on the number of objectives which are simultaneously optimised, the optimisation can be classified as single- or multi-objective. The distinction between the two is relevant in choosing the search algorithm which will be utilised, as well as in formulating the fitness and objective functions.

When a problem involves multiple objectives, there are two potential approaches which are typically utilised, scalarisation or vector optimisation (Marler and Arora, 2004). Scalarisation is the more classical approach, wherein the components of a vector of objective functions are combined to form a single objective (scalar). Vector optimisation, on the other hand, is characterised by independent treatment of objective functions.

The essence of the classical approach to multi-objective optimisation is that a number of objectives are converted into a single objective. Many methods exist for this conversion, with one of the most popular being the weighted-sum approach. However, the solution of the equivalent single-optimisation problem is conducted with a standard single-objective search algorithm and is often very dependent on the parameters used for the conversion (Deb and Goel, 2002). Furthermore, with a classical optimisation method, only one solution can be found per simulation run.
When multiple objectives are considered simultaneously, there may not be a single solution that is best with respect to all objectives. Therefore, a set of solutions comprise the Pareto-optimal front. For Pareto-optimal, or non-dominated, solutions no improvement is possible in any objective without worsening the value of another objective (Sivanandam and Deepa, 2008).

If a scalarising approach is chosen, the optimisation will need to be run many times in order to produce a set of trade-off solutions, or Pareto-optimal front. A vector optimisation approach, on the other hand, would produce a Pareto front in a single run.

**Objective Function**  The objectives of the optimisation are calculated through one or more objective functions. An objective function relates a given design variable or set of variables (solution) to an objective value. The form of the objective function is key in determining which type of search algorithms may be used with a problem.

**Design Variables**  Decision alternatives for the problem are described using a set of design variables (Rothlauf, 2011). Design variables are independent decision variables which are created by the designer of the optimisation and acted upon by the search algorithm. The designer will decide their number, and may assign each of them a value, type (discrete, continuous, combinatorial), and allowable range (Van Oers, 2011). During the optimisation, the search algorithm will continuously alter the values of the design variables in an effort to find the combination of variables, or solution, which will result in optimal objective values.

A distinction is made between optimisations with discrete and continuous design variables. Discrete optimisation may be distinguished as the case where variables only take on discrete, typically integer, values (Gould, 2006). In discrete optimisation, all possible solutions comprise a finite set. Thus, the best feasible solution is known to be one of the solutions of the set. Because of the finiteness of the solution set, exhaustive or explicit enumeration approaches tend to be favoured (Hurkens, 1999). These approaches consist of generating all feasible solutions, evaluating their fitness, and then selecting the best one. However, exhaustive or explicit enumeration approaches can be very time-consuming ways to solve discrete optimisation problems, and specific algorithms are usually required to reduce optimisation time (Hurkens, 1999). In continuous optimisation, on the other hand, variables are allowed to take on any values permitted by the constraints (Gould, 2006) This flexibility results in an increase in the variety of algorithms which may be applied.
Constraints The constraints of the optimisation reflect non-negotiable requirements for the problem, which limit the solutions generated by the search algorithm to feasible ones (Abt, Harries, and Hochkirch, 2004). Van Oers (2011) suggests that constraints express characteristics which are to be safeguarded during the search process, to ensure that they lie within the bounds defined by the designer.

Search Algorithm The search algorithm uses the values of objectives and constraints to generate a new set of design variables. The algorithm is comprised of defined, mathematical rules used to conduct the search process systematically. A multitude of search algorithms exist, with varying levels of complexity, and they are utilised based on the characteristics of the variables, objectives and constraints which have been developed for a given problem. Search algorithms are discussed more extensively in section 4.1.2.4.

Fitness Function The fitness function measures the quality of the solutions generated by the search algorithm and is used to convert an objective function value into a measure of relative fitness or suitability (Chipperfield, 1997).

The names of the objective and fitness functions are sometimes used interchangeably, and it is indeed possible that the two are equivalent. For instance, in a maximisation problem, the solution with the largest objective function value is also the fittest. In minimisation problems, however, the mapping of the objective function to a relative fitness through the fitness function is necessary, as lower objective function values correspond to fitter individuals (Chipperfield, 1997).

4.1.2 Modelling the Ballasting Problem as an Optimisation

Based on the brief discussion of the main elements of an optimisation, it is now possible to describe the modelling of the ballasting problem as an optimisation.

4.1.2.1 Design Variables

The first step in the modelling was to select the design variable which would be used to affect change to each ballasting configuration. The selection was made based on the practical problem of preparing loading conditions and ballast plans for the PS. When preparing a loading condition,
the naval architect affects changes to the attitude and stressing of the vessel by altering the amount of water in each ballast tank as required. Consequently, the same design variable was chosen for the optimisation.

The independent design variable for the ballasting problem was defined as an array, $x_{ij}$, with a column of ballast tank fillings for each loading condition considered. The index $i$ was chosen to represent the ballast tank of interest ($i = 1, 2, 3, ..., tnks$) and the index $j$ to represent the loading condition ($j = 1, 2, 3, ..., lc$). The variable $tnks$ is the total number of ballast tanks in the PS and equals 83. The variable $lc$ refers to the total number of loading conditions considered simultaneously.

In the final version of the optimisation, the design variable was defined as discrete, with 25% filling increments used for each ballast tank. In section 4.3, the impact of the discretisation of the design variable is discussed, as is the reasoning behind the selection of the aforementioned increment-size.

**Bounds** The bounds of $x_{ij}$ were based on the physical filling constraints of the ballast tanks, with a minimum filling of 0% corresponding to a lower bound of 0 and a maximum of 100% corresponding to an upper bound of 1. Thus, $0 \leq x_{ij} \leq 1 \ \forall i, j$.

Furthermore, as was mentioned in section 3.1.2.1, the ballast stripping system, which removes the last 1 m of water from each tank, is assumed idle for time-sensitive operations because of its much lower pumping capacity. The stripping system constraint was modelled by altering the expression of the lower bound. Namely, for $i = 1, 2, ..., tnks$ and $j > 1$:

- if $0 \leq x_{ij} \leq \text{minimum filling level}_i$, then $x_{ij+1} \geq x_{ij}$
- if $x_{ij} \geq \text{minimum filling level}_i$, then $x_{ij+1} \geq \text{minimum filling level}_i$

**4.1.2.2 Objectives**

As stated in the problem definition, the primary objective of the optimisation method was to minimise ballasting time for a topsides lifting operation. However, in addition to this objective, two additional objectives were also modelled, based on the preliminary performance of the optimisation. Specifically, the draught and trim requirements for each loading condition were
incorporated as objectives rather than constraints. The following paragraphs elaborate on the modelling of the three optimisation objectives.

**Minimise ballasting time**  The duration of ballasting during lifting operations was initially considered in terms of two essential questions: (1) how much water must be transferred between loading conditions? (2) based on the utilisation of ballast zones and pumping capacity, how much time does it take for the necessary water to be transferred?

Plainly put, if the amount of water transferred between loading conditions decreases, for a given pump utilisation, then the total ballasting time required decreases. If for a given amount of water transferred, the speed with which the ballast is transferred to the proper tanks is increased, either by a more efficient combination of zones or by increased pump capacity, then the total ballasting time also decreases. Thus, it was hypothesised that the optimisation of ballasting time could be subdivided into two problems, (i) a search for minimal water volume transfer between loading conditions and (ii) a search for optimal pump utilisation for maximum transfer speed.

**(i) Minimise water volume transfer**  Water volume transfer was estimated based on an expression from Kurniawan and Ma (2009). To minimise the water volume transferred between loading conditions, the ballast added to, removed from, or shifted internally within the vessel should be reduced as much as possible. This is expressed mathematically in equation 4.1.

\[
f_1 = \begin{cases} 
\sum_j \left[ \sum_i (x_{ij+1} - x_{ij}) \cdot v_i + \sum_k [x_{kj+1} \leq s_{kj}] (x_{kj} - x_{kj+1}) \cdot v_k \right], & \text{if } \sum_i x_{ij+1} > \sum_i x_{ij}, \\
\sum_j \left[ \sum_i (x_{ij} - x_{ij+1}) \cdot v_i + \sum_k [x_{kj} \geq s_{kj+1}] (x_{kj+1} - x_{kj}) \cdot v_k \right], & \text{if } \sum_i x_{ij+1} \leq \sum_i x_{ij}.
\end{cases}
\]

(4.1)

where \(v_i\) represents the volume of ballast tank \(i\) in \(m^3\). A lower limit to the water volume transfer was established as the volume required for draught change, based on the vessel hydrostatics.

The concept of minimising water transfer was implemented in preliminary versions of the optimisation, but left out of the final model. It was determined that sufficiently good results could be obtained by focusing on minimising pumping time, as described in the next paragraphs. It was also determined that minimising water transfer in itself did not necessarily result in significantly improved times, possibly because of the inhomogeneity of the pumping capacity among the ballast tanks.
(ii) **Minimise pumping time** The calculation of pumping time for ballasting and de-ballasting operations was outlined in section 3.3. The procedure for the calculation was described as complex and non-intuitive, and it was mentioned that MATLAB scripts were developed by the author to automate the search for an optimal pumping configuration.

Indeed, two scripts were developed to facilitate the calculation of pumping time, for ballasting and de-ballasting, respectively. Both scripts were relatively primitive in their approach to the calculation, taking advantage of the computational capability of the computer to enumerate the potential configurations faster and more systematically than could be done by hand.

In the case of ballasting, the time required for the script to enumerate all possible solutions and identify the optimal one was negligible, being possible almost instantaneously. However, for de-ballasting, the enumeration of all possible solutions given the 16 subgroups of tanks and multiple pump room usages per subgroup required considerably more runtime. Two versions of the de-ballasting code were written, one for a quick estimation of de-ballasting time without the full list of pump room usages (runtime $\approx 35$ sec) and another with all possible pump room usages (runtime $\approx 37$ min).

Both of the de-ballasting scripts were deemed too slow to be included in the optimisation. Consequently, another approach was used to give an estimate of the pumping time objective. For ballasting, the script was used normally to find a pumping time for the tanks filling. The same ballasting script was used to calculate the time for pumping time for tanks emptying. However, the time calculated by the ballasting script was then corrected by a factor, $c$, to give an estimate of de-ballasting time.

So, a “pseudo” de-ballasting time was calculated, as in equation 4.2, for the tanks emptying in transition $l_{c_j}$ to $l_{c_{j+1}}$.

$$pseudodeballast\text{time}_{l_{c_j\rightarrow l_{c_{j+1}}}} = c \cdot ballast\text{time}_{l_{c_j\rightarrow l_{c_{j+1}}}}$$ (4.2)

The correction factor, $c$, was determined from a series of calculations of ballasting and de-ballasting time. For 150 randomly-generated filling transitions, the pumping time was calculated first with the ballasting script and then with the de-ballasting script. For each of the 150 pairs of times, a $c$ value was calculated as in equation 4.3.

$$c = \frac{deballast\text{time}_{l_{c_j\rightarrow l_{c_{j+1}}}}}{ballast\text{time}_{l_{c_j\rightarrow l_{c_{j+1}}}}}$$ (4.3)
An average value of $c = 1.022$ resulted from the 150 values, with a standard deviation of approximately 6%.

The pseudo-time objective was used in the optimisation as a quick estimation of pumping time, with the actual value of pumping time calculated in post-processing for promising solutions.

$$f_1 = \sum_{j=1}^{lc-1} (\text{ballast time} + \text{pseudo deballast time})_j$$

(4.4)

**Minimise Draught Difference from Target** For lifting operations, it is imperative that the PS achieve a sequence of specified draughts. These draughts are determined based on the dimensions of the topsides to ensure appropriate clearances between the vessel and the platform, as well as to ensure that the TLS can attach to the structure as needed to complete the lift. Thus, ideally:

$$\text{draught}(x_{ij}) = \text{target draught}_j, \forall j = 1, 2, 3, ..., lc$$

(4.5)

However, the initial performance of the optimisation indicated that this requirement was difficult to meet outright, even when it was converted to an inequality, of the form:

$$|\text{draught}(x_{ij}) - \text{target draught}_j| \leq \text{eps}, \forall j = 1, 2, 3, ..., lc$$

(4.6)

where $\text{eps}$ represents the maximum allowable difference from the target draught.

Finally, it was decided to implement the draught requirement as an additional objective to be minimised, as shown in equation 4.7.

$$f_2 = \sum_{j=1}^{lc} |\text{draught}(x_{ij}) - \text{target draught}_j|, \forall j = 1, 2, 3, ..., lc$$

(4.7)

The resulting improvement in the number and quality of feasible solutions produced may be said to justify this decision. In section 4.3, the effect of constraints on the optimisation’s performance is discussed more extensively.

**Minimise Trim** The PS is required to maintain an even keel in the majority of the loading conditions of the lifting operation. The result is a constraint on the trim for each loading condition:

$$|\text{trim}(x_{ij})| \leq \text{trim limit}_j, \forall j = 1, 2, 3, ..., lc$$

(4.8)
Originally, trim was implemented as a constraint in the optimisation. However, the more string-
gent the trim limit, the more difficult for the optimisation to produce a sufficient amount of
feasible solutions and to progress the search at an acceptable rate. Consequently, trim was also
converted into an objective of the optimisation, as shown in equation 4.9.

\[
f_3 = \sum_{j=1}^{lc} |trim(x_{ij})|, \forall j = 1, 2, 3, ..., lc
\]  

(4.9)

As with the draught requirement, the optimisation performance improved with the implementa-
tion of trim as an objective.

4.1.2.3 Constraints

It is now possible to describe the mathematical formulation of the problem constraints. These
are subdivided into hydrostatic, structural loading, pumping system and stability constraints.

**Hydrostatic Constraints**  Although draught and trim were modelled as objectives, the heel
requirement was retained as a constraint. The heel is required to be approximately zero, and
it was modelled within the optimisation as a limitation to the allowable transverse centre of
gravity (TCG) values, as follows:

\[
g_1 : |TCG(x_{ij})| \leq TCG \text{ limit}_j, \forall j = 1, 2, 3, ..., lc
\]  

(4.10)

Heel was modelled in terms of TCG separation because the LCT codes used to calculate the
ship characteristics for each loading condition were not equipped to estimate it directly. Within
the LCT, transverse centre of buoyancy (TCB) is always zero, and balance in the transverse
direction is obtained by requiring that TCG is also zero.

In the optimisation, the limit to TCG was set to 0.5 m in order to allow sufficient diversity in the
initial phases of the search. The effect of constraints on the performance of the optimisation is
further discussed in section 4.3. The heel angle of final solutions was checked to ensure feasibility
based on the formula for small angles of inclination (equation 4.11).

\[
\phi = \arctan \left( \frac{GG'}{GM_t} \right)
\]  

(4.11)
where $GG' = TCG$ and $GM_t = KM_t - KG$. $KM_t$ is the height of the transverse metacentre above the keel, and is known for a given draught from the ship’s hydrostatics. $KG$ is the height of the centre of gravity above the keel and is calculated for each loading condition.

**Structural Loading Constraints** Due to the PS’s unique length-to-breadth ratio, it is admittedly an oversimplification to evaluate the structural loading of the vessel based solely on a one-dimensional (1-D) Bernoulli beam approach, as in conventional vessels. Within Allseas, the structural loading of the PS has usually been determined using finite element method (FEM) analyses for critical phases of lifting operations. These analyses have been used to develop a set of allowable longitudinal loading envelopes for different operational phases (i.e. transit, lifting, etc.).

The allowable longitudinal loading envelopes are stored within Allseas’ LCT. When a loading condition is prepared by hand, the stillwater longitudinal structural loading of the PS (shear force, bending moment) is calculated based on a 1-D approach and checked against the envelopes. Once a loading condition is prepared such that meets longitudinal strength checks, data from the LCT is exported to Femap, and the structural loading is more thoroughly evaluated.

Currently, research is being done within Allseas to develop a non-FEM approach for evaluating the global hull structural integrity of the PS. Once such an approach has been devised, any additional structural constraints will also need to be included in the ballast optimisation method.

Within the scope of this project, only stillwater longitudinal structural loading was considered. The structural loading was evaluated based on the conventional 1-D hull girder approach, as in the LCT. The actual longitudinal shear force (SF) and bending moment (BM) for each loading condition of the lifting operation was required to be within the allowable envelopes. The requirement is implemented in the constraints:

\[
g_2 : \max SF(x_{ij}) \leq \text{allowable envelope}_j, \forall j = 1, 2, 3, ..., lc \tag{4.12}
\]
\[
g_3 : \max BM(x_{ij}) \leq \text{allowable envelope}_j, \forall j = 1, 2, 3, ..., lc \tag{4.13}
\]

**Stability Constraints** Due to the large breadth of the PS (124 m), the vessel’s intact stability has been found to be non-critical. Consequently, no intact stability constraints have been implemented in the ballast optimisation method. In the future, an evaluation of the free surface
effect of ballast configurations produced could be included, though it is not expected that it will render stability critical.

Damage stability criteria are still being developed for lifting operations with the PS. Given the ample intact stability and the large number of compartments, it is generally found that the damage stability of the vessel is also considerable (PS Intact Stab. & Long. Strength, 2011).

A few situations are outlined in which damage stability may influence the ballasting configuration selected. In the initial steps of the lifting operation, the PS is ballasted to a deeper draught to sail in around the platform, beneath the topsides. In this stage, the filling level of the ballast tanks in the forward hulls is not to exceed the external sea level. In the event of a hull breach and ingress of water, the draught of the vessel will increase, preventing the collision of the TLS with the underside of the topsides (PS WB System FMEA, 2010).

Another critical phase is when the vessel is sailing-out from around the platform, upon having lifted the topsides. The PS is transporting the full topsides load while sailing at a very shallow draught to clear the substructure (jacket) as it moves away. Water levels within ballast tanks on the inside of the slot are to be maintained at the same height as the external sea level. In this way, a collision resulting in a hull breach will not cause the ship to sink at the fore and the lifted topsides to strike the top of the jacket.

Given the ongoing analyses regarding damage stability criteria for the PS, no relevant constraints were implemented in the ballast optimisation method. As with structural loading, once the damage stability criteria for the PS are fixed, they may also be implemented as constraints in the ballast optimisation.

4.1.2.4 Search Algorithm

In this section, the selection of the search algorithm for the optimisation is described, including the problem requirements, an overview of algorithms surveyed, and a brief description of the algorithm chosen for implementation.

Requirements for the Search Algorithm   A preliminary step in selecting a search algorithm for the ballasting problem was to enumerate the requirements for the algorithm in the scope of this project. The decision to use an available algorithm, rather than develop one from scratch, significantly influenced the selection process. Choosing an existing algorithm rather
than writing and debugging a new one, meant that the project could focus on the more practical issue of accurately modelling the ballasting problem and tailoring the chosen algorithm to produce usable solutions, rather than expending effort on the mathematics of the search process. Thus, the first important requirement in selecting the search algorithm was that it should be readily available for use in a numerical computing environment, like MATLAB.

A second important requirement was that the algorithm chosen should be capable of global optimisation without requiring a significant amount of problem-specific information. The search algorithm had to be selected when the mathematical modelling was at a very initial stage and subject to alterations in the course of the project. Consequently, it was deemed prudent to prefer search algorithms which could be implemented quickly and produce valid results, without requiring a smooth, well-mapped solution space. Also, given that the initial modelling might, and did, change in the course of the project, it was important that the algorithm chosen was flexible enough to accommodate modifications. Given the modelling, it was vital that the search algorithm should be capable of multi-objective optimisation.

Finally, the ballast optimisation method was not intended for immediate real-time use in operations, but rather as an aid during the planning stage. As a result, the time required by the search algorithm for solving was not to be excessive, but it was also not required to produce solutions instantaneously. Ideally, the optimisation could be completed on the scale of hours rather than days. It would also be beneficial if the algorithm chosen could be implemented with some form of parallel computing or as part of a hybrid method to reduce the necessary computational time for finding a solution.

**Survey of Search Algorithms** Having expounded on the general requirements, it is now possible to present an overview of the most relevant search algorithms surveyed for application.

There are a multitude of ways to classify the different types of optimisation methods, depending on the primary characteristics of interest (Yang, 2013). One major distinction is that of local versus global optimisation. A local optimum is better than all neighbouring points in the search space, whereas a global optimum is better than all other points in the search space (Burke and Kendall, 2014). For the ballasting problem, the optimisation is intended to improve on the solutions generated by hand. The goal is to find an acceptably optimal solution, if not the global optimum, but certainly not to become trapped in local optima along the way.
A second distinction is made between gradient- or derivative-based and derivative-free methods. Methods are categorised based on whether or not they use the derivative of the objective function to guide the search.

A third distinction is made between deterministic and stochastic optimisation methods. Deterministic methods proceed with the search in a “mechanically deterministic manner without any random nature,” and will end up at the same final solution given the same initial point (Yang 2013). Stochastic methods, on the other hand, include a random element, and typically come to a different final solution despite using the same initial point. When randomness is introduced within a specific component of an algorithm, in a systematic way, the method is called heuristic, or when implemented at a higher level of sophistication, metaheuristic (Yang, 2013).

A survey of global optimisation methods was conducted, including a combination of classical and newer optimisation methods.

1. Classical Optimisation Techniques
   One of the older groups of optimisation methods is that of classical optimisation. Wehrens and Buydens (2000) describe these as techniques which rely on a representation of the solution space. Often some sort of sampling of possible solutions is applied by these methods in order to get an understanding of the geometry of the solution space or of the direction which is most promising for the search.

   Classical optimisation methods are divided into two categories, weak and strong methods (Wehrens and Buydens, 2000). Strong methods require a great deal of information about the problem parameters and the solution space, but if the information is correct, it results in very good performance of the optimisation. A typical example of a strong optimisation method is gradient-based optimisation. This method relies on a correct approximation of the local gradient, meaning the objective function must be differentiable.

   Weak methods make no assumptions about the solution space, and often rely on chance. Newer optimisation methods, such as those described below, are typically a combination of strong and weak methods, seeking to compromise between chance and problem information (Wehrens and Buydens, 2000). Random search is a typical example of a weak method. It is essentially as simple as it sounds, entailing a consecutive evaluation of randomly selected solutions, with the best solution found replacing the current solution until time is up.
To summarise, strong classical optimisation methods are very effective, but require a lot of problem-specific information to be available. They have been widely implemented in numerical computing software, and so can be applied relatively quickly to a model. However, as newer optimisation methods are developed which combine qualities of weak and strong classical optimisation, the classical techniques tend to be combined with newer approaches to form hybrids, which are more effective for engineering optimisation (Andersson, 2000).

2. Simulated Annealing

Simulated annealing (SA) is a stochastic, non-gradient-based optimisation method inspired by the annealing of solids. Annealing is the process of heating and then cooling a material so as to change its physical properties due to rearrangement of its internal structure. The changes to the system occur randomly, but stay in a given arrangement with a probability depending on the energy of the system and its temperature. The probability that the system will retain a given configuration is based on Gibb’s law (Sivanandam and Deepa, 2008):

$$p = e^{\frac{E}{kT}}$$  \hspace{1cm} (4.14)

where $E$ represents the energy of the system, $k$ is Boltzmann’s constant, and $T$ represents the temperature of the system.

During an iteration, a random new solution in the neighbourhood of the current solution is chosen. If the fitness value of the new solution is better than the current solution’s fitness, then the new solution replaces the current one. If the fitness value of the new solution is worse than that of the current solution, then it is assigned a probability based on a slightly altered version of expression ??:

$$p = e^{-\frac{\Delta \text{fitness}}{kT}} = e^{-\frac{(f(\text{new}) - f(\text{old}))}{kT}}$$  \hspace{1cm} (4.15)

A fraction of worse solutions are accepted in order to ensure that the algorithm does not become trapped in local optima (Nikolaev and Jacobson, 2010).

Usually, the SA algorithm starts from a high temperature which exponentially decreases with every iteration. It has been shown that improved performance is attained through slower cooling (Sivanandam and Deepa, 2008). When the temperature of the system is high, worse solutions are accepted by the algorithm with greater frequency, ensuring that it may move away from any local optimums encountered early on (Jacobson, 2013). As
the temperature decreases with each iteration, so does the probability that worse solutions will be accepted.

According to Nikolaev and Jacobson (2010), SA has not generally been the preferred method for engineering problems because it is very slow at solving complex systems, requiring a considerably larger amount of processor time than conventional algorithms. Some efforts have been made to improve its performance and reduce solution time, but such efforts also result in more difficult escape from local optima. SA has been applied mostly for discrete optimisation problems in fields like process engineering, operations research, and smart materials, and used in both single- and multi-objective optimisations (Nikolaev and Jacobson, 2010). Once extremely popular for combinatorial optimisation problems, today it is most frequently combined with other algorithms in hybrid approaches.

3. Genetic Algorithms

Genetic algorithms are the most prominent members of a wider group of techniques known as evolutionary algorithms. The latter are inspired by the process of evolution and natural selection. Sivanandam and Deepa (2008) describe evolutionary algorithms as iterative and stochastic processes that operate on a group of individuals, each of which represent a potential solution to the optimisation problem. The initial group of individuals, or population, is randomly generated and assigned a “measure of goodness” by a fitness function. The quantitative fitness value is then used to guide the search.

Genetic algorithms are the most well-known of the evolutionary techniques, primarily because of their robustness. Robustness refers to the algorithm’s ability to perform consistently well for a variety of problem types. Because they do not make use of objective function derivatives, GAs have no concrete mathematical restrictions as to the form of the objective or fitness function. Indicative of their flexibility is the fact that they have been used in many different applications, ranging from design problems, to scheduling, signal processing and combinatorial optimisation. For many researchers, they are one of the best methods to apply when little is known about the search space (Sivanandam and Deepa, 2008; Mitchell, 1999).

The wide applicability of GAs is also one of the shortcomings of this method in relation to other, more problem-specific algorithms. In fact, Mitchell (1999) warns that if a
space is well-understood, small, smooth or unimodal, other tailored heuristics may outperform a general-purpose GA. Also, as in most heuristic methods, there is no guarantee of optimality, but only of a solution that is sufficiently good.

GAs have some similarity to SA. Both methods are inspired by natural phenomena and may be used in both single- and multi-objective optimisations. Sivanandam and Deepa (2008) argue that several features make GAs more effective. First, GAs evaluate an entire population of solutions at each iteration, while SA only considers a single individual. This means that GAs are able to investigate a large part of the solution space at each iteration. Furthermore, GAs use recombination operators which allow good characteristics from highly-fit individuals to be used to form the next individuals.

Since SA considers only one individual at a time, iterations are done more quickly. However, GAs can be parallelised to improve speed. Both methods have been implemented over a range of problems.

4. **Tabu Search**

Tabu search is a local search method which is guided by a form of memory about past alternatives investigated. The meaning of the word *tabu* in Webster’s Dictionary is given as “something banned as constituting a risk.” In terms of searching the solution space, Glover and Laguna (1997) describe this risk as that of a “counterproductive course, including one which may lead to entrapment without hope of escape.” The idea for tabu search is partly inspired by human behaviour, in that straying from the accepted course may be regarded as a source of error, but also as a potential source of improvement (Glover, 1990).

Tabu search was envisioned as an evolution of hill-climbing search techniques, which would avoid becoming trapped in local optima. An outline of the tabu search algorithm is given in Glover and Laguna (1997). Beginning from an initial current solution, at each iteration the algorithm generates a *candidate list* of possible moves. The best admissible candidate from the list is selected as the new current solution. The admissibility of a candidate is evaluated based on two aspects.

First, an admissible candidate should not be on the *tabu list*, which enumerates the solution points or move attributes which are to be avoided. The tabu list is continually updated based on a form of short-term memory, and is used to prevent the algorithm from reverting back to previous solutions (Gendreau and Potvin, 2014). Additionally, if the candidate
is on the tabu list, it may still be admissible if it meets one or more of the aspiration criteria. These criteria allow solutions to be exempted from the tabu list, if they seem to lead to promising solutions. The algorithm stops when it meets the stopping criteria, which are typically based on the total number of iterations or the number since the last best solution was found.

According to Gendreau and Potvin (2014), since its inception by Glover in 1986, the method has been mainly applied to combinatorial optimisation problems in operations research. Glover and Laguna (1997) list many other fields where tabu search has been used, including design, resource planning, telecommunications, financial analysis and scheduling, among many others.

Glover and Laguna (1997) base the algorithm’s efficiency on the fact that it “incorporates adaptive memory and responsive exploration” of the solution space, rather than being “memoryless” like SA and, to a certain extent, GAs. However, Gendreau and Potvin (2014) warn that significant knowledge of the problem is essential to correctly implementing even basic steps of the tabu search procedure.

**Conclusion** Of the methods reviewed, genetic algorithms were found to most effectively meet the requirements enumerated. They are a global optimisation method requiring little problem-specific information to begin implementation. GAs are available for single- and multi-objective optimisation in MATLAB. Tabu search is not available in MATLAB, so it would require considerably more time to begin implementation. In terms of availability for quick implementation in MATLAB, GAs, SA and classical optimisation methods are all included as part of the Global Optimisation Toolbox.

GAs, SA, and tabu are global optimisation methods which each have a mechanism for avoiding being trapped in local minima. This is not the case for the classical methods. Tabu search and classical methods generally require more problem-specific information than GAs and SA, constituting the latter methods more favourable.

It is possible to conclude that the choice is primarily between GAs and SA given the characteristics of the problem as they are currently known. In terms of computational time, GAs are population-based, meaning that they require more time for each iteration than SA, which evaluates a single solution at a time. However, GAs may be used with parallel processing, which allows a reduction to the computational time. Furthermore, SA performance benefits from a
slow cooling scheme and degenerates as it is sped up, meaning that what is gained in time may be lost in performance of the algorithm.

There is no concrete rule for choosing a specific optimisation problem, but given the requirements and judging the sum of the method’s attributes, it is decided to make a first optimisation attempt using GAs.

**Description of Genetic Algorithms (GAs)** The algorithm starts by randomly generating an initial population, or group of individuals (solutions). In each step, or generation, the algorithm uses individuals from the current generation to create a subsequent one. The creation of each generation is conducted through a combination of steps.

First, each individual in the current population receives a score based on its fitness function value. Then, fitness scores are converted to a usable range of values. In the selection phase, individuals are chosen as parents based on their scaled fitness, with a higher probability of selection corresponding to higher scaled fitness values. Some very fit individuals are chosen as elite and passed on the the next generation as they are.

Individuals of the next generation, or children, are produced from parents either by mutation or crossover. Mutation involves making random changes to the genome of the parents, whereas crossover involves the combination of pairs of parent genomes. Stopping criteria are evaluated for each generation, and the algorithm is either stopped or another generation is created.

Figure 4.2 gives a graphical representation of the simple GA algorithm.

![Algorithm for simple GA](image.png)

**Figure 4.2:** Algorithm for simple GA (adapted from Andersson, 2011)
4.2 Programming Implementation

4.2.1 Numerical Computing Environment

The optimisation was implemented in the MATLAB numerical computing environment, with two different search algorithms. The MATLAB environment was chosen based on the author’s familiarity with the software and the availability of an existing toolbox for global optimisation. Additionally, scripts from a preliminary attempt to implement Allseas’ LCT in MATLAB would be compatible with any tool for ballast optimisation developed in the same environment.

The problem was initially optimised using the genetic algorithm solver from MATLAB’s Global Optimisation Toolbox. The toolbox provides solvers which search for global minima or maxima, and supports various levels of algorithmic customisation. Consequently, though it is easy to begin implementing a prepared model with a simple genetic algorithm, it is also possible to develop a more sophisticated implementation with custom-written constraint handling and fitness scaling functions, among other characteristics. The toolbox also allows for parallel processing, in combination with MATLAB’s Parallel Computing Toolbox, a useful capability given the computationally-intensive nature of genetic algorithm optimisation.

In a later stage of the research, for reasons discussed in chapter 4, it was decided to switch to the non-dominated sorting genetic algorithm (NSGA-II), originally developed by Deb et al (2002). A MATLAB implementation of the algorithm was made available to the author through the Ship Design Lab at Delft University of Technology, where it is used in a packing approach for early stage ship design (see Van Oers, 2011).

The decision to use an existing GA, rather than to code and debug an algorithm from the beginning, meant that implementation began immediately once the basic structure of the mathematical model was defined. This reduced the time required for the creation of the solver, allowing the focus of the project to shift from the mathematics of numerical optimisation to the modelling of the ballasting problem as an optimisation in a global sense.
4.2.2 Non-linear Constrained GA Optimisation with MATLAB’s Global Optimisation Toolbox

MATLAB uses an Augmented Lagrangian Genetic Algorithm (ALGA) to solve non-linear constraint problems. The optimisation problem solved is defined:

\[
\min_x f(x)
\]

such that

\[
A \cdot x \leq b \quad (4.16)
\]
\[
A_{eq} \cdot x = b_{eq} \quad (4.17)
\]
\[
c_k(x) \leq 0 \quad \text{for} \quad k = 1, 2, \ldots, m \quad (4.18)
\]
\[
ceq_k(x) = 0 \quad \text{for} \quad k = m + 1, \ldots, n \quad (4.19)
\]
\[
lb_i \leq x_{ij} \leq ub_i \quad \forall i, j \quad (4.20)
\]

where \(m\) and \(n\) are the total number of non-linear inequality and equality constraints respectively. In order to begin the optimisation, the GA problem structure requires the definition of a design variable, a fitness function, bounds, and linear/non-linear constraints.

The design variable must be a one-dimensional vector, with a length corresponding to the number of problem variables. The design variable is the input vector to the fitness function. The upper and lower bounds of the design variable are also defined in vector form.

If there are linear constraints, values are assigned for \(A, b, A_{eq}\) and \(b_{eq}\). The non-linear constraints are defined in a separate function, which is called in the problem structure. The fitness function is also included in the problem structure through a function call. The input to the fitness function is the design variable vector, and the output is the fitness function value, which the optimisation tries to minimise.

With the ALGA approach, non-linear constraints are handled differently than linear constraints and bounds. Namely, they are combined with the fitness function to form a separate subproblem using Lagrangian and penalty parameters. The subproblem is formulated

\[
\Theta(x, \lambda, s, \rho) = f(x) - \sum_{i=1}^{m} \lambda_i s_i \log(s_i - c_i(x)) + \sum_{i=1}^{n} \lambda_i c_{eqi}(x) + \frac{\rho}{2} \sum_{i=m+1}^{n} c_{eqi}(x)^2 \quad (4.21)
\]
where $\rho$ is the positive penalty parameter, the elements $s_i$ of the vector $s$ are non-negative shifts, and the components $\lambda_i$ of the vector $\lambda$ are non-negative and are known as Lagrange multiplier estimates.

The GA minimises a sequence of subproblems, each with a fixed value of $\lambda, s, \text{ and } \rho$. The Lagrangian estimates are updated once the optimisation has found a feasible solution for the subproblem with the required accuracy. If a feasible solution is not found, the penalty parameter is increased, and a new optimisation is run. This process is continued until one of the stopping criteria are met.

The output of the GA is a solution vector, $x$, corresponding to the final fitness value.

**Setting Parameters of the GA** When using the GA solver, there are many parameters of the algorithm which may be customised to suit the problem at hand, including population size, crossover fraction, fitness scaling method, and stopping criteria tolerances, among others. Once the basic modelling of the problem as a GA optimisation is complete, these parameters must be systematically investigated. The most favourable values should be identified, and their sensitivity evaluated.

**Stopping Criteria** Convergence, or stopping, criteria define when the optimisation will terminate. A number of criteria are evaluated, with the general intention being that the optimisation stops when no further improvement is obvious in the fitness functions. For the MATLAB GA, the stopping criteria are the following:

- Generations - When the set value of generations is reached, the algorithm stops.
- Time limit - The algorithm stops after a given amount of time.
- Fitness limit - When the value of the fitness function for the best point in the current population is less than or equal to this, the algorithm stops.
- Stall generations - The algorithm stops when the average relative change in the fitness value over a given number of generations is less than the “function tolerance.”
- Stall time limit - When there is no improvement in the fitness function during a period of time equal to this, the algorithm stops.
• Function tolerance - The algorithm stops when the average relative change in the fitness value over ‘stall generations’ is less than this.

4.2.3 NSGA-II

The non-dominated sorting genetic algorithm (NSGA-II) was developed by Kalyanmoy Deb and his students at the Indian Institute of Technology. Before the procedure followed by the algorithm is outlined, two concepts essential to understanding the NSGA-II are briefly described.

Non-dominated Sorting  For a non-dominated, or Pareto-optimal, solution, no improvement is possible in any of the objective functions without worsening at least one of the other objective functions.

A group of solutions may be divided into multiple non-dominated fronts. First, the entire group of solutions is considered at once, and a single non-dominated front is identified. Then, the solutions belonging to the first non-dominated front are removed from the group, and the remaining solutions are re-evaluated. The non-dominated front of the reduced group is identified as the second non-dominated front. The sorting continues until all solutions have been assigned to a front. Figure 4.3 shows the first non-dominated or Pareto fronts for a group of solutions (from Van Oers, 2011).

![Figure 4.3: Non-dominated solution fronts, from Van Oers (2011)](image)

In the NSGA-II, the front to which a solution belongs is used as a measure called “rank”. Lower non-dominated fronts are superior to higher ones.
Crowding-distance Calculation  In order to evaluate solutions belonging to the same non-dominated front, the NSGA-II uses a “crowding-distance” calculation. This measure is intended to preserve the diversity or spread of the solutions, without requiring a user-defined parameter. As may be seen in figure 4.4 from Van Oers (2011), the crowding distance of solution $i$ in its front is the average side length of the cuboid, formed by using the nearest, neighbouring solutions as vertices.

![Crowding distance calculation for NSGA-II, from Van Oers (2011)](image)

**Figure 4.4:** Crowding distance calculation for NSGA-II, from Van Oers (2011)

Outline of the Algorithm  The procedure followed by the NSGA-II is briefly described based on Deb et al (2002). The algorithm begins by randomly generating a parent population, $P_0$ of size $N$. The population is ranked according to non-domination, and then the tournament selection, recombination and mutation operators are used to create an offspring population, $Q_0$, of the same size.

It is now possible to describe the core of the NSGA-II procedure, illustrated in figure 4.5. At generation $t$, the parent, $P_t$, and offspring, $Q_t$, populations are first combined into a single population, $R_t$, of size $2N$. Elitism is included in the algorithm through this combination, as the parent population competes with the child population produced from it. The combined population then undergoes non-dominated sorting. Each individual is assigned to a non-dominated front, $F_i$.

To create the next population of parents, $P_{t+1}$, $N$ individuals must be chosen from $R_t$. Population $P_{t+1}$ is filled up with individuals beginning with best non-dominated set, or front, $F_1$. 


Assuming that $F_1$ has less than $N$ members, individuals are then taken from the next-best non-dominated front, $F_2$. By $F_l$ we denote the last non-dominated front from which members can be chosen to fill population $P_{t+1}$. In figure 4.5, $F_l = F_3$. The solutions in the last front are sorted based on crowding-distance, with less-crowded regions considered preferable. The best-ranked individuals in the final front fill the last slots remaining in population $P_{t+1}$. The new population then undergoes selection, crossover and mutation to produce the next population of offspring, $Q_{t+1}$.

**Figure 4.5:** Overview of NSGA-II procedure (from Deb et al, 2002)

**Constraint Handling** In the presence of constraints, the definition of domination between two solutions, $i$ and $j$, is defined in Deb et al (2002) as follows: A solution $i$ constrained-dominates a solution $j$ if at least one of the following conditions are met

1. Solution $i$ is feasible and solution $j$ is not.
2. Solutions $i$ and $j$ are both infeasible, but solution $i$ has a smaller overall constraint violation.
3. Solutions $i$ and $j$ are both feasible, and solution $i$ dominates solution $j$.

There is no need for a penalty parameter, as in MATLAB’s GA, since objectives and constraints are compared to one another.

**GA Parameters and Stopping Criteria** As with MATLAB’s GA, the NSGA-II offers the possibility to alter certain GA parameters such as population size, mutation and crossover
rates, among others, as necessary. Termination is imposed by setting a maximum number of
generations.

### 4.2.4 Ballast Method Implementation

It is now possible to give an overview of the programming implementation of the ballast method
in the MATLAB environment. In the following paragraphs, a description is given of the input,
the main body of the code, the important functions, the output, the validation process and
post-processing. Figure 4.7 gives a graphical overview (flowchart) of the program as an aid to
understanding.

It is noted that the current version of the optimisation code has been constructed so as to
accommodate any number of loading conditions. In general, this is achieved by ensuring that
all arrays and loops are constructed in a parametric way, so as to be able to be sized according
to the number of loading conditions required.

**Input**  The input to the fitness and constraint functions comes mainly from Allseas’ LCT.
Within the LCT, important information is stored, such as PS dimensions, hull geometry and
buoyancy distribution, lightship weight distribution, location and capacity of ballast and other
tanks, location and dimension of lifting systems, and weights and distribution of external loads
(topsides). The LCT also contains data on allowable shear force and bending moment envelopes
for different operational phases, such as transit, lifting, and transport.

A "read & prepare” function was created to read the relevant data from the LCT (Excel) for
each loading condition and store it to MATLAB structural arrays. The succession of target
draughts, and the limits for trim and heel should also be inputted separately. For 11 loading
conditions, the function takes approximately two minutes to run, primarily due to the delay
in interfacing with Excel. In order to reduce the total run time of the optimisation, the read
& prepare function is only run once at the beginning of the analysis. The data is stored once
and then loaded at each subsequent optimisation run. In the following paragraphs, the data
inputted from the LCT is referred to as “inputted ship data”.

**Main Body of GA Code**  As mentioned in section 4.2.1, a MATLAB implementation of
the NSGA-II was made available to the author through Delft University of Technology. The
NSGA-II codes perform the search according to the procedure described in section 4.2.3, with subroutines for initialisation, mutation, crossover, tournament selection and non-dominated sorting, among others. The logic of the NSGA-II code is illustrated in figure 4.6, which is taken from Van Oers (2011). The fitness and constraint functions developed by the author for the ballasting problem are called by the main body of the NSGA-II to conduct the optimisation.

1. Initialise population by generating random values.

2. Evaluate individuals: calculate objectives and constraints.

   For current_gen = 1:number of generations

   3. Create child population using

      - Selection
        - Tournament selection
        - Choice of parents based on
          - Feasibility
          - Rank
          - Crowding distance
        - Cross-over
        - Mutation

      For 1:population size

      4. Evaluate designs (calculate numerical ratings)

   End

5. Ensure elitism:

   - Merge parent and child population
   - Perform priority rating according to
     - Feasibility
     - Rank
     - Crowding distance
   - Create new parent population by retaining
     individuals with highest priority ratings, as shown in Fig. 5.5

   End

Figure 4.6: Outline of NSGA-II search algorithm, from Van Oers (2011)

Design Variables In section 4.1.2.1 the design variable, \( x_{ij} \), was defined as an array of ballast tank filling levels, with physical bounds of 0 to 1. However, the GA requires a one-dimensional vector as a design variable. Consequently, for the programming implementation, the design variable, \( x \), is a one-dimensional vector of length \( nvar \), with \( nvar = tnks \cdot lc \). In order to
optimise using discrete values of the design variable, $x$ is rounded as in equation 4.22.

$$x_{\text{new}} = \text{round}\left(\frac{x}{\text{inc}_x}\right) \cdot \text{inc}_x$$  \hspace{1cm} (4.22)

Solutions are evaluated and repaired to be discrete, then replaced in the original population.

**Fitness Function** The fitness function calculates the quality of the design alternatives, or solutions, generated by the GA. As the optimisation has three objectives, three fitness functions are evaluated for each solution.

The first objective, pumping time, is calculated by its own script, which is called by the main body of the GA. As explained in section 4.1.2.2, the pumping time objective is actually a pseudo-time objective, since de-ballasting time is estimated from ballasting time with a correction factor, $c$. For the second and third objectives, draught and trim are calculated by a *ship characteristics function*. Both the *pumping time* script and the *ship characteristics function* are further described in the following paragraphs.

**Constraints** The TCG for each loading condition is calculated by the *ship characteristics function*. The girder loads are also calculated within the *ship characteristics function*, with the final check that they are within allowable envelopes being conducted by the *structural check function*.

**Auxiliary Functions** To improve the speed of the optimisation, MATLAB suggests that the body of the code is broken into smaller functions, rather than consisting of a single, longer code. Thus, in addition to the fitness and non-linear constraints functions, the code is further subdivided into the following functions, which are called as needed.

**Ship Characteristics Function** This function is vital in updating the ship’s weight distribution, hydrostatic and structural loading characteristics for each new value of filling levels. The function is constructed from several subfunctions, which were originally created as an attempt to code the LCT in MATLAB.

- *Update Water Ballast Data*

  Based on updated filling levels and inputted ship data, the function calculates the new longitudinal distribution of water ballast weights. From the longitudinal distribution, the total mass, LCG and TCG of the water ballast are found.
• **Update Ship Center of Gravity**
  The longitudinal distribution of weights, total vessel mass, LCG, and TCG for the PS are updated based on the output of the previous function.

• **Update Draught and Trim**
  Given the updated weight distribution from the previous function, lookup tables for buoyancy and LCB are utilised to calculated the new draught and trim for the vessel. The mass and LCG of the vessel are assigned as target values and the correct draught and trim are interpolated from the stored table. The table is created for a range of draughts from 0 to 30 m, and a range of trims from -15 to 15 m. In its current form, the LCT only provides trim and draught data for zero heel, thus it is not possible to make a fully accurate calculation for varying angles of heel. This should eventually be addressed, but until then it is assumed that for the small angles of heel which are required in solutions the error incurred is within acceptably low levels.

• **Calculate Girder Loads**
  Based on the updated draught and trim values, the buoyancy of the vessel is calculated. Given the weight distributions outputted by the previous functions, it now possible to calculated the longitudinal load distribution. Numerical integration of the load distribution gives the longitudinal shear distribution, and a second integration gives the longitudinal distribution of the bending moment.

**Structural Check Function** This function compares the longitudinal shear force and bending moment for a solution to their respective allowable envelopes. The shear and bending distributions are calculated by the *ship characteristics* function. Where their values exceed the allowable values, the magnitude of the violation is recorded and non-dimensionalised with respect to the allowable value. The output of the function consists of two vectors, one with non-dimensionalised shear force violations along the length of the ship and the other with non-dimensionalised bending moment violations.

**Pumping Time Function** Based on the method described in section 3.3 and the modelling of the pseudo pumping time objective (4.1.2.2), pumping time scripts were written to automate the ballasting and de-ballasting time calculations. For a given set of tank fillings, the scripts identify the most favourable pump assignments for minimum transitional pumping time between the fillings. Both the ballasting and de-ballasting functions use an enumerative approach, calculating the pumping time for every subgroup of tanks and each possible pump room assignment.
Chapter 4. Method

The pumping time for each set of pump room assignments is calculated given pumping system constraints. Namely, tanks are considered in subgroups. For each subgroup, the pump rooms which may feasibly be used, independently or in combination with other pump rooms, for ballasting or de-ballasting the subgroup are enumerated. For example, if a pump room cannot effectively empty a group of tanks that is very far away, due to suction limitations, it is not included in the list of potential pump room assignments to be considered for the tank group by the scripts.

Given the feasible pump room assignments for each tank group, all possible combinations of the assignments are considered. If two tank groups require the use of the same pump room, their total pumping time will be the sum of their respective pumping times. If two tank groups use different pump rooms, their total pumping time will be the maximum of the two individual times, as it is possible for the tank groups to be pumped simultaneously. In this manner, the total pumping time for each string of assignments is calculated, and the minimum of all possible times is retained as the best use of pump rooms for the tank fillings considered.

During runtime of the optimisation, only the quick ballasting script was called to evaluate the pumping time for each individual. The coefficient, \( c \), described in section 4.1.2.2, was incorporated in the de-ballasting time calculation to give an approximate time estimation. The slower, but more accurate, de-ballasting script was used in post-processing, in combination with the ballasting script, in order to give a more accurate estimation of total pumping time for promising solutions.

**Validation of Functions** In general, the functions described were validated with existing hand calculations done by naval architects of the PS engineering team, as well as with any calculations also done in the LCT. More specifically, ballast tank filling levels which had been prepared by hand for a topsides lifting operation were inputted into each function, and the resulting output values were checked for conformity. The functions were found to be very accurate for all of the calculations implemented.

**Output & Post-processing** At each generation, the parent and child populations are stored. A convergence plot is generated for all objectives, including best and mean individual values. A pareto front is also generated between each pair of objectives.
As post-processing, solutions are first filtered in order to find those which are feasible, and within 0.5 m of total draught and trim requirements. Then, a few solutions are selected from the Pareto front and evaluated with the ship characteristics and pumping time scripts.

### 4.3 Influence of GA Parameters on Optimisation

A distinct characteristic of GAs is their sensitivity to parameter settings. They are generally regarded as robust and widely applicable, but to truly excel at the optimisation of a specific problem, they require some fine-tuning. Considerable research is conducted focusing solely on the influence of the GAs’ parameters on their performance. Within the timeframe of the current project, an exhaustive evaluation was not possible. However, a discussion follows of some of the more relevant parameters and their effect on the GA. The discussion is of a very general nature, not presuming to address fully the finer details of each parameter’s influence on the algorithms, but rather to introduce the phenomena which might have a bearing on the optimisation results.
4.3.1 Design Variable Discretisation

The number of design variables, \( nvar \), is dependent on the total number of ballast tanks, \( tnks \), and loading conditions, \( lc \), considered simultaneously, as expressed in equation 4.23.

\[
\text{nvar} = lc \cdot tnks
\]  

(4.23)

For each design variable, \( x \), the number of possible values it may take depends on the chosen increment size of the variable. Assuming an increment for \( x \), \( inc_x \), and an upper bound for the variable, \( ub_x \), the total number of possible values, \( vp \), is found from equation 4.24.

\[
vp = \frac{ub_x}{inc_x} + 1
\]  

where it is required that the ratio \( \frac{ub_x}{inc_x} \) result in an integer value. For an increment size of 25%, as was used in the optimisation, and an upper bound of 100%, there are 5 possible values for each design variable, \([0 \ 25 \ 50 \ 75 \ 100]\).

Given \( vp \) and \( nvar \) it is possible to calculate the total number of possible design alternatives, \( alt \), for the problem (equation 4.25).

\[
alt = vp^{nvar}
\]  

(4.25)

For example, given \( nvar = 2 \) and \( inc_x = 100 \), there are two possible values for each variable to take, 0 and 100. Thus, there are \( 2^2 = 4 \) possible alternatives: 0-0, 0-100, 100-0, 100-100.

The following conclusions may be drawn:

- For constant \( lc \), \( tnks \) and \( ub_x \), the number of possible design alternatives for the GA to investigate depends on \( inc_x \). The smaller the increment size, the more possible values for each design variable to take and the greater the number of possible design alternatives.

- For constant \( inc_x \) and \( ub_x \), an increase in the number of variables, either due to an increase in \( tnks \) or \( lc \) means an increase in the number of possible design alternatives.

In figure 4.8, the number of alternatives is plotted as a function of discretisation and number of loading conditions. Figure 4.8(a) shows the drastic increase in alternatives for decreasing increment size of the design variables, for a constant number of variables. Figure 4.8(b) illustrates the rapid increase in alternatives for additional loading conditions considered simultaneously, for
constant tank number and increment size. From figure 4.8(b) it is particularly noteworthy that, for the given modelling of tanks, the runtime will increase significantly with number of loading conditions due to the corresponding increase in alternatives. Consequently, a modification to tank modelling such as grouping of tanks into subsets may be relevant for future versions of the method.

![Graphs showing number of alternatives vs discretisation and loading conditions](image)

**Figure 4.8:** Number of alternatives possible as a function of discretisation and number of loading conditions

For the GA, a greater number of possible design alternatives to investigate may lead to better overall results, due to the increased diversity of the solutions considered. However, an increased number of alternatives will also mean that the GA may take longer to converge.

### 4.3.2 Population Size

Population size has been documented as having a significant effect on the performance of the optimisation. In general, if the population size is too small for the number of variables, the optimisation will converge more quickly, but there is the possibility that it will have found a local optimum because it has not searched enough of the solution space. If the population size is too large, the chance of finding the global optimum is better because more of the solution space will be searched, but the convergence speed will be reduced.

Thus, tuning the population size for the optimisation involves a compromise between the quality of the final solution and the time required for the search. A balance must be found between
the size of the population relative to the number of variables and the number of generations required for convergence. An indication that the population size might not be large enough for the number of variables and discretisation being used is that the number of feasible individuals comprises a very small percentage of the population.

It is noted that these statements on the effect of population size are more accurate when considering the extremes, namely very large or very small populations. However, there is a fuzzy area in between extremes, within which changes in the population size may not have as marked an effect.

In the optimisation, a population size approximately 30% larger than the number of design variables was used. A sufficient number of feasible solutions was produced in each population (approximately 30%), however further experimentation with varying population sizes is suggested to determine optimal performance.

4.3.3 Mutation Percentage

Mutation is one of the key operators of the GA. The mutation percentage determines the ratio of the population that is generated randomly during reproduction. It is a deciding factor in producing new genetic traits and retaining diversity. When the mutation percentage is high, a very large proportion of the population is altered at random. Taken to the extreme, the large mutation rate relegates the optimisation to a random search of the solution space. On the other hand, if the mutation percentage is low, subsequent populations are less diverse, and the optimisation converges quickly. However, the convergence may occur prematurely, to a local optimum, since little of the solution space will have been investigated.

For the ballast optimisation, a mutation percentage of 2.5 was used. A few consecutive runs of the optimisation for a range of mutation percentages, from 2.5 to 10%, were tested out for the same total number of generations. It was observed that for 2.5% mutation, a sufficient number of individuals was produced with better time objective values than achieved with the other mutation percentages in the same number of generations. More systematic experimentation with the mutation rate would be advisable and is recommended for further tailoring of the optimisation’s performance.
4.3.4 Effect of Constraints on Population Feasibility

The average percentage of feasible individuals in the population has a direct effect on the performance of the GA. When there is not a sufficient pool of feasible individuals in the population, there is a good chance that in each subsequent generation the few feasible designs will be combined with the many infeasible designs. The GA will need longer to improve on the small group of feasible designs.

For the ballasting optimisation method presented, the draught and trim requirements were modelled as objectives rather than constraints. The decision to change the modelling was based on the few, if any, feasible solutions produced while these two constraints were active.

When the draught, trim, TCG and structural loading constraints were imposed, it became very difficult for the optimisation to find any feasible solutions. Even after some of the constraints were relaxed by adding tolerances, the number of feasible solutions remained very small. The draught requirement seemed to be the most difficult to satisfy, followed by the trim requirement. Consequently, to improve the number of feasible solutions identified, these two requirements were converted from constraints to objectives to be minimised.

Initially, only the draught difference was added as an objective. However, the PS is required to retain a trim of approximately zero and when the trim constraint was made to reflect this, the feasibility of the population drastically decreased. By making trim the third objective, an effort could be made to minimise it as much as was needed without deteriorating the population feasibility.

The average feasible population for the three-objective optimisation was approximately 30%, compared to about 0-1% feasibility for the single- or two-objective optimisation with stringent constraints.

4.4 Switching GAs: From MATLAB’s GA to NSGA-II

Originally, the optimisation method used MATLAB’s single-objective GA to search the solution space. However, in the course of the project, the NSGA-II was adopted as the search algorithm. The key reasons for switching to the NSGA-II were to increase oversight, improve constraint-handling and take advantage of the algorithm’s multi-objective optimisation capabilities.
Increased Oversight Though the MATLAB GA was able to find reasonably good solutions for the water transfer objective, which was used in an earlier stage of the project, it did not perform well on the pumping time objective. The optimisation-produced results were unable to improve upon the time estimated for the hand-calculated loading conditions.

While searching for the cause of the poor results, some unexplained errors were discovered within the GA solver. For instance, it was discovered that some child populations were produced by the solver which were all non-numerical ("NaN"). The problem became a search for understanding as to whether the poor performance of the optimisation for the time objective was due to an ineffective fitness function, or to programming bugs.

In order to ascertain which of the aforementioned issues, modelling or bugs, were to blame for the poor performance of the optimisation, the MATLAB GA was abandoned and NSGA-II was adopted for the optimisation. As mentioned earlier in this chapter, a MATLAB implementation of the algorithm was made available to the author through the Ship Design Lab at Delft University of Technology. The NSGA-II implementation allowed more direct access to the underlying operators of the algorithm. Furthermore, it was more flexible than the MATLAB GA, making discretisation of the design variables and the addition of objectives relatively easy.

Improved Constraint Handling Given the difficulty in finding feasible solutions for the ballast problem constraints, it is also possible that the constraint-handling method used by the MATLAB GA was less effective than that of the NSGA-II. As already described in section 4.2.2, MATLAB uses an augmented Lagrangian genetic algorithm (ALGA) for optimisations with nonlinear constraints.

In the ALGA, the original fitness function is combined with the nonlinear constraint function using Lagrangian and penalty parameters. Then, the algorithm attempts to minimise the new fitness function. Since the GA is only focussed on minimising the fitness function, it is possible that the effect of a constraint violation is to improve the fitness to such a degree that it counterbalances the effect of the applied penalty. When constraints are hard to meet, this may lead the GA to stumble around the search space until it finds a feasible region (Pearce, 1997).

The NSGA-II uses a different approach to constraint-handling, wherein infeasible solutions are assigned a lower rank than feasible ones. Constraints are not compared with fitness function values, so there is no need for the inclusion of a penalty parameter.
Benefits of a Multi-Objective Approach  It is also hypothesised that the conversion of the optimisation from single- to multi-objective may have played a role in improving its performance. As was already discussed in section 4.3.4, by converting difficult-to-meet constraints to objectives, the optimisation could work to minimise the violation of requirements while retaining the population feasibility needed for the GA to make progress on the time objective. A supplementary benefit of the multi-objective approach is that it allows for the evaluation of trade-offs between the different objectives.

As evidenced by the results (Chapter 5), the NSGA-II was able to produce good solutions for the pumping time objective. It allowed more control and insight into its internal operations compared to the less-accessible GA solver provided by the MATLAB toolbox.
Chapter 5

Results

Once the ballast optimisation problem was implemented in MATLAB, it was necessary to determine a test case to be used to check the validity of the optimisation results and gauge the efficacy of the method compared to hand calculations. An actual topsides removal operation to be performed by the PS was chosen as the test case. The specific operation was chosen because hand-calculated loading conditions had already been prepared for the operation and made available to the author.

In this chapter, the test case utilised to validate and evaluate the performance of the optimisation method is introduced in section 5.1. Then, in section 5.2, the hand-calculated loading conditions prepared for this case are presented, followed by the optimisation-produced results in section 5.3. A comparison of the hand-calculated loading conditions to the optimisation-produced results is conducted in section 5.4. Finally, in section 5.5, a brief discussion of the results is included.

5.1 Description of Test Case

The full topsides removal operation from which the test case was developed, was described in section 1.2 and summarised in table 1.2. Although the operation includes 11 loading conditions in total, a subset of three conditions was chosen for the test case. By using only three conditions, sufficient variety in draught requirements and loading could be included without overly complicating the model in terms of computational time. Furthermore, it was found that it was easier to retain oversight of the problem and complete any hand calculations required to validate the generated solutions with a smaller set of loading conditions.
A few goals were set for the use of the test case. It was considered important to establish a basic working model for the optimisation, the results of which could be quickly verified with available hand calculations. Additionally, the influence of individual parameters of the GA on the efficacy of the optimisation should be understood. The test case would be used to experiment with different approaches regarding the formulation of the fitness function, the modelling of objectives and constraints, and the inclusion of one or more optimisation objectives.

The test case was comprised of loading conditions 1-3 of the full topsides removal operation, with the requirements for draught, trim, TCG separation and structural loading summarised in table 5.1. These loading conditions were selected because they require significant changes in the draught of the PS and include both ballasting and de-ballasting operations. The trim and TCG requirements are taken from Allseas’ LCT.

<table>
<thead>
<tr>
<th>Loading Condition No.</th>
<th>Phase Description</th>
<th>Required Mean Draught [m]</th>
<th>Maximum Absolute Trim/TCG [m]</th>
<th>Maximum Absolute Shear Force [ton]</th>
<th>Maximum Absolute Bending Moment [ton* m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>L.C. 1</td>
<td>Transit to Platform</td>
<td>12.00</td>
<td>0.05 / 0.04</td>
<td>25009</td>
<td>1800615</td>
</tr>
<tr>
<td>L.C. 2</td>
<td>Positioning Around Platform (sail-in)</td>
<td>22.00</td>
<td>0.05 / 0.04</td>
<td>67023</td>
<td>1800615</td>
</tr>
<tr>
<td>L.C. 3</td>
<td>Attain Connecting Draught</td>
<td>19.08</td>
<td>0.05 / 0.04</td>
<td>67023</td>
<td>1800615</td>
</tr>
</tbody>
</table>

Table 5.1: Loading conditions used as test case to validate and evaluate performance of ballast optimisation

5.1.1 Threshold for Minimum Pumping Time

A theoretical threshold, or lower bound, was estimated for the pumping time per transition based on the minimum water volume transfer required to change draught. From the hydrostatic table of the PS, it was possible to find the approximate displacement of the vessel at each draught. Then, given the necessary water volume for each draught change, and assuming that all of the pumping capacity of the ballast system is available (regardless of the way pump rooms
are utilised), a lower bound for pumping time was estimated. The lower threshold for each transition is given in table 5.2.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>L.C. 1</td>
<td>12.00</td>
<td>353997</td>
<td>364822</td>
<td>37200</td>
<td>9.81</td>
</tr>
<tr>
<td>L.C. 2</td>
<td>22.00</td>
<td>718819</td>
<td>108098</td>
<td>37200</td>
<td>2.91</td>
</tr>
<tr>
<td>L.C. 3</td>
<td>19.08</td>
<td>610721</td>
<td></td>
<td></td>
<td>12.71</td>
</tr>
</tbody>
</table>

Table 5.2: Estimation of lower threshold for pumping time per transition

5.2 Hand-Calculated Loading Conditions for Test Case

In section 1.3, the procedure for preparing a loading condition by hand was outlined. Loading conditions 1-3 were prepared according to this procedure. The hand-calculated ballast configurations for the test case are depicted in figure 5.1. A colour scheme is used to show the variations in filling levels for the ballast tanks, based on the legend included in the figure.

Table 5.3 gives actual values calculated for requirements such as draught, trim, and TCG. Structural loading curves for longitudinal shear force and bending moment are given in figure 5.2.

The pumping times calculated for each transition are presented in table 5.4. The table also gives a comparison of pumping times calculated by hand to pumping times calculated by the MATLAB scripts written by the author for the optimisation method. It is noteworthy that for the same ballast configurations, the scripts are able to come up with better pumping times than were found by hand.
Figure 5.1: Hand-calculated ballast configurations for loading conditions 1-3

<table>
<thead>
<tr>
<th>Requirement</th>
<th>LC 1</th>
<th>LC 2</th>
<th>LC 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draught [m]</td>
<td>12.00</td>
<td>22.00</td>
<td>19.08</td>
</tr>
<tr>
<td>Trim [m]</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>TCG [m]</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Maximum absolute shear force [ton]</td>
<td>16858</td>
<td>32172</td>
<td>25715</td>
</tr>
<tr>
<td>Maximum absolute bending moment [ton*m]</td>
<td>1462568</td>
<td>1022580</td>
<td>1495717</td>
</tr>
</tbody>
</table>

Table 5.3: Actual values calculated for requirements for hand-prepared loading conditions
Figure 5.2: Longitudinal structural loading distributions for hand-calculated LC 1-3

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>L.C. 1-2</td>
<td>12.44</td>
<td>11.30</td>
</tr>
<tr>
<td>L.C. 2-3</td>
<td>4.13</td>
<td>3.10</td>
</tr>
<tr>
<td>Total, L.C. 1-3</td>
<td>16.57</td>
<td>14.40</td>
</tr>
</tbody>
</table>

Table 5.4: Pumping times found for hand-calculated ballast configurations for test case, both by hand and using pumping scripts
5.3 Optimisation-Produced Loading Conditions for Test Case

The ballast optimisation method described in chapter 4 was used to generate ballast configurations for each loading condition in the test case. Since the method was modelled as a multi-objective optimisation, the output was not a single solution, but rather a front of possible alternatives. Figure 5.3 presents the feasible solutions found by the optimisation for trim and draught difference objectives less than 0.5 m. The time objective is plotted on the horizontal axis, the draught difference objective on the vertical axis, and the trim objective based on a colour scale.

![Figure 5.3: Optimisation results for pumping time, draught difference and trim objectives, with an arrow indicating the solution selected for further analysis](image)

One of the benefits of a multi-objective approach is that the trade-off between the draught-trim and time objectives is visualised, and solutions with favourable times and varying draught-trim violations can be considered. From the many solutions in figure 5.3, those of particular interest are bounded by two limits. First, in terms of the pumping time objective, the theoretical threshold for pumping time of 12.71 hrs (section 5.1.1) serves as a lower bound for possible solutions. Second, the minimum hand-calculated pumping time of 14.40 hrs acts as an upper bound, as the objective is to improve on hand-calculated configurations. For the draught difference and trim constraints, filtering has already been applied to retain only those solutions with a violation less than SI0.5m. More stringent requirements for these constraints may be implemented for these values through further filtering.
In figure 5.3, it may be observed that some of the feasible solutions produced by the optimisation are less than the theoretical threshold based on minimum water volume transfer for draught change. The threshold of 12.71 hrs was assumed to be a theoretical lower bound for pumping time based on the minimum water volume transfer for draught change. Consequently, the reader may wonder why the optimisation appears to have produced feasible solutions below that threshold. However, the threshold was calculated from vessel displacements for a specific set of draughts and approximately zero trim and heel. If the draught and trim of the vessel differ from the original values, then the water volume needed to move from one displacement to the next will also differ, resulting in a slightly different value for the threshold.

A single solution lying within the bounds described is chosen for closer investigation. The solution selected is indicated by an arrow in figure 5.3. The filling levels produced for the test case were inputted into Allseas’ LCT in order to visualise the ballast plans and compare them to the hand-calculated configurations. The optimisation-produced configurations are depicted in figure 5.4. The same colour scheme is used to illustrate the variations in fillings levels as for the hand-calculated ballast configurations in figure 5.1.

Table 5.5 gives actual values calculated for requirements such as draught, trim, TCG and structural loading. Structural loading curves for longitudinal shear force and bending moment are given in figure 5.5.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>LC 1</th>
<th>LC 2</th>
<th>LC 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draught [m]</td>
<td>12.17</td>
<td>22.00</td>
<td>19.07</td>
</tr>
<tr>
<td>Trim [m]</td>
<td>-0.19</td>
<td>-0.03</td>
<td>0.02</td>
</tr>
<tr>
<td>TCG [m]</td>
<td>-0.32</td>
<td>0.25</td>
<td>-0.41</td>
</tr>
<tr>
<td>Heel [deg]</td>
<td>-0.2</td>
<td>0.2</td>
<td>-0.3</td>
</tr>
<tr>
<td>Maximum absolute shear force [ton]</td>
<td>17720</td>
<td>22798</td>
<td>22726</td>
</tr>
<tr>
<td>Maximum absolute bending moment [ton*m]</td>
<td>1800615</td>
<td>1647524</td>
<td>2288614</td>
</tr>
</tbody>
</table>

Table 5.5: Actual values calculated for requirements for optimisation-produced loading conditions

The pumping times calculated for each transition are presented in table 5.6. The pump rooms used to ballast each tank group for the transition LC 1-2 (12 to 22 m) are listed in table 5.7.
Chapter 5. Results

Figure 5.4: Optimisation-produced ballast configurations for loading conditions 1-3

The pump rooms used to de-ballast each tank group for the transition LC 3-4 (22 to 19.08 m) are listed in Table 5.8.

<table>
<thead>
<tr>
<th>Transition</th>
<th>Pumping Time for Optimisation-Produced LCs [hrs]</th>
</tr>
</thead>
<tbody>
<tr>
<td>L.C. 1-2</td>
<td>9.80</td>
</tr>
<tr>
<td>L.C. 2-3</td>
<td>3.23</td>
</tr>
<tr>
<td>Total, L.C. 1-3</td>
<td>13.03</td>
</tr>
</tbody>
</table>

Table 5.6: Pumping times found for hand-calculated and optimisation produced ballast configurations for test case

These results were produced using the NSGA-II. The optimisation was run on a single core due to limited parallel processing capabilities of the student version of MATLAB. Initially, the optimisation required a total runtime of approximately 39 hours (28 seconds per generation...
Figure 5.5: Longitudinal structural loading distributions for optimisation-produced LC 1-3

Table 5.7: Ballasting pump room assignments per tank group for optimisation-produced LC 1-2
Table 5.8: De-ballasting pump room assignments per tank group for optimisation-produced LC 2-3

To ensure that the results produced by the optimisation were reasonable and accurate, two validation steps were carried out. First, the ballast configurations prepared by hand were used as input to the optimisation model in order to check that ship characteristics and pumping
times were being calculated correctly by the model. The Excel spreadsheets prepared for the loading conditions and pumping time calculations were used to confirm results.

Additionally, the ballast configurations produced by the optimisation were inputted into Allseas’ LCT to verify that the requirements for draught, trim, heel, and structural loading were indeed met. The pumping time calculations for the most promising solutions were also checked by hand to ensure the validity of results.

5.4 Comparison of Hand-Calculated and Optimisation-Produced Results

Given the results presented in sections 5.2 and 5.3, it is now possible to more closely compare the loading conditions (LCs) prepared by hand and by the optimisation.

**Ballast Configuration** The characteristics of the water ballast of the hand-calculated and optimisation-produced configurations are compared in table 5.10. The major difference between the hand-prepared ballast fillings and the optimisation-produced ballast fillings is that the former are much more concentrated than the latter. For all of the loading conditions of the test case, the optimisation results use a greater number of tanks. However, the total weight of water ballast is approximately the same for both sets of loading conditions, with a maximum difference in LC 1 of 6988 t. From this, it is possible to conclude that in the optimisation-produced LCs, more tanks are filled to a lower level than in the hand-calculated LCs.

<table>
<thead>
<tr>
<th>GA Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>population size [-]</td>
<td>320</td>
</tr>
<tr>
<td>mutation percentage [%]</td>
<td>2.5</td>
</tr>
<tr>
<td>crossover percentage [%]</td>
<td>100</td>
</tr>
<tr>
<td>x-increment size [%]</td>
<td>25</td>
</tr>
<tr>
<td>total generations [-]</td>
<td>5000</td>
</tr>
<tr>
<td>cores used (in parallel) [-]</td>
<td>1 (no parallel processing)</td>
</tr>
</tbody>
</table>

**Table 5.9:** GA parameters used for test case
The longitudinal and transverse centres of gravity (LCG, TCG) are located at about the same place for both sets of LCs. There is more variance in the vertical centre of gravity (VCG), which is typically lower for optimisation-produced loading conditions. The largest difference between the two sets of conditions is approximately 3.31 m in LC 1.

<table>
<thead>
<tr>
<th>Water Ballast Characteristics</th>
<th>Hand-Calculated LCs</th>
<th>Optimisation-Produced LCs</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC [-]</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Tanks Filled [-]</td>
<td>23</td>
<td>47</td>
</tr>
<tr>
<td>Weight [t]</td>
<td>138584</td>
<td>145572</td>
</tr>
<tr>
<td>LCG [m]</td>
<td>119.15</td>
<td>119.69</td>
</tr>
<tr>
<td>TCG [m]</td>
<td>7.69</td>
<td>6.47</td>
</tr>
<tr>
<td>VCG [m]</td>
<td>9.59</td>
<td>6.28</td>
</tr>
<tr>
<td>LC [-]</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Tanks Filled [-]</td>
<td>62</td>
<td>83</td>
</tr>
<tr>
<td>Weight [t]</td>
<td>528446</td>
<td>528804</td>
</tr>
<tr>
<td>LCG [m]</td>
<td>146.92</td>
<td>146.86</td>
</tr>
<tr>
<td>TCG [m]</td>
<td>2.02</td>
<td>2.38</td>
</tr>
<tr>
<td>VCG [m]</td>
<td>13.97</td>
<td>11.78</td>
</tr>
<tr>
<td>LC [-]</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Tanks Filled [-]</td>
<td>62</td>
<td>83</td>
</tr>
<tr>
<td>Weight [t]</td>
<td>412774</td>
<td>412539</td>
</tr>
<tr>
<td>LCG [m]</td>
<td>143.67</td>
<td>143.71</td>
</tr>
<tr>
<td>TCG [m]</td>
<td>2.68</td>
<td>2.03</td>
</tr>
<tr>
<td>VCG [m]</td>
<td>11.43</td>
<td>9.76</td>
</tr>
</tbody>
</table>

Table 5.10: Comparison of hand-calculated and optimisation-produced ballast configurations

**Meeting Requirements** The hand-prepared loading conditions strictly adhere to all of the hydrostatic and structural loading requirements set out for the test case. The optimisation-produced loading conditions meet structural constraints, but they are somewhat more relaxed in hydrostatic constraints.

With respect to draught, there is a slight violation of 0.17 m in LC 1. The trim values in LC 2 and 3 of the optimisation results are acceptable, with a slight violation of 0.14 m in LC 1. Heel is the constraint with the greatest violation in the optimisation-prepared LCs. This is to be expected, as the TCG limit was set to 0.5 m within the optimisation in order to sustain the desired population feasibility. All of the LCs exhibit a small degree of heel, not more than 0.3 deg.
It is not expected that the aforementioned constraint violations will have a significant effect on the operability of the vessel and lifting systems. Given the improvement possible in the time objective, these small constraint violations could be considered justifiable. However, if another solution is desired which more strictly adheres to one or more of the constraints, it may be easily selected from the Pareto front of solutions produced by the optimisation. The latter emphasises the benefit of multi-objective optimisation compared to a single-objective one, namely that the trade-off between the main objective of time and the constraints is clearly visualised.

**Pumping Time**  The optimisation-produced LCs result in a significant improvement in the overall ballasting time for the test case. Table 5.11 gives an overview of the pumping times for four cases. Column 2 gives the times for the hand-calculated loading conditions and pumping times. Column 3 gives the times for the hand-calculated loading conditions when the pumping codes implemented in MATLAB are used for the time calculation. Column 4 gives the times for the optimisation-produced loading conditions calculated by the MATLAB pumping codes. Column 5 gives the times according the the theoretical lower bound mentioned in section 5.1.1.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>L.C. 1-2</td>
<td>12.44</td>
<td>11.30</td>
<td>9.80</td>
<td>9.81</td>
</tr>
<tr>
<td>L.C. 2-3</td>
<td>4.13</td>
<td>3.10</td>
<td>3.23</td>
<td>2.91</td>
</tr>
<tr>
<td>Total, L.C. 1-3</td>
<td>16.57</td>
<td>14.40</td>
<td>13.03</td>
<td>12.71</td>
</tr>
<tr>
<td>Δ Lower Bound [%]</td>
<td>30</td>
<td>13</td>
<td>2.5</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5.11: Pumping times found for hand-calculated and optimisation produced ballast configurations for test case

In table 5.11, for the transition LC 1-2, the time for the optimisation-produced LCs seems to fall below the theoretical lower bound. This is justified by the fact that LC 1 had a small draught and trim violation, although the threshold was originally calculated for the exact values of target draught and trim. Thus, the theoretical lower bound for the slightly altered draught and trim values would be somewhat different from the original one.
In an effort to make an improved comparison, the optimisation-produced solutions may be penalised in time for the slight constraint violations. In table 5.12, time penalties are calculated for the trim and draught difference violations of LC 1 of the optimisation-produced solution. The penalties are estimated based on the additional water volume that would need to be pumped at nominal capacity to eliminate constraint violations.

<table>
<thead>
<tr>
<th>Value</th>
<th>Violation [m]</th>
<th>Δ Volume [m³]</th>
<th>Time Penalty [hrs]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draught Difference</td>
<td>0.17</td>
<td>23854</td>
<td>0.64</td>
</tr>
<tr>
<td>Trim</td>
<td>-0.14</td>
<td>369</td>
<td>0.01</td>
</tr>
<tr>
<td>Total Time Penalty</td>
<td>-</td>
<td>-</td>
<td>0.65</td>
</tr>
</tbody>
</table>

Table 5.12: Time penalties for constraint violations of LC 1 of the optimisation-produced ballast configurations

Thus, the corrected time for the transition LC 1-2 is 10.45 hrs. Even with the added time penalty, the optimisation-produced solution still results in reduced ballasting time, of 13.68 hrs, compared to the hand-calculated solution.

5.5 Discussion

From the results presented, it is concluded that it is possible to improve upon the ballasting time required for the test case by using the optimisation. It is also possible to reduce the analysis time needed for the preparation of ballast configurations for the PS from weeks to hours, or at most days. Many solutions are generated as output from the multi-objective optimisation. In looking more closely at one of the solutions with a very favourable pumping time value, a general comparison could be made to the hand-calculated configurations prepared for the same case.

An improvement to the procedure of preparing ballast configurations by hand is achieved with the use of the MATLAB scripts developed in the course of this thesis. In creating a set of functions to automatically find the optimal pump room assignments, using the same logic and assumptions as by hand, the process is significantly accelerated. It is also relatively simple for the computer to try out all possible pump assignments to ensure that the time found is indeed the minimum possible for the given set of fillings.
A limitation of both the hand and optimisation approaches to calculating pumping time, is that neither considers hydrostatic and structural constraint values during the transition from one loading condition to the next. The only constraints considered are those that govern the operation of the pumping system, such as which pump rooms may be pumping at the same time and acting on which subgroups of tanks. As is, the optimisation method developed cannot determine whether a constraint violation in between loading conditions will restrict the prescribed use of pumps.

Furthermore, for hand and optimisation-produced conditions, the pumping time is calculated assuming nominal pumping capacity, and without simulating the flow in pipework. This may be an oversimplification of the phenomena, and it is suggested that the time calculation should be re-evaluated when a more accurate pumping time calculation is possible. A dynamic ballast simulator is currently under development within Allseas, which will be able to simulate the flow of water ballast in the piping and verify the time calculation of optimisation-produced configurations. With more accurate data about the realistic pumping capacity of the system, an improved estimate of the time will be possible. However, the core functionality of the optimisation will remain valid even for a revised pumping time function.

Finally, it is noted that problem constraints played an important part in developing the ballast optimisation model and in finding feasible solutions. When preparing loading conditions by hand, it is a tedious process to find a single ballast configuration that ensures a LC meets hydrostatic and structural constraints. Finding a sequence of configurations, or LCs, which are also optimised for time is even more time-consuming. Thus, it is conjectured that the ballast problem is inherently difficult, not only for the genetic algorithm.
Chapter 6

Conclusions & Recommendations

In this chapter, the conclusions of the current research are presented, and some recommendations are made as to directions for further investigation and application of the proposed method. The conclusions are summarised in section 6.1, wherein the success of the project in meeting the main objective is evaluated and the supporting research questions are addressed. In section 6.2, directions for continued development of the ballast optimisation method are suggested and the further improvement of ballasting time for lifting operations is discussed.

6.1 Conclusions

The main objective of the current project was to develop a ballast optimisation method, to be implemented in a tool, which would facilitate the preparation of loading conditions for Pieter Schelte lifting operations, with the primary goal of minimising total ballasting time for an operation.

As described in this report, a ballast optimisation method was developed and implemented in MATLAB. The optimisation generates ballast tank configurations which meet requirements for draught, trim, heel and structural loading. It has been modelled so as to be able to consider multiple loading conditions at one time, as will be needed for topsides lifting operations with the PS.

The optimisation method was evaluated for a test case of three loading conditions taken from an actual topsides lifting operation. For the test case, the optimisation was able to produce
loading conditions which improved upon the time found for loading conditions calculated by hand. A theoretical threshold was also estimated as a lower bound for ballasting time, based on the minimum water volume transfer required for draught change. The optimisation-produced loading conditions were able to come very close to this theoretical lower bound.

The time required for the optimisation to produce the loading conditions is reasonable. When running on a single core, and for three loading conditions considered simultaneously, the optimisation requires approximately a three hours to converge. When additional loading conditions are considered at once, it is expected that the total runtime of the optimisation will increase. The total runtime will depend on the calculation time for each generation and the total number of generations required for convergence. The runtime may be mitigated with the use of parallel processing. However, with respect to the time needed for analysis, it is noteworthy that a required time of hours, or even days, is still an improvement over the original preparation time which was on the order of weeks.

To conclude, the main objective of this project was met, in that a ballasting optimisation method was developed which achieved the goal of producing loading conditions that sufficiently minimise operational ballasting time. The method allows for the preparation of ballast configurations in an automated way and more quickly than can be done by hand.

The following supporting research questions were considered in addition to the main objective:

1. Which optimisation approach is best-suited to the ballasting problem?
   
   It is difficult to argue in an absolute sense that a single approach is the best possible for the modelling of a problem. On the contrary, many different approaches could prove equally effective. In light of this, it is only possible to examine how successfully the chosen approach, of multi-objective GA optimisation, was able to meet the predefined objective. The approach selected was able to produce feasible ballast configurations which improved upon the ballasting time required by the corresponding hand-calculated configurations. The nature of the GA allowed the development of a working model for the problem without requiring thorough knowledge of the solution space beforehand. The flexibility of the GA and the implementation also allowed for improvement of the model as understanding of the problem increased. Furthermore, the use of a multi-objective approach was critical in maintaining population feasibility and giving an indication of the potential trade-offs between pumping time, required draught and required trim.
For further application of the method, some attempt could be made to mitigate the required runtime. In addition to taking advantage of the potential for parallel processing on multiple cores, a hybrid GA approach might be considered to further reduce runtime. A possible drawback of GAs is the need for tuning of parameters, such as population size, to ensure optimal performance. With additional test cases, the sensitivity of the optimisation’s performance to parameter-values could be determined.

2. **How should functional requirements and design specifications of the existing system be modelled as boundary conditions or constraints?**

The modelling of the ballast optimisation method was presented and discussed in chapter 4. It is possible to briefly reiterate the main conclusions. The current modelling of the problem, with three objectives and three constraints, has been effective in finding feasible solutions and meeting the main objective of the project. Given the choice of GAs for the optimisation, it was found that too stringent constraints can impede the progress of the optimisation and result in worse solutions. For this reason, the most difficult-to-meet constraints, namely draught and trim, were converted to objectives.

Ballasting time was originally considered in terms of water transfer and pumping time, but it was concluded that the consideration of water transfer was superfluous for the overall objective. MATLAB codes were written to calculate ballasting and de-ballasting time, wherein the best time was found by enumerating all possible pumping assignments. The codes were necessary in order to automate the tedious and complex analysis of pumping assignments and ensured that time was calculated for the optimal use of pump rooms. However, the enumerative approach resulted in long runtimes for the de-ballasting code, and required the use of a pseudo time objective within the optimisation. Fortunately, the pseudo-time objective was sufficiently effective in finding good solutions.

3. **How should the problem be modelled in order to find a set of \( N \) acceptable (not necessarily optimal) ballast configurations for \( N \) different loading conditions, which combined give optimal ballasting duration for an operation?**

In order to consider \( N \) different loading conditions (LCs) at one time, it was decided to incorporate multiple LCs into the design variable. In this way, all LCs of interest were acted upon simultaneously by the optimisation. The large number of variables corresponding to additional LCs may pose a problem for convergence time, as mentioned in section 4.3.1, but this should be investigated further.
4. How can the ballast optimisation method be developed in such a way as to allow for general use or parameterisation?

Both the mathematical modelling and the programming implementation (described in chapter 4) were developed so as to allow for general use of the method, for any topsides lifting operation and any number of loading conditions. All the necessary data about the requirements for a loading condition is inputted directly from Allseas’ LCT. With respect to the coding all variables and arrays are parametric.

6.2 Recommendations

6.2.1 Further Development of Ballast Optimisation Method

Refining the Model The modelling of the ballast optimisation could be extended for increased accuracy and completeness, as follows:

- Additional constraints should be added to ensure that the solutions produced are as suited to the real problem as possible. Constraints could be included to evaluate transverse structural loading, free surface effect, damage stability, and structural loading at intermediate filling levels (between loading conditions).

- Within the scope of this thesis, pumps were assumed to operate at approximately nominal capacity when used in parallel. The pumping time calculations should be compared with the pumping time estimation of a dynamic flow model of the PS ballast system, and eventually with actual operational data from the vessel, in order to confirm their validity.

- An improved version of the pumping codes should be developed to consider ballasting and de-ballasting simultaneously, rather than as two distinct optimisations taken in sequence. Every effort should be made to reduce the runtime for the pumping code, so as to incorporate it into the optimisation in place of the pseudo-time objective currently being used.

- The sensitivity of the optimisation to the GA parameters used should be determined, through investigation of their effect on the problem and experimentation with other test cases. Ideally, the parameter values selected should either be robust enough for generic use, or sufficiently mapped and understood so as to be quickly redefined as needed in the model.
• The possibilities for decreasing runtime through parallel computing should be explored.

• The model should be extended to consider tidal compensation as part of the lifting operation.

**Additional Applications**  With the current optimisation method as a basis, several extensions or additional applications beyond minimising ballast time are proposed:

• Extend the optimisation to include one ore more relevant objectives such as minimising power, minimising structural loading or minimising draught. These objectives are further discussed in appendix D.

• Use the optimisation method for other PS operations like jacket-lifting.

• Develop a similar optimisation method for the cargo barge, and consider optimising operations where PS and barge are used at the same time, i.e. offloading of topsides onto cargo barge.

• Eventually, develop a version of the method for real-time use onboard the PS, possibly to interface with the ballast control system.

• Use the optimisation method to investigate the effect of modifications to the pumping system, like improved pumping capacity or segregation valves.

**6.2.2 Further Improvement of Ballasting Time for Lifting Operations**

A theoretical threshold was calculated for ballasting time based on the water volume transfer required to change draught. Assuming that all of the PS’s pumping capacity could be used, without considering pump room assignments, in order to ballast or de-ballast the water, this threshold sets a lower bound to the possible improvement in ballasting time. It has been found that the ballast optimisation method developed can produce solutions very close to the threshold. Consequently, it could be argued that modifications to the ballast system will need to be considered if the times are to be reduced drastically, i.e. cut in half.

It is possible that the ballast optimisation method developed in this thesis could be used to investigate the effect of such modifications to the ballast system. For instance, changes to the pumping system such as increased pumping capacity or additional segregation valves would only
require alterations to the pumping time codes. The rest of the optimisation modelling would
remain the same. Thus, a relatively quick estimation of the improvement to ballasting time
could be attained for each modification proposed. Additionally, in light of the more general
problem of reducing ballasting time for the vessel, operational measures might be considered,
such as simultaneous navigation and ballasting.
Appendix A

Coordinate Systems

Figure A.1 shows the coordinate system used in this thesis, which is taken to be the same as the system used in Allseas’ LCT.

\[\textbf{X-axis} \text{ Taken from the aft perpendicular, positive pointing forward.}\]

\[\textbf{Y-axis} \text{ Taken from the centre line, positive to port side.}\]

\[\textbf{Z-axis} \text{ Taken from the base line, positive pointing upwards.}\]
Appendix B

Ship Hydrostatics

To briefly illustrate the way the ship’s hydrostatics change with changing draught, the graphs in Figure B.1 are included. The graphs give the changing values of displacement, LCB and LCF, for draughts in the PS’s lifting range.

Figure B.1: Hydrostatics of Pieter Schelte for lifting draught range, 12-25 m
Appendix C

Pumping Capacity Calculation

The nominal pumping capacity of each centrifugal ballast pump of the PS is $3100 \text{ m}^3/\text{h}$. In each pump room, two pumps are configured in parallel. Theoretically, the total capacity of two identical parallel pumps is found to be nearly twice their individual capacities. Within the scope of this thesis, the pumping capacity of two parallel pumps was taken to be exactly twice their independent capacity. Justification for the doubled capacity is included in the following chapter, based on analyses conducted within Allseas and by external consultants.

The pump curves are plotted based on data from the manufacturer. Given the performance curve for a single pump (head [m] v. flow rate [$\text{m}^3/\text{h}$]), the curve for two centrifugal pumps is found by summing the flow rates at each head. The point of operation of the pumps is found at the intersection of the combined pump curve and the system curve. The system curve is a parabolic curve that defines the relationship between the flow quantity through the system and the “resistance” offered by the system. It is completely independent of the pumps acting on the system.

Within Allseas, the system curve of the ballast system was approximated by considering a theoretical system consisting of a “general” tank. The general tank was defined with a capacity similar to ballast tank 1102, which has the maximum tank height of 30 m. The location of the general tank was assumed to be that of ballast tank 900 (with LCG=134.25 m, TCG=0 m, and VCG=9.35 m at 100% filling). This tank location was selected in order to consider one of the “worst-case” scenarios in terms of distance from the pump room (P.R. 6) and resistance due to fittings and length of pipework. For the given tank, the system curves were calculated for a
range of filling levels, namely fillings of 2 and 25% at an external draught of 11.8 m, and 50 and 75% at an external draught of 16.91 m.

The pump and system curves resulting from the analysis are shown in figure C.1. In the figure, the “average” system curve intersects the combined pump curve at approximately $6400 \text{ m}^3/\text{h}$. Since the general tank evaluated is taken to be worst-case in terms of resistance, the system curves for other tanks are expected to encounter less resistance (i.e. have a less-steep slope, intersect pump curve further to the right). Thus, it was assumed that using the total capacity of $6200 \text{ m}^3/\text{h}$ for the two pumps in parallel would be reasonably accurate for the preliminary evaluation of pumping times for each transition between loading conditions.

![Figure C.1: Pump and system curves for ballast system analysis and pump capacity calculation (PS WB Head Calculation, 2014)](image)

The exact analysis of the ballast system resistance was carried out using an industry-standard fluid dynamics package, Fluidflow (Fisher, 2007). The goal of the analysis was to ensure that the basic design specification of the ballast system would be satisfactory from pumping aspects, as required to meet the vessel’s operational parameters. The ballast system was modelled within the Fluidflow software, and the performance of the pumps was assessed against system flow resistance characteristics.

In the analysis, the ballast system was assumed symmetrical about the ship’s centreline, and only the starboard side of the main ring was modelled. An emphasis was placed on evaluating “worst-case scenarios”, with respect to the effectiveness of the system when dealing with tanks furthest from their respective pump room. From a fluid dynamics aspect, it was concluded
that the ballast system is suitable for satisfying the ballast transfer requirements of the system (Fisher, 2007).
Appendix D

Additional Optimisation Objectives

It was decided that the optimisation model would initially focus on the objective of minimising ballasting time. Given the model developed, modifications could be made to consider these additional objectives.

- **Minimise structural loading** An additional optimisation objective is to minimise structural loading of the hull. As discussed in section 4.1.2.3, in this project we consider only longitudinal structural loading through longitudinal shear force and bending moment. Structural loading is particularly critical for topsides removal operations after the lift has been completed, when the PS is carrying the topsides with the TLS. At this point, the PS is sailing-out from around the substructure (jacket) at a very shallow draught, to ensure that no collision occurs between the topsides and the substructure.

- **Minimise ballasting power** Minimising the power required for ballasting during the operation is considered as another optimisation objective. Big draws on the available power are made by the dynamic positioning (DP) and active-motion compensation (AMC) systems. The DP system is vital when the PS is sailing-in around the platform and sailing-out from the platform. AMC capabilities are critical during connection of the TLS beams to the platform.

- **Minimise draught** The relevance of minimising draught may not be as readily apparent as that of the previous objectives. This objective is of interest when investigating the connecting draught of the vessel during lifting. When lifting a topsides, it is preferable to attach the topsides lift system (TLS) to the platform legs at a point as close as possible
to the underside of the topsides, which means at shallower PS draughts depending on the platform’s air gap. Of course, the shallower draughts would need to meet the remaining hydrostatic and structural constraints. Thus, it would be valuable to be able to identify the minimum draught for a loading condition given these considerations.

The operational phases in which each of the aforementioned objectives are considered most critical are summarised in Table D.1.

<table>
<thead>
<tr>
<th>Operational Phase</th>
<th>Ballast Needs</th>
<th>Critical Optimisation Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transit → Sail-In</td>
<td>ballast</td>
<td>time</td>
</tr>
<tr>
<td>Sail-In → Connect</td>
<td>de-ballast</td>
<td>time, power</td>
</tr>
<tr>
<td>Connect → Pre-Tension</td>
<td>de-ballast</td>
<td>time</td>
</tr>
<tr>
<td>Pre-Tension → Fast-Lift</td>
<td>de-ballast</td>
<td>time</td>
</tr>
<tr>
<td>Fast-Lift → Sail-Out</td>
<td>de-ballast</td>
<td>time, structural, power</td>
</tr>
<tr>
<td>Sail-Out → Transit</td>
<td>ballast</td>
<td>time, structural</td>
</tr>
</tbody>
</table>

Table D.1: Critical objectives per operational phase

Having established that the objectives enumerated are those most relevant to the ballast optimisation problem, it must now be decided whether they will be optimised individually or in combination with one another. An important consideration, particularly for the latter case, is which of the objectives are conflicting. Two objectives may be considered conflicting when they are inversely proportional to each other. For instance, minimising the vessel’s draught will mean increased hull structural loading, thus the objectives *minimise draught* and *minimise hull structural loading* are conflicting. Table D.2 gives an overview of relationships between the proposed objectives.

<table>
<thead>
<tr>
<th></th>
<th>Time</th>
<th>Structural</th>
<th>Power</th>
<th>Draught</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>–</td>
<td>May conflict</td>
<td>Conflicting</td>
<td>–</td>
</tr>
<tr>
<td>Structural</td>
<td>May conflict</td>
<td>–</td>
<td>Uncertain</td>
<td>Conflicting</td>
</tr>
<tr>
<td>Power</td>
<td>Conflicting</td>
<td>Uncertain</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Draught</td>
<td>–</td>
<td>Conflicting</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Table D.2: Potential conflicts between objectives
Appendix E

Dimensions of Ballast Tanks

Figure E.1: Maximum dimensions of ballast tanks
Bibliography


