



FACULTY OF ENGINEERING AND THE ENVIRONMENT

MENG SHIP SCIENCE

GDP 40

Final Report

The Development of an Autonomous Self-Propulsion Vessel for
Powering and Manoeuvring Tests in an Uncontrolled
Environment

Authors:

Mr. Enrico ANDERLINI
Miss. Heather CROSSLEY
Mr. James HAWKES
Mr. Hieu LE
Mr. James MOZDEN
Miss. Kimberley NEALE
Mr. Ben THORNTON

Supervisors:

Prof. Stephen TURNOCK
Dr. Dominic HUDSON
Dr. Alex PHILLIPS

April 15, 2013



Photo: The model performing a circle manoeuvre.

Abstract

With rising fuel costs and a changeable climate, ship owners are always looking for ways to improve the efficiency of their old ships. BP Shipping, the sponsor for this project, aims to improve the efficiency of their fleet of twelve Virtue Class tankers by adding retro-fit appendages to their vessels or by changing the operational conditions, such as trim, to decrease the required power.

There are many characteristics of a ship's behaviour that may be altered when attempting to improve the efficiency, not just calm water powering performance. The sea-keeping, manoeuvring and stability characteristics may be adversely affected, and the ship's performance in a seaway may also be changed.

Normally, testing for these characteristics would require the use of expensive ocean basins, but this project aimed to remove the need for these facilities by enabling the tests to be performed on natural bodies of water such as lakes, rivers and coastal waters.

In order to achieve this, a 1/60 scale model of a Virtue tanker was adapted into an autonomous model, capable of navigating itself through test routines and recording information about its performance using on-board sensors.

The testing focused on trying to measure manoeuvring and powering characteristics in calm water, to compare against towing tank experiments which were also performed in calm water. In order to facilitate testing in a seaway, a wave buoy had to be developed which could measure the sea-state.

Although some efficiency improvements were tested in this project, the overall aim was to create a test platform suitable for in-depth testing of such improvements in the future.

By the end of the project, a complex test platform had been built into the existing model hull form. Aside from some issues with the torque sensor and speed sensor which could be rectified in the future, the system performed well during lake testing. It executed test matrices with high repeatability and collected accurate data which was used to assess its manoeuvring and powering characteristics. The wave buoys were also successful in measuring directional wave information, although they were tested independently.

The project shows that it is possible to bypass expensive tank facilities by developing an autonomous model, although more work is required before the quality of the results can be fairly compared.

Acknowledgements

The group would like to acknowledge the three supervisors for this project, for their continued support throughout the year: Professor Stephen R. Turnock, Dr. Dominic Hudson and Dr. Alex Philips.

The group would also like to thank BP for their kind sponsorship and cooperation, without which this project would not have been possible.

The group are grateful to Mr. Chris Richardsen of QinetiQ, for his outstanding cooperation and generosity in organizing student-priced towing tank facilities; and Gordon Maxwell from the Warsash Maritime Academy at Timsbury Lake for allowing the group to use their lake, workshop and support boats for a week.

The group would like to extend thanks to certain members of the University of Southampton: Mr. William Ashley, for his assistance with rapid prototyping; Mr. Philip Herring for his expertise and guidance in foam-sculpting; and the Institute for Sound and Vibration for their patient guidance with electronics.

CONTENTS

1	Introduction	1
1.1	Introduction	1
1.2	Aim	3
1.3	Objectives	4
1.4	Report Structure	5
1.5	Literature Review	6
1.5.1	Developing a Self-Propulsive Model	6
1.5.2	Developing a Self-Measuring Model	7
1.5.3	Developing an Autonomous Model	9
1.5.4	Testing	10
1.5.5	Wave Measurements	11
2	Preliminary	14
2.1	Evaluating Efficiency	14
2.2	Efficiency Improvements	16
2.3	Preliminary Calculations	22
2.3.1	Resistance and Scaling	22
2.3.2	Rudder Discussion	27
3	Design & Build	31
3.1	System Overview	31
3.2	Power Train	33
3.2.1	Design Concept	33
3.2.2	Motor Selection	34
3.2.3	Thrust Measurement	34
3.2.4	Additional Components	35
3.2.5	Power Train Unit	35
3.2.6	Shaft Alignment Procedure	36
3.3	Torque Dynamometer	39
3.3.1	Design of the Structure	39
3.3.2	Flexure Design	40
3.3.3	Dynamometer Electronics	42
3.4	Electronics	44
3.4.1	Overview	44
3.4.2	Powering	46
3.4.3	Module: Master	48
3.4.4	Module: Motor Controller	50
3.4.5	Module: Thrust Block	55
3.4.6	Module: Bilge Pump	57

3.4.7	Module: Rudder Controller	58
3.4.8	Sampling	59
3.4.9	Summary	60
3.5	Rudder System	61
3.5.1	Control method	62
3.6	Autonomy System	65
3.6.1	Introduction	65
3.6.2	Global Overview	66
3.6.3	Compass	67
3.6.4	GPS	69
3.6.5	Arduino	70
3.6.6	Heading Controller	71
3.6.7	Nodes dedicated to Testing	74
3.6.8	Smach States Machine	76
3.6.9	Back-seat Driver	78
3.6.10	High Level Library	78
3.6.11	Logger and Average-calculator	79
3.6.12	Shut-down Function	79
3.6.13	Launch File	80
3.6.14	Manual Control	80
3.6.15	Communications with Ashore Computer	81
3.7	Miscellaneous	82
3.8	Weights And Trim	84
3.8.1	Loading Conditions	84
3.8.2	Inclining Experiment	85
3.8.3	Trim Conditions	89
3.8.4	Weight Positioning	91
3.8.5	Moment of Inertia	91
4	Wave Buoys	94
4.1	Introduction	94
4.2	Design	96
4.2.1	Sensor Requirements	96
4.2.2	Electronics	96
4.2.3	Buoys	100
4.3	Data Processing	104
4.3.1	IMU Processing	104
4.3.2	Wave Spectra Processing	108
4.3.3	Software	111
4.4	Testing	113
4.4.1	OrcaFlex	114
4.4.2	Lamont Tank	115
4.5	Results	117
4.5.1	Dean Stream Waves	117
4.5.2	JONSWAP Waves	120

4.5.3	Spectral Densities	120
4.5.4	Extreme Tank Conditions	124
4.5.5	Mooring Tension	126
4.5.6	WABDAP Results	127
4.5.7	Other Results	127
4.5.8	Sources of Error	128
4.5.9	Summary	128
5	Testing	131
5.1	Towing Tank Tests: QinetiQ	131
5.1.1	Test matrix	131
5.1.2	Bollard Pull	133
5.1.3	Shaft Efficiency	135
5.1.4	Naked Hull	137
5.1.5	Self-Propulsion Tests	138
5.1.6	Summary	140
5.2	Lake Testing: Timsbury Lake	141
5.2.1	Bollard Pull	142
5.2.2	Straight Line Tests	144
5.2.3	Manoeuvring Trials	145
5.2.4	Summary	148
6	Results and Evaluation	149
6.1	System Performance	149
6.1.1	Power Train	149
6.1.2	Torque Dynamometer	149
6.1.3	Electronics	149
6.1.4	Autonomy System	150
6.2	Sensor Performance	151
6.3	Resistance and Propulsion	154
6.3.1	QinetiQ: Naked-hull tests	154
6.3.2	QinetiQ: Self-Propulsion Tests	155
6.3.3	QinetiQ: Comparison of Different Loading Conditions	157
6.3.4	QinetiQ: Discussion	160
6.3.5	Lake Tests (Straight Run)	161
6.3.6	Error Analysis	166
6.3.7	Future Testing	167
6.4	Manoeuvring	168
6.4.1	Full-Scale Ship Results	168
6.4.2	Model Ship Results	168
6.4.3	Discussion	172
6.5	Rudder Comparison	177
6.5.1	Results	177
6.5.2	Data Analysis	177
6.6	Summary of Results	179

7 Conclusion	180
7.1 Group Management	181
7.2 Budget Summary	181
8 Further Work	183
8.0.1 Power Train	183
8.0.2 Torque Dynamometer	183
8.0.3 Electronics	183
8.0.4 Autonomy System	184
8.0.5 Weights and Centres	184
8.0.6 Wave Buoys	185
8.0.7 Testing	185
9 Bibliography	186
10 Appendix: System Overview	192
11 Appendix: Power Train	193
12 Appendix: Torque Dynamometer	194

LIST OF FIGURES

1.5.1	Previous self-propulsion rig from 2005	6
1.5.2	Thrust measurement techniques	8
2.2.1	Efficiency improvements	17
2.2.2	Grim van wheel	19
2.2.3	Schneekluthduct	19
2.2.4	Promas rudder system	20
2.3.1	Full scale sea-trial results	23
2.3.2	Zigzag manoeuvring	25
2.3.3	Comparison of rudder lift coefficients	29
2.3.4	Zigzag manoeuvring	30
3.1.1	Comparison between 3-D model and completed model	32
3.1.2	3-D model displaying internal layout of the model	32
3.2.1	Initial concept diagram for power train assembly	33
3.2.2	Photographs of completed thrust load cell arrangement	35
3.2.3	Photographs of completed power train unit	36
3.2.4	Photographs detailing each stage of the shaft alignment procedure	38
3.3.1	3D CAD drawing of completed dynamometer	40
3.3.2	Schematic of the amplifier circuit.	42
3.3.3	Schematic of the Wheatstone bridge wiring	43
3.3.4	Photograph of the completed Wheatstone bridge wiring	43
3.4.1	Arduino Pro Mini MCU and perfboard	45
3.4.2	Electrical overview	45
3.4.3	Electronics enclosure	47
3.4.4	Master module	48
3.4.5	Motor module	51
3.4.6	Rotary encoder	52
3.4.7	Motor module	56
3.4.8	Amplified signal responses from the load cell	57
3.4.9	Bilge pump module	58
3.4.10	Rudder controller module	58
3.5.1	Design of the spade rudder for the model	62
3.5.2	Rudder profile for the large rudder	62
3.5.3	Comparison of the two rudders	63
3.5.4	Data transfer between central unit and rudder control unit	63
3.5.5	Rudder system's diagram	64
3.5.6	Picture of the model's rudder system in operation	64
3.6.1	Main computing unit	65

3.6.2	Diagram of the control system	68
3.6.3	Pelicase box and its components	69
3.6.4	GPS unit	69
3.6.5	Simple unity feedback system diagram	72
3.7.1	Superstructure CAD and model	83
3.8.1	The inclining rig	86
3.8.2	Heeling Graph	88
4.1.1	Example directional wave spectrum	95
4.2.1	Wave buoy components	98
4.2.2	Wave buoy PCB design	98
4.2.3	Complete wave buoy electronics	99
4.2.4	Buoy design process	100
4.2.5	CAD profiles of buoy hulls	101
4.2.6	Mooring line arrangement for the buoys	103
4.2.7	Styrofoam buoy hulls	103
4.3.1	Example of magnetometer calibration process	105
4.3.2	Example of accelerometer cumulative error and the correction process	108
4.3.3	Sensor outputs in WABDAP	112
4.3.4	3D output of WABDAP	113
4.3.5	Calibration interface of WABDAP	114
4.4.1	An example OrcaFlex cylinder breakdown	115
4.4.2	Experimental setup for the Lamont tank tests	116
4.5.1	elevation vs. time for all buoys in OrcaFlex	118
4.5.2	Elevation vs. time for all buoys in the Lamont tank	119
4.5.3	Elevation vs. time for all buoys in OrcaFlex	121
4.5.4	Elevation vs. time for all buoys in the Lamont tank	122
4.5.5	Elevation vs. time for all buoys in OrcaFlex	123
4.5.6	Spectral densities for the BS buoy	124
4.5.7	Elevation vs. time in the Lamont tank	125
4.5.8	Orcafex mooring arrangement simulation	126
4.5.9	Effective mooring line tension during OrcaFlex simulations	129
4.5.10	2D spectrum for the SC buoy	130
4.5.11	3D spectrum for the SC buoy	130
5.1.1	Thrust block calibrations	133
5.1.2	QinetiQ tank thrust and RPM ² relationship	134
5.1.3	Shaft efficiency experiment	135
5.1.4	Diagrammatic representation of a self-propulsion test result [Molland et al., 2011].	138
5.2.1	Lamont tank carriage calibration graph	142
5.2.2	Lamont tank bollard pull thrust calibrations	143
5.2.3	Circle manoeuvre diagram	145
5.2.4	Zig-zag manoeuvre heading and rudder angles	146
5.2.5	Derivation of K from a zigzag manoeuvre	147
5.2.6	Derivation of K/T from a zigzag manoeuvre	147

6.2.1	Fourier series of an RPM, thrust, current and speed data set	151
6.3.1	Full-load self-propulsion results	156
6.3.2	Self-propulsion test results	158
6.3.3	Experiment results for the naked-hull tests	159
6.3.4	Propeller thrust vs RPM	163
6.3.5	Comparison of power requirements for different bows	166
6.4.1	Model path during a circle manoeuvre	170
6.4.2	Model heading during a circle manoeuvre	170
6.4.3	Model path during a circle manoeuvre	171
6.4.4	Model heading during a circle manoeuvre	171
6.4.5	Heading and rudder angle during a zigzag trial	172

LIST OF TABLES

2.3.1	Principal particulars	22
2.3.2	Full-scale power requirements and non-dimensional resistance coefficients	23
2.3.3	Model propeller particulars	24
2.3.4	Estimation of the model self-propulsion point	26
2.3.5	Estimating resistance using different form factors	27
2.3.6	Rudder principal dimensions	27
2.3.7	Rudder torque calculations	28
2.3.8	Lift coefficients generated by XFLR5.	29
3.2.1	Operating requirements for the motor	34
3.3.1	Properties of the chosen strain gauge	40
3.3.2	Flexure design calculations	41
3.3.3	Flexure design calculations	41
3.3.4	Comparison of 4- and 8-gauge Wheatstone bridge circuits	41
3.3.5	Power consumption of torque dynamometer components	42
3.3.6	Battery Properties Comparison.	42
3.6.1	PID controller responses to different inputs	72
3.6.2	PID controller gains and filtering intervals	74
3.6.3	ROS terminal commands	81
3.8.1	Particulars at full and model scale for loaded and ballast conditions.	84
3.8.2	Model scale draught mark readings.	85
3.8.3	Table of Weights for Inclining Experiment.	87
3.8.4	Inclining Experiment Results.	89
3.8.5	Full scale trimming limits obtained from the loading computer.	90
3.8.6	Trim Condition Particulars	90
3.8.7	Final model scale loading conditions	91
3.8.8	Lightship Weights Table.	91
3.8.9	Weight Positions - Ballast Condition.	92
3.8.10	Weight Positions - Full Load.	93
4.1.1	Desired wave heights for model-scale testing	95
4.2.1	Buoy weights and dimensions	101
4.2.2	Buoy component weights	101
4.4.1	Buoy calculations for Orcaflex	115
5.1.1	Test matrix: QinetiQ	132
5.1.2	Test matrix: QinetiQ (continued)	132
5.1.3	Test matrix: QinetiQ (continued)	132
5.1.4	Test matrix: QinetiQ (continued)	132

5.1.5	QinetiQ thrust calibration coefficients.	133
5.1.6	Qinetiq bollard pull data (full load, level trim).	134
5.1.7	Qinetiq bollard pull data (ballast load, level trim).	134
5.1.8	Motor current and shaft efficiency during bollard pull tests	136
5.2.1	Lamont tank carriage calibration coefficients.	142
5.2.2	Lamont tank bollard pull data.	143
5.2.3	Lamont tank thrust calibration coefficients.	143
6.2.1	A sample of straight-line lake-tests showing statistical information	153
6.3.1	Model full load power requirements	154
6.3.2	Form factor derivation.	154
6.3.3	Typical results for the FL_{DT} condition.	155
6.3.4	Interpolation value at self-propulsion point.	155
6.3.5	Propulsive efficiency η_D and delivered power P_D for three different form factors.	157
6.3.6	Model effect power requirements	158
6.3.7	Full load effective power requirements	159
6.3.8	Ballast effective power requirements level trim	159
6.3.9	Ballast effective power requirements -2m trim	159
6.3.10	Typical results for the FL_{T1} condition.	159
6.3.11	Typical results for the BL_{DT} condition.	160
6.3.12	Typical results for the BL_{T1} condition.	160
6.3.13	Calculation propulsive efficiency η_D	160
6.3.14	Interpolation for the performance of the model at 1.029 m/s.	163
6.3.15	Data from open-water chart.	163
6.3.16	Delivered power requirement for ship in waves calculated by traditional method.	164
6.3.17	Delivered power requirement for ship in waves calculated by form factor approach.	164
6.3.18	Direct comparison between bows using mean values.	165
6.3.19	Performance of the model with normal bow.	165
6.3.20	Performance of the model with 1st alternative bow.	166
6.3.21	Performance of the model with 2nd alternative bow.	166
6.4.1	Summary of results for two circle manoeuvres	168
6.4.2	Circle Manoeuvre Settings	169
6.4.3	Circle manoeuvre results	169
6.4.4	Circle manoeuvre results (continued)	170
6.4.5	Circle manoeuvre turning radius	170
6.4.6	Zigzag manoeuvre test matrix	171
6.4.7	Zigzag manoeuvre results	171
6.4.8	Zigzag manoeuvre results (continued)	172
6.4.9	Nomoto coefficients calculated from zigzag manoeuvres	172
6.5.1	Zigzag manoeuvre results	177
6.5.2	Zigzag manoeuvre results (continued)	177
6.5.3	Zigzag manoeuvre results (continued)	177
6.5.4	Nomoto coefficients from zigzag manouevres	178
7.2.1	Summary of funds and expenditures	182

INTRODUCTION

In this chapter, the motivation for the project will be introduced; the project will be broken down into detailed aims and objectives and a literature review will be performed.

1.1 Introduction

[HAWKES & MOZDEN]

Whilst ships are widely regarded as the most fuel-efficient means of bulk transportation [IMO, 2012a], they are also the most widely employed: ninety percent of all worldwide bulk cargo is transported by sea [IMO, 2012b]. The slightest improvements to ship efficiency could result in huge monetary gains for the owners, especially with rising fuel costs; and reduce the carbon footprint of the industry significantly.

The International Maritime Organization (IMO) has introduced the Energy Efficiency Design Index (EEDI) for shipping, making the ship industry the first to agree to a global carbon dioxide reduction strategy: from the 1st of January 2013, new ships must be built according to a stringent efficiency index. Although there is hot debate about the unfair application of the index to different parts of the industry, it does provide a well-recognized means of judging efficiency, η .

$$\eta = \frac{\text{Output}}{\text{Input}} = \frac{\text{Deadweight} \times \text{Speed}}{(\text{Power} \times \text{Specific Fuel Consumption}) + \text{Other Costs}} \quad (1.1.1)$$

When it comes to ship design, many of these factors are immutable. The deadweight, additional costs and ship speed are often determined by the economies of the company or market and the specific fuel consumption limited by engine technology. This leaves one remaining parameter, the vessels power requirements.

Many companies, faced with increasing fuel costs, are trying to reduce the power requirements of their older vessels by changing the operating conditions of the vessel, or fitting retro-fit devices to the hull - to make the vessel more efficient. One of the major problems with such modifications is that their performance and influence on other aspects of the ship's behaviour is difficult and expensive to predict. In this project, the effect of efficiency improvements on a scale model of a BP Virtue Ice Class tanker will be investigated using novel, experimental methods which aim to reduce the cost of these predictions.

In developing these methods, it is important to consider the characteristics of the vessel that should be measured. In searching for an optimum design this could include powering (hence efficiency), manoeuvring and sea-keeping characteristics, to mention but a few.

During the design cycle for a vessel, there are various methods for predicting calm-water powering characteristics: using empirical methods such as standard series data; theoretical methods such as computational fluid dynamics; or experimental methods such as model testing. These methods are all reasonably straight-forward, since the problem is steady-state - *i.e.* constant in time.

When considering manoeuvring, sea-keeping and powering characteristics in a seaway, the problem becomes dynamic (*i.e.* changes in the time-domain). Most dynamic problems are avoided in classic ship design, typically because: most empirical calculations are too simplistic to handle dynamic behaviour; numerical methods become prohibitively computationally expensive; experimental methods become very expensive.

For the dynamic tests, model testing is currently regarded as the most reliable method. Typically these tests involve multi-directional wave machines, complicated tow-post arrangements and/or large ocean basins, which are very costly. One of its nearest competitors, computational fluid dynamics, may require weeks of computational time to solve even the simplest of dynamic problems without simplifications - such as assuming inviscid flow or removing the propeller.

This project aims to reduce this problem, by using lakes, rivers and other inland or coastal waters as a replacement for expensive towing tanks. The major advantage is the cost saving, but there are numerous disadvantages. Firstly, the environment is uncontrollable and relies on favourable weather conditions; the model can no longer be tethered to the a tow post, so motions (such as heave and pitch) and forces (such as resistance) can no longer be measured against a fixed reference; and the model must be self-supporting - carrying its own power source, control system, signal processing system and wireless communications to shore.

In order to test whether this technique is reliable and useful, typical efficiency improvement methods (henceforth EIs) will be investigated. This could include retrofit devices or operating conditions; in fact, a parallel project is also investigating alternative bow-forms which could be tested. It is generally believed that an achievable efficiency gain would be of the order of 2-10% through the use of EIs, which can be measured via the powering characteristics.

As well as efficiency improvements, the IMO also specifies various manoeuvrability standards for all vessels constructed after 2004 [IMO, April 2005]. Indeed, the project's primary sponsor (BP) has raised particular concerns over the manoeuvring of their vessels after design modifications. So as a further test of the functionality of the "open-water" techniques, manoeuvring trials will be performed.

Like other projects of this nature, the test platform will need to be extensible so that it can easily be adapted and improved in the future. For example, the vessel could be fitted with advanced inertial measurement sensors for sea-keeping characteristics; or fitted with surveying equipment for research less related to the vessel itself.

This project will continue work begun by a previous project which looked at a low cost approach to improving the energy efficiency of existing ships. The test platform developed was a 1/60th scale model based on an existing ice class tanker, suitable for self-propulsion towing tank tests. This model will be adapted by replacing the self-propulsion rig; installing a full suite of sensors to measure the powering and manoeuvring characteristics; building an advanced control system; and developing an autonomous control unit.

1.2 Aim

[HAWKES]

The main aim of the project was originally to investigate efficiency improvement methods for a tanker operating in a seaway. To be able to perform these investigations it was necessary to build a test platform that could be operated free from expensive towing tanks and ocean basins. In fact, the task of building such a complex test platform became the principal part of the project, and the aim was revised. The aim of the project became:

“To develop an autonomous self-propulsion model capable of self-measuring small-scale changes in powering and manoeuvrability characteristics in an open-water environment - based upon a 1/60 scale model tanker.”

This aim can be broken down into its individual components. Firstly, consider the open-water environment - this implies lakes, rivers or coastal waters which may provide an environment (*i.e.* wind, waves) which is applicable to the model-scale ship. The main issue with using these environments is that they are uncontrollable, unlike a towing tank or ocean basin. The advantage of using open-water environments is that they are practically free to use, compared to expensive tank facilities. However, it will be necessary to build devices, such as wave buoys, that can accurately assess the environment.

The changes in efficiency are expected to be small - typically smaller than environmental ‘noise’ in the experiments, such as gusts or particular wave patterns. This does not reflect badly on the open-water testing method, since this noise is encountered at full-scale too. However, to obtain reliable performance predictions the tests must be repeated many times to filter out the noise. To ensure high repeatability the vessel must be made autonomous in an attempt to remove human (‘driver’) error - where autonomous implies operating without continuous human guidance in an unstructured environment. Correctly implementing an autonomous system is difficult, but is another step towards removing the need for expensive towing tanks and ocean basins.

In the past, the main sponsor for this project (BP) has used appendages on the full-scale ship in a bid to improve propulsive performance. However, concerns were also raised over the manoeuvrability of their vessels when these appendages were installed because many of their captains had reported significant changes in handling. This sets the precedence to measure both manoeuvring and powering characteristics. Since the vessel is to be run in large open-water environments, proper manoeuvring tests should be undertaken using the autonomous control system.

Since the vessel will be operating in open-water and not connected to a tow-post, it must be capable of measuring its own characteristics - including thrust, shaft power, position and orientation. Normally the signal processing of the sensors (*i.e.* strain gauges) would be off-loaded to expensive, bulky, on-carriage analog-to-digital converters and computers; it will be a significant challenge to reimplement this on-board the vessel, especially considering that the model must also carry its own power source.

1.3 Objectives

[HAWKES]

Following from the previous section, the aim can be broken down into significant objectives.

- Develop a means of measuring the test environment.
- Develop the model into a vessel that is:
 - (a) self-propelled (*i.e.* providing its own powering and steering, with suitable controls);
 - (b) self-measuring (*i.e.* able to measure its key characteristics);
 - (c) autonomous (*i.e.* able to use its controls systematically to perform repeatable tests).
- Perform dry calibration of relevant sensors and control systems.
- Perform towing-tank tests to:
 - (a) obtain reliable results which can be used to infer the performance of the vessels sensory system and to perform any necessary calibrations;
 - (b) obtain results for at least one efficiency improvement method, as a standalone result.
- Perform testing in an open-water environment to:
 - (a) show that tests can be run with high repeatability using the autonomous system;
 - (b) obtain results for the unmodified vessel as a comparison to the towing tank tests;
 - (c) obtain results for the vessel with various appendages or modifications to measure small changes in manoeuvring and powering.

1.4 Report Structure

[HAWKES]

The report begins with a literature review which investigates different techniques for completing the objectives outlined above. This section represents the early stages of research, before a decision was made on the relevant methodologies for each objective. More honed research followed, but for the purposes of the report this is kept within the relevant sections.

The following chapter (2), details preliminary calculations and research that were required. For example: estimations of the model resistance were needed for the design of the power train.

Chapter 3 consists of a section for each major part of the model, and the first section of this chapter provides an overview of the whole system. As a parallel to this chapter, Chapter 4 covers the design and build of the wave buoys.

The testing process for both the towing tank and lake is detailed in Chapter 5, with results and evaluation in the following chapter. The report finishes with a conclusion and suggestions for further work.

1.5 Literature Review

[MOZDEN et AL.]

This section discusses the research performed prior to starting the rest of the project, by focusing on each objective from Section 1.3. There are various methodologies which could be followed to fulfil each objective, understanding the advantages and disadvantages of each will help to avoid potential errors and prevent a repeat of previous mistakes.

1.5.1 Developing a Self-Propulsive Model

[MOZDEN]

For the purpose of this report, a self-propulsion or self-propelled model is one which provides its own powering and steering based on input commands given by a control system. Self-propulsion models are not a new development, they have been used for many years to investigate ship speed and powering requirements; many systems have been built to propel model ships with varying degrees of success. There are two key areas: the powering system to drive the ship forwards, and a steering system to enable the ship to perform controlled manoeuvres.

A previous self-propulsion rig was built in 2005 by Bowen et al. [2005] (Figure 1.5.1), it was designed as a self-contained unit capable of measuring thrust and torque which could be transferred between suitable hulls. The system was attached to a solid base plate and supported the shaft system with several plain bearings. The motor, a 12 V DC motor from a Black & Decker drill, was too large to be fitted in-line with the shaft so was attached via pulleys and a belt drive. The choice of motor had been limited by cost, it was important to find one which had a relatively large torque to overcome shaft resistances and a simple speed control system; a drill motor was chosen and linked with a 4-speed gearbox which was used to step the motor output so that the required torque could be produced for the RPM. The motor was not ideal but was the only option with the available funding. A smaller, more powerful motor would be able to sit in-line with the shaft, hence minimizing mechanical losses by belt slippage and making the whole rig smaller and lighter [Bowen et al., 2005].

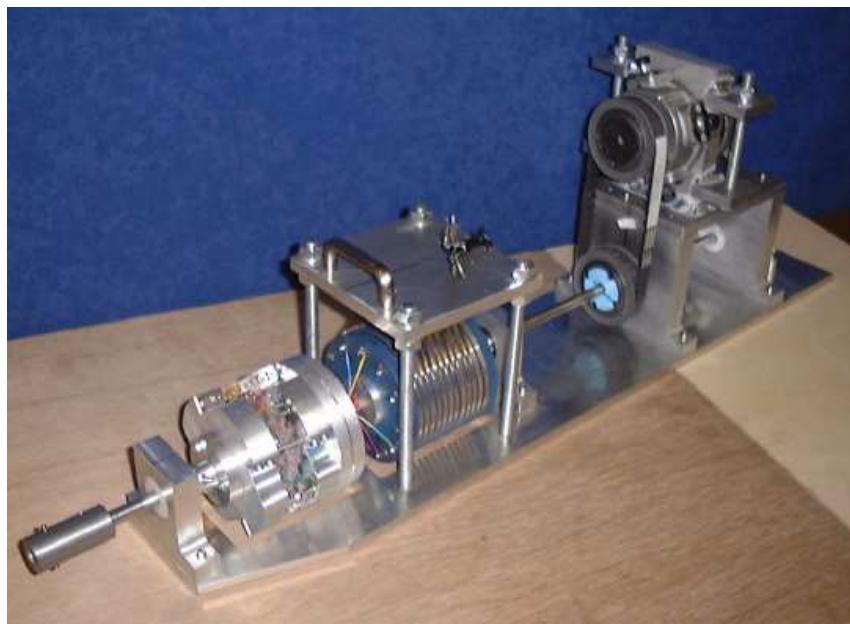


Figure 1.5.1: Self-propulsion rig used in 2005 GDP.

Even with the issues outlined, the system was reused in 2012 for an energy efficiency investigation

looking at cost-effective retro-fit devices to reduce the resistance around a hull. Using the 2005 self-propulsion rig in a tanker hull form, the rig shaft line was 20 mm higher than the stern tube even though it was positioned as low as possible in the hull. To combat this the shaft was angled, inducing a vertical thrust component and hence incurred losses; a three piece step system could have removed the angle, but this would have resulted in increased friction in the system [Collison et al., 2012].

There are a number of key areas to consider when developing a self-propulsion model; the main focus must be on ensuring that the rig does not inhibit the quality of the measuring system and hence it needs to represent a full scale ship as accurately as possible. The main issues discovered in the previous system were in finding a suitable motor and gearing system capable of producing high torque at low RPMs; and minimising frictional loses. It is important to achieve a level shaft line too, as this will ensure there is only a horizontal thrust component acting on the ship and is representative of the full-scale vessel.

1.5.2 Developing a Self-Measuring Model

Thrust Measurements

[MOZDEN]

Free running models need to be able to measure their behaviour using on board sensors. Unlike models used in a towing tank, there is no external system, such as a carriage, attached to the vessel - meaning that there is no reference frame the model can be measured against.

Self-propulsion tests are one of the main techniques used to evaluate the performance of a vessels propulsive system; they measure the thrust and torque produced by the propeller, by comparing the shaft power to the hull resistance as it travels down a towing tank.

On a propeller shaft, thrust is the force trying to push the shaft forwards through the hull. The only reason it does not move, in relation to the hull, is because the reaction from the bearings hold it in place and transfer the force to the hull. There are various techniques used for measuring the thrust produced by a rotating shaft, these are outlined below and are shown in Figure 1.5.2:

- Strain gauges

A strain gauge measures the strain of an object, which is to say the deformation of a body. There are several types of strain gauges, but the most common consists of a metallic foil mounted on a flexible backing which is attached to the object. As the object deforms, the foil is deformed causing its electrical resistance to change, this change is measured by a Wheatstone bridge circuit and equates it to strain. Multiple strain gauges can be attached to opposite sides of an object; as the object bends one side is put into tension, increasing the resistance because the area of the gauge narrows, whilst the opposite side compresses, decreasing the resistance as the area increases.

To measure thrust, the stain gauge arrangement can be fitted in a load cell; generally four strain gauges are used, although one or two strain gauge systems are also possible. By fixing one end of the load cell and allowing the other end to deflect, the strain gauges can monitor the linear deformation caused by the thrust acting on the shaft [A & D Ltd., 2012].

An alternative method is to use a thrust dynamometer, which is essentially a wheel with four (or eight, for a greater cost) 'spokes' to act as the flexures holding the strain gauges. As the thrust is produced, the shaft (running through the centre of radially aligned spokes) causes the spokes to deflect. This deformation is then translated to measure the displacement. This concept has a good shear resistance due to the applied torque, however it may be unsuitable for large stress levels and deflections as they could cause large membrane stresses, and resist true flexure bending [Molland and Turnock, 1990].

- Linear variable differential transformer

There are several electrical components which can be used to measure thrust, a linear variable differential transformer (LVDT) measures linear displacement of a shaft. It consists of one primary and two secondary coils symmetrically placed around a central magnetic core fitted on a shaft. By determining the difference between the induced voltages on the secondary coil, the output gives the position of the central core, and hence the linear shaft displacement [Dunn and Mueller, 1990]. Although this system is very accurate, the sensors are extremely expensive; a quote for a small unit was in excess of £2500.

- Optical techniques

There are a number of leading sensor companies starting to develop optical sensors; these are extremely accurate and require no physical contact between the shaft and the sensor resulting in zero frictional losses and a maintenance-free system. An example of these instruments is the TT-Sense produced by VAF instruments; LEDs on the shaft and high accuracy optical sensors around it are used to detect small axial displacements (radial displacements can also be measured for monitoring torque). The power to run the LEDs is transmitted onto the shaft via induction, allowing contact-less transmission [VAF Instruments, 2012]. This type of thrust measurement has been mainly developed for full size ships; smaller scale systems use very similar principles.

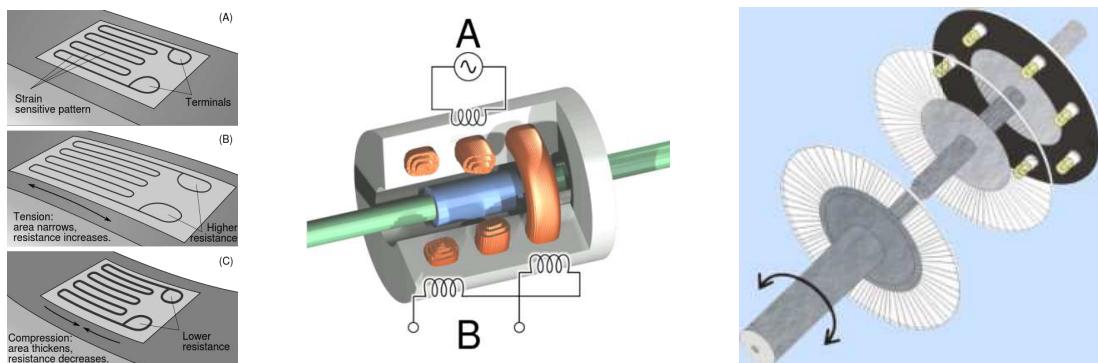


Figure 1.5.2: Diagrammatic representations of the 3 thrust measurement techniques: strain gauges (left), LVDT (centre), optical devices (right) [images from public domain].

Previous GDPs have used a thrust bridge and a load cell mounted on the hull. In 2005, the thrust dynamometer consisted of four flexures and was designed for a thrust loading of 6 N. As the system was mounted on the shaft line, slip rings were needed to transfer the signals and power; this caused significant friction and due to its large diameter (10 mm) limited the overall height of the rig [Bowen et al., 2005]. The system was refurbished by a GDP in 2012; the thrust bridge was tested but found that the weight of the slip rings and dynamometer had been carried by the flexures, rather than the support bearings, resulting in an unpredictable cyclic loading while the shaft rotated. As a result, the dynamometer was bypassed and a Honeywell load block was added to measure the movement of the shaft, by monitoring the deflection of the thrust bearing. The results showed that this was unreliable, mostly because the physical interface to the load cell was unsatisfactory. The load cell itself had been bench tested earlier in the project to verify its accuracy [Collison et al., 2012].

Torque Measurements

[CROSSLEY]

There are various methods of measuring the torque from a rotating shaft. The previous model [Bowen et al., 2005] used strain gauges mounted on flexures which gave a strain reading proportional to the

change in voltage caused by the deformation due to bending. The construction of a suitable device for this method was outlined in a paper by Molland and Turnock [1990]. The strain gauges were powered using sliprings which also take the output voltage off the shaft to be processed through amplification and analogue to digital conversion.

A similar approach using capacitors instead of strain gauges was described by [Falkner, 1996] although this was only at a prototype level. However, the design was shown to be simple and relatively straightforward to manufacture. The prototype achieved good signal responses to small changes in torque. This concept has not been environmentally tested and Falkner [1996] stated that changes in humidity may affect the capacitors, therefore making it unsuitable for the model.

Another option available on the market is to use an optical transducer, which works by having two discs with segmented gratings which are attached to the shaft in such a way that opaque sections on one disc partially obscure the clear sections on the other [Sensor Technology, 2012]). When light passes through the discs, the intensity is picked up by photovoltaic detectors so that when a torque is applied, the intensity of light changes and the electrical output will be different, hence giving an accurate torque measurement. The main advantage of this system is that it is non-contact and so does not need complex electronics on the shaft and the components will not affect the torque measurement in any way. This method has been shown to be accurate in a paper by [Dunn and Mueller, 1990] at a small scale for a model propeller shaft. However, it is unsuitable for this project due to the high costs of off-the-shelf units.

Overall, the only feasible option for the measurement of torque is to improve upon the design of the current torque dynamometer, as this is a proven method that is relatively easy to manufacture and is inexpensive. There are number of improvements outlined in last year's report[Collison et al., 2012]that will be taken into consideration.

1.5.3 Developing an Autonomous Model

[ANDERLINI]

An autonomous model is a vessel that is capable of performing its required tasks (e.g. manoeuvring trials, moving to desired location etc.), which have been previously been built in its software, without any direct human control. Its main advantage over a radio-controlled model is that it provides high repeatability, by removing the errors associated with constant human action, over the course of many tests.

The most common examples of autonomous vessels are AUVs (Autonomous Underwater Vehicle), which have become the platforms for some of the most complex robotic applications. One of these vehicles (Delphin2) has been produced at the University of Southampton [Philips et al., 2010]. Concerning surface vessels only, some autonomous models have been developed to acquire water data in lakes [Hitz, 2010] or to enter robotics competitions [Pickem et al., 2012]. However, what is particularly interesting from a naval architecture point of view is the model of the tanker Esso Osaka built jointly by researchers from the Technical University of Lisbon and NTNU [Moreira and Guedes Soares, 2011]. This ASV (Autonomous Surface Vessel) was designed purely to estimate its manoeuvring capabilities, thanks to built-in GPS [Moreira and Guedes Soares, 2011], and to aid the research in marine control systems [Moreira et al., 2007]. Nevertheless, it is thought that the inclusion of a self-propulsion rig to measure thrust and torque [ITTC, 2002b] would greatly enhance the measuring possibilities of such a model.

As aforementioned, such an ASV would require a robust and simple software architecture, which would allow the integration of multiple systems (e.g. sensors, computers, actuators) and their communication. On the Esso Osaka, a program written in Matlab/Simulink offers a relatively intuitive user-interface from which the various sensors and actuators are called. A more interesting solution is presented by Robot Operating System or ROS.org [2013], which is a collection of libraries specifically designed to develop

the most complicated robotic applications and enable the sharing of programs among users. Such a system would therefore represent a more robust platform (it has been used successfully on numerous robotic applications, including by Philips et al. [2010] and Pickem et al. [2012]), with a greater amount of information and help available in the initial stages of the project available on-line and at the University of Southampton itself.

1.5.4 Testing

[ANDERLINI]

Self-propulsion Tests

The aim of self-propulsion tests is to check whether the propeller is able to absorb the delivered power and whether the ship speed requirements are met [Lewis, 1989]. Additionally, from a self-propulsion test, it is possible to obtain the wake fraction, the thrust deduction factor, the relative rotative efficiency and the model and ship self-propulsion points [Lewis, 1989]. These points are represented by the propeller rotational speed that results in the thrust being equal to the vessel resistance, for a given speed, at model- and full-scale [Lewis, 1989] (the latter point is obtained by considering the difference in total resistance coefficient between the model and the full-scale ships). Therefore, it would be particularly interesting to estimate the model self-propulsion point in the initial stages of the project, so that the corresponding value for the propeller speed could be specified in the controls of the ASV.

From the full-scale ship powering data, the model self-propulsion point can be estimated empirically by applying the inverse process required to predict the ship self-propulsion point from model tests as described in Molland et al. [2011]. Additionally, recent developments in numerical methods mean that CFD software may be used to obtain the model wake fraction and thrust deduction factor quite accurately [Molland et al., 2011]. However, the most reliable results would be produced by experimental model self-propulsion tests, whose procedure is described in ITTC [2002b].

Manoeuvring Trials

According to Bishop and Parkinson [1976], there are two main types of manoeuvring trials:

- stability trials: ‘pull out’ and ‘spiral’ manoeuvres;
- control trials: ‘circle’ and ‘zig-zag’ manoeuvres;

These tests can be performed both with free-running models and full-scale ships [Bishop and Parkinson, 1976]. As can be deduced from their names, the first type of trial is to be performed to check the stability of the vessel (its ability to return to its original course after disturbances have died out), while the second type is studied to assess the manoeuvring capabilities of the ship. An accurate description of these procedures can be found in Lewis [1989]. Additionally, they have been standardized in ITTC [2002a].

Although it has been possible to fit mathematical models to the manoeuvres (e.g. Nomoto equation for the zigzag manoeuvre [Journée and Pinkster, 2002]), these models are highly approximated and the tests are mostly designed to enable the comparison of the current design with successful designs [Bishop and Parkinson, 1976].

Examples of manoeuvring trials performed with a free-running tanker model are given by Moreira et al. [2007] and Moreira and Guedes Soares [2011]. In addition, Steenson et al. [2011] provides a good instance of circle manoeuvres performed by an AUV on the surface.

1.5.5 Wave Measurements

[THORNTON]

There are a number of methods currently in use around the world to measure wave characteristics. This ability is important and plays a key role in many applications such as the design of ships and offshore structures and wave forecasting, providing the capability of describing sea states using a directional wave spectrum [Wiegel, 1982].

According to Stewart [2009] the more commonly used techniques include:

- Estimates from observations at sea – The most common technique used in early tabulations of wave heights, before the age of satellites.
- Synthetic Aperture Radars (SAR) on Satellites – This technique maps the radar reflectivity of the sea surface, indicating wave-like features relating to real waves. It maps some waves clearer than others however, with no exact relationship between wave height and density. Hence, this technique is better applied for determining additional information such as the spatial distribution of wave directions in shallow water.
- Satellite Altimeters – A radio pulse is emitted from the satellite altimeter, which reflects first from the wave crests, then the troughs. The reflected altimeter pulse is stretched in time, which is detected by the satellite and measured to calculate wave height. Accuracy of $\pm 10\%$ can be achieved.
- Wave gauges – Various gauges can be mounted either on platforms above or under the sea surface to measure the height of the wave or the subsurface pressure, which is related to wave height, providing they do not interfere with the wave. Pressure gauges must be located within $\frac{1}{4}$ wavelength of the surface for accurate readings as wave-induced pressure fluctuations decrease exponentially with depth. Other types of gauges include sound, infrared beams and radio waves. Arrays of gauges are widely used to determine the direction of waves, as well as height. Accuracy of $\pm 10\%$ can be achieved. This method may be more feasible for tank environments, proving difficult to implement in the open water.
- Accelerometers mounted on buoys – This method is often used for measuring waves during short experiments at sea. The most accurate measurements are made using accelerometers coupled with gyroscopes for stability. Displacement measurements are achieved by double integration of the vertical accelerations. Drawbacks to this method, however, are that the double integration amplifies low-frequency noise and the buoys' sensitivity to heave is limited to wavelengths greater than the buoys' diameter. Accuracy of $\pm 10\%$ can be achieved with careful measurements.

In Terray et al. [1999] the use of a conventional bottom-mounted, upward-looking Acoustic Doppler Current Profiler (ADCP) provides measurements for both wave height and direction in shallow water (8-11 m). Both height and directional spectra compared well with a co-located array of pressure gauges. Wave estimation techniques were based on the implication of a single bottom-mounted ADCP with the "Janus" beam geometry, as this is what most current ADCPs utilize. This method employs acoustic beams that are inclined at an angle typically 20-30 degrees with respect to the vertical axis, measuring and averaging the instantaneous velocity component using sonar. Though it is not possible to separate the horizontal and vertical wave velocity components, the measured wave field is statistically stationary in time; hence the cross-spectra between measured velocities contain information about wave direction. This was proven to be a successful method for measuring wave parameters such as significant height, peak period and peak direction, however may prove difficult to implement during open water model tests.

The adequacy of the transfer function used between subsurface pressure and surface wave height in the linear wave theory is investigated in Bishop and M.A.Donelan [1987] through the use of pressure transducers and capacitance wave probes in a wind-wave flume. Extensive information was attained confirming that linear wave theory, through the use of pressure transducers, is adequate in obtaining reliable surface wave height estimations from pressure data. It is noted that a well-designed pressure transducer system with proper analysis techniques should give accurate wave heights to $\pm 5\%$. However, this method requires a closely controlled and monitored environment in which many possible sources of error may become apparent, particularly when currents and wave non-linearities are present.

Various radar techniques have also been implemented over time to measure wave height and direction. One such example is the measurement of ocean wave directional spectra using scanning-beam microwave radars on aircraft and satellites [P.L.Baker, 1985]. Another commercial example is the RS Aqua WaveRadar Rex [RS AQUA Ltd] which implements a downward looking non-contact sensor using a microwave radar technique to measure distance to the sea surface, providing a surface profile. Other radar techniques are available; however this is clearly not a viable option for this project due to cost, logistics and complexity.

According to Berteaux [1976] buoy systems can be summarized into 2 main areas; moored and free drifting systems, both of which can be further classified into surface and subsurface buoy systems. Moored systems are used for long-term measurements using instrumented buoys capable of monitoring the spatial and temporal distribution of environmental variables. Free drifting systems are more often used for measurements of surface or deep water currents, with simultaneous measurements of oceanic parameters and position providing a record of time and space variation of the recorded measurements. It is necessary to implement a surface buoy system when monitoring parameters at the air-sea interface, such as wave measurements, and where radio telemetry is required. Surface buoys can be either surface following, i.e. discus or boat hull shape; or surface decoupled, i.e. spars. Surface following buoys have a tendency to closely follow the waves, hence are the popular choice in instrumented wave buoy designs.

Moored floating data buoys are a commonly used ocean observing system, with various programs in place such as the National Oceanic and Atmospheric Administration's National Data Buoy Center (NOAA NDBC) [National Data Buoy Center, 2012]. There are currently 110 moored buoy systems globally. There are 6 types of moored buoys in the NDBC's fleet, ranging from 12 m discus hulls to 6 m boat-shaped hulls and smaller 1.5-2.5 m coastal buoys. Capabilities range from the measurement of wave energy spectra where significant wave heights and periods are obtained, to direction of wave propagation. Various other measurements are also taken, such as barometric pressure; air and sea temperature; and wind direction and speed.

There are commercially available wave buoys designed to measure the surface sea state, with ease of deployment and retrieval in mind. One such range of buoy designs is the Datawell Waverider series [Datawell, 2012]. This range of surface contouring buoys includes aluminum spherical buoys of diameters 0.9, 0.7 and 0.4 m, measuring waves using a single GPS receiver and transmitting directional and spectral wave data via high frequency (HF) link to an onshore receiver. Wave resolutions of 1cm can be achieved, with a directional resolution of 1.5 degrees and spectral data frequency range of 0.025 Hz – 0.60 Hz.

The Triaxys Directional Wave Buoy [Triaxys Technologies Inc.] encompasses a similar design to the Waverider, but uses a sensor unit comprising of 3 accelerometers, 3 rate gyros and a Fluxgate compass to measure the buoys' heave, pitch and roll responses. Data can be transmitted from the buoy via options such as VHF or GPRS, and includes information such as wave statistics; heave, North and East displacements (HNE); wave direction and energy as a function of frequency (MeanDir); and directional and non-directional wave spectra. It can measure heave of $+/-20$ m with a resolution of 1 cm and accuracy

of better than 2%, period of 1.5-33 seconds with a resolution of 0.1 second and accuracy of better than 2%, and wave direction over 360 degrees with a resolution of 1 degree and accuracy of 3 degrees.

Other similar wave buoys are available on the commercial market, such as the SEAWATCH Wavescan Buoy [Fugro OCEANOR, 2010] and the Wavetector 2 [Independent Maritime Assessment Associates, 2012], which upon contact with the company via email was confirmed to cost around £4500 for the basic tethered buoy with no directional capability, and around double that for the telemetry version (not currently offered commercially).

Simpler, more economical, options for wave buoys have been explored, such as in Wellicome et al. [1999] where a wave buoy was designed based on a circular lifebelt for ship model tests in open water. It is a free-drifting system implementing a drogue positioned upwind of the buoy to slow down the buoy drift to leeward, and an umbilical to supply power to the sensors and transmit data to a support boat downwind. Sensory equipment included 3 accelerometers positioned at 120 degree intervals around the lifebelt, and a Fluxgate compass suspended in the center of the ring. Some satisfactory results were obtained for the dimensional frequency spectra, however it was noted that further work was needed with regards to obtaining the mean wave direction and directional spread.

There was also a study conducted in Sundar et al. [2008] where the performance of a 2.2 meter diameter discus-shaped buoy was investigated in long and short wavelength directional seas using a 1:8 model scale buoy. 4 accelerometers were mounted on the buoy to measure pitch and roll, the results of which were validated using a polygonal array of 5 wave gauges. The performance of the discus buoy was found to be good for directional wave measurements with significant wave heights as low as 8 cm and period as low as 1.5 seconds, agreeing closely with results obtained from the array of wave gauges.

PRELIMINARY

This chapter covers preliminary work that was required, prior to beginning work on the project. It includes a considerable amount of research into the definition of ‘efficiency’ and the types of efficiency improvement that could be tested in the future. Powering performance, when attempting to design efficiency improvements, is the most critical characteristic to measure - and these sections set a benchmark for the design of sensors and signal processing systems that must be designed to measure small changes in powering.

Also, a number of calculations were required regarding powering and manoeuvring to ensure a successful design. For example, power estimates were required in order to choose a suitable motor for the vessel.

2.1 Evaluating Efficiency

[MOZDEN]

The propulsive efficiency of a ship can be judged by its power requirements and reducing the requirements leads to a higher efficiency. The overall efficiency can always be improved, this is led by economic and environmental factors as the ship owner endeavours to make a better return on their investment, whilst minimising pollution to the marine environment as per shipping regulations.

There are two key ways to reduce the power requirements for a ship at the design stage:

- Increase propulsive efficiency
- Reduce ship resistance.

The propulsive efficiency can be increased by improvements to the propeller for optimal flow and better propeller-hull interactions. The hull resistance reductions can be achieved by optimising the principle shape (either the main form i.e. U- or V-shape sections or locally, for example bulbous bows) or by the addition of appendages to improve the flow around the hull. Once a vessel has been built it can be difficult to improve the efficiency if it does not meet its initial design criteria; Efficiency Improvement (EI) devices can be retro-fitted to the ship, these tend to focus on reducing losses due to wave generation, hull friction and propeller losses and can produce improvements by up to 10% in some instances, and will be discussed more thoroughly in the next section.

The propulsive efficiency can be determined by investigating the total propulsive efficiency (η_T) of the craft (Equation 2.1.1), which is a function of the effective power over the delivered power. To analyse this experimentally, the effective thrust (T_E), torque (Q), advance velocity (u) and propeller angular velocity (n) need to be recorded [Muntean, 2008]. By comparing different conditions, the comparative performance of the different EI devices can be assessed.

$$\eta_T = \frac{T_E u}{Q n} \quad (2.1.1)$$

When a system accelerates, the force applied to induce this motion is thrust. In a ship, the thrust is generated by the ships propeller(s) accelerating the water, resulting in a thrust force propelling the vessel in the opposite direction, equal to the change in momentum of the water flow. The thrust of the propeller (T) needs to be greater than the resistance (R) of the hull without the propeller in order to create forward

motion, the relationship between propeller thrust and this difference is the thrust deduction factor (t) (Equation 2.1.2) [Carlton, 2012].

$$t = \frac{(T - R)}{R} \quad (2.1.2)$$

Torque is the tendency of a force to rotate an object about an axis; it acts to twist an object, as opposed to thrust which is a pushing force. The power to produce the thrust must also be used to rotate the shaft connecting the engine to the propeller. It is a function of the applied force (F) and the radius (r) which is it acting (Equation 2.1.3).

$$Q = F.r \quad (2.1.3)$$

The performance of a propeller is determined by its open water behaviour; when it is fitted to a ship it needs to be positioned in close proximity to hull in order to be attached to the shaft. The flow into the propeller is influenced by the hull as it is disturbed so no longer uniform. This results in the average flow speed into the propeller (V_a) differing from the advance speed of the hull (V_s); the difference in speed is the wake fraction (w_T) (Equation 2.1.4) [Molland et al., 2011].

$$w_T = \frac{(V_s - V_a)}{V_s} \quad (2.1.4)$$

Using Equations 2.1.2 and 2.1.4 the thrust power (P_T) developed by the propeller is a product of the propeller thrust and flow into the propeller (Equation 2.1.5); the effective power (P_E) is the power needed to move at the operating speed (Equation 2.1.6).

$$P_T = T.V_a \quad (2.1.5)$$

$$P_E = R.V_s \quad (2.1.6)$$

The hull efficiency is the relationship between P_E and P_T (Equation 2.1.7); this should generally equal unity as it indicates that the all usable power from the engine which has been delivered to the propeller is to be used to overcome hull resistance, propelling the ship forward [Rawson and Tupper, 2001]

$$\eta_H = \frac{P_E}{P_T} = \frac{(1 - t)}{1 - w_T} \approx 1 \quad (2.1.7)$$

Thrust can be related to the propeller properties using the thrust coefficient (Equation 2.1.8) and torque using the torque coefficient (Equation 2.1.9); K_T is the thrust coefficient, K_Q is the torque coefficient, ρ is the fluid density, n and D are propeller properties rotational velocity and diameter respectively. These relationships allow scaling to be used for predicting thrust and torque behaviour for full-scale from geometrically similar models as K_T and K_Q should be the same for both.

$$T = K_T \cdot \rho \cdot n^2 \cdot D^4 \quad (2.1.8)$$

$$Q = K_Q \cdot \rho \cdot n^2 \cdot D^5 \quad (2.1.9)$$

2.2 Efficiency Improvements

[NEALE]

As discussed previously, pressure on the shipping industry to reduce its carbon emissions has culminated most recently with the implementation of the EEDI. This regulation requires that all new build ships conform to a certain level of energy efficiency. Furthermore, the high cost of fuel, which affects both new builds and existing vessels alike, has instigated additional interest in reducing fuel consumption through energy efficiency improvements.

The variety of methods available with the potential to reduce emissions and/or fuel consumption is vast, with efficiency improvements making up just one area. The IMO breaks emission reducing methods into four components [Buhaug et al., 2009]. These are energy efficiency improvements, renewable energy resources, fuels with lower emissions per unit of work (e.g. natural gas and bio-fuels) and emission reduction technologies.

Focussing on efficiency improvements (EI's), two areas may be considered: design improvements and operational improvements. A full breakdown of the different components of these areas is presented in Figure 2.2.1.

This project was limited to exploring efficiency improvements through the use of model testing. The hull form of the model was fixed with the exception of the bow which was removable. As the bow design was being investigated in a parallel project, two topics were chosen as being most suitable for examination: retro-fit devices and trim optimisation.

Less intrusive than many other energy saving methods, retro-fit efficiency improvement devices have attracted growing attention over the last few decades. Applicable to both new builds and existing vessels, they may be used to rectify or avert design imperfections. A wide array of devices have been designed, generally focussed on reducing losses in three areas: wave-making along the hull, friction of the hull and propeller, and propulsive losses [de Jong, n.d.].

Losses caused by wave-making can be decreased by reducing the peaks and troughs that occur in the pressure distribution around the hull. One common device used to reduce the pressure peak at the bow is the bulbous bow. These are regularly included in the original design of the hull. Less common, however, are devices used to reduce the pressure trough at the stern. One example of these is a trim wedge. Originally used on small planing vessels to adjust the trim, they are increasingly being used on larger ships such as destroyers. However, instead of adjusting the trim, wedges on larger ships modify the flow around the stern, creating an increase in pressure and directing flow downwards which decreases the size of the stern wave. Studies have found a power reduction of 6% for destroyers with a trim wedge [Karafiath and Fisher, 1987]. A similar device is an interceptor, which consists of a vertical plate attached to and protruding below the transom. This has been found to reduce the power requirements of a cruise vessel by 3% [Allema, 2005]. An alternative device which may be fitted to the transom of a vessel is a ducktail. A ducktail consists of a 3-6 meter long horizontal plate which increases the length of the effective waterline and reduces the wetted transom area. These effects also lead to a reduction in wave-making drag and have been reported to reduce fuel consumption by around 4 % [Pleiss, 2011]. Little is known about the application of these devices to larger vessels such as tankers, however it has been shown that these devices have a greater effect at higher speeds. The wave-making resistance is a relatively small component of the overall resistance for a tanker and therefore they may not be an economically viable solution.

Losses due to friction from the hull and propeller can be reduced through the use of surface treatments and regular cleaning. Nevertheless, one retro-fit solution currently being developed is air bubble lubrication. Air-bubble lubrication involves the injection of air bubbles along the hull. It is not known

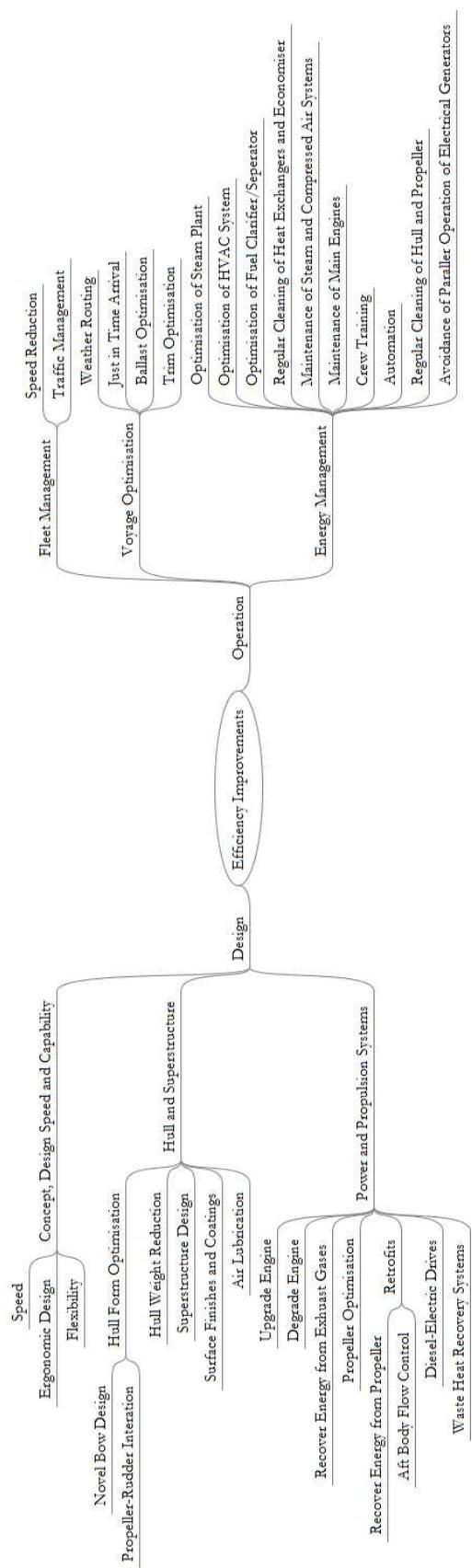


Figure 2.2.1: Breakdown of available efficiency improvements.

exactly how air bubbles reduce the frictional drag, although one theory suggests that their deformation causes the flow to become more isotropic, thereby reducing turbulent stresses. Recent studies have so far failed to produce significant resistance reductions; a full scale investigation conducted by the Maritime Research Institute (MARIN) produced a maximum propulsive power reduction of 2% [Foeth et al., 2010]. After taking into account the power required to produce the bubbles a net increase of 0.6% was obtained. However this study was conducted on a 109.8 m inland shipping vessel; as the reduction of friction due to air bubbles has been found to continue to the aft end of the vessel, it has been suggested that the system would be more efficient on longer ships [National Maritime Research Institute, Japan, n.d.]. Manoeuvring and sea-keeping trials were conducted, with little effect observed in manoeuvring and no effect observed in sea-keeping. It was concluded that no extra training would be required to operate a ship with air bubble lubrication [Foeth et al., 2010]. This device could potentially be well-suited to a tanker hull, as the bottom of the hull tends to be wide and flat. Additionally 60-65 % of the total resistance of a tanker is caused by friction, resulting in a high potential for drag reduction [Turnock, 2012]. Unfortunately the prospect of air bubble lubrication reaching a state of commercial viability is currently very poor.

The majority of retrofit devices are designed to improve the energy efficiency associated with the propeller. They generally operate by recovering the rotational energy of the flow or by modifying the inflow to the propeller.

Swirl devices aim to recover these losses by inducing a rotational flow in the opposite direction. Pre-swirl stators are stationary blades or fins fixed in front of the propeller. They trigger a rotation of the flow in the opposite direction to the propeller which creates an additional blade loading and increased thrust [Zondervan et al., 2011]. Successful full scale trials have been conducted which indicate a propulsive power reduction of 4.5% [SSPA, 2011] for a VLCC. They are relatively less expensive than other swirl devices such as contra-rotating propellers and can be installed without great difficulty. However, as they also generate a certain amount of drag are only suitable for low-speed, full-hulled vessels such as tankers.

Post-swirl stators are fins attached to the rudder which convert the rotational energy of the flow into thrust. Consequently, in contrast to pre-swirl devices they are better suited to faster, slender-hulled vessels. Full scale trials conducted by Mitsubishi Heavy Industries on a 167.2 m car carrier resulted in a power reduction of around 4%, with no adverse effects on strength or vibration observed [Hoshino et al., 2004].

A better known post-swirl device is the Grim vane wheel, seen in Figure 2.2.2. It comprises of a freely rotating wheel situated behind the propeller. The inner section of the vane acts as a turbine, whereas the outer section acts as a propeller. As the diameter is larger than the main propeller it is able to convert energy from the flow into additional thrust. Studies have shown reductions in propulsive power of up to 10% [Buhaug et al., 2009]. The confidence in vane wheels was damaged after their installation on the Queen Elizabeth II, which failed due to strength and vibration issues. However research and redesign since have made the concept more feasible, although the vanes may be damaged in heavy seas.

An alternative method to recover rotational losses is through a co-axial contra-rotating propeller. A second propeller is situated directly behind the main propeller and rotates in the opposite direction, recovering rotational energy from the main propeller's slip stream. Although, as mentioned, this device is costly in comparison, it also has the potential of offering the best returns. Successful full scale trials have resulted in improvements in power consumption of around 15%, with the best results obtained for heavily loaded propellers used on cargo ships and ro-ro ferries [Buhaug et al., 2009].

Energy may also be lost through vortex shedding at the propeller hub. Propeller boss cap fins are able to recover some of these rotational losses, thereby reducing power consumption by 5%. Easy to install and currently fitted to thousands of existing ships, including tankers, this is a well established method of



Figure 2.2.2: Example of a Grim vane wheel [image from the public domain].

improving energy efficiency [Hansen et al., 2011]. Model tests have been conducted in order to investigate the effects of this device in waves, with very little difference observed from the calm water state.

So far devices which recover rotational losses have been discussed. However devices also exist, such as nozzles and ducts, which seek to modify and accelerate the inflow to the propeller in order to produce a better operating environment

Wake equalising ducts such as the Schneekluth WED comprise of a hydrofoil section which is bent to form two half-ring ducts, fixed either side of the hull in front of the propeller (see Figure 2.2.3). The flow through the duct is accelerated whereas the flow beneath the duct is slowed down. Although the average wake is not greatly affected, the wake field forwards of the propeller becomes more homogeneous. Additionally a more axial flow is created improving the efficiency of the propeller. Full scale trials on a tanker have resulted in power savings of 3-8%. This device has been installed on over a thousand vessels and is therefore well established. Further benefits include a reduction of flow separation, reduction of propeller induced vibrations and improved steering due to the straightened flow to the rudder [Schneekluth Hydrodynamik].



Figure 2.2.3: Example of a Scheekluth duct [Schneekluth Hydrodynamik].

Ducts are often used in combination with other devices such as pre-swirl fins. An example of this is the Mewis duct, a fairly recent invention which has been installed on ships ranging from small bulk carriers to large tankers. This device equalises the wake in addition to producing a pre-swirl effect. It

has also been found to improve cavitation behaviour and reduce hub vortex losses. An average power saving of around 6% is reported, although this value is very dependant on the propeller thrust loading. With regards to manoeuvring and sea-keeping behaviour, crew reports have conveyed an improvement in course keeping and a more stable propeller rotation in a seaway [Svardal and Mewis, 2011]. Another device often combined with ducts are Grothues spoilers. Best suited to tankers and full hulled ships, they consist of a set of curved vanes attached in front of the propeller. They aim to prevent cross-flow occurring in front of the propeller, leading to reported efficiency increases of between 3 and 9% [Renilson Marine Consulting Pty Ltd, 2009].

Additional efficiency improvement devices focus on the rudder. Asymmetrical or 'twisted' Rudders have been found to improve cavitation behaviour and convert rotational energy in the fluid into thrust [Turnock and Molland, 2010]. Often they are designed as part of an integrated unit with the propeller, with a bulb connecting the propeller and the rudder. This allows the propeller and the rudder to be designed together to optimise performance. One example is the Promas system (see Figure 2.2.4) designed by Rolls-Royce who have reported improvements in efficiency of 2 to 9%, depending on the fullness of the hull. They also claim that the system is able to improve manoeuvrability at low speeds.

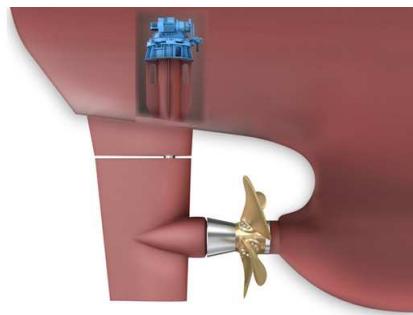


Figure 2.2.4: The Promas system [Rolls Royce].

It was previously stated that trim optimisation could be an alternative method of improving a vessels efficiency. The optimum trim for a vessel is dependant on many factors including the hull shape, speed, displacement and water depth. Usually a ship will be optimised for performance at its design displacement and speed, however may often operate outside of these conditions [DNV]. To remedy this vessels may be fitted with technology enabling the correct trim for the particular operating condition to be identified. This technology may be based on pre-programmed data obtained from CFD analysis, or may be based on direct analysis from on-board monitoring. One system produced by Eniram is claimed to have potential to reduce fuel consumption by 2.6% [ENIRAM, 2012].

A number of energy saving devices and methods have been considered, although the range is by no means fully comprehensive. Although promising results have been achieved careful consideration is required before installing such devices. Performance can be dependant on many factors including the hull form and different operating conditions. Installation of an incompatible device could even impair efficiency; for example, a pre-swirl device fitted to a fast cargo ship may generate more drag than the extra thrust gained from the propeller.

It is also important to consider the effect these devices may have on the operation of the ship, including manoeuvring, sea-keeping and vibration characteristics. Often these are considered as an after-thought, with crew reports relied on as a means to assess the performance. Port access may also be an issue for any installations which extend beneath or behind the hull.

An additional problem is confidence in the reported power savings of such devices. Accurate and

transparent full scale results are notoriously difficult to obtain. Long-term studies may be inconclusive due to vessels using the devices to travel faster rather than to save energy. Furthermore trials are often conducted by the developers of the system rather than an independent source.

Because of these issues, the devices can not be produced 'off the shelf.' For each application the device must be optimised using a combination of CFD, model tests and full scale trials. This may be a laborious process; unfortunately due the complex nature of the shipping industry it can often be unclear as to whose responsibility this process is [de Jong, n.d.].

In conclusion, the devices investigated were generally reported to improve efficiency by values ranging from 2 to 10%. Any platform designed to test these devices would therefore be required to have the capability to detect changes in efficiency of this magnitude. With regards testing a model-scale tanker, the efficiency improvement devices with the most promise include pre-swirl stators, propeller boss cap fins, ducts, spoilers and integrated rudder units. Furthermore trim optimisation may produce satisfactory results.

2.3 Preliminary Calculations

[LE]

The purpose of this section is to introduce important calculations regarding resistance, powering and handling. These will be used for the design and testing of the model.

For reference, the principal particulars are presented in Table 2.3.1.

Table 2.3.1: Principal particulars of the full-scale and model-scale BP Virtue Ice Class vessel.

	Unit	Ship	Model
L_{OA}	m	183.88	3.06
L_{PP}	m	174	2.9
B	m	32.2	0.54
Depth	m	18.8	0.31
Draft Mean (Full Load)	m	11.02	0.1837
Draft Mean (Ballast)	m	6.91	0.1152
Displacement (Full Load)	tonnes	49969	0.2257
Displacement (Ballast)	tonnes	29773	0.1345
Service speed	knots	15.5	2.000
Service speed	m/s	7.974	1.029
F_n		0.188	0.188
WSA (FL)	m^2	7960.6	2.211
WSA (BL)	m^2	6231.6	1.731
C_B		0.7994	0.7994

2.3.1 Resistance and Scaling

[LE]

The aim of this section is to provide a rough estimation of the resistance of the model, which was used for the motor selection and propeller calculations.

In a normal design process, a model is tested to measure its resistances at different speeds. The data would then be scaled up to estimate the resistance of the full-scale ship. However, in this project, the power requirements of the full-scale ship at different speeds were known, whilst the power requirement for the model were unknown. Therefore, an inverse process must be applied. This meant the power requirements of the full-scale ship at different speeds were be read from Figure 2.3.1 and then scaled down to estimate power requirements of the model.

It should be noted that the full-scale ship's power requirement in Figure 2.3.1 includes roughness allowance, added air resistance, appendage drag, etc. However, there was no information about the method that the shipyard used to calculate the ship's resistance and how the resistance curve was created. Thus, it was difficult to quantify the added resistance caused by the aforementioned factors. The purpose of this section is to provide a first estimation of the maximum power required by the model. Therefore, at this early stage, the contribution of air resistance, appendage drag and roughness allowance could be assumed negligible.

Estimating Form Factor

[LE]

The first stage was to estimate the form factor, and two methods were used: Hughes' method and Watanabe's method.

- Hughes' Method

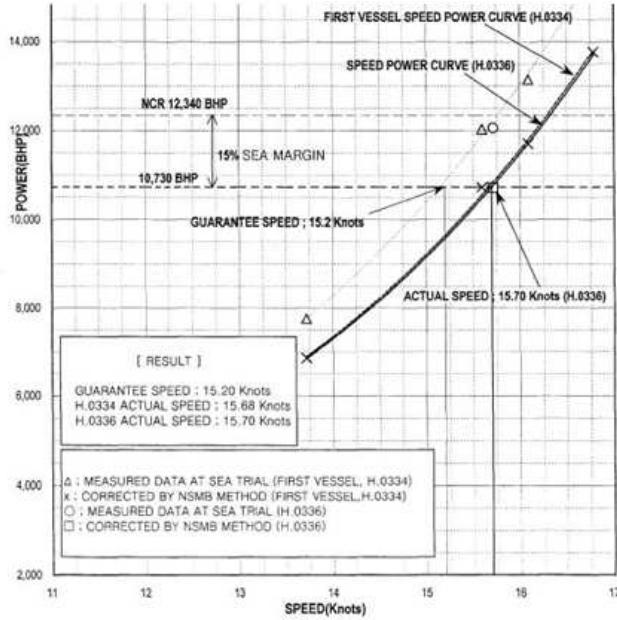


Figure 2.3.1: Full scale sea-trial results.

Two speeds were chosen from the sea-trial results so that the Froude numbers were between 0.1-0.2 as shown in Table 2.3.2. Then the form factor could be found by determining the coefficient of skin frictional drag from the ITTC correlation line:

$$C_F = \frac{0.075}{(\log R_n - 2)^2} \quad (2.3.1)$$

in which the Reynold's number R_n is given by

$$R_n = \frac{\nu L_{PP}}{\mu} \quad (2.3.2)$$

and μ is the dynamic viscosity of salt water (1.19×10^{-6} Ns/m²). This coefficient can then be used with the Hughes' breakdown of total resistance:

$$\frac{C_T}{C_F} = (1 + k) + \frac{AF_n^4}{C_F} \quad (2.3.3)$$

which can be rearranged to find the form factor, $1 + k = 1.425$.

Table 2.3.2: Full-scale power requirements and non-dimensional resistance coefficients for different speeds. Assumption: QPC (Quasi Propulsive Coefficient) is 0.68.

Fn	Velocity		Delivered Power	PE	Resistance	C_{Ts}	R_n	C_F
	knots	m/s	BHP	kW	W	N		
0.1696	14	7.202	7250	5437.5	3697500	513428	2.43E-03	1.05E+09
0.1902	15.7	8.076	10730	8047.5	5472300	677594	2.55E-03	1.18E+09

- Watanabe's empirical formula

An empirical formula for form factor was used to compare with the calculation above [Molland et al., 2011].

$$k = -0.095 + \frac{25.6C_B}{(\frac{L}{B})^2 \sqrt{\frac{B}{T}}} = 0.347 \quad (2.3.4)$$

hence, the form factor was calculated to be $1 + k = 1.347$.

According to [Collison et al., 2012], the form factor of the naked hull is 1.388 which is nearly identical to Watanabe's empirical formula.

Propeller Discussion

[ANDERLINI]

On a full-scale ship it is fundamental to match the propeller revolutions, torque and delivered power to the safe operating range of the installed propulsion engine [Molland et al., 2011]. The same problem applies to a model designed for free-running tests, although slightly different constraints apply.

In the current project, the tight economic budget demanded the use of 'off-the-shelf' solutions whenever possible. For this reason, the propeller was taken from the self-propulsion rig produced as part of a previous GDP project [Bowen et al., 2005]. Its selection was based on its number of blades and its diameter, which corresponds exactly to the scaled dimensions (1:60) of the full-scale ship propeller, in the absence of any other information, such as Blade Area Ratio (BAR) due to confidentiality issues. The main characteristics of this propeller can be seen in Table 2.3.3.

Table 2.3.3: Number of blades, diameter, expanded blade area ratio, pitch over diameter ratio and type of the selected model propeller.

No. of Blades	Diameter (mm)	Expanded BAR	Pitch/Diameter	Type
4	100	0.55	0.875	Wageningen B series, FP

As can be seen from Table 2.3.3, the propeller belongs to a very famous propeller series, for which the open water data is readily available in both printed [Van Lammeren et al., 1969] as well as electronic format [Wolfson Unit, 2012]. In this project, the numerical implementation in the propeller design software WinProp of the Wolfson Unit [2012] has been adopted. Additionally, a Boundary-Element Momentum Theory code has been developed at the University of Southampton, whose working mechanism is clearly explained in chapter 15 of Molland et al. [2011]. Since the geometric sectional data was available for the selected propeller [Bowen et al., 2005], it was possible to use the BEMT algorithm to quickly and reliably estimate the propeller self-propulsion point and subsequently validate the result with the experimental open water data.

It should be noted that due to the very short chord (always less than 0.1 m) and the slow model speed, the flow around the propeller was laminar. Hence, the sectional data and in particular the lift and drag coefficients (e.g. derived from xFoil [Drela and Youngren, 2001]) should have been modified in the BEMT program accounting for the laminar flow. However, considering the very high motor power available and the high number of revolutions it could achieve, it has been preferred to dedicate the resources to the development of other sections (e.g. software, electronics, rudder development etc.) during the preliminary stages of the project and derive the actual model self-propulsion point experimentally during towing tank testing.

Figure 2.3.2 shows the open water chart for a Wageningen B series propeller with $P/D = 0.875$ and $\text{BAR} = 0.55$. The two curves are very similar for their central portion and present the same maximum open water efficiency. However, the torque is slightly overestimated by the BEMT code for low advance numbers, where the underlying theory starts to fail. The open water curves plot the thrust and torque

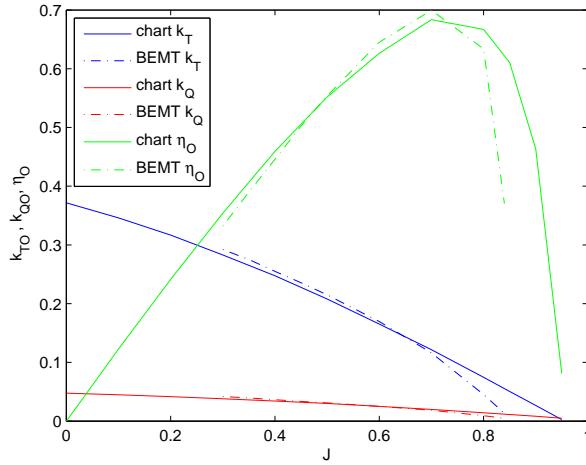


Figure 2.3.2: Open-water chart for a Wageningen B series propeller with BAR=0.55 and $P/D = 0.875$. Both experimental and numerical (BEMT code) data have been presented.

coefficients and the efficiency of the propeller operating in open water against the advance coefficient. These values are obtained as follows [Krueger, 2004] [Krueger, 2008] [Molland et al., 2011]:

$$\text{Advance number : } J = \frac{v_A}{nD} \quad (2.3.5a)$$

$$\text{Thrust coefficient : } k_{TO} = \frac{T_O}{\rho n^2 D^4} \quad (2.3.5b)$$

$$\text{Torque coefficient : } k_{QO} = \frac{Q_O}{\rho n^2 D^5} \quad (2.3.5c)$$

$$\text{Open - water efficiency : } \eta_O = \frac{J k_{TO}}{2\pi k_{QO}} \quad (2.3.5d)$$

In Equations 2.3.5a, 2.3.5b, 2.3.5c and 2.3.5d, T_O is the propeller open-water thrust, Q_O the propeller open-water torque, $\rho = 1000 \text{ kg/m}^3$ the water density, D the propeller diameter, n the number of revolutions (Hz) and $v_A = v_S(1 - w)$ the velocity in the propeller plane (m/s), where v_S is the ship (model) forward speed and w the wake fraction [Krueger, 2004] [Krueger, 2008] [Molland et al., 2011]. The wake fraction has been taken to be $w = 0.4386$ from Collison et al. [2012], where it has been estimated by a numerical simulation in OpenFOAM [OpenCFD Ltd, 2012] of a wind tunnel experiment. Therefore, $v_A = 0.5854 \text{ m/s}$ for the model operational speed. Additionally, it is possible to estimate the model thrust from the predicted model resistance, R_T (taken from Table 2.3.5), as $T = \frac{R_{T_m}}{(1-t)}$ [Krueger, 2004] [Molland et al., 2011], where t is the thrust deduction, which can safely be assumed to be 0.2 for a tanker model [Krueger, 2004].

In order to estimate the model self-propulsion point, it was assumed that the model thrust is in fact equal to the open-water thrust. Hence, it was possible to calculate the open-water thrust from the chart for every advance ratio, which is selected by varying the propeller revolutions only. From Equation 2.3.5b it is then possible to calculate the open-water chart and compare it with the predicted model thrust (hence, the name thrust equality [Molland et al., 2011]). This iterative procedure is continued until convergence occurs. The predicted model self-propulsion points are reported in Table 2.3.4 for all possible wake fractions. The results have been calculated from both the open-water chart data and the BEMT code.

Table 2.3.4: Estimation of the model self-propulsion point for the three predicted form factors.

	Experiment	BEMT	Experiment	BEMT	Experiment	BEMT
1+k	1.425	1.425	1.347	1.347	1.388	1.388
Desired T (N)	8.6125	8.6125	8.3375	8.3375	8.4750	8.4750
J	0.333	0.333	0.338	0.338	0.335	0.335
k_{T_O}	0.2797	0.2808	0.2782	0.2792	0.2791	0.2802
k_{Q_O}	0.0400	0.0402	0.0398	0.0399	0.0399	0.0401
T (N)	8.6450	8.6779	8.3452	8.3751	8.5234	8.5563
%-error in T	0.38%	0.76%	0.09%	0.45%	0.57%	0.96%
Q (Nm)	0.1242	0.1206	0.1193	0.1197	0.1219	0.1225
η_O	37.07%	37.06%	37.63%	37.64%	37.29%	37.29%
N (RPM)	1055	1055	1039	1039	1048	1048

It is recognised that the estimated advance number of the self-propulsion point is very low. Nevertheless, the high power of the chosen motor means that it can be achieved without problems. The only issue associated with such a high propeller speed is the vorticity it creates in the flow impinging on the rudder, which may cause an increase in resistance as compared with the full-scale ship. Also, this may cause problems when scaling EI devices that are aimed at reducing propeller swirl.

The development of a new propeller has not been addressed as part of this project, as it was not considered to be cost-effective in terms of time and effort required.

All of the described results were also computed with a 15% margin allowance for weather conditions, but this data is not included in the report for brevity.

Power Estimation

[LE]

The following formulae were used to obtain the power requirement for the model.

$$C_w = C_{Ts} - C_V = C_{Ts} - (1 + k)C_F = 8.29 \times 10^{-5} \quad (2.3.6)$$

$$R_n = \frac{vL_{model}}{\mu_{freshwater}} = \frac{1.029 \times 3.06}{1.14 \times 10^{-6}} = 2.76 \times 10^6 \quad (2.3.7)$$

$$C_{Fm} = \frac{0.075}{(\log R_n - 2)^2} \quad (2.3.8)$$

$$C_{Tm} = C_{Fm}(1 + k) + C_w \quad (2.3.9)$$

$$R_{Tm} = C_{Tm} \times \frac{1}{2} \rho \times WSA_m \times v_m^2 \quad (2.3.10)$$

$$P_E = R_m \times v_m \quad (2.3.11)$$

According to Collison et al. [2012], the wake fraction of the model was predicted to be 0.4386 and thrust deduction was assumed to be 0.2 as already described. Relative rotative efficiency was assumed to be approximately 1: $\eta_R \sim 1$.

Propeller efficiency η_O can be derived from the average of the results in the Table 2.3.4: $\eta_O = 0.366$. From this, the Quasi-Propulsive Coefficient (QPC), η_D , of the model can be calculated:

$$\eta_H = \frac{1 - t}{1 - w_T} = \frac{1 - 0.2}{1 - 0.4386} = 1.425 \quad (2.3.12)$$

$$\eta_D = \eta_O \eta_H \eta_R = 0.523 \quad (2.3.13)$$

$$P_D = \frac{P_E}{\eta_D}(W) \quad (2.3.14)$$

Table 2.3.5 illustrates the estimated power requirements assuming different form factors.

Table 2.3.5: Estimating resistance using different form factors.

1+k	1.425	1.347	1.388
C_w	4.088E-04	5.266E-04	4.651E-04
R_n	2.62E+06	2.62E+06	2.62E+06
C_{Fm}	3.84E-03	3.84E-03	3.84E-03
C_{Tm}	5.89E-03	5.70E-03	5.80E-03
R_m	6.89	6.67	6.78
P_E	7.08	6.86	6.98
P_D	13.54	13.12	13.34

2.3.2 Rudder Discussion

[LE]

The purpose of this section is to calculate the rudder torque requirement for the model in circle and zigzag manoeuvring tests.

Dimensions of the model's rudder were calculated and shown in Table 2.3.6. It should be noted that the in zig-zag manoeuvring test, the rudder was required to turn 20° to either side only. However, it was envisaged that for the model scale, an angle of 20° may not be large enough to create an adequate response. Therefore, to be safe, the torque requirement were calculated for the extreme case of turning 35° to either side. The result was then used to choose a suitable servo, and model's rudder stock diameter.

In addition, a comparison of lift coefficients generated by a spade rudder and a semi-balanced rudder will be shown to prove that the difference is negligible allowing a spade rudder to be used without significant errors.

Table 2.3.6: Rudder principal dimensions.

	Real ship	Model		
A	40445540	mm ²	0.0112	mm ²
\bar{x}_1	1600	mm	26.67	mm
\bar{x}	1128	mm	18.80	mm
CPs \bar{y}	4689	mm	78.15	mm
CPc	2728	mm	45.47	mm
C_R	6265	mm	0.1044	m
C_T	4750	mm	0.0792	m
Mean chord	4514	mm	0.0752	m
Rudder span	8960	mm	0.1493	m
Taper ratio	0.758			
Stock	35	%aft		

Formulae for Calculating Rudder Torque

The rudder of the full-scale ship is semi-balanced, however, for this project, a spade rudder was used for simplicity. K_T/J^2 was found from the propeller open water chart $\frac{K_T}{J^2} = 2.57$. From that K_T/J^2 , C_L , C_D , CPs and CPc could be interpolated and then using the following formulae, rudder torque and bending moment were estimated. [Molland and Turnock, 2007]

$$\bar{x} = \left(\frac{CPC}{100} \times \bar{c} \right) - x_1 \quad (2.3.15)$$

$$C_N = C_L \cos \alpha + C_D \sin \alpha \quad (2.3.16)$$

$$N = C_N \times 0.5 \rho A v^2 \quad (2.3.17)$$

$$Q_R = N \times \bar{x} \quad (2.3.18)$$

$$C_R = \sqrt{C_L^2 + C_D^2} \quad (2.3.19)$$

$$R = C_N \times 0.5 \rho A v^2 \quad (2.3.20)$$

$$M = R \times \bar{y} \quad (2.3.21)$$

$$BM_E = \frac{M}{2} + \sqrt{M^2 + Q_R^2} \quad (2.3.22)$$

The rudder stock requirement was as follows:

$$D = \sqrt[3]{\frac{BM_E \times 32}{\pi \times \sigma}} \quad (2.3.23)$$

where σ is the allowable stress in the stock material.

Results

Interpolated values of C_L , C_D , CPs and CPc and corresponding rudder torques and bending moments are presented in Table 2.3.7. According to the results, the servo was required to provide at least 3.2 kg·cm of torque to the rudder. The diameter of a steel rudder stock would need to be larger than 4.45 mm. The available stainless steel rod for rudder stock had a diameter of 8 mm and was therefore considered to be suitable.

A similar table of result was obtained for the semi-balanced rudder and the comparison is shown in Figure 2.3.3. From this figure, it can be seen that the lift coefficient generated by semi-balanced rudder is slightly higher than by the spade rudder, which may have been the reason why the full-scale vessel was designed with a semi-balanced rudder. However, for the purpose of this project, a spade rudder should be sufficient.

Table 2.3.7: Rudder torque calculations, results are shown in bold font.

	Unit	30°	20°	10°	-10°	-20°	-30°
C_L		4.009	2.595	1.276	-1.476	-3.190	-4.604
C_D		1.257	0.530	0.121	0.121	0.530	1.257
C_N		4.101	2.620	1.278	-1.475	-3.179	-4.616
N	N	29.751	19.008	9.270	-10.699	-23.065	-33.490
\bar{x}		-0.009	-0.016	-0.020	-0.012	-0.012	-0.012
Q_R	Nm	-0.260	-0.306	-0.184	0.124	0.270	0.407
	kgcm	-2.655	-3.119	-1.877	1.267	2.749	4.150
C_R		4.202	2.649	1.282	1.481	3.234	4.773
R	N	30.484	19.217	9.300	10.745	23.463	34.628
\bar{y}		0.082	0.085	0.087	0.060	0.063	0.063
M	Nm	2.497	1.632	0.807	0.642	1.472	2.172
BM_E	Nm	2.504	1.646	0.817	0.648	1.484	2.191
Rudder stock	m	0.005	0.004	0.003	0.003	0.004	0.005

It was acknowledged that the lift coefficient generated by the airfoil in laminar flow around the model rudder is smaller than the lift coefficient generated in the turbulent flow at full scale. Initially, the software XFOIL [Drela and Youngren, 2001] was used to predict lift coefficients of NACA 0020 with different angles

of attack and given Reynolds number. However, it failed to provide lift coefficients with the angle of attack higher than 10° . Fortunately, a similar software tool called XFLR5 was able to calculate the lift coefficients as can be seen in Table 2.3.8. The Reynolds number was calculated using the mean chord length of the rudder.

$$Re_{r_m} = \frac{Mean chord \times model speed}{Dynamic viscosity of fresh water} = \frac{0.0752 \times 1.029}{1.14 \times 10^{-6}} = 67908 \quad (2.3.24)$$

$$Re_{r_s} = \frac{Mean chord \times ship speed}{Dynamic viscosity of salt water} = \frac{4.514 \times 7.97}{1.19 \times 10^{-6}} = 3.02 \times 10^7 \quad (2.3.25)$$

Table 2.3.8: Lift coefficients generated by XFLR5.

Angle of attack	Model	Ship	Ratio
10	1.08	1.2	1.11
15	1.5	1.7	1.13
20	1.62	2.1	1.30
25	1.42	2.2	1.55
30	1.28	1.8	1.41

It was shown in Table 2.3.8 that the lift coefficient generated by a NACA 0020 section in flow near rudder of the real ship is about 1.4 times higher than in the flow near rudder of the model. Therefore, the area of the model rudder should be 1.4 times larger than the geometrically scaled rudder. The depth of the rudder should not be changed, as it would be deeper than the keel and cause much additional resistance. Therefore, it was decided that the length of the airfoil should be increased by 1.4 times.

To test whether this analysis was correct, and also to observe changes in rudder size on the manoeuvres, two rudders were built: one to scaled dimensions, and one with an increased chord.

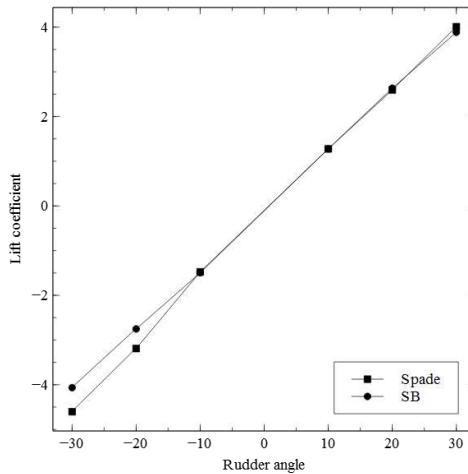


Figure 2.3.3: Lift coefficients of a spade rudder and a semi-balanced (SB) rudder.

Rudder Rate of Turn

From the sea trials data provided by BP, it was found that the full-scale rudder was able to turn from -10° to 10° within about 5 seconds, or rather, $4^\circ/s$ as can be seen in Figure 2.3.4. It is noted that the dimension of the rate of turn r_t is degree/second. Therefore, the non-dimensional group $\frac{r_t \times L_{BP}}{v}$

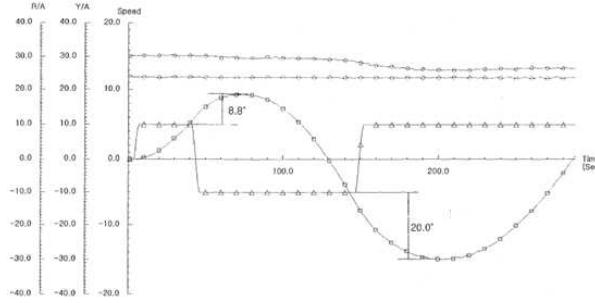


Figure 2.3.4: Zigzag manoeuvring test results of real ship.

assumed to be constant in order to scale the rate of turn of rudder to model's scale. With this assumption, the required rate of turn of the model rudder was calculated as:

$$r_{tm} = \frac{v_m}{v_s} \times \frac{L_s}{L_m} \times r_{ts} = \sqrt{60} \times r_{ts} = 31^0/s \quad (2.3.26)$$

DESIGN & BUILD

This chapter covers the design and build of the test platform, broken down into a number of individual sections, such as the power train, autonomy unit and rudder system. The buoys will be covered in their own chapter (4).

3.1 System Overview

[HAWKES & MOZDEN]

This section aims to give an overview of the whole system and how it fitted together. A 1/60 scale model had already been built, and after stripping out the old powertrain and other fittings the design and build of a new system proceeded. The strip-planked, cold-moulded, hollow model was approximately 3 m long, with a removable bow section and plenty of attachment points inside the hull.

It was possible to break the design & build process down into a number of sections. The objective of building a self-powering, self-measuring, autonomous vessel gave rise to some obvious sections: such as developing a new power train with thrust and torque sensors; developing a rudder system that could steer the model; and developing an autonomy unit that could control the entire system and perform the necessary test procedures.

However, there were a number of other sections which were not apparent at the outset, but required careful development and considerable time. This included weighing the entire model and its components accurately and creating a ballast system to match the full-scale vessel's displacement and trim conditions.

The interface between the motors, servos and sensors and the high-level autonomy unit also needed tackling - since motors or strain gauges do not simply plug into a computer. This was covered by an electronics section, which also dealt with the powering of the entire vessel.

The design and build was thus split into seven sections, as follows, which were very cross-dependent:

- Power Train: This covered the fitting of the motor, drive shaft and propeller, and included issues such as building a watertight stern-tube, aligning the shaft and measuring thrust. The thrust sensor was encompassed in this section, since it was fundamental to the design of the power train. This section was covered by two group members [Mozden & Crossley].
- Torque Dynamometer: This sensor deserved a section of its own, since it was a modular addition to the power train and was also particularly complex. As it turned out, this sensor would not be connected to the electronics system, so its design was dependent on the design of the autonomy unit; as well as the physical design of the power train. [Crossley]
- Electronics: This section included the powering of the vessel and the connection between the autonomy computer and all the sensors and drivers on the vessel. It was particularly dependent on the power train, since it had to interface with the motor and thrust sensor; the rudder system which utilized a servo and rotation sensor; and the autonomy unit, from which it would receive commands and send sensor results. [Hawkes]
- Rudder System: This covered the design and build of two rudders which scaled to the full-scale ship and the creation of a system to control it. [Le & Crossley]

- Operating System: This included the autonomy unit and its peripherals, which provided the ‘brain’ to the vessel. It was dependent on the electronics system to provide its power and communication to the devices that it was trying to control. [Anderlini]
- Miscellany: This included the design and construction of a watertight cover and suitable superstructure. [Neale & Thornton]
- Weights and Trim: This section covered the weighing and ballasting of the model to match the full-scale ship. It was dependent on every other section of the design and build since each section added weight to the vessel, and the ballast weights competed for space inside the model. [Neale]

Apart from the technical cross-dependence of these sections, there was also the physical challenge of fitting all the hardware into the vessel. The vessel was modelled in 3-D software prior to construction; to ensure there would be no clashes between different sections. [Mozden, Hawkes, Neale]



Figure 3.1.1: [Left] 3-D model and [right] actual model at ballast condition in Timsbury Lake.

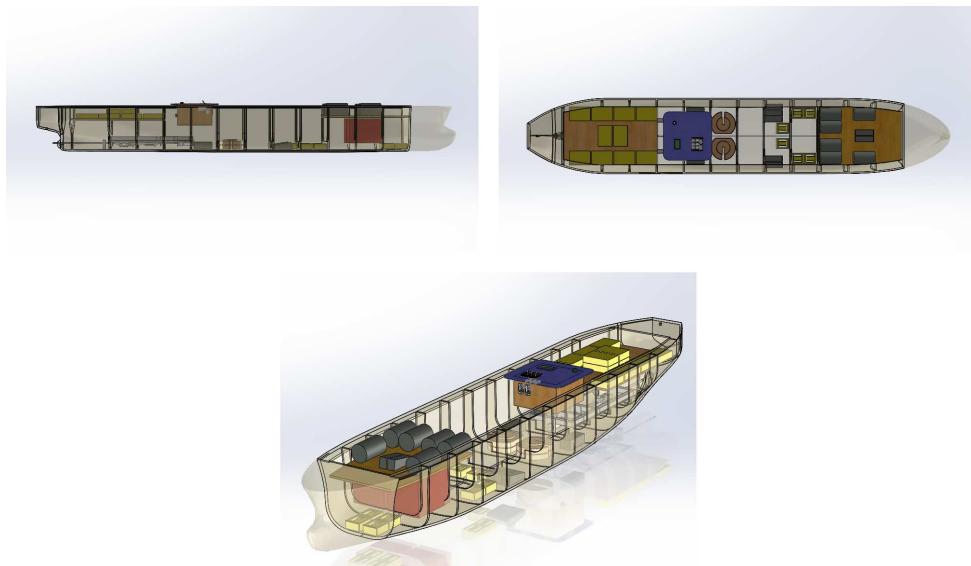


Figure 3.1.2: 3-D model of the internal layout for towing tank testing, showing the positioning of all the weights at full load condition. Different permutations arose for different test conditions. Note that for tank testing, the tow post attachment requires most of the midships space.

3.2 Power Train

[MOZDEN]

This section discusses the design and construction of the power train unit. As discussed in the Literature Review, it was necessary to redesign the power train rig to improve the quality of the resistance data; a new, smaller and more efficient motor was also incorporated. The purpose of the power train unit was to allow the shaft line to ‘free-float’ within the hull ensuring the thrust from the propeller was transferred solely through the thrust block, allowing the full force to be measured, minimising losses via other contact points for most accurate data.

3.2.1 Design Concept

[MOZDEN]

The power train unit contained the motor and the torque dynamometer (detailed in Section 3.3). In a typical ship, the propeller shaft is supported by several bearings and the engine is mounted on rubber feet to reduce vibrations. The majority of the thrust is transmitted through the thrust block, but every point of contact between the shaft and hull will transmit some of the load; for this model platform it is vital that all the thrust is transferred through a single point so it can be accurately measured. Hence, a ‘free-floating’ design concept has been used.

The concept was to mount the complete power train on a rig which sat on a pair of linear bearings. This would be able to slide with low friction along the ship; the only fixed contact between the rig and the hull would be through the thrust block which transmitted the force through a strain gauge. The strain gauge would record the end deflection, corresponding to the distance the propeller shaft had moved forward in the hull before the thrust force has overcome the hull resistance and begun to propel the ship. The linear bearings needed to have as little friction as possible, so that they did not damp the thrust force; allowing it to be transferred to the hull via the thrust block. The schematic layout for the unit can be seen in Figure 3.2.1.

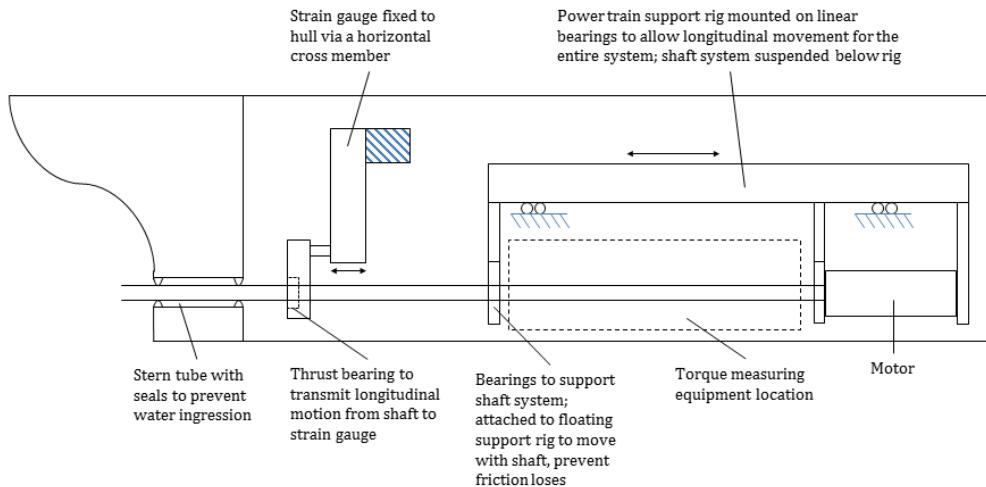


Figure 3.2.1: Concept diagram for power train assembly.

3.2.2 Motor Selection

[MOZDEN]

The motor is the most important component within the shaft line system; it is necessary to get a suitably scaled torque range in order to get data in the correct powering ranges. The previous rig used a large diameter drill motor in order to simplify the control system, however this did not have a suitable torque range and poor high RPM performance. It was also too large to sit in-line with the shaft, resulting in a pulley and belt drive to be installed which caused many mechanical losses. It was important that the new propulsion rig did not repeat the problems previously experienced.

The Preliminary Calculations Section 2.3 discussed the model-ship scaling methods in order to estimate the resistance and propulsion requirements of the model. These needed to be used for determining the correct motor choice. Table 3.2.1 lists the relevant powering requirements which would scale up to the operating conditions of the full ship.

Table 3.2.1: Operating requirements for the motor.

Required values		
Thrust	(N)	10
Torque	(Nm)	0.14
Angular velocity	(RPM)	1100
Delivered power	(W)	13

It was important to select a motor which could comfortably achieve these specifications and not be operating at its maximum capacity as it did so. All motors have an operating range and by increasing the electrical power input to them they can be sped up; to ensure the system was efficient a high performance motor was selected which could be operated at a lower capacity during the testing.

A Maxon 12 V DC motor (#148866) with a 3.5:1 reduction gearbox (#203113) was selected. It is a very powerful motor but extremely small (40 mm diameter) which allowed it to be directly attached to the shaft, removing the need for a complex linkage. It was rated at 150 W and required a 12 V DC supply for maximum RPM. After the reduction gearbox, this maximum RPM was expected to be 1977 RPM; this would allow for some overhead in the system in order to perform faster trials than the operating conditions if necessary. At the default self-propulsion point (1100 RPM), the motor was expected to produce approximately 0.2 Nm of torque, based on the speed-torque gradient 4.05 RPM/mNm. This motor almost perfectly meets the self-propulsion criteria, hence was selected for the model.

3.2.3 Thrust Measurement

[MOZDEN]

The various options for measuring thrust were discussed in the Literature Review (Section 1.5). The different options were investigated, but the high precision LVDT and optical sensors were far too expensive for this project (quotes were in excess of £2000 for each) and the accuracy would have been unnecessary as losses from other parts of the system would reduce the degree of accuracy. In order to measure the thrust generated by the propeller a load cell was used; it was rigidly fixed at one end to the hull, whilst the other end sat in contact with the propeller shaft via a thrust bearing. As the force pushed the shaft forwards, the load cell deflected at one end, transferring this to an electrical signal which was sent to the electronics system for processing; as this was the only fixed point of contact between the shaft and the hull, all the thrust was transferred through this unit and hence accurately measured.

The load cell used was for this system was taken from the self-propulsion rig from the previous GDP, a Honeywell load block with a load capacity of 25 N. The load cell consisted of 2 gauges measuring the deformation of each side of the cell and produced a signal output proportional to the deformation, and

hence thrust, experienced. In the previous project there were concerns regarding the accuracy of the cell, however it was the linkage between the load cell and the shaft causing the poor data rather than the cell itself. As discussed in Section 2.3 the thrust produced by the propeller was predicted to be of the order of 10 N; this cell was deemed to have the correct capacity range for this system.

Figure 3.2.2 shows the final load cell arrangement; a thrust bearing was pressed onto shaft which transmitted the thrust force off the shaft and onto the plate attached to the end of the load cell. To ensure the plate only exhibited a linear transformation and did not bend, it was reinforced with a metal frame; the location of the plate relative to the load cell was adjustable to ensure there was no initial preloading.

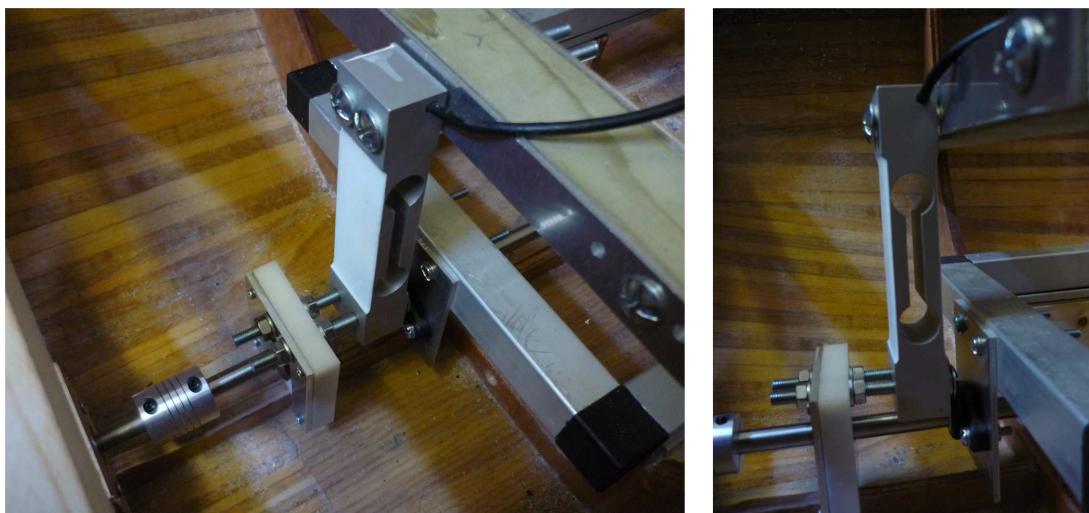


Figure 3.2.2: Final arrangement of the thrust load cell.

3.2.4 Additional Components

[MOZDEN]

Several additional components were needed to fit the power train system into the model tanker. Firstly, the stern tube needed to be modified to ensure it would be watertight, as it is the only hole in the hull below the waterline, it was important to prevent water ingress. A nitrile rubber seal was fitted in the outer end of the stern tube; at the internal end a ball bearing was fitted to support the shaft. This was necessary as the weight of the propeller would cause a large bending moment, damaging the shaft and potentially inducing cyclic loading.

The shaft was broken into several sections, joined by linear shaft connectors. This enabled the rig to be removed for maintenance without disturbing the stern tube. A second shaft connector was used to attach the motor spindle to the shaft. It was important to ensure each shaft section was fully inserted in the connector to prevent slip and longitudinal movement which could lead to shaft slippage.

3.2.5 Power Train Unit

[MOZDEN]

The power train unit was constructed of *EasyFix* aluminium tubing. This is a system consisting of an aluminium square cross section that can be cut to size and joined using a selection of 90° connection blocks which push into the ends of the tubing. It was a very quick construction method and ensured the structure was square and rigid. Once the framework was assembled, additional plates could be bolted to the aluminium tubing to suspend the shaft system below.

The framework was mounted on a *Drylin N* low-profile linear bearing system. This consisted of two anodised aluminium rails fixed to the hull, each with two plain linear bearings fixed to the power train frame. This bearing system was chosen as it was maintenance free and self-lubricating, and is suitable for high speed and acceleration systems. The plain bearings sat within a slot in the linear rails, they ‘float’ within the gap to provide a small degree of flexibility if the rails were not perfectly parallel.

The complete unit can be seen in Figure 3.2.3, and the full technical schematics of the power train system are in the appendix (11).



Figure 3.2.3: Complete power train unit; mounted on two linear bearing rails with the shaft system hung below. The thrust block transfers the full thrust via a vertical strain gauge to the hull enabling accurate measurement. The images were taken during alignment so the dial gauge is shown (discussed later).

3.2.6 Shaft Alignment Procedure

[MOZDEN]

It was vital to ensure the shaft was as flat as possible, relative to the keel line, to ensure there were no vertical thrust components which would equate to losses and not be recorded fully by the strain gauge in the thrust block. Along the shaft there were several bearings and components to support the load (as discussed in the previous sections) which need to be perfectly level, both horizontally and vertically, to minimize shaft whirling and rotating imbalances. This would lead to uneven wear on the supports and over time - mechanical failure, as well as varying frictional forces during each revolution.

The shaft was aligned using a Bosch Quigo self-leveling cross-line laser level projected down the shaft line with the bearings located round it and being fixed into place. The full procedure is outlined below (see Figure 3.2.4 for images of the process):

1. The hull was levelled using precision spirit levels to achieve longitudinal and transverse level.
2. A self-levelling cross-line laser was positioned at the location of the propeller and projected through the stern tube along the length of the hull onto graph paper at the far end.
3. Bearing plates were mounted to the power train chassis which blocked the laser, allowing only the central cross to reach the graph paper; the visible cross was measured and the plates positioned to ensure the bearings were central to the laser’s centre.
4. The bearing plates were fixed into place, working from stern towards the front of the vessel, the motor attachment plate were fixed last.

5. The shaft was inserted and a DTI dial gauge was used to ensure the shaft span true, individual adjustments for each bearing could be made to remove any local offsets to minimise shaft whirling; the final maximum offset read by the gauge for the final shaft line was ± 0.05 mm diameter.



Figure 3.2.4: Breakdown of each step of the alignment procedure required to ensure the shaft was horizontal with minimal lateral offset which could induce shaft whirling.

3.3 Torque Dynamometer

[CROSSLEY]

This section examines the design and construction of a torque dynamometer. As discussed in the Literature Review (Section 1.5), it was necessary to build a new dynamometer, although the current method of using strain gauges mounted on flexures to measure the torque was to be kept.

3.3.1 Design of the Structure

[CROSSLEY]

There were two key design constraints that had to be addressed during the development of the torque dynamometer. These were the space available; and how to transfer the power and generated signal to and from the rotating shaft.

The previous group [Collison et al., 2012] had used slip rings, which are a series of metal brushes that pass onto a series of pads to transmit the power and signal. These are very expensive to buy, as the contacts must be made of precious metal in order to avoid any noise inclusion into the signal. The slip rings employed previously were housed in a unit with a large diameter that meant the shaft had to be angled, thus resulting in a loss in propulsive efficiency due to the generation of a vertical thrust component.

It was initially looked into purchasing a new set of slip rings with the correct diameter. However, the quotes came back at around £1000, which exceeded the available project budget. Slip rings from car alternators were also considered since they were cheap, as they were comprised of copper contacts (rather than superior silver). Nevertheless, it would be difficult to fit these. Another idea was to make a set of basic slip rings using ball bearings to transmit the power onto a conductive metal ring set into the shaft isolated using araldite infused epoxy paper. These two options were not used as the quality of transmission would be too poor, leading to an inconsistent power supply to the strain gauges as well as noise inclusion for the generated signal; both issues leading to unreliable torque readings.

The final decision was to make the torque signal processing a separate unit with its own power supply in the form of a battery and have the signal transmitted via Bluetooth. The big advantage of this is that the generated signal from the strain gauges could be immediately amplified and converted from analog to serial thereby reducing the chance of noise affecting the quality. The other advantage is that the battery will provide a steady and uninterrupted power supply, again improving the quality of the data obtained from the strain gauges.

The space available for the torque dynamometer was constrained by the spacing between the ribs of the model, as well as the clearance from the shaft - limiting the maximum diameter. The material chosen was aluminium as this has a lower density than steel and is therefore lighter. The dynamometer was suspended from two bearings and it was important to minimize the effect of the weight of the unit itself upon the flexures, as this can give an error over time and cause permanent deflections to the flexures, as noted in the 2012 GDP [Collison et al., 2012]. This effect must be considered when calibrating the sensors.

The final design of the dynamometer housing consisted of seven pieces bolted together as shown in Figure 3.3.1. The electronics mounting platform was rapid prototyped in two parts due to printing limitations. This method of construction was selected as it permitted the construction of a lightweight component that has exactly the correct dimensions, whilst offering a platform to which electronics can easily be attached. There was also a concern that too much metal close to the Bluetooth transmitter might affect the signal. For this reason, the metal parts have been designed to allow the dynamometer to be assembled easily with the bolts as well as providing gaps for the electronics wires to be passed

through.

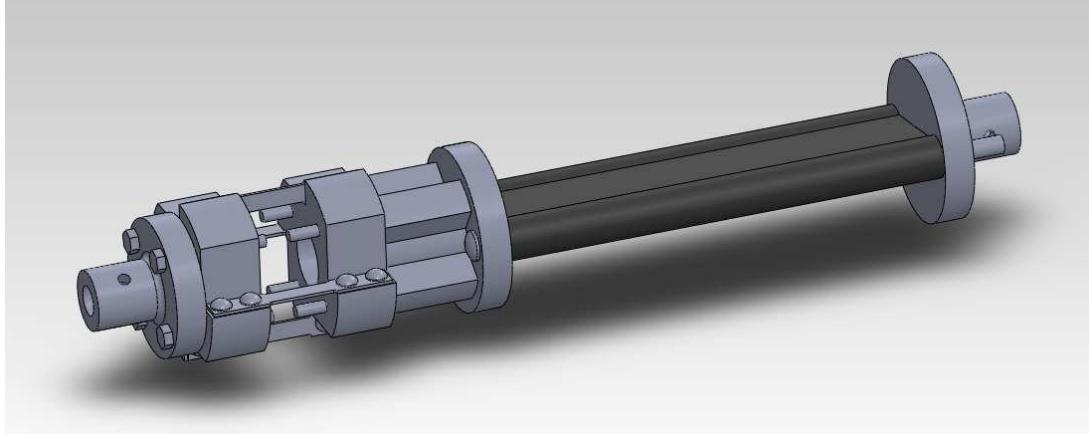


Figure 3.3.1: 3D CAD drawing of completed dynamometer.

3.3.2 Flexure Design

[CROSSLEY]

The flexures are the four pieces of metal that have the strain gauges mounted upon them within the diameter. They are thin pieces of aluminium designed to give the greatest range of deflection for the expected range of torque to be experienced, in order to achieve the best signal output. The limitations for the design are the material constraints, as there must be a significant safety margin before failure for the loads expected as well as considering manufacturing limitations. The aluminium used was taken to have a Young's Modulus of 69×10^9 N/m² [Calvert and Farrar, 2008].

The design torque is taken to be 0.14 N and the maximum permissible mean flexure separation radius is 0.025 m, which is taken as the distance between the centrelines of opposing strain gauges. This was calculated based on the 5 cm clearance between the shaft centreline and hull floor as well as 1.5 cm high centre rib giving a maximum radius of 3.5 cm. Allowing for a clearance of 1 cm, this gave the final radius as 2.5 cm.

The selected strain gauge was a basic 120 Ω resistance aluminium foil strain gauge from RS components. Its properties are shown in Table 3.3.1 and it was chosen because it had the greatest gauge factor available, which is important for signal generation, as it is a measure of sensitivity.

Table 3.3.1: Properties of the chosen strain gauge.

Property	Value
Gauge Factor	2.1
Gauge Length (mm)	5
Resistance (Ω)	120
Width (mm)	3.5
Overall Length (mm)	9.5

A number of different dimensions were investigated for the length, width and thickness of the flexure in order to find the best output voltage for the design torque, as shown in Tables 3.3.2 and 3.3.3. These results were obtained following the procedure described in a paper by Molland and Turnock [1990].

As can be seen from Table 3.3.2, a flexure with the dimensions of 21 × 6 × 1 mm gives a high output voltage whilst still maintaining a high moment of inertia, which is important for strength properties. It is important to check that the torque expected does not cause stress that will exceed the material

Table 3.3.2: Flexure design calculations.

L (mm)	W (mm)	T (mm)	I/Y (mm ³)	I (mm ⁴)	Torque Force per Flexure (N)	Root Strain	Deflection (mm)
10.0	6.0	1.0	1.00	0.50	1.4894	0.000108	0.003597
20.0	6.0	1.0	1.00	0.50	1.4894	0.000216	0.028780
21.0	6.0	1.0	1.00	0.50	1.4894	0.000227	0.033316
21.0	5.0	1.0	0.83	0.42	1.4894	0.000272	0.039980
21.0	7.0	1.0	1.17	0.58	1.4894	0.000194	0.028557
21.0	6.0	2.0	4.00	4.00	1.4894	0.000057	0.004165
21.0	6.0	0.5	0.25	0.06	1.4894	0.000907	0.266531

Table 3.3.3: Flexure design calculations.

L (mm)	W (mm)	T (mm)	Distance from Root (mm)	% of Root Strain	Maximum Strain at Gauge	Output Voltage (mV)
10.0	6.0	1.0	1.00	30.00	0.000032	503.15
20.0	6.0	1.0	1.00	65.00	0.000140	2180.30
21.0	6.0	1.0	1.00	66.67	0.000151	2348.01
21.0	5.0	1.0	1.00	66.67	0.000181	2817.61
21.0	7.0	1.0	1.00	66.67	0.000130	2012.58
21.0	6.0	2.0	1.00	66.67	0.000038	587.00
21.0	6.0	0.5	1.00	66.67	0.000604	9392.04

properties of the flexure. The yield strength of aluminium was taken to be 160 N/mm² [Calvert and Farrar, 2008]. For the chosen flexure, using an initial thrust estimate of 8.6 N, the tensile stress due to thrust is 0.358 N/mm² and the shear stress is 0.233 N/mm². Both of these values fall well below the yield strength even with a considerable safety margin. Hence, the flexure dimensions can be considered as acceptable.

There was an option to have either a total of four strain gauges or eight strain gauges; the latter situation being with a gauge either side of the flexure in series. According to Molland and Turnock [1990], the recommended power dissipation should lie in-between 0.0031 – 0.0078 W/mm². Therefore, the eight strain gauge configuration is suitable for this application as shown in Table 3.3.4. The other advantage of this arrangement is that the bridge response is independent of the direction of application of torque. This would be very useful for the torque analysis of manoeuvres such as crash stop tests of the model.

Table 3.3.4: Comparison of 4- and 8-gauge Wheatstone bridge circuits.

Property	4 Gauge	8 Gauge
Equivalent Resistance (Ω)	240.0	120.0
I (A)	0.03083	0.06167
I per arm (A)	0.01542	0.03083
Power per Gauge (W)	0.02852	0.11408
Gauge Area (mm ²)	17.50	17.50
Power Dissipation(W/mm ²)	0.00163	0.00652

3.3.3 Dynamometer Electronics

[CROSSLEY]

The voltage output from the strain gauges is amplified from microvolts to volts in order to minimize the effects of noise on the clarity of the signal. This is done using an amplifier circuit, shown in Figure 3.3.2, designed by the electronics laboratory in the Institute of Sound and Vibration Research (ISVR) at the University of Southampton.

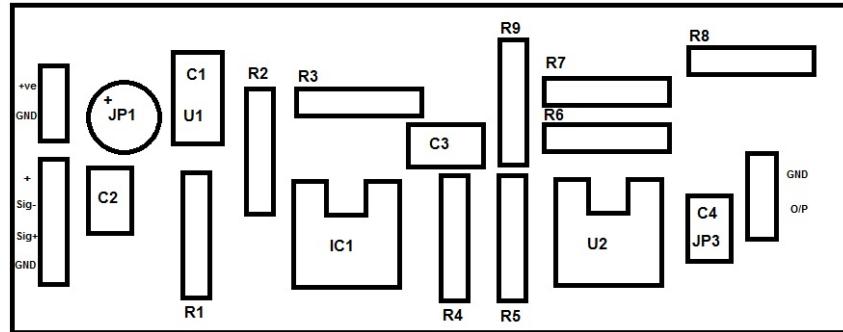


Figure 3.3.2: Schematic of the amplifier circuit.

The amplified signal is then passed to an Arduino Pro Mini which controls an attached Bluetooth module in order to transmit the signal in serial to an external dongle. The Bluetooth module was selected for ease of use, as it is designed specifically to attach directly onto the Arduino, and to save space, since both components are very small.

The selection of the batteries was based on the power consumption of the electronics and strain gauges as well as taking into account space restrictions within the dynamometer. Table 3.3.5 outlines the expected power consumptions of the components. A number of different batteries were considered as outlined in Table 3.3.6.

Table 3.3.5: Power consumption of torque dynamometer components.

Component	Current (mA)
Arduino Pro Mini	150.00
Bluetooth Mate Gold	50.00
Strain Gauges	61.67
Total	261.67

Table 3.3.6: Battery Properties Comparison.

Battery	Voltage (V)	Capacity (mAh)	Hours (h)	Length (mm)	Weight (g)
4 x Alkaline single use AAA	6	4000	15.29	44.5	46
4 x NiMH rechargeable AAA	4.8	2400	9.17	44.5	44
Alkaline single use PP3	9	550	2.10	48.5	48
NiMH pack	6	3300	12.61	115	325
Energ-pro 25C LiPo	7.4	400	1.53	54	32
Axion R3 LiPo	7.4	1300	4.97	70	73

The chosen battery was the Energ-pro 25C LiPo battery for a number of reasons. A rechargeable battery was deemed to be a better choice for this application, as it can be ensured that the maximum capacity is available before each set of testing. Additionally, the cost of replacing four AAA batteries before each test run would soon add up and thus would not be an economic choice. The NiMH rechargeable

batteries unfortunately did not yield the required minimum voltage output of 5V and therefore were not a viable option. Both the NiMH and Axion R3 LiPo battery packs were too long to fit in the space allocated. This left the Energ-pro LiPo battery pack as the only viable option and although it has the lowest capacity, it was the best choice.

The strain gauges were fixed onto the flexures using super-glue. Even though specialist glue is available, standard super-glue works just as well because the flexibility of the glue is greater than that of the flexures, thus not affecting any strain reading.

The strain gauges were wired up in a four-arm Wheatstone bridge configuration as shown schematically in Figure 3.3.3. The finished product can be seen in Figure 3.3.4.

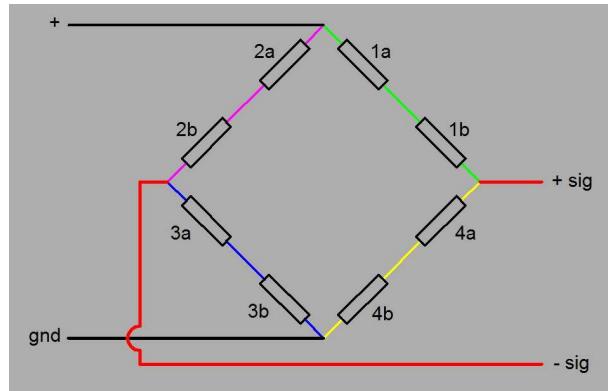


Figure 3.3.3: Schematic of the Wheatstone bridge wiring connecting the 8 strain gauges.

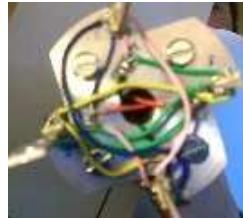


Figure 3.3.4: Photograph of the completed Wheatstone bridge wiring.

A basic Bluetooth dongle was used to connect to the Bluetooth module. The Arduino was programmed to send out an increasing counter value through the serial line in order to test the Bluetooth connection. The serial speed must be set to 115200 baud rate as this is the default setting for the Bluetooth Mate. A basic program called PuTTY [Tatham, 2013] was used to receive the data on the laptop as it is free and simple to use as well as capable of logging all the data transmitted for analysis.

Unfortunately, there was a bad connection somewhere within the wiring of the dynamometer - either in the Wheatstone bridge or the amplifier provided by the ISVR. Due to time constraints this connection could not be found, and there has been no success in transmitting a signal generated from the strain gauges through the Bluetooth Mate. Therefore, the strain gauges may need rewiring in the future. A method for attaching the electronics components to the plastic holder should also be developed, as this stage was not reached.

Fortunately, there was a backup to the torque dynamometer in the form of a current sensor, which could measure the electrical current consumption of the motor and relate this to propeller torque. This will be discussed later, in Sections 3.4.4 and 5.1.3.

3.4 Electronics

[HAWKES]

'Electronics' covers the hardware-software interface between the autonomy unit, which will be introduced in Section 3.6, and all the sensors, motors, servos and other hardware on the vessel; whilst also dealing with the power supply for the vessel.

In a normal tank environment, the raw analog signals from various sensors on-board would be sent on-shore, where they would be dealt with using bulky, expensive signal processing units. Similarly, shore-based power would be supplied to the vessel and most hardware controllers would also be mounted on the carriage. The challenge with this project was to rebuild this equipment into a small, portable system without sacrificing quality.

Most of the electronics were prototyped on breadboard, which is a solderless board on which electrical components and jumper cables can be slotted together to create a working circuit. Ideally, the whole electronics system would have been compressed onto a printed circuit board using surface-mount components. This type of system would be expensive, but would have shrunk the size of the entire system considerably. As a compromise, the electronics were moved from breadboard to perfboard (sometimes called prototype board) where the components were soldered into place. This was highly robust, since the solder maintains a secure connection, but allows for modification during the project and expansion in the future.

3.4.1 Overview

[HAWKES]

To deal with the conversion from high-level logic (*i.e.* the autonomy computer) to low-level logic (*i.e.* raw analog and digital signals) suitable microcontrollers (MCU) were required. For this project, Arduinos were chosen, which are 'Atmel AVR' MCUs pre-assembled to provide easy access to the input/output pins. They are programmed using a specialist Arduino development environment that provides a C library to wrap around the raw AVR programming language [Arduino.cc, 2012].

Specifically, the Arduino Pro Mini (5 V/16 MHz) MCU was used because it operates at a suitable speed, was cheap (£14) and came in a DIP format, which is a physical footprint which mates with breadboard and perfboard. It features 32 kB of flash memory, where the program is stored, and 2 kB of SRAM, where run-time variables are stored. Programming of the MCU, and USB communications to a high-level computer, were performed using a USB-to-serial converter, whereby the serial signal is handled by the MCU's on-board transceiver. Figure 3.4.1 shows an Arduino Pro Mini with a serial converter attached, and some perfboard, on which the electronics were assembled.

The speed of these MCUs (16MHz) was more than adequate for this project, but the SRAM was quickly exhausted. To circumvent this problem multiple Arduinos were used, where each one formed a 'module' that would handle a specific part of the system.

These modules communicated through a digital communications protocol called the two-wire I2C interface[Philips Semiconductors, 2000]. This allowed the modules to be located close to their respective sensors or controllers, reducing noise on analog signals - since I2C can handle cable lengths of several metres. Additionally, the I2C bus is a master-to-multi-slave network - meaning that one MCU could be designated as the master, and multiple MCUs could connect to that master to send and receive data. This allows up to 127 devices to be connected to the master MCU, unlike serial which is strictly 1-to-1 communication. This was useful considering the number of modules communicating in this project, and allows additional modules to be added easily in the future.

A breakdown of the system is shown in figure 3.4.2, which illustrates the different modules and their primary functions. In the following sections each module will be discussed by considering its design,

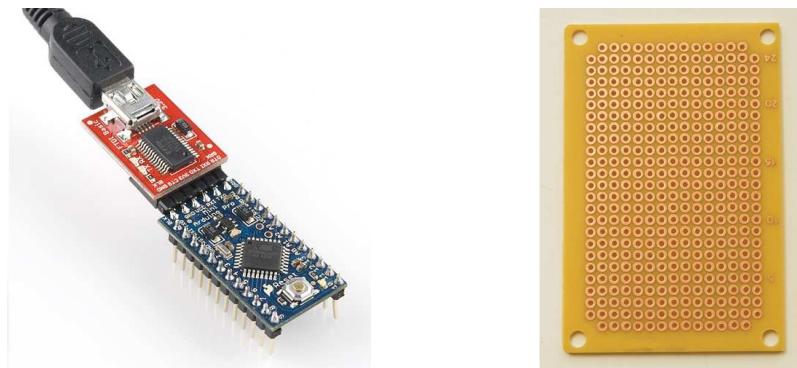


Figure 3.4.1: [Left] The Arduino Pro Mini, based on the ATmega328, shown with a USB-serial converter. [Right] A bare prototyping board or perfboard ready for 0.1" through-hole components such as DIP packages.

construction and programming.

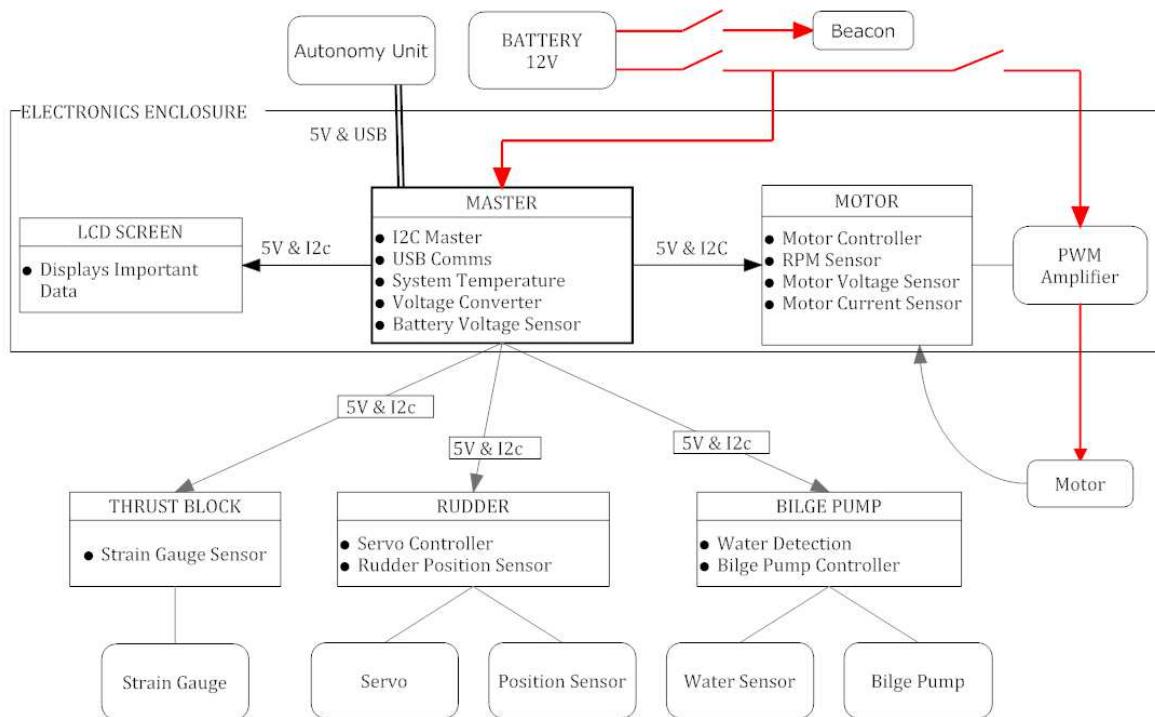


Figure 3.4.2: Overview of the electronics system. The various modules and peripherals will be discussed in this section.

The electronics were mostly built into an enclosure which formed one of the hatch covers on the main deck. The various modules connect to this enclosure using four-core speaker cable, which carries the power and I2C communications. These cables are rated up to 10 A and plug into the enclosure using 4-pin ATX connectors, as found in most desktop computers.

The electronics system has three main isolation switches - one for the MCUs, LCD screen and sensors ('control'); one for the amplifier and motor ('motor'); and one for the flashing beacon ('light'). These were added for safety, so that the entire system could be shut down instantly in case of emergency; and

also for convenience, so that the electronics systems could be tested or remain powered whilst work was carried out on the vessel. The motor cannot be powered-on when the control system is turned off to prevent any unexpected motor activity and consequent risk of injury.

Figure 3.4.3 shows the main electronics enclosure, which was neat and robust - although this is difficult to illustrate.

3.4.2 Powering

[HAWKES]

The motor, which was selected in section 3.2.2 is a 12 V DC motor; and the computer, which performs the autonomy tasks, requires 5 V. Therefore, two voltage levels were used in this system - 12 V and 5 V. A DC-DC converter was used to step-down the voltage from a 12 V supply. The 12 V circuitry was all contained within the main electronics enclosure, which was positioned near the motor.

It was important to estimate the power consumption of the vessel, in order to choose a suitable 12 V power source.

The motor is a 150 W motor, which means it has a $\frac{150W}{12V} = 12.5$ A continuous current draw, which increases slightly as the coils inside the motor heat up. However, the motor was likely to draw much less than this, since it exceeded the power requirements of the vessel considerably (Table 2.3.5) - a generous estimate of 4 A was assumed, which included frictional and electrical losses (estimated as a 70% loss), for continuous operation. A 12 V beacon was also installed on the vessel, drawing about 0.5 A.

At the 5 V level there were a number of MCUs and other devices, which were estimated to consume 2 A of current (for reference, each MCU draws a maximum of 200 mA). However, the servo on full-lock was rated up to 4 A, the bilge pump up to 2 A and the autonomy computer up to 1.25 A (with USB peripherals). Since the servo and bilge pump would not be drawing significant current most of the time, it was reasonable to estimate a total continuous current draw of 4.25 A and a peak current draw of 9.25 A. The DC-DC converter that was chosen to handle this 5 V current will be introduced in Section 3.4.4. It advertises a an efficiency of 93% when operating with a 12 V input and 5V output (at 4.25 A), meaning that its current consumption at the 12 V supply would be close to 2 A for continuous operation.

This brought the total estimated continuous current draw to 6.5 A, but 7.5 A was assumed as a safety margin.

There were several 12 V power sources that could have been appropriate, including lead-acid batteries, lithium-ion batteries or lithium-polymer batteries. For this project it was decided to use car batteries (lead-acid). Lead-acid batteries were actually cheaper, per-weight, than solid lead, and at 14 kg each, made for good ballast; whereas lithium batteries are intended for low-weight applications, and would have been an expensive and unnecessary option for this project. The best price-to-capacity ratio for car batteries meant that 60 amp-hour batteries were the best option, each of which would supply at least 8-hours of charge to the model.

It was estimated that a full day of testing (\approx 8 hours) would not require more than 6 hours of continuous operation so one battery would have been adequate. However, due to the internal resistance of the batteries and inevitable circuit resistance, the supply voltage wanes at high-load. Additionally, as a battery reaches the end of its capacity, the supply voltage may drop to < 11 V, even at zero-load. This could have had a negative impact on the testing since the supply voltage cannot be kept constant throughout the day. To alleviate this problem, an extra car battery was added in parallel: this gives the model at least 16-hours continuous use and reduces the voltage losses in a typical day of testing.

The logic circuits should never suffer due to this voltage drop, since the DC-DC converter provides a constant output so long as the input voltage (12 V) is higher than the output voltage (5 V). However, it is not ideal to allow the maximum motor voltage to decrease throughout the day, simply for consistency.

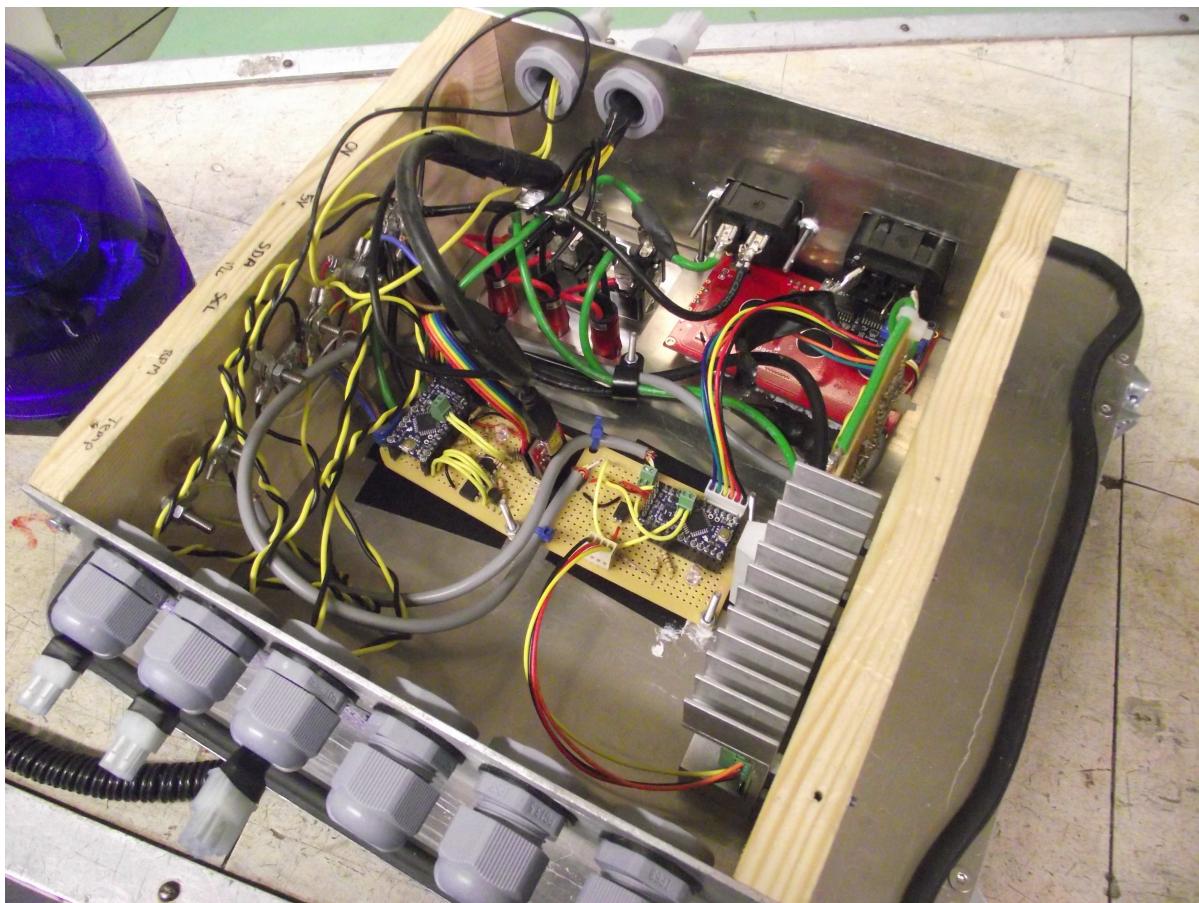


Figure 3.4.3: [Bottom-right] Initial design of the electronics enclosure. [Bottom-left] A picture of the final enclosure from above and below [top], with the base removed. In the top picture two modules are visible: the master module on the left and the motor module on the right - which is connected to the large PWM amplifier. The backs of the switch panel and LCD are also visible. The main power cables (green and black) were crimped to the switches and sockets as necessary and every attempt was made to route them neatly and securely. The thinner black and yellow cables were data and power connections to the external modules, which terminate on the side of the enclosure to large bolts. Various rainbow cables connected the modules to their peripherals such as the USB converter, amplifier and sensors. All external connections were sealed using silicon sealant or cable glands.

For the purposes of functionality, two extra car batteries were also purchased so that a healthy pair of batteries could be in the model at all times. As it turned out, it was more effective to simply place all four batteries in parallel, greatly improving the consistency of the supply voltage. The performance of the batteries will be examined during the test process.

The price of four car batteries, totalling 240 Ah, was £208. Little research was performed into the viability of lithium batteries since the same amount of power would have cost over £2500 using off-the-shelf units.

3.4.3 Module: Master

[HAWKES]

The master module was designed to communicate with the autonomy unit using a USB-serial converter. The communication is two-way - the autonomy unit sends commands to the master module, and the master module sends sensor results to the autonomy unit. Additionally, the master module acts as a relay - distributing commands to the appropriate modules and collating sensor results from the different modules via I2C.

The master module was designed to be the only critical module in the electronics system; such that any other module could be removed without sacrificing the integrity of the system (albeit with a loss of functionality). As such, all critical circuitry was placed on the master module - which included the DC-DC converter, system temperature sensor, battery monitoring sensors and I2C pull-up resistors (which will be introduced in this section).

Figure 3.4.4 shows a circuit diagram of the master module and a photo of the complete unit.

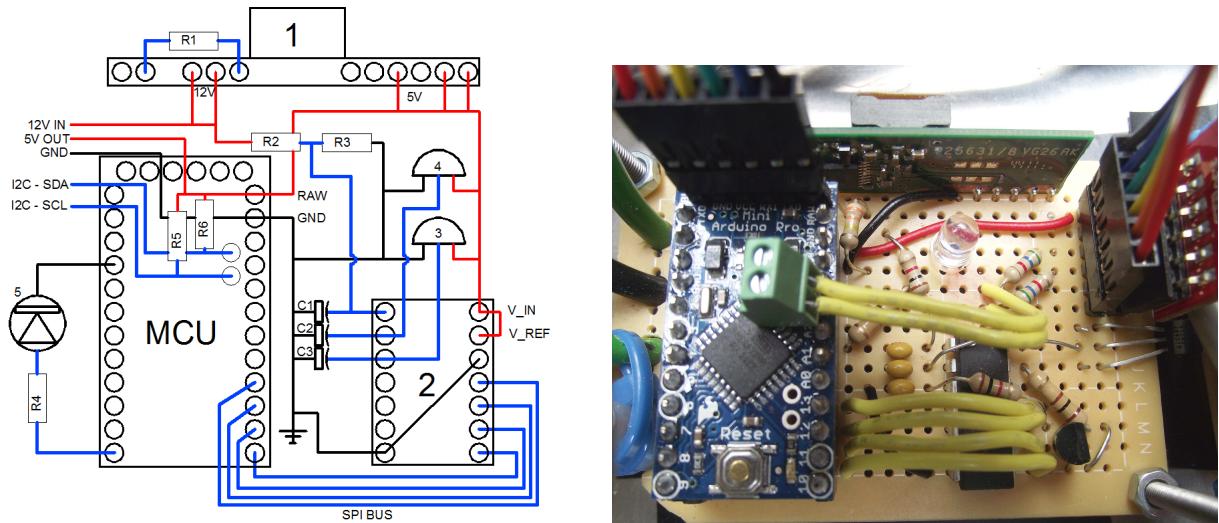


Figure 3.4.4: [Left] Circuit diagram of the master module, showing supply voltage in red, data lines in blue, and ground as black. [Right] A photo of the complete module, note that much of the circuitry is completed underneath the board, out of sight.

Item ① was a variable-output DC-DC converter¹ rated to 10 A in accordance with the 9.25 A requirement calculated in Section 3.4.2. R1 was chosen to trim the output voltage to 5 V using the manufacturers guidelines, the actual output was measured as 5.04 V (independent of battery voltage) due to resistor rounding error.

$$R1 = \left[\frac{10500}{V_O - 0.7525} - 1000 \right] = \left[\frac{10500}{5.0 - 0.7525} - 1000 \right] = 1.472k\Omega \approx 1K5 \quad (3.4.1)$$

¹12 V Austin Lynx SIP Non-Isolated Power Module; 10V - 14V input; 0.75V-5.5V output; 10 A output current.

Item ② was an analog-to-digital converter (ADC), specifically an MCP3204, and was used many times in this project. Although the chosen MCU has dedicated analog input pins, they are only 10-bit, giving 4.8 mV resolution. The MCP3204 provides 4 channels with 12-bit resolution (1.2 mV) and communicates using the Serial Peripheral Interface (SPI) bus which was connected to pins 10 through 13 on the MCU. The MCP3204 is a successive approximation ADC which utilizes a digital-to-analog converter and comparator to perform a binary search for the voltage of the measured value. The measured voltage should be low impedance ($< 1k\Omega$), since the MCP3204 does not have built-in buffer (voltage-following/repeating) op-amps (operational-amplifiers), and the sample rate should be less than 100 ksps (kilo-samples per second).

Item ③ was a 2.5 V voltage regulator (TL431) which provides a low-impedance voltage to channel 2 of the ADC. In 12-bit notation, the value of this measurement should be 2047 - if it is not, then the supply voltage can not be 5 V. In effect, this voltage regulator provides a means of calculating the voltage produced by the DC-DC converter, with a 1.2 mV resolution; and hence the reference voltage for other sensors on the master module.

Resistors R2 (560K) and R3 (330K) created a voltage divider acting on the 12 V (battery) supply voltage reducing the typical 12 V supply to a $12 \times \frac{330K}{560K+330K} = 4.4V$ signal, which can be used to measure the battery voltage. This divider allows some head-room in case the battery voltage is higher than 12 V, which can be the case with car batteries (fully charged = 12.7 V). This signal was *not* low impedance due to the high resistors and should ideally have been passed through a buffer op-amp. However, the battery voltage was not a critical measurement, so the impedance error was calibrated out for simplicity. Resistor error (which can be up to 5%) and impedance error added up to a 10.6% under-estimate of the battery voltage which was calibrated-out reliably, using an oscilloscope for the reliable measurement of the battery voltage. This voltage divider was connected to channel zero on the ADC, giving a 3.24 mV resolution of the battery voltage - which is a 0.027% resolution, based on the expected value (12 V).

Item ④ was a temperature sensor (TMP36) which provides a pre-calibrated output voltage proportional to the local temperature. This is attached to channel 2 of the ADC and provides information about the temperature inside the electronics enclosure. This temperature rose by several degrees over the course of a day, and was approximately 10 degrees above ambient temperature. It was worth monitoring the temperature because the motor amplifier and DC-DC converter are both inside the electronics enclosure and create considerable heat. At 15 degrees above ambient, in worst-possible conditions (motor: maximum continuous rating, servo: full-lock, bilge pump: on) and assuming 80% and 93% efficiency from the DC-DC converter and amplifier respectively the heat dissipation was estimated to be around 48 W.

$$[5V \times 9.25A \times (1 - 0.93)] + [12V \times 12.5A \times (1 - 0.80)] = 48.23W \quad (3.4.2)$$

The enclosure was mostly made from 2 mm aluminium and was sealed for safety (of the users and the system). It should dissipate 48.23 W of heat at 12.5° above ambient temperature which is considered safe, based on the surface area and specific heat conductance of aluminium. Some headroom should also be allowed here though, since the convective heat transfer from the electronics to the aluminium is probably the limiting factor in the heat transfer process, rather than the conductance of the aluminium.

R4 and an orange LED ⑤ created a status light controlled by the MCU. R4 was chosen as 100R to limit the current across the 3 V LED to 20 mA. This LED was set to flash every two seconds to indicate normal operation, henceforth called the ‘heartbeat’.

R5 and R6 were I2C ‘pull-up’ resistors. I2C requires careful balancing of the capacitance and resistance on the two communication wires (SCL - clock; SDA - data) by adding pull-up resistors connected to the

positive rail (5 V). According to the Arduino datasheet the capacitance of each Arduino's I2C bus is 10 pF, as is the I2C bus on the LCD screen. With these 6 devices the total capacitance was 60 pF. According to the I2C protocol [Philips Semiconductors, 2000] the maximum resistance R_p for the pullup resistors should be chosen so that it matches the maximum rise time, which is the time for the digital signal to switch from negative to positive. Operating at 100 KHz this was about 1 μ s ($\approx 10\% \times 1/100$ kHz, as a rough estimate). And using the identity $t = RC$ (time constant = resistance \times capacitance) the maximum resistance can be found:

$$R_{p_{max}} = \frac{1 \times 10^{-6}s}{60 \times 10^{-12}F} = 16.6k\Omega \quad (3.4.3)$$

The minimum pull-up resistance at this voltage is $\approx 1.53k\Omega$ according to the I2C protocol [Philips Semiconductors, 2000]. It was decided to use 5K6 resistors, since this gave some head-room for expansion but should also yield good range (higher resistance reduces noise susceptibility).

Capacitors C1, C2 and C3 were all 0.1 μ F decoupling capacitors. In general, capacitors can be used to filter AC (noise) but allow DC signals. In this case they served to cancel any noise created by fluctuations in the supply voltage by decoupling it to ground [Horowitz and Hill, 1989, p.42].

The LCD screen can also be considered part of the master module. The LCD consisted of a standard 4x20 character SPI-type screen connected to an I2C converter, and ran at the standard 5 V logic voltage. The master module sent signals to the screen via I2C, updating the display with relevant information that it obtained from the other modules.

The program uploaded to the MCU used a memory-intensive library (rosserial_arduino) to communicate with the operating system on the autonomy unit. Although quite straight-forward it relied on the ability to save a number of constant variables into the flash memory to save SRAM. The communications were performed at 10Hz in this project due to limitations of the autonomy computer, which could be remedied in the future. This module has many free clock cycles, and could easily increase the communication rate to 1 kHz or more (if the autonomy unit can handle it) - mostly limited by the rate of communications via serial, which was capped at 14.4 kB/s (115200 baud rate) by the hardware. This module should also be able to cope with expansion if more modules are added in the future.

In the future, it would be possible to add to this module in the form of additional sensors - although processor-intensive tasks should be kept off this module to keep the communications flowing smoothly. It would also be possible to add error checking across all the modules, so that the master can better monitor the health of the electronic system and report it to the autonomy unit.

3.4.4 Module: Motor Controller

[HAWKES]

The motor controller module's purpose is to interface with a PWM (Pulse Width Modulation) motor controller, measure the motor voltage and current (hence power) and measure the shaft speed (RPM).

The motor controller module is a slave module in the system, communicating to the master module using the I2C protocol. In contrast to the master module, the sensors implemented in this module are critical to the measurement of the model, so had to be designed and built much more carefully.

Figure 3.4.5 shows the circuit diagram and completed pictures of this rather complex module.

Item ⑦ is the PWM amplifier - used to amplify small logic signals into a large supply signal for the motor. PWM is a type of digital signal that consists of a voltage that is pulsed on and off very quickly (in this case: 32 KHz). The time for which the signal is on or off corresponds to the width of the pulses - hence *pulse width* modulation. A PWM signal that is half on and half off is said to have a duty cycle of 50%. The motor will simply 'average' this 50% duty cycle into an analog voltage of half the maximum

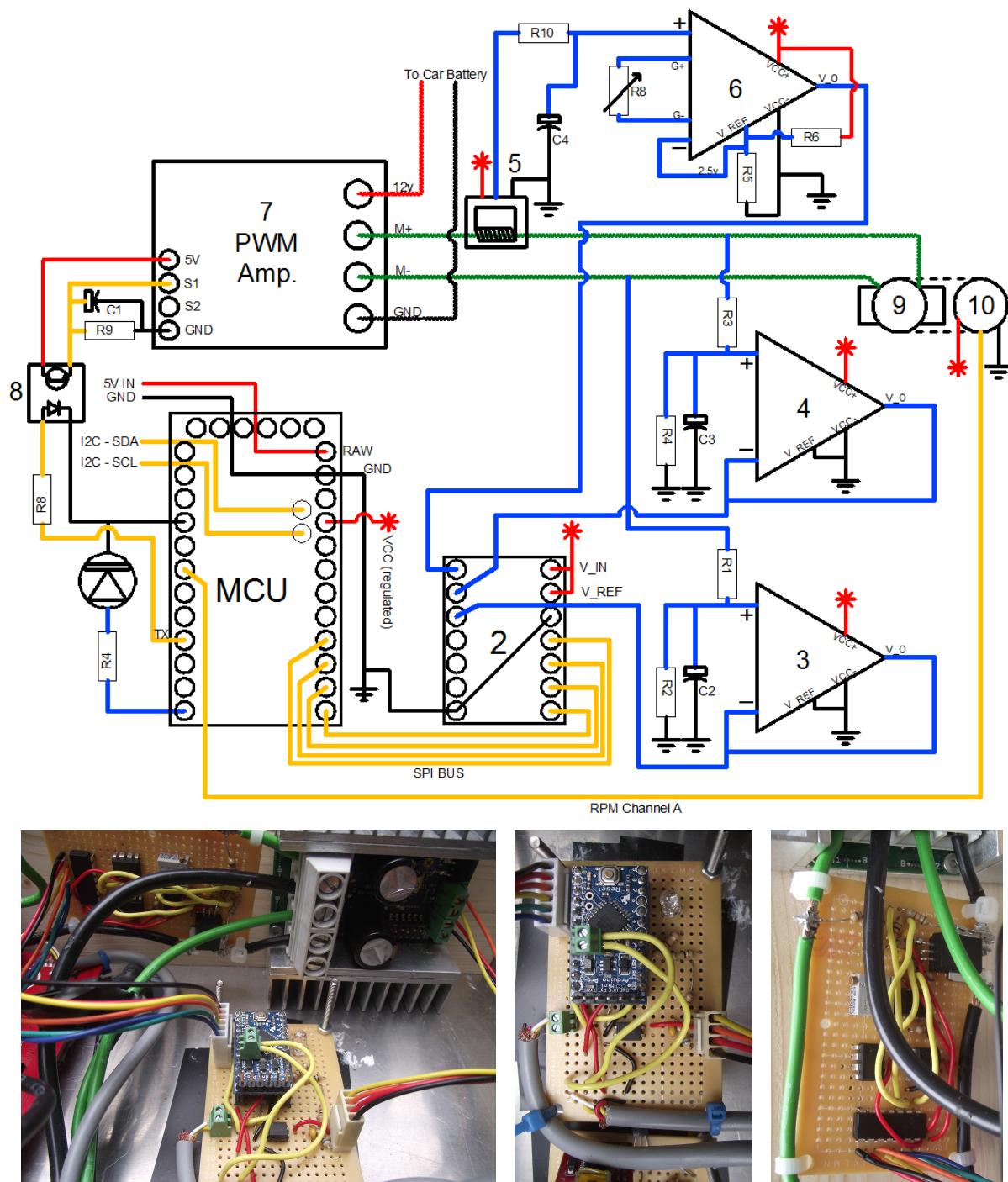


Figure 3.4.5: [Top] Circuit diagram of the motor controller module, showing 5 V in red, analog data lines in blue, digital data lines in orange, amplified motor supply in green, and ground in black. [Bottom] Photos of the complete module, which was split onto two perfboards - one containing the MCU and control circuitry and one containing the sensors and ADC. The PWM amplifier is clearly visible in the top right of the first picture. Much of the circuitry is completed underneath the boards, out of sight.

voltage.

The motor is rated as 12.5A-continuous, but has a stall current of 103 A. The stall current is the maximum possible current draw, which will occur if the shaft is stationary but the motor is at full power. In this project this can only occur when the motor starts. The PWM controller has to be able to handle this load for a few seconds at most. The chosen PWM amplifier was a Syren 50 A, which can handle a 100 A peak load. A typical car battery has a very high discharge rate (over 400 A) which would normally be used to start a car, so is likely to supply the full current to the motor during a stall condition. The amplifier contains a current limiter, to prevent damage to the amplifier or motor in case the shaft becomes locked or jammed.

Communication to the Syren amplifier is performed using one-way serial. Pin 3 on the MCU (another Arduino) was set as a serial transmission pin (TX). The motor controller produces its own small (500 mA) 5 V DC power source, which could cause problems if it is connected to the much larger power converter on the vessel: when the ‘control’ switch is on and the motor turned off, the master module would act as a power source for the logic circuits on the PWM amplifier, which would usually get its power straight from the batteries - this could damage the amplifier. Also, the small voltage inconsistencies could cause communications to fail - potentially damaging the motor and causing injury. To avoid this problem, an opto-coupler was used.

This device (item ⑧) isolates the two circuits, but couples the communication channel using an LED and optical switch. When the switch is open, S1 (the serial receiver on the Syren) is pulled to ground, but when closed is shorted to the 5 V supply. This type of circuit can often be found in transistor-type relays, as in Horowitz and Hill [1989, p.363, fig. M]. This allows serial communications to be transmitted from the MCU to the motor controller safely. R9 and C1 were chosen to give a small time constant suitable for serial communications, and R8 was chosen to match the LED requirements.

The motor (item ⑨) was fitted with a 1024 counts per revolution rotary encoder (item ⑩). Inside the rotary encoder is an optical device which worked by counting contrasting-coloured segments of a rotating disk, using an optical sensor.

The encoder, a Maxon MR Encoder (225787), had three output channels (Index, A and B). The index pulsed once per revolution; channels A and B pulsed with a phase difference at 1024 pulses per revolution, allowing the direction and speed of rotation to be found (Figure 3.4.6).



Figure 3.4.6: The rotary encoder [left] creates a digital signal [right] by detecting colour patterns on a rotating disk using multiple photo-optic devices.

Channels A and B produced 1024 rising edges (signal switching from low to high voltage) and 1024 falling edges (high to low) per revolution. If both channels were counted using both rising and falling edges this could total 4096 counts per revolution. However, after some experimentation it was determined that the MCU could only handle 1024 counts per revolution. The encoder connected to the motor controller module in the form of a 10-pin connector. Five volts and ground were passed to the encoder, and the

interrupt pin (2) of the MCU was connected to channel A, with rising edges triggering the interrupt routine on the MCU.

It is worth noting that the RPM is measured inside the motor, before the gearbox which had a 3.5:1 reduction ratio. The resultant counts per *shaft* revolution was 3584. Using a sensible time-out period in the code, it was possible to measure rotational speeds as low as 20 RPM.

The encoder was verified using a strobe light pulsing once per second when the shaft was rotating at 60 RPM; and twice per second when the shaft was rotating at 120 RPM. Unfortunately there were no available means to perform an external calibration at higher speeds, since any external optical device would have been far less accurate than the rotary encoder that was used. The only possibility in the future would be to attach the encoder to a commercial-grade decoder to verify the signal processing that the motor controller module performs.

Item ② was a 10-bit ADC (MCP3204), which was introduced in Section 3.4.3. Its role was to obtain better resolution for the channels it would be measuring.

Channels 1 and 2 of the ADC were connected to the outputs of the PWM amplifier, named M+ and M-. Depending on the demand value sent to the amplifier, these values vary between 0 and 12 V (PWM). At full ahead, M+ is ≈ 12 V and M- is ≈ 0 V. As the demand value reduces the voltage difference decreases, until full reverse when M+ is ≈ 0 V and M- is ≈ 12 V. The potential difference between M- and M+ creates the motor voltage, which had to be measured to obtain the power consumed by the motor.

The resistors R1, R2, R3 and R4 form voltage dividers to split the 12 V motor supply into a low-current 5 V signal. The resistors are the same as the battery voltage sensor in Section 3.4.3: 560K and 330K necessarily. C2 and C3 are $0.1\ \mu F$ decoupling capacitors that form an R-C (resistor-capacitor) filter acting on the signal where R is the parallel sum of the two resistors ($R = 1/(\frac{1}{330K} + \frac{1}{560K}) \approx 200K$). This filter is a low-pass filter, which attenuates signals of a period shorter than the R-C time constant $t = RC \approx 200K \times 0.1\mu F = 20ms$, but allows *low* frequencies to *pass*. This effectively removes any noise carried on the signals with a frequency higher than ≈ 50 Hz. This type of filter essentially acts as a short time-averager, and would not be suitable for fast-fluctuating signals. The filter removes noise such as the PWM frequency, which is far in excess of 50 Hz (32 KHz) but also leaves fluctuations below the sample frequency intact [Horowitz and Hill, 1989, p.263].

To remove the problem of high impedance to the ADC, the signals are passed through a buffer operational op-amp - items ③ and ④. This effectively repeats the high-impedance signals to low-impedance signals which are fed to the ADC. This was done to retain high accuracy on these measurements, which are critical to the assessment of vessel power. The battery voltage sensor does not use this arrangement, since it was not so critical.

Calibration was performed using an external voltmeter. Since the resistors were not error-free corrections were necessary, but the error was linear, small and easy to remove by finding the response gradient and zero-offset.

Channel zero of the ADC is connected to a hall-effect current sensor (item ⑤), which measures the current through the motor circuit (shown in green) while introducing negligible voltage drop². The voltage supplied to a DC motor will be directly related to the rotational speed, whereas the current will be proportional to the torque. The manufacturer supplied a torque constant of 16 mNm/A, which provided a means of converting electrical input to the motor to shaft torque. This torque value was used as a backup for the torque dynamometer.

²An alternative method would have been to use a small but high-power (*i.e.* 0.1R, 25W) resistor, across which a very small voltage drop could be measured (then $I = V/R$). In early experiments this proved unreliable because the resistor was not capable of maintaining its rated power for more than a few seconds, resulting in a dramatic failure of the resistor.

The hall-effect sensor uses the electromagnetic signature of the motor circuit to measure the current up to ± 100 A. The sensor creates a 2.5 V signal at zero current, increasing to 5 V at 100 A and decreasing to 0 V at -100 A. An instrumentation amplifier (specifically an INA122PA, item ⑥), which is a very high-quality type of op-amp designed for sensor amplification, is used to amplify the signal to improve the resolution, across a smaller range of currents (*i.e.* ± 5 A). The instrumentation amplifier features a very high common-mode rejection ratio (CMRR) which means it is good at detecting small differential voltages carried on a higher common voltage, whilst maintaining low noise and drift - in this case: a small signal carried on a 2.5 V common voltage. The op-amp also features built-in buffer op-amps on the two input signals to remove impedance error. The op-amp, item ⑥ was set up with a 2.5 V negative signal input and reference voltage (via a 1K-1K potential divider) and a variable gain resistor.

Another R-C filter was used, this time with $R_{10}=100\text{K}$ and $C_4=0.1 \mu\text{F}$ which created a 100 Hz low-pass filter to reduce noise.

The current sensor was pre-calibrated by the manufacturer so that the response was guaranteed to within 0.3% (constant error). However, errors are introduced by the amplification, due to inaccurate resistors and inexact gain settings, so calibration had to be performed.

The motor did not create a significant current draw when running dry, and it was impractical to calibrate the current in the towing tank, so calibration was performed using two car sidelights. These ran at 12 V and drew approximately 2 A each. These lights were placed individually and in parallel, as a dummy load (replacing the motor), at various voltages up to 12 V. The external measuring device was a hand-held hall-effect ammeter. The gain resistor (R8) was tuned to $11.19 \text{ k}\Omega$ which gives an appropriate gain, calculated using the op-amp manufacturer's guidelines:

$$G = 5 + \frac{200\text{k}\Omega}{R_G} = 5 + \frac{200\text{k}\Omega}{11.19\text{k}\Omega} = 22.873 \quad (3.4.4)$$

Upon calibration this gain was found to be very accurate. The sensor gave results correct to the fourth significant figure when compared to an ammeter, with a very small zero-offset which was removed through calibration.

This gain allowed a range of current of ± 4.37 A, which proved to be suitable for the model, which drew about 3.5 A at maximum load. With 12-bit resolution this gave a resolution of 0.2% which would be suitable for detecting small changes in current. The gain should be reduced for future testing, if the vessel is pushed beyond its design speed, since the motor can reach up to 12.5 A continuous current.

In figure 3.4.5 all the positive rails marked with a red asterisk are connected to the regulated 5 V voltage of the MCU: including the sensors, op-amps, ADC and reference voltages. This meant that even large fluctuations in this absolute voltage did not effect the results at all, since everything was proportional to it, including zero offset error and scalar error. This was a luxury not often afforded to typical carriage-based systems, which must attempt to compensate for differences in reference voltage across multiple devices.

A status LED, similar to the master module's status LED, was also implemented. The master module emits an I2C heartbeat signal every two seconds, such that the LED would flash if the module was working correctly and communicating.

The motor module was programmed so that the motor speed could be controlled via RPM, voltage, current, power or pulse-width duty cycle. A simple feedback loop was implemented to perform this targeting, which also provided a ramp function for the motor so that it could never jolt (*i.e.* switch speed suddenly), which would have caused damage to the motor and power train.

The RPM counter works by timing 30 pulses of the encoder. To reduce the error in the timing

function, all other processes were suspended when the Arduino was counting. To count 30 pulses at 20 RPM took 25 ms, and this was programmed as a ‘timeout’ so that the function would not block other functions for more than 25 ms (*i.e.* when the shaft is stopped). At a medium RPM of 500 the function took 1 ms to count 30 pulses. For higher sample rates on other sensors it would have been necessary to reduce the ‘timeout’, to prevent the RPM counter from locking the processor for too long, which would have reduced the minimum measurable RPM.

Realistically, the maximum sample rate required would be approximately 200 Hz (5 ms period) and the minimum measurable RPM at this sample rate would have been about 100. This could be improved by moving the RPM sensor to a separate module, where it could block the processor for as long as required.

Upon orders from the autonomy unit, the master module sends two values to the motor module. An integer message called ‘motorTargetMethod’ which corresponds to the type of motor setting (*i.e.* via RPM, voltage, etc.) and a floating-point message called ‘motorTarget’ which is the actual target to which the feedback loop works towards.

When the master module sends a request for data, the motor module responds with six values which are the calibrated sensor values for motor current, motor voltage, motor power, shaft RPM and PWM duty cycle as well as a repeat of the target motor value for verification.

3.4.5 Module: Thrust Block

[HAWKES]

The thrust block module was the first module that was built external to the electronics enclosure. It was built into a small enclosure of its own, which is watertight and just large enough to accommodate the electronics. As explained earlier, the module connects to the main electronics enclosure via 4-core speaker cable and an ATX-style 4-pin plug.

The purpose of the thrust block module is to process the signal from the strain gauge and communicate the results to the master module, and hence the autonomy unit, which logs the data.

In Figure 3.4.7 there are a number of items that will be discussed, which make up the thrust block module.

The strain gauge (item ①) is a cantilever beam load-cell that was introduced in Section 3.2.3. According to the manufacturer the excitation voltage should be 10 V, but it was also possible to run the load cell using the 5 V system already in place. A higher excitation voltage means a larger signal which is less susceptible to noise, but a lower voltage means less heat is generated within the strain gauge - and heat distortions, which affect the zero-point and response gradient of the strain gauge, are reduced. The strain gauge will always be susceptible to environmental changes in temperature, and this model of load cell does not contain temperature-correction resistor to quantify it.

The strain gauge inside the load-cell consists of a Wheatstone bridge, as shown in the circuit diagram. The resistance of the four resistors changes with strain in the load cell - altering the two signal voltages (S1 & S2). In the no-load condition the two signals should be equal to half the supply voltage (*i.e* 2.5V).

The potential difference between the two signals is linearly proportional to the strain, to within 0.02% (according to the manufacturer), but the potential difference is typically very small. Initial testing indicated that a load of $\pm 25\text{N}$ resulted in a $\pm 20\text{ mV}$ potential difference (0.8 mV/N), so this had to be amplified in order to be useful.

Item ③ is another instrumentation amplifier (INA122PA), which is used as a differential amplifier by connecting the positive and negative signal inputs to S1 and S2. The gain was controlled using R4, a 200K variable resistor (trimpot), using equation 3.4.4. A reference voltage of 2.5 V was supplied via

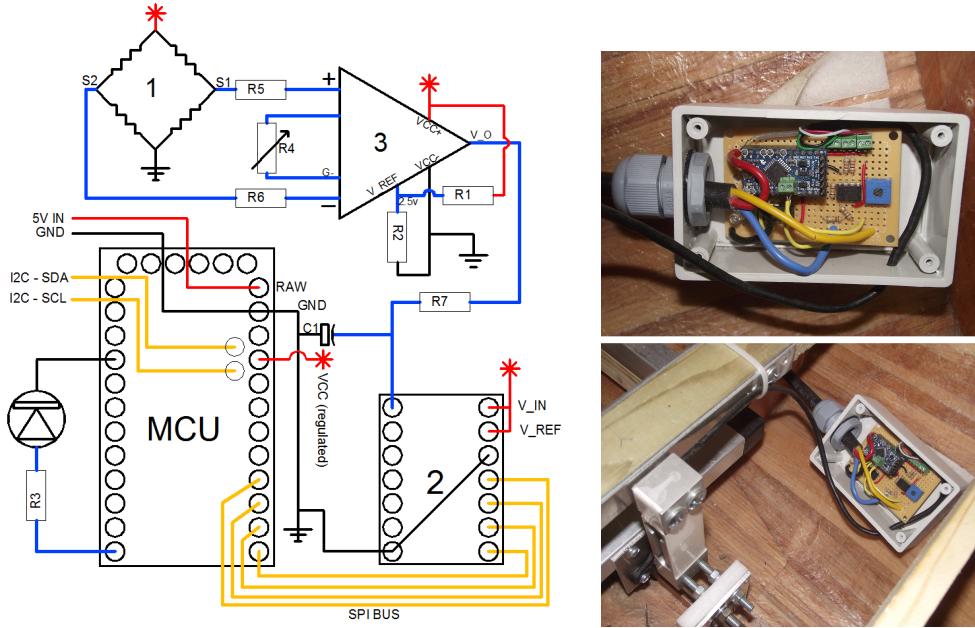


Figure 3.4.7: [Left] Circuit diagram of the thrust block module, showing 5V in red, analog data lines in blue, digital data lines in orange, and ground in black. [Right] Photos of the complete module, showing the electronics and the connection to the strain gauge. The enclosure usually has a watertight lid fitted.

a 1K-1K potential divider. This meant, similarly to the current sensor, that the differential voltage is amplified around a 2.5 V reference.

This voltage was then fed to the ADC (item ②) which was another MCP3204, and recorded by the MCU. An R-C filter was used between the op-amp output and the ADC to provide some noise filtering, this created a 100 Hz low-pass filter. This was particularly useful for this sensor, because the strain gauge readings were noisy.

Expecting a thrust of approximately 10 N a suitable gain would be approximately 100, which would give a response of 80 mV/N. At 10 N, using the 12-bit ADC this resulted in a 0.15% resolution which was more than adequate for detecting the thrust changes expected.

One significant problem was encountered with the amplification. Because the signal was small a large amplification was required - significantly larger than for other sensors in the electronics system. Also, the strain gauge was very susceptible to zero offset error, whereby S1 and S2 were not equal in the no-load condition. When amplified, this error became very large and caused the zero point (where thrust was zero) to create a signal voltage that was not 2.5 V. In fact, as the gain increased (by changing the variable resistor) the zero point could be pushed outside the measurable range, where it would be truncated - as shown in Figure 3.4.8

The gain was tuned roughly to provide suitable gain across the measurable range, but a small mistake in the initial testing shifted the zero-point out of the measurable range. This was because the power train was not truly in its neutral position when the gain was set. This caused some problems for the tank testing since the zero-point could not be tracked throughout the day, but the gain was reduced before later testing. Ideally, R2 would be replaced with a variable resistor which could shift the zero-point.

The thrust block module had to be calibrated because the relationship between thrust and strain was unknown. This was done by means of bollard pulls against a calibrated tow post. This allowed the zero offset and gradient of the response to be calculated which allowed the digital signal to be related to actual

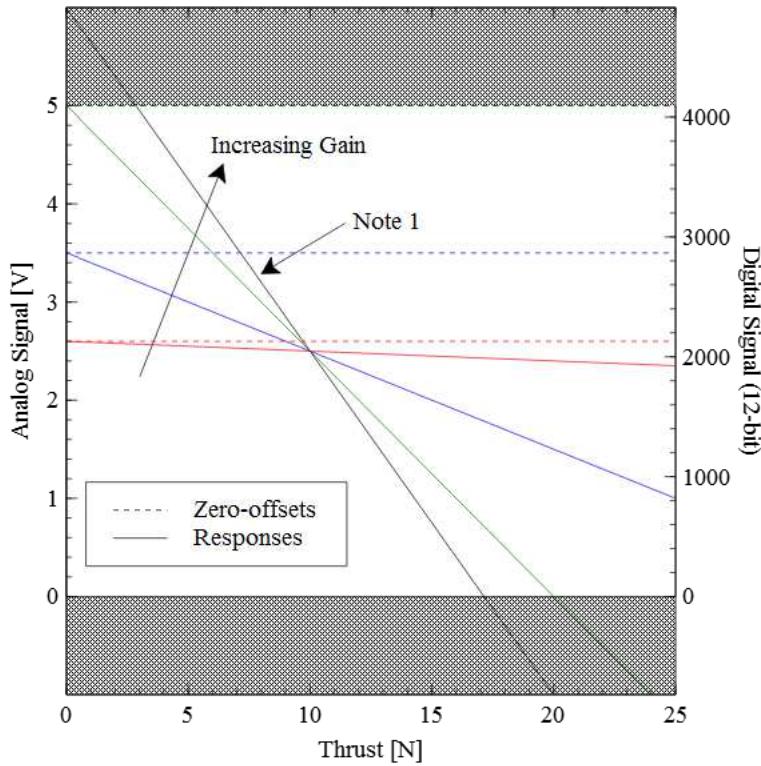


Figure 3.4.8: Amplified signal responses from the load cell.

thrust.

Once again, all the reference voltages in the thrust block module were supplied from the same voltage which greatly reduced the potential for error. However, the strain gauge was susceptible to other sources of error, such as drift error - which occurs mostly due to temperature changes. This will be investigated in later sections.

The algorithm for the thrust block was simple. The ADC measurements were read using the SPI bus and sent to the master module. Really, an entire MCU was not required for such a trivial task but since they were cheap it made sense to keep the modular pattern - and it allowed the thrust block module to be placed near the load cell, reducing the length of the analog signal cables (therefore reducing noise).

3.4.6 Module: Bilge Pump

[HAWKES]

The bilge pump module was rather simple. It consisted of an Arduino, a water detection switch and a 6 V water pump that would be run from the 5 V supply. The switch triggered a connection from five volts to an analog input on the MCU, when water to be detected.

The bilge pump itself was powered via a 10 A DC relay - allowing the high-power pump to be turned on and off from a low-power signal.

A status LED for the heartbeat was also fitted, as with all modules. This module was never permanently fitted to the model, since it was not clear where the best location would be until after testing. The bilge pump was used as a roving module, and could be installed permanently in the future.

The bilge pump reported to the master module (and autonomy unit) when water was detected and when the bilge pump was turned on. The pump can only be run for 30 seconds every 5 minutes, so these two conditions were not always synchronous.

The complete module is shown in Figure 3.4.9.

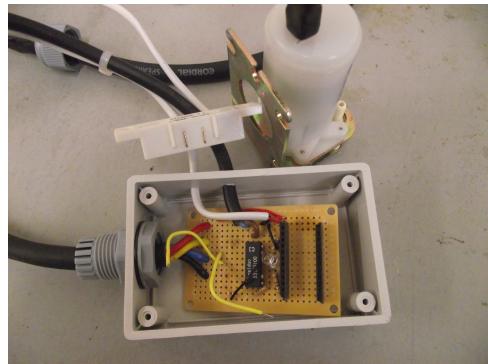


Figure 3.4.9: The completed bilge pump module, shown here without the MCU. Clearly visible are the white water detection switch, the bilge pump and the DIP relay. All the MCUs were socketed for easy removal, so they could be upgraded or replaced easily, and programmed ex-situ.

3.4.7 Module: Rudder Controller

[HAWKES]

The rudder module was another slave module, external to the main enclosure and was the final module to be built and tested - since it was not required for towing tank testing.

Its role was to control the rudder using a servo and to verify the position of the rudder using a rotary potentiometer, since the servo's built-in position sensor was not accurate enough.

The circuit diagram and completed unit can be seen in Figure 3.4.10 and each element of the module will be explained in this section.

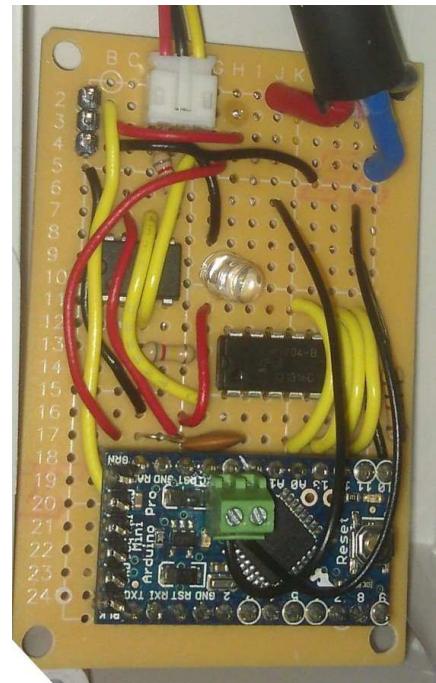
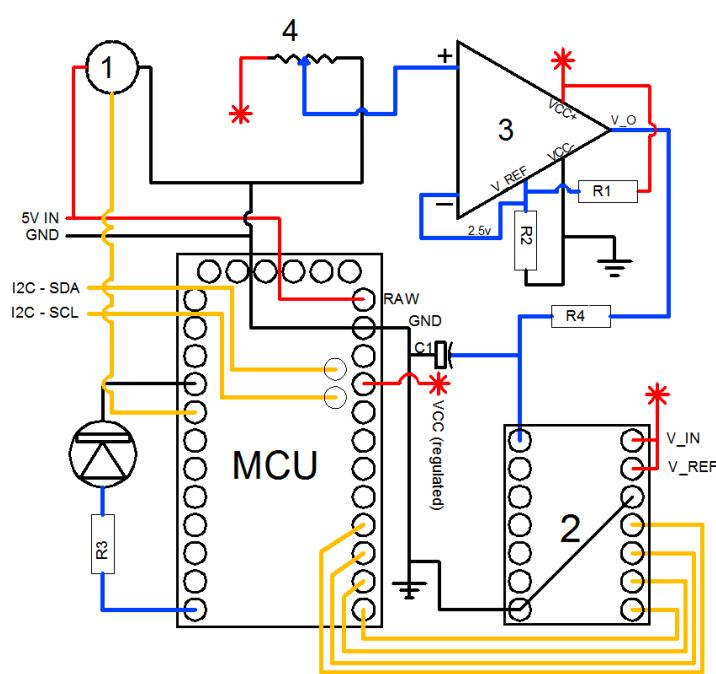


Figure 3.4.10: [Left] Circuit diagram of the rudder controller module. [Right] Photo of the completed rudder controller module. Two different 3-pin connectors can be seen on the top-left of the perfboard, where the servo and rotary potentiometer plug in.

The servo, item ① is a waterproof PWM-controlled servo (Hitec HS-5646WP) which takes a high-power 5 V supply and a low-power PWM signal to set the position. This will be introduced in more detail in Section 3.5.

The rotary potentiometer (item ④) was a simple device that produced an analog signal between ≈ 0.5 V and ≈ 4.5 V depending on the rotary position - between approximately $\pm 150^\circ$. Another option could have been to use a rotary encoder, but these are better suited to speed-measurements (*i.e.* shaft speed) than position sensing.

Since the rudder actually only turns through a small amount of this 300° range a five-times gain was applied, about a reference voltage of 2.5 V which corresponds to a centred rudder (0°). An instrumentation amplifier (again, the INA122PA) was used (item ③) where the gain connectors were disconnected. According to Equation 3.4.4 this gives a gain of 5 since $R_G = \infty$.

Similarly to the other modules, a 12-bit ADC (MCP3204 again) was used to gain further resolution of the signal, and a 100 Hz low-pass filter was applied using an R-C filter (R4 and C1).

Calibration was performed by forcing the rudder to its central position and its most extreme positions and recording the digital signal and physical angles. There was some zero-offset error due to a misaligned rotary potentiometer, but this was calibrated out.

Of course, a heartbeat LED was also implemented.

The rudder module functions like the other modules in terms of I2C communications. On receiving a target rudder angle from the autonomy unit (via the master module) the MCU performs a feedback loop which smoothly corrects the servo position until the reported angle from the rotary potentiometer is correct.

To prevent overshoot, the correction applied to the servo signal is decreased depending on the difference between the target angle and actual angle. This meant the rudder slowed down as it approached the target value, which was actually quite realistic, when compared to the real ship.

The update speed could be adjusted in the code so that the rotational speed of the rudder could be altered to match the real ship as detailed in Section 2.3.2.

A detailed breakdown of the proportional feedback controller will be presented in Section 3.5.

The absolute rudder angle and a verification of the target rudder angle was sent to the master module on request, and relayed to the autonomy unit for logging and real-time processing.

3.4.8 Sampling

[HAWKES]

The autonomy unit was only requesting data from the master module at a rate of 10 Hz (which in turn requested sensor updates from the slave modules at 10 Hz), which was not ideal. In the future it is hoped to increase this to at least 100 Hz since the autonomy unit should be capable of this.

Sampling all the sensors (on various modules) at 10 Hz would have been inefficient since the MCUs ran at 16 Mhz and could sample at a much higher rate. To utilize the full power of the MCUs and to produce better results, the sample rate of the MCUs was increased and a software-based low-pass filter was applied by creating a moving average of the data.

For the thrust block, which was the noisiest sensor in the system and the most under-worked MCU, the sample rate was increased to 500 Hz with a 50-point moving average; and similar rates were also applied to other sensors. Due to time constraints and not wishing to jeopardize valuable testing, the MCUs were not pushed to their limits. It is expected that with some careful coding this sample rate could be increased to around 10 KHz.

The other modules also had their sample rates increased to between 200 and 500 Hz, depending on

the MCU load and the amount of noise created by various sensors.

3.4.9 Summary

[HAWKES]

The electronics has been discussed in detail, showing an overview of the electronics system and how it cooperated; as well as the detailed design of each module.

All told, the bill of materials for the electronics contained over 500 items, of which about 150 were unique. The complexity was necessary in order to match commercial-grade signal processing devices, and early calibrations suggested that it had succeeded. Indeed, one of the biggest problems when calibrating the electronics was in finding external sensors that were deemed more accurate than the created system itself - most notably in the case of the RPM sensor.

3.5 Rudder System

[LE]

Taking into consideration the operating conditions, a waterproof servo was required. Therefore, the Hitec HS-5646WP Waterproof Servo servo was purchased. Its speed at 6.0 V is $300^\circ/s$ which was much quicker than required ($31^\circ/s$), but the operating speed could be tuned in software to match the rudder scaling. Also, it could provide 11.3 kg·cm torque, which is significantly higher than the estimated required torque, allowing for a safety margin.

A 4.7 k Ω rotary potentiometer was used to measure the rudder angle to provide feedback to the controller.

A simplified rudder form was produced to minimise construction time. The first rudder was built with its dimensions scaled geometrically from the real rudder; the second rudder was built with increased chord to compensate for the loss of lift coefficient due to the difference in Reynold's number (*i.e.* due to the nature of the flow around the rudder). The profile of the rudder was a NACA 0020.

The preliminary calculations section showed that the semi-balanced rudder and spade rudder only gave a small difference in performance. It was clear that a semi-balanced rudder has an advantage in terms of sustaining higher angle of attack before stall. However, this project only requires the rudder to turn 20° to either side so it was decided that a spade rudder would be sufficiently accurate to model the system. The two options for 3D printing are shown below and it should be noted that it would be very difficult to print a semi-balance rudder with the first simple method.

1. The first method is called *bits from bytes*, which essentially uses a nozzle to produce layers of hot material which hardens on cooling. This method is cheap (*i.e.* only about £5 for an object of this size) and the machine can be controlled by students so it does not have any lead time for the production. However, it can only print a simple and hollow structure because this method subjects the material to high thermal stress. This meant that the top layer of the rudder was curved because of the thermal stress during cooling. It is possible to reduce these stresses by using a thermal heated plate to slow the rate of cooling. The construction of the larger rudder used a thermal plate, which resulted in a better quality finish.
2. The second method is to use an *envisionTEC 3-dimensional printer ULTRA*. The material is envisionTEC's LS600 which is one of the extremely durable photopolymers for use in producing very accurate parts with high feature detail. It is an ABS-like material, which could sustain high temperature and has high impact resistance like thermoplastics, making it very suitable for this application. This method allows users to print a solid rudder with no limitation of complexity hence it is very easy for the design step. However, it is a costly and time-consuming process requiring technical personnel to operate. The rudder costs £60 and the process from submitting a drawing to the time it was printed took more than 2 weeks.

The rudders were initially printed using the first *bits from bytes* method and the second method were only used once the design had been proved to work. The rudder design can be seen in Figure 3.5.1, designed to a NACA 0020 profile. The nut in the bottom of the rudder is used to connect the rudder with the rudder stock. This system allows the ship to change its rudder quickly and have potential to use different rudders in the future to test their effectiveness. The two products will be shown and compared in Figures 3.5.2 and 3.5.3.

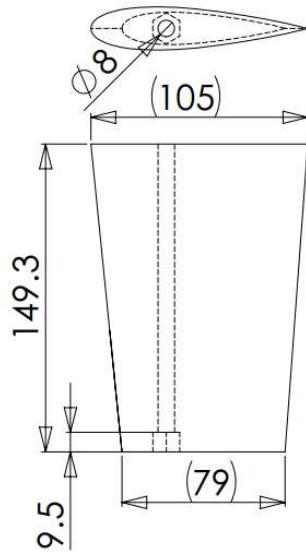


Figure 3.5.1: Design of the spade rudder for the model.



Figure 3.5.2: Profile of the big rudder printed by the first *bits from bytes* method, different layers can be observed.

3.5.1 Control method

[LE]

In the highest level, the rudder control unit receives a reference angle from the centre unit (ROS system) and implements that angle on the rudder stock, the real-time rudder angle will then be measured by potentiometer and returned to the central unit (as can be seen in Figure 3.5.4)

In a more detailed diagram as in Figure 3.5.5, a loop is used to ensure the real rudder angle is the same as reference angle. The algorithm is as following:

1. The rudder control unit takes the reference angle from the central unit.
2. It implements the angle on the rudder stock using the servo.
3. The real-time rudder angle is measured by the potentiometer that is coaxial with the rudder stock. The rudder control unit then compares the rudder angle and the reference angle.
 - If the rudder angle is smaller than the reference angle then increase the set angle.



Figure 3.5.3: The left rudder was geometrically scaled from the full-scale rudder and printed using second method (ULTRA), the right rudder was 1.4 times bigger than the first one taking into consideration loss of lift coefficient, manufactured using the first *bits from bytes* method.

- If the rudder angle is larger than the set angle then decrease the set angle.

The step of increase or decrease is adjusted so that the overshoot is negligible.

4. When the rudder angle is identical to the reference angle (the difference is smaller than acceptable error) then the loop is stopped and the task is completed.

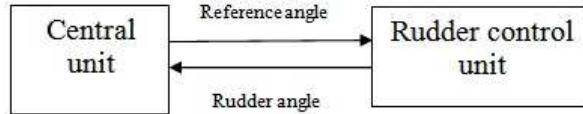


Figure 3.5.4: Data transfer between central unit and rudder control unit.

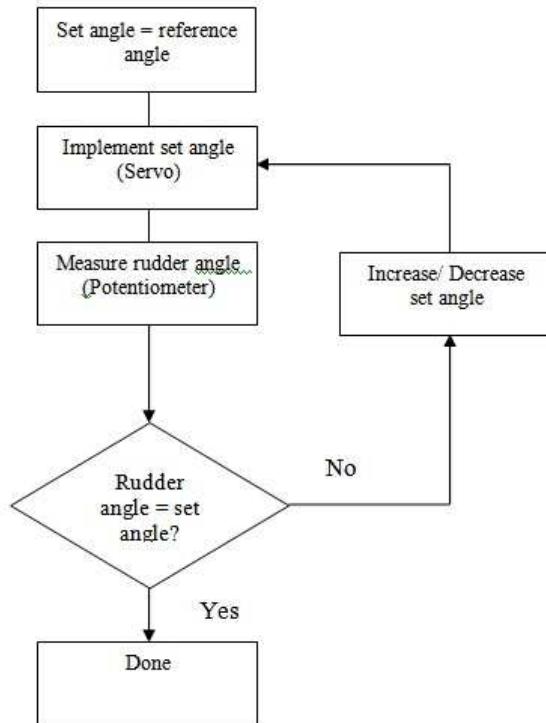


Figure 3.5.5: Rudder system's diagram.



Figure 3.5.6: Picture of the model's rudder system in operation.

3.6 Autonomy System

[ANDERLINI]

3.6.1 Introduction

The main processing unit of the model was a Beagleboard XM [Beagleboard.org, 2011], shown in Figure 3.6.1, although it was used only at the very end of the project. For initial testing and code de-bugging, a Dell Latitude E4200 [Dell, 2013] (provided free of charge) proved to be a much better platform due to its much greater processor speed, RAM and hard-disk capacity; and the presence of the built-in screen and battery.

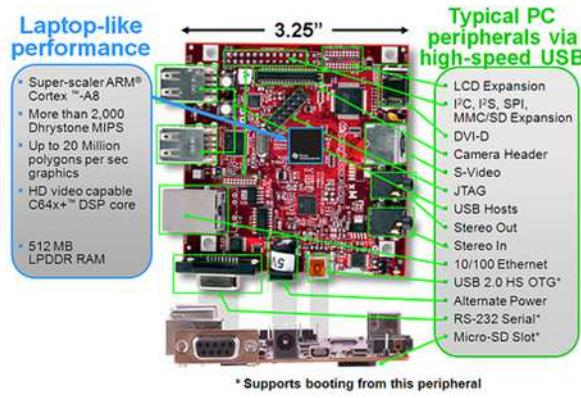


Figure 3.6.1: Beagleboard XM. This picture has been taken from Beagleboard.org [2011].

The operating system adopted on both computers was Ubuntu 12.04 [Canonical, 2013.], since it would provide an easier platform for the subsequent installations, whilst still being free. The operating and control system for the autonomous surface vessel (ASV) has been written using the Robot Operating System (hereafter referred to as ROS) [ROS.org, 2013], which is an open-source assembly of python and C++ libraries specifically designed to help developing robot applications. ROS provides hardware abstraction, device drivers, libraries, visualizers, message-passing and package management, as well as tools and libraries for obtaining, building, writing, and running code across multiple computers [ROS.org, 2013]. It has previously been adopted by researchers at the University of Southampton for the creation of the control system for the AUV Delphin2 [Phillips et al., 2010] due to its BSD (Berkeley Software Distribution), open-source and re-distributable licence.

Some preliminary code had been prepared by Dr A. B. Phillips (University of Southampton) and full access was granted to the Delphin2 code; this was extensively modified, enlarged and adapted for use in the ASV model but still retains the philosophy behind the development of the operating system. The main reasons for the changes are the different purposes and operations of surface ships when compared to the Delphin2. The AUV was designed for sensing data mainly in the vertical plane, has 9 degrees of freedom, relies heavily on side thrusters for manoeuvring and mostly navigates by dead-reckoning, employing the GPS signal only on the surface. The ASV operates purely on the surface, is subjected to aerodynamic forces due to the wind, employs solely its rudder for manoeuvring and is able to exploit the GPS signal extensively for navigation. In addition, changes in the hardware required significant modifications to the original system adopted for Delphin2, especially considering the interactions between the main computer and the master Arduino, which were completely different. Furthermore, some adjustments have been necessary because of the update of Python to version 2.7 from 2.5 due to the installation of Ubuntu 12.04

rather than 10.04.

3.6.2 Global Overview

[ANDERLINI]

Before proceeding to the actual description of the code, some of the most common terms associated with the selected operating system will be explained.

A “package” is the folder and all executable programs, datasets and configuration files within it that are necessary for the operation of a robot [ROS.org, 2013]. In this case, the package is the complete system ASV. The goal of packages is to create minimal collections of code for ready re-use [ROS.org, 2013]. They are created automatically by specifying other packages on which they are dependent (known as “dependencies”) and can be modified by user-input afterwards.

The executable codes within a package are known as “nodes”, which may be written in either Python or C++ [ROS.org, 2013]. Multiple nodes can be launched at the same time, by writing a suitable “launch” file in XML format that contains their names and details.

One of the main advantages of ROS is that it provides a very simple way to exchange information, or “messages”, between nodes, even when running on different machines. Hence, for instance, if the node reading the output from the compass sensor “publishes” the actual heading of the model as a “topic”, this can be read, or “subscribed to”, by the node operating as heading controller, which then in turn publishes a rudder angle demand to the node running on the Arduino which subscribes to this topic in order to keep the desired course by changing the rudder angle. All this exchange of information occurs in real time, with a minimal time delay (which depends on the hardware). A message can contain multiple topics.

ROS is designed to allow re-use, development and re-distribution of software between developers. Hence, it is possible install packages created by other users, which may be distributed under different licences (typically BSD licence) [ROS.org, 2013]. Such packages are collected in “stacks” according to their functionality (e.g. a navigation stack or a manipulation stack) and “repositories” (groups of stacks) and can be found on ROS.org [2013]. Each stack has an associated version and can declare dependencies on other stacks to provide greater stability in development [ROS.org, 2013]. Hence, their goal is to simplify the process of code sharing.

In order to use third-party packages, it is necessary to specify these as “dependencies”, i.e. stacks the newly created package is dependent on, in the manifest file (created automatically) of the current package (in this case ASV) [ROS.org, 2013]. This permits the use of functionalities and tools contained within these packages [ROS.org, 2013]. In this project, the ASV package has the dependencies specified as:

- `std_msgs` - enables the use of messages for the exchange of topics and strings
- `rospy` - imports the ROS Python library
- `roscpp` - imports the ROS C++ library to be used on the Arduino
- `roslib` - lists basic dependencies and support libraries
- `smach`, `smach_viewer` and `smach_ros` - allows the creation of complicated robotic tasks
- `rosserial_arduino` and `rosserial_python` - communicates with the master Arduino [ROS.org, 2013].

The software developed in ROS for the ASV has a hierarchical structure. On the bottom level, there are the nodes that interact with the USB sensors and the master Arduino. Their main objectives are to open the respective serial port (e.g. GPS, compass or Arduino), read the serial data, separate the single variables (e.g in the case of the compass heading, pitch, roll etc.), transform them into integer or floating

values and then publish them as a topic to the other nodes. In this project, the serial port names were fixed for each sensor (including the Arduino) in the `/udev` local rules [Drake, 2006]. This ensured that the sensor kept the same label, no matter which computer port was connected or which time order.

On a higher level, there is the node acting as the heading controller; it receives the current heading from the compass node and the heading demand from the higher level nodes and calculates the required rudder angle to be passed to the Arduino in order to achieve the desired bearing.

The level above the heading controller are the nodes dedicated exclusively for performing tests (such as the straight-run, self-propulsion tests or the circle and zigzag manoeuvres) and more complicated robot missions, such as moving between specified points. These latter types of nodes, which include some of the manoeuvres, are known as “smach states” from the homonymous ROS stack (collection of packages) [ROS.org, 2013]. Their main advantage is that they can be assembled in another, higher-level node, the mission script, to achieve complicated tasks.

The highest level node is the back-seat-driver, which continuously checks the system performance. It reads the hardware-communications nodes status (e.g. whether the compass is actually on or not), the presence of errors in the system, the pitch, roll and temperature within the electronics box and Pelicase and compares them against limit values (e.g. a maximum temperature). If any of these limits are exceeded, the back-seat driver will stop the system. This node must be run from the launch file, as the limit values are to be written here.

Finally, the system contains other three fundamental nodes: the high-level library, the logger and a node to compute the mean of the desired variables (e.g. motor voltage and current, propeller revolutions etc.). The software has also been designed to allow for the use of a bilge pump.

There are a number of dummy nodes within the ASV package which were created for debugging other codes within the package. They are very simple programs, whose purpose is to publish specified topics (e.g. the compass output for the dummy compass node). Their values may be fixed (as per the dummy heading controller node) or time varying, as in the case of the dummy GPS node (every second the X-variable changes by 1 m).

All nodes were provided with a “sleep” statement to tell the program to wait a specified time interval before performing the commands. A time interval of 0.01 seconds was specified in all nodes (0.5 s for the heading controller), in order to prevent them using 100% CPU. Nevertheless, the Arduino software was provided with time averaging (see Section 3.4.8), as was the navigation smach state. This was not deemed necessary in the other codes, but could be easily incorporated.

The folder ASV was created as a ROS package, and can be found in the CD attached to this document. Figure 3.6.2 shows the flow of most topics between nodes when a simple mission is performed (moving to a predefined point). In the following subsections the key points for each node class are described, with the exception of the dummy ones.

3.6.3 Compass

[ANDERLINI]

The magnetometer used was an OceanServer OS-5000US tilt-compensated, 3-axis digital compass [OceanServer, 2010], which can be seen in the bottom right corner of figure 3.6.3. It used a very similar code to the one developed for Delphin2 used to create the node that opens the compass serial port, reads the value and publishes the compass output (heading, pitch, roll, magnetic field vectors, and acceleration vectors) as a single topic (`compass.py`). Simple as it may sound, this process actually requires deep knowledge of the Python language. Some modifications were required for reading the serial data due to compatibility issues associated with Python version 2.7.

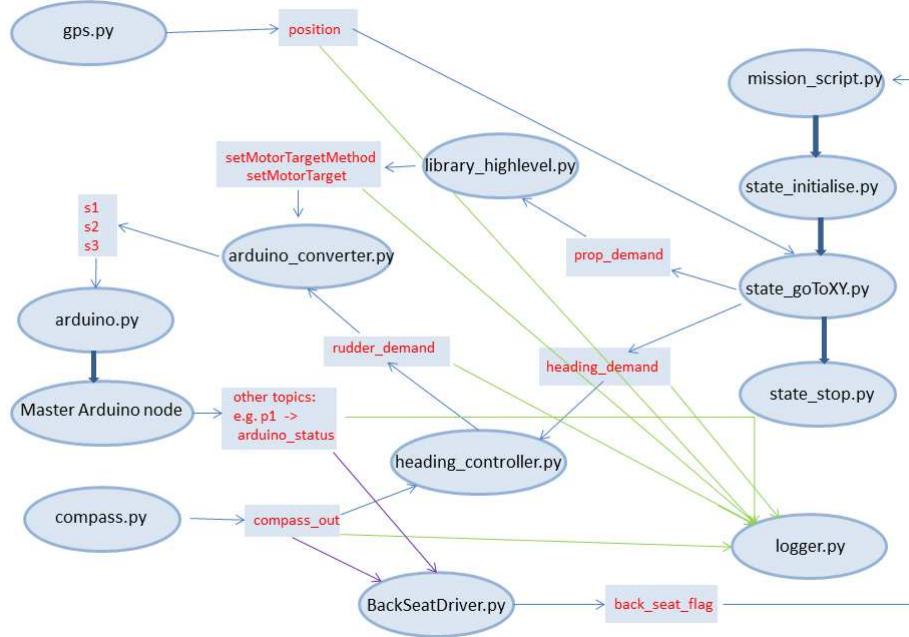


Figure 3.6.2: Diagram of the control system under the operation of a mission that requires the model to move between pre-defined points. Node names are written in black, while topic names in red. Green lines lead to the logger, while purple lines to the back-seat driver nodes respectively. Blue lines indicate the sharing of topics between adjacent nodes. The list of topics published by the Arduino is too long to be included fully. These change label after they pass through the heading controller node as described in Section 3.6.5.

A function is dedicated to the opening of and the reading of stream data from the compass port. A further function is designed to check (up to 5 trials allowed) whether the format of the data is correct. This is based on the total string length and the nature of the first character, which should be “\$”. If the port is opened and the stream of data is in the correct format, the compass node status is published in the main function as being on, read by the back-seat driver. It is then possible to call a further function to read the data continuously through a while-loop and divide it into single numbers in an array (columns) in floating point format. This function was much simpler than the original version in Delphin2, which contained complex calibration functions and changes of sign. These were deemed unnecessary, as the compass was fixed in the correct upright position and calibrated in an external program, CuteCom [Neundorf, 2009], following the instructions provided by the manufacturer [OceanServer, 2010]. This calibration process consists of pressing Esc+C (1B + 43), fully rotating the compass on a level plane and when finished pressing the Tab key (20). Afterwards, it is necessary to tilt the compass to the side, press Esc+Z (1B + 5A), rotate the compass fully and when finished press the Tab key (20). The keys in brackets correspond to the commands in hexadecimal language, as used on CuteCom. Since the rate of publication of the compass variables is 100 Hz, no time averaging of the values was deemed necessary. Furthermore, the node is also provided with a shut-down function to close the serial port if the node is closed by the user.

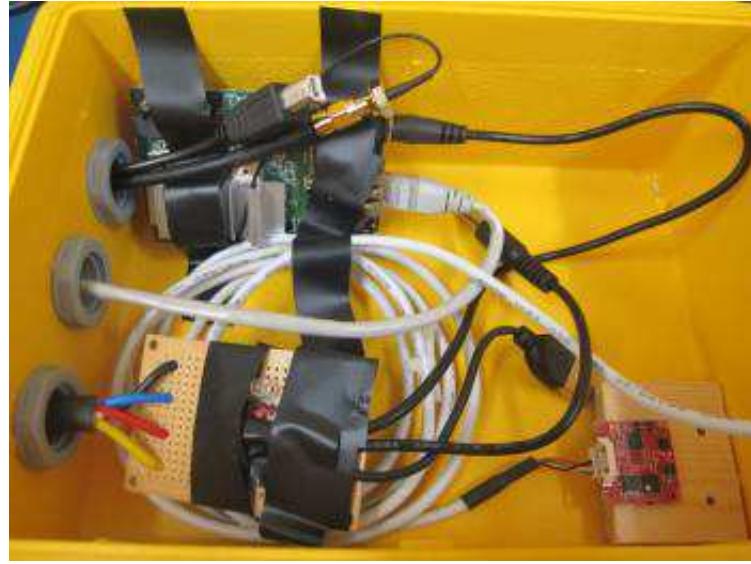


Figure 3.6.3: Pelicase box and its components. Inn the bottom right corner it is possible to see the compass unit, while inn the top left corner the WIFI unit is just visible, partly covered by electric tape. On top of it, there is the extension cable for the WIFI aerial, which passes through the same cable gland as the plug of the USB hub, which is required only when the laptop is employed. The ethernet CAT5, grey cable passes through the middle cable gland. The last cable gland (at the bottom of the figure) is used for the USB connections to the electronics unit as well as the powering of all components within the Pelicase. The Pelicase is oriented towards the bow of the model in the direction of the compass (hence, the bow would be to the right of the figure).

3.6.4 GPS

[ANDERLINI]

The GPS unit adopted in this project was a GlobalSat [GlobalSat, 2013] BU-353S4 with USB interface, which can be seen in Figure 3.6.4. It has an acquisition rate of approximately 35 s (time required to acquire signal) and a sampling rate of 1 Hz. This latter value is particularly low so a future update should substitute this GPS with a quicker one. Indeed, the signal was lost three times for periods of approximately 20 s when the model was travelling at speeds greater than 1.3 m/s.



Figure 3.6.4: GPS unit. This figure has been taken from GlobalSat [2013].

As the same GPS unit is mounted on the Delphin2, this node (`gps.py`) was not modified other than to

solve a bug in Python version 2.7 when reading the stream of serial data. Like the compass node, specific functions are provided for opening the serial port, checking whether the data has the correct format and closing the port if the program is aborted by user input. In addition, four extra functions are required to identify different types of incoming messages. Hence, the message type can be:

GPGGA (global positioning system fix data), where the string includes the latitude, the longitude, the GPS time, whether a GPS fix has been achieved and the number of satellites contacted,

GPGSA (GPS active satellites), which contains the identity of the contacted satellites,

GPRMC (recommended minimum specific GPS/transit data), which includes the GPS time, the latitude, the longitude and the speed (originally in knots, but immediately converted into m/s)

GPGSV (GPS satellites in view), where only the number of satellites is present [Baddeley, 2001].

In the main function, the serial data is continuously read through a while-loop provide the port is open and the data is in the correct format. These strings are then subdivided into the corresponding variables and the latitude and longitude are converted into decimal degrees. Moreover, by exploiting an original position (reference position, input into the code by user input), the latitude and longitude in decimal degrees and the radius of the Earth (6371000 m), assuming the Earth is a perfect sphere [Movable Type Scripts, 2013], the x - and y -coordinates (expressed in metres) of the current position are calculated.

At the end of the process, the node continuously publishes (with a sampling period of 1 s) as a single topic the latitude and longitude in decimal degrees, the GPS time, the number of satellites, the presence of a GPS fix (0 negative and 1 affirmative), the model speed and the x - and y -coordinates as compared to a reference point in m.

3.6.5 Arduino

[ANDERLINI]

The master Arduino was also connected to the computer via a USB port. The communications are slightly more complicated than for the compass and GPS (purely reading serial data), and three nodes are necessary for the exchange of data. The information passed between the two systems was a form of serial input and output, which exchanged floating and integer values as topics. This was enabled by the stack `rosserial` [ROS.org, 2013], which allowed the running of nodes on the master Arduino and was described in Section 3.4.3. There was a C++ node running on the Arduino implemented by the `rosserial_arduino` library. It read the value picked by its sensors (e.g. motor voltage and current, battery voltage, etc.) and published them as separate topics. It also subscribed to the topics published by the main computer, which were the rudder angle demand, the motor demand (voltage, power or propeller revolutions, etc.) and the motor demand method (choosing between any of the previous methods). These were contained in a single topic per variable (e.g. motor voltage is a topic, etc.) and exchanged separately using Python and C++ `Float32` and `Int8` formats.

There were 16 publishers and 3 subscribers operating on the master Arduino; due to the limited size of the SRAM on the master module, their names were very short (s_i and p_n , where i is the number of the respective publisher and n the number of the respective subscriber). For this reason, an additional node was required on the main computer (in Python, `arduino_converter.py`), to act as a converter for the names of the topics. Hence, this node simply subscribed to all topics published by the Arduino and published the same topics under more descriptive names (e.g. “p4” becomes “`caseTemperature`”) to make the de-bugging easier and more reliable. Similarly, the code subscribed to the rudder angle demand, motor method and motor target topics, from the other nodes in the software, and published them as “`s1`”, “`s2`” and “`s3`” respectively.

One final Python node was required on the main computer to open the Arduino serial port to permit the exchange of data (`arduino.py`). A very specific language had to be used to enable the `rosserial_arduino` package on the Arduino, open its port and fork server, so it was preferred to copy the python node from the `rosserial_python` package to the package `ASV` and modified the port name according to the `/udev` local rules settings and a shut-down function has been added.

3.6.6 Heading Controller

[ANDERLINI]

The heading controller node (`heading_controller.py`) received a heading demand from the higher level nodes, calculated the required change in heading from the current heading angle (from the compass), estimated the necessary rudder angle to achieve such change and publish it, to be picked up by the master Arduino. Hence, this node subscribed to the compass output and the heading demand topics and published a rudder angle demand. As in the case of the logger, these publishers and subscribers were created directly in the node itself, rather than calling the functions within the high-level library.

Different types of controller were considered, including PID, IMC (Internal Model Control) and LQG (Linear Quadratic Gaussian) controllers [Franklin et al., 2010]. State-space methods provide the possibility of creating more adjustable and reliable digital controllers, but their use would require complex modelling of the equations of motions which could not be achieved as part of this project. Hence, it was decided to adopt the simplest controller with a frequency-domain analysis, namely the PID controller due to its ease of development and relative robustness. It has been successfully used on Delphin2, [Philips et al., 2010], and a very similar tanker model [Moreira et al., 2007].

In order to design a successful controller, it was necessary to represent the motions of the ASV with a mathematical model. The simplest model specifically developed to represent the manoeuvring (or control) characteristics of a ship is the Nomoto 1st order model [Fossen, 2011]:

$$Tr + Kr = \delta \quad , \text{ or in non-dimensional format : } T'\dot{r}' + K'r' = \delta \quad (3.6.1)$$

In Equation (3.6.1), $r = \dot{\psi}$ is the rate of turn where ψ is the heading angle, δ is the rudder angle and T and K are the Nomoto coefficients. In order to obtain the non-dimensional form, $r' = rL/\bar{U}$, $T' = T\bar{U}/L$ and $K' = KL/\bar{U}$, where L is the ship (or model length) and \bar{U} its average forward speed. Similarly, the non-dimensional time is given by $t' = t\bar{U}/L$. Due to the form of Equation 3.6.2 it did not matter whether the dimensional or non-dimensional form of the equation was employed in the development of the control system for the model, as long as the scaling was based on the Froude number equality. The non-dimensional form has been used within this section, so that the results from the full-scale and the model-scale vessels could directly be compared. The Nomoto coefficients have been calculated from the full-scale and model-scale manoeuvring trials as described in Section 5.2.3. If the Laplace transform of Equation (3.6.1) is taken, then the result is the celebrated auto-pilot model [Fossen, 2011] in the frequency domain:

$$\frac{\psi}{\delta}(s) = \frac{K'}{s(1 + T's)} \quad (3.6.2)$$

Now that the transfer function of the plant is available (Equation (3.6.2)), it was possible to design a compensator to stabilize it. The name of the PID controller derives from the fact that the input required (rudder angle) to obtain the desired output (change in heading) is equal to the sum of a proportional (P), an integral (I) and a derivative (D) term [Franklin et al., 2010]. The general plant structure of a simple unity feedback system is shown in Figure 3.6.6, where e is the tracking error (required change in

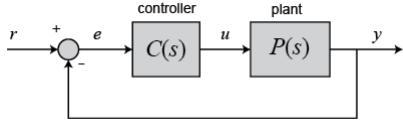


Figure 3.6.5: Diagram of a simple unity feedback system (taken from University of Michigan [2012]).

Table 3.6.1: Controller response for single proportional, integral and derivative controllers with respect to response rise time, overshoot, settling time and steady-state error. The table has been taken from University of Michigan [2012].

Controller Response	Rise Time	Overshoot	Settling Time	S-S Error
K_P	Decrease	Increase	Small change	Decrease
K_I	Decrease	Increase	Increase	Eliminate
K_D	Small change	Decrease	Decrease	No change

heading), which is equal to the difference between the desired input value, r (actual heading), and the actual output, y (desired heading angle) [University of Michigan, 2012]. The control signal to the plant is equal to [University of Michigan, 2012]:

$$u = K_P \cdot e(t) + K_I \int e(t)dx + K_D \frac{de}{dt} \quad (3.6.3)$$

In Equation (3.6.3), K_P is the proportional gain, K_I the integral gain and K_D the derivative gain. The control signal u was sent to the plant and as a consequence the new output signal y was calculated and then fed back and compared with the desired value to estimate the new error signal e . The process was repeated, with the controller taking the new error signal and computing its first derivative and its integral [University of Michigan, 2012]. Taking the Laplace transform of Equation (3.6.3) produced the transform function of a general PID controller [University of Michigan, 2012]:

$$G(s) = \frac{K_D s^2 + K_P s + K_I}{s} \quad (3.6.4)$$

Considering the three individual components of a PID controller, the proportional controller (K_P) reduces the rise time and the steady-state error, although it will never completely cancel the latter [University of Michigan, 2012]. The integral control (K_I) eliminates the steady-state error for a constant or step input, but it results in a slower or more cumbersome response [University of Michigan, 2012]. The derivative control (K_D) typically augments the stability of the system, reduces the overshoot and improves the transient response [University of Michigan, 2012]. This is summarized in Table 3.6.6. However, this general behaviour of each single controller may not be exactly accurate when the three controllers are combined, as they are dependent on each other [University of Michigan, 2012]. The main drawback associated with the derivative term is that it has a very high gain for high frequency signals so noise would generate large variations in the control signal [Murray, 2006], a first-order filter is typically used for this term and the transfer function of the PID controller was modified as follows [Murray, 2006]:

$$G(s) = K_P + \frac{K_I}{s} + \frac{\frac{K_D}{K_P}}{1 + sT_f} \quad (3.6.5)$$

In Equation 3.6.5, $T_f = \frac{K_D}{K_P N}$ is the filtering time, where N lies in the range 2-20. Although Equations

3.6.3 and 3.6.5 are expressed in continuous-time form, this is not the actual case for the control system of the model, since the computer has a digital nature with a sampling rate for both the actuators and the sensors [Murray, 2006]. Therefore, it is possible to obtain the transfer function of the discrete-time system through the bilinear transformation $s = \frac{2}{T_s} \cdot \frac{z - 1}{z + 1}$ [University of Michigan, 2012], where $T_s = \delta t$ is the sampling period. In its digital form, Equation 3.6.3 becomes [Murray, 2006]:

$$u(t_i) = P(t_i) + I(t_i) + D(t_i) \quad (3.6.6)$$

where:

$$P(t_i) = K_P e(t_i) \quad (3.6.7)$$

$$I(t_i) = I(t_{i-1}) + K_I \delta t e(t_{i-1}) \quad (3.6.8)$$

$$D(t_i) = \frac{T_f}{T_f + \delta t} D(t_{i-1}) - \frac{K_D(e(t_i) - e(t_{i-1}))}{T_f + \delta t} \quad (3.6.9)$$

In Equations 3.6.6-3.6.9, i refers to the current time iteration step, while $i-1$ refers to the previous one. As such, a simple first-order backward differencing scheme has been used to approximate the derivative term, which also presents a first-order filter. This digital PID controller has been implemented within the main function of the Python node. This means that the error, e , is calculated as the difference of the heading demand and the current heading (ensuring it lies between the -180° and 180° bounds) and that the time difference, δt , is computed as the current time less the time of the previous step. The value of the error of the previous time-step is also stored in order to calculate the derivative term. The required rudder angle demand corresponds to the output u . A function ensures the rudder angle magnitude does not exceed the limiting value of 35° . An external while-loop ensures the required rudder angle is calculated continuously, and published periodically.

Only the proportional controller had been implemented for the testing, as the optimal controller should be as simple as possible. This meant that, for instance, if a PI controller was sufficient for the desired response, there would be no need for a PID controller [University of Michigan, 2012]. A value of 2 was assumed for the proportional gain based on previous experience with Delphin2. This controller has been very successful in practice, being only slightly too responsive at high model speeds.

Nevertheless, after the Nomoto coefficients have been calculated in sections 6.4.1 and 6.4.2, it has been possible to obtain K_P , K_I and K_D by tuning the controller to the mathematical model of the ASV. The tuning process is very complicated and time-consuming; it can be performed by trial and error using Table 3.6.6 or can rely on more advanced techniques based on Bode and Nichols plots (e.g. Ziegler-Nichols' tuning [Murray, 2006]). Within this project, for simplicity, the automated tuning design toolbox **sisotool** of Matlab has been adopted for the derivation of the transfer function of the required P, PI and PID controller required to stabilise the plant provided by Equation 3.6.1. This was calculated using a robust response tuning algorithm, which ensures a good compromise between robustness and performance. In addition, the optimal value of N to be used in the derivative filter has been calculated automatically by Matlab. From the transfer function, the gains have been obtained from Equation 3.6.5. The results are shown in Table 3.6.6. The first three controllers were obtained from the full-scale ship data (Section 6.4.1), while the latter three from the model data itself (Section 6.4.2). From Table 3.6.6, it is clear that the model data provided the most sensible results as compared with the empirical gain used and the literature [Moreira et al., 2007]. In particular, the proportional gain for a pure P-controller is very close to the empirical one used throughout the project. Furthermore, its being slightly smaller should be even more beneficial, considering the excessive response at high model speeds. However, the

Table 3.6.2: Proportional, integral and derivative gains and number of filtering intervals that have been obtained from the non-dimensional Nomoto coefficients for different controller types at both the full-scale and the model-scale ships.

Controller type	K'	T'	K_P	K_I	K_D	N
P (full-scale)	-0.1801	0.7609	-4.8646			
PI (full-scale)	-0.1801	0.7609	-4.2194	-0.04964		
PID (full-scale)	-0.1801	0.7609	-14.9313	-0.3015	-7.7890	2.0866
P (model)	-0.7952	1.1597	-0.72287			
PI (model)	-0.7952	1.1597	-0.6292	-0.0048399		
PID (model)	-0.7952	1.1597	-2.1991	-0.0294	-1.7743	2.1233

gains of the full-scale ship are not too dissimilar from those of the model and their larger magnitude reflects the poorer manoeuvrability of the full-scale ship, as shown by its lower K/T ratio (see Section 6.4.3) [Fossen, 2011].

The three controllers that use the model-scale ship data (P, PI and PID) have been implemented in the software for future validation during testing (`heading_controllerP.py`, `heading_controllerPI.py` and `heading_controllerPID.py`). Nevertheless, it should be noted that the gains are positive in the software, as a positive rudder angle corresponds to a counter-clockwise rotation (to starboard).

This node does however have an intrinsic issue that cannot be removed. If it is run while no heading angle demand is being publish, the model will automatically steer northwards, as this corresponds to the 0° value specified in the software to initialise the node. Other more complicated controller options are recommended for future use in Section 8.

3.6.7 Nodes dedicated to Testing

[ANDERLINI]

Initially, no use of the ROS smach package has been made to prepare the nodes dedicated to performing the straight run tests and circle and zigzag manoeuvres. This solution offered greater simplicity and robustness (the smach states function only with the back-seat-driver node, hence requiring the whole system) as well as improved debugging capabilities (as the whole system is run with smach states, debugging is much more difficult, with the largest amount of problems deriving from the interactions with the back-seat driver). All nodes in this section have common characteristics: the call of functions from the high-level library (Section 3.6.10), the publishing of the node status (whether it is on or not), the use of the time to divide the code execution in different parts and a main while-loop that ensures that the demands (motor or rudder) are constantly being published (the concept is to perform actions as long as the package `rospy`, continuously used by the Python programs, is running). In particular, due to its presence, simple if-loops contained within it are sufficient, removing the need for further while-loops (with the exception of the circle and zigzag manoeuvres, as will be discussed).

Self-propulsion tests Considering the nodes employed during the self-propulsion tests at QinetiQ (`qinetiq.py`), their operating principle is very simple. They set a motor method (or setting) and motor demand after 5 seconds from the start of the node and finish publishing these values after a predetermined time interval. Although initially a motor voltage was specified, after the first tests at the Lamont towing tank, the desired propeller revolutions number (RPM) could be accurately set. The predetermined stopping time (from the start of the program) was typically either 117.2 s or 175.2 s, corresponding to the time required by the model to travel 100 and 170 m respectively at a speed of 1.0294 m/s (corresponding to the operational speed, 15.5 knots full scale) both increased by 10 seconds, approximately equalling

the time required by the operator to start the carriage. Originally, an extra loop was added to decrease the model speed before stopping the motor (with the appropriate function from the high-level library), but this has been removed as the Arduino software already has a ramp function that serves the same purpose. Different copies of this node have been produced, each with a different demand for the propeller revolutions so that they were ready to be launched during the self-propulsion tests without the need to modify them at the end of each run. For this reason, the files name also contains this value (e.g. `qinetiq_400.py`).

Circle trial Considering the node dedicated to the circle manoeuvre (`circle.py`), immediately after the while-loop statement, a propeller demand was set and kept (thanks to the while-loop) until the end of the run. An if-loop ensured that the model was running straight (rudder angle demand is 0°) for the first 10 s so that it reached a steady-state condition. After this a rudder angle demand was set (for simplicity 20° , 25° or 30° , was chosen) and the code waited other 5 s according to a further if-loop. This pause was fundamental, as it ensured a consistent change in heading (typically 30° , but in any case still larger than 5° even with strong wind) from the initial position. A further while-loop was necessary, to check whether the absolute value of the difference of the current heading from the zero heading (heading error) is greater than 5° . Any other value could be used, but smaller values could have been problematic because there is the small possibility that the compass readings may not have been fine enough if there is wind. In addition, the model still has the necessary inertia to finish (and overshoot) the circle manoeuvre. If the error in heading is still greater than 5° , the rudder angle demand is still being published and the model continues its turn. Once the error drops below the limit, the stop function of the high-level library is called. Circle manoeuvres were only performed towards the starboard side to decrease the possibility of human error and collisions in the lake. Nevertheless, this can very simply be modified. Moreover, the rudder angles can also be changed, but it should be noted that values equal to or greater than 20° were employed because of the relatively narrow space available at the lake.

Zigzag trial The zigzag manoeuvre is similar to the circle manoeuvre but with some obvious differences. After the initial run, identical in nature to the one designed for the circle manoeuvre, a positive rudder angle demand (20° , towards starboard) was set and a while-loop continuously checked whether its value was larger than the difference between the current and original heading. Once this changed, the rudder angle demand was inverted in sign (-20°). The same process was repeated on the other side, but the new while-loop checked whether the error (difference of current and zero headings) was greater than the rudder angle demand because of the sign change. Originally, only one turn per side was programmed according to the ITTC [2002a] recommendations and the full-scale ship tests (`zigzag.py`). However, as the results were not very clear, this process was extended to two turns per side (simply extending the technique described above (`zigzag2.py`)). Although the manoeuvres were performed to the same side for simplicity, they could be started towards the port side, noting the required change in sign. The rudder angle demand value may also be changed (according to the ITTC [2002a] recommendations 10° - 10° and 20° - 10° zigzag manoeuvres are possible), although a relatively large value is advised to see any significant course changes.

During the lake testing, three speeds, in the region of the expected model self-propulsion point, were used for both manoeuvres. For both the circle and zigzag manoeuvres, having the heading controller node running simultaneously had advantages and drawbacks. On the one hand it ensured a straight initial run irrelevant of the previous manoeuvre, so was possible to immediately begin another manoeuvre in the same direction and, in the case of the circle node, approximately the same location provided the wind

was not too strong. On the other hand, since the heading controller also used a while loop to publish the rudder angle demand, it was found to override the rudder angle demand every time it published (approximately 0.5 second intervals). However, this should not have affected the manoeuvres, as the servo feedback loop is not as responsive and the subsequent demand restored it to the original rudder angle.

3.6.8 Smach States Machine

[ANDERLINI]

“Smach” is a stack (collection of packages) that provides the task-level framework for the rapid development of complex robot behaviour [ROS.org, 2013]. It is a Python library that should be used to create hierarchical state machines [ROS.org, 2013], for every state it is necessary to establish the possible outcomes [ROS.org, 2013] so that the software is able to autonomously decide how to proceed. In the software developed as part of this project, the three possible outcomes are “succeeded”, “pre-empted” (the initial conditions have not been met, e.g. the Arduino status is off) and “aborted” (mission time-out or the process has been stopped by user input). Each state is a Python class and requires an initialization and execution function, where the possible outcomes are to be found. In total, only five state nodes were prepared for this project, namely an initialization, a navigation, states to execute circle and zigzag manoeuvres, and a finalization node. However there was still many possible missions, as the states could be combined.

The main advantage of smach states is that multiple states can be run, their order may be changed, they can be repeated, the positions where the model should move to are defined for the navigation state and it is also possible to specify a time interval after which the smach state will be aborted [ROS.org, 2013]. Hence, it is possible to build complex missions, where the model moves in a predefined path and executes the manoeuvres autonomously.

All states can only be launched from an external Python node, the mission script (`missionscript.py`). In this file the smach state machine is created with the final outcome “finish” and the desired states added to it. However, during the process it was necessary to specify some parameters required by the states for their operation. All states required the high-level library (to access its functions) and the time-out to specify its execution in the mission script node. The navigation state also needed the *x*- and *y*-positions to travel to, including permitted tolerance. All these values were expressed in metres. Furthermore, the manoeuvring states needed the motor and rudder angle demands. During the specifications of the states within the state machine, it was necessary to express their execution order, and a call to a following state for each outcome. For instance if a very simple mission is written, where the model moves to one point, the first state will be the initialization, whose success calls the navigation state, but an “aborted” or “pre-empted” outcome calls the stopping state. The navigation state will also call the finalization state to publish the completion of the mission to the logger. In addition, the mission script is also provided with an introspection server for visualization and debugging purposes and a shut-down function in case the program is closed by user input.

Several mission scripts (e.g. straight runs, combination of manoeuvres etc.) were prepared before lake testing. They were only briefly tested because of the limited time-scale and the presence of other vessels at Timbsbury lake. It is recommended to verify these scripts in a future open water test.

Initialization The initialization node (`state_initialize.py`) checks whether the Arduino, compass and GPS nodes are on-line and if not the case, the state is aborted. It also checks the battery voltage and if it is below 10 V, the state is pre-empted. Otherwise, the state exists with success and the mission script automatically moves to the next state.

Manoeuvres The circle and zigzag manoeuvres smach states (`state_circle.py` and `state_zigzag.py`) are identical to the codes previously described. In addition, the motor (e.g. propeller revolutions) and rudder demands (e.g. -35° for a 35° circle manoeuvre to the port side) need to be specified in the mission script, so that this greatly increases their flexibility. However, such codes are run only, returning “succeeded”, if the state has not timed out. Otherwise, the state is aborted. In case the back-seat driver flag is 1 (an error has occurred in the software), the state is pre-empted.

Navigation The navigation, or “go to”, node is one of the most complicated (`state goToXY.py`). Its purpose is to drive the model from the origin to a specified point. Numerous alternatives could be found in the literature, especially concerning AUVs, however in order to perform manoeuvres and straight run self-propulsion tests consistently rather than following gymkhana tracks, one of the easiest solutions was employed. The track-follow algorithm based on the article by McPhail and Pebody [1998] originally employed on the AUV Delphin2 [Philips et al., 2010] was simplified based on first principles. The node uses the coordinates of the destination and the GPS output to compute the quickest path to the final point and thus the required heading demand, which is transformed into a rudder angle demand by the heading controller node. By using the current heading value from the compass output, it determines the motor demand (expressed as desired propeller revolutions) by adopting a stepped method, where the motor power is increased for smaller heading errors (difference between the heading demand and the current heading).

A while-loop is employed to perform all calculations repeatedly and continuously publish the heading and motor demands. The range to the final destination is calculated as:

$$\text{range} = \sqrt{dx^2 + dy^2} \quad (3.6.10)$$

In Equation 3.6.10, dx is the difference of the x-position of the target and current GPS x-coordinate, while dy the difference of the target y-position and the current GPS y-coordinate. An if-loop is used to check whether the range is smaller than the specified tolerance, thus returning the “succeeded” flag. Otherwise, the bearing is calculated, which is the angle between the x-axis (positive Eastwards) and the line that connects the current position with the target destination:

$$\text{bearing} = \arctan\left(\frac{dx}{dy}\right) \quad (3.6.11)$$

The bearing does not immediately correspond with the heading demand (new desired heading) because of the employed axis system. While the bearing value is $\frac{\pi}{2}$ (the Python library `numpy`, uses bearings in radians) along the y -axis, this corresponds to 0 on the compass readings (it was assumed that the magnetic and real magnetic poles coincide, even though this is not exactly the case). It should be noted that the heading readings throughout the software range from 0 to 360° or 2π , whereas the arctangent function is positive for points in either the first or third quadrant and negative for points in either the second or fourth quadrant, to correct for these, the heading demand was calculated as follows:

$$\text{heading} = \psi = \begin{cases} \frac{\pi}{2} - \text{bearing}, & \text{if } dx \geq 0 \\ \frac{3\pi}{2} - \text{bearing}, & \text{if } dx < 0 \end{cases} \quad (3.6.12)$$

The heading demand angle is then transformed in degrees and the heading error computed, which is the difference between the heading demand and the current heading angle as published in the compass

output. A function is employed to ensure the error lies in the (-180°) - 180° range. At this point, if-loops are used to specify four values for propeller revolutions depending on the absolute value of the error as follows:

$$\text{propeller RPM} = \begin{cases} 400, & \text{if } \tilde{\psi} = 180^\circ \\ 500, & \text{if } 90^\circ \leq \tilde{\psi} < 180^\circ \\ 800, & \text{if } 45^\circ \leq \tilde{\psi} < 90^\circ \\ 1000, & \text{if } \tilde{\psi} \leq 45^\circ \end{cases} \quad (3.6.13)$$

In Equation 3.6.13, $\tilde{\psi}$ represents the heading error. This solution avoids the model travelling too far from the target when facing the wrong direction and consuming valuable energy.

Finally, the heading and propeller revolutions demands are published by calling the specific functions from the high-level library. This process is continuously repeated due to the while-loop until the range falls below the tolerance and the states exits with success. The smach state can be pre-empted if the backseat driver presents errors or aborted if the mission has timed out.

Finalization The finalization node has only one outcome: “succeeded”. It publishes a string describing the conclusion of the ASV mission to the logger and stops the motor and rudder using the stop function of the high-level library. In the mission script node, this state is called either at the end of the mission or whenever one of the previous states returns the “aborted” or “pre-empted” flags.

3.6.9 Back-seat Driver

[ANDERLINI]

This node (`BackSeatDriver.py`) is vital, as it monitors the status of the vehicle, checking that everything is running smoothly. In case of any errors, it publishes a Boolean operator to shut down the mission script (after stopping the motor and the rudder). If Arduino warnings are issued, they are printed to the screen, but the flag (Boolean operator) is left unchanged at zero. The same is true if water is detected, as it would not be possible to drive the model home if the system was shut down. Errors are displayed when the maximum roll and pitch, the minimum battery voltage, the maximum temperature within the Pelicase and electronics box or the current time exceed predefined values. Similarly, a back-seat driver flag of 1 is published if the Arduino node is down or unresponsive.

3.6.10 High Level Library

[ANDERLINI]

The high level library is a Python class (`library_highlevel.py`) which subscribes to all topics being published and publishes all required topics, such as the motor method, the motor demand (e.g. propeller rpm, motor voltage etc.), the rudder angle demand, the heading demand etc. It contains an ordered list of all publishers and subscribers, which can be called when necessary from the other nodes by specific functions. Every function also contains a text string that prints to the screen, simplifying debugging. To read data, “getter” functions were designed, while “setter” functions ensure required data is publishing. Typically, each function handles one variable, except the stop function which sends a zero to the Arduino for the rudder angle demand, the motor demand and the motor method.

3.6.11 Logger and Average-calculator

[ANDERLINI]

The logger node (`logger.py`) provides a solution to save all data during the tests. Without this process everything may run smoothly but not log any data, resulting in a wasted test session, as a result priority has been given to robustness rather than appearance. Instead of calling the functions from the high-level library, the subscribers to the required topics were included within the program. This ensured continuous working even if any of the topics it subscribes to were not being published (a zero would appear in the log files). The stored variables are the compass and GPS output, the heading demand, the rudder angle demand, the rudder angle demand as stored on the Arduino at the same time, the motor method request (i.e. idle, motor voltage, motor power or propeller rpm), the actual motor method, the motor target (i.e. the desired value of the selected variable), the motor target as seen by the Arduino, the actual propeller rpm, the actual motor voltage, current and power, the actual thrust, the battery voltage, the case temperature and the mission strings (e.g. when each mission and/or manoeuvre is started). These were all saved separately with a time stamp (computed from the start of the run) in CSV-formatted files. The only exception were the topics published from the compass and the GPS, which were stored in only two (one for each file). Hence, the heading log file (from the compass) contains a column for time, heading, roll, pitch, temperature in the Pelicase (where the compass is stored), four columns for the magnetic field vectors (three) and their sum and four columns for the acceleration vectors and their sum. The GPS position log includes the time, the *x*- (positive Eastwards) and *y*-positions (positive Northwards), the model speed, the latitude and longitude and a Boolean operator which determines whether a valid GPS fix was obtained (1) or not (0). The CSV-formatted files were stored within a common folder, named at the time the logger was started. To ensure greater robustness of the code, a shut-down function was been included, to close the files and containing folder if is exited by user input or the node encounters an error. Note that the node starts logging five seconds after launch to ensure that all systems are on. However, this time delay may be decreased or removed, as necessary.

A further logging node (`averager.py`) was created specifically for steady-state testing (it was originally designed for the self-propulsion towing-tank tests, but could be used during straight line runs and circle manoeuvres). It was very similar to the main logger: same approach to the subscribers, similar shut-down function and the same folder per experiment structure. However only one file was saved, containing one column for the time (from the zero time), the motor setting, the motor target, the propeller revolutions, the motor voltage, current and power, the battery voltage, the case temperature and the thrust. Hence, only the essential variables were stored and in this case, only the mean of each variable (with the exception of time), averaged over 20-seconds intervals, was saved. Note that the denominator of the mean, calculated at the end of each interval, was increased by 1 because the first value for each variable is 0 (necessary as an initial value to start up the file). This logger does not have the 5 seconds waiting time (it immediately begins logging as soon as it is launched) and has a total time length of 160 s, to match the length of the runs at QinetiQ. Nevertheless, this can be easily modified and the file can be closed by user input with the shut-down function, but the mean of the current interval will be lost.

3.6.12 Shut-down Function

[ANDERLINI]

All higher level nodes, such as the nodes designed for the testing and the mission script were provided with a shut-down function. This function, also known as a “shut-down hook” (ROS.org, 2013), is called if the program (or node) is closed by user input (`Ctrl+C`), and ensures that the master Arduino receives a motor method, a motor demand and a rudder angle demand value of 0, to turn the motor off, by calling the stop function from the library high-level. A similar tool was used in the original code, but some

modifications were necessary due to the newer Python version (2.7), where it has to be clearly stated in the code that the function takes no arguments. Nevertheless, with the current software version the shut-down function, when called, executes its loops and prints the messages to the screen (e.g. the motor demand is 0, as written in the stop function in `library_highlevel.py`) irrespective of the selected time delay (even zero) within the shut-down function; however it does not publish the required topics. This problem has been observed by other users and the ROS documentation [ROS.org, 2013] warns developers about the bug. So far no solution has been found other than manually turning off the motor and rudder from separate windows and user input. In any case, it is strongly advised to find a neater alternative solution in the future, unless the improvement is brought about by Python version 2.8.

3.6.13 Launch File

[ANDERLINI]

A launch file (`asv_mission.launch`), written in XML format, was used to run multiple programs at once. This reduces the possibility of human error by minimising the number of tasks to be performed. It is fundamental for the back-seat driver node, which can only be run from a launch file in the current configuration. The launch files contain the required limiting values for maximum pitch and roll (both 45° but unnecessary as the ASV does not have the same issues as the AUV), minimum battery voltage (10 V), maximum temperature within the electronics box and the Pelicase (50°C), the mission time out (typically 4 hours, but longer when the Beagleboard XM is running) and the reference GPS positions (also present within the GPS node).

In order to launch the nodes from this file, it is necessary to specify their name (note that this is the name of the node itself rather than the Python program which may differ), the package they belong to (ASV), whether their results should be printed to the screen (for rapid debugging) and if they are required. By selecting this last option, if the node exits or is pre-empted, then the whole system is finished. The only node that has been necessary is the back-seat driver to increase the robustness of the system (e.g. sometimes the GPS signal was lost for a few seconds). The other nodes included within the launch file are the logger, the compass, the GPS, the node that communicates with the Arduino, the node that converts the subscribers and publishers to and from the Arduino and usually the heading controller. The mission script has seldom been launched, but it has been noticed that its inclusion within the launch file greatly augments the stability of the navigation smach state.

3.6.14 Manual Control

[ANDERLINI]

It was possible to send direct commands in ROS on the terminal to manually steer the model. This was adopted for avoiding collisions at lake testing, but in the future it is foreseen that only automatic missions will be launched, with manual override being used as the last option to prevent disasters.

Such commands are rather long because of how ROS is designed. Within this section, only the most fundamental commands are included. They can be seen in Table 3.6.14 [ROS.org, 2013]. The `rostopic` publishing command was sufficient to steer the model in conjunction with the launch file as the only topics required were the heading demand (if the heading controller is running, otherwise the rudder angle demand), the motor method (e.g. propeller revolutions) and the motor target (e.g. RPM).

For future use, a GUI program launched on the receiving computer (see next section) could greatly simplify this process, where buttons on a window corresponds to commands on the terminal window of the sending computer.

Table 3.6.3: Useful ROS terminal commands.

Command	Description
<code>roscore</code>	It launches ROS. First command to be run.
<code>rosrun [package] [node]</code>	It launches a single node (to be substituted with the file name, e.g. <code>arduino.py</code>) of a package (change with package label, e.g. ASV).
<code>roslaunch [package] [launch file]</code> <code>rostopic pub [topic] std_msgs[message type] [value]</code>	It runs a launch file (e.g. <code>asv_mission.launch</code>) It publishes a topic, where the topic name (e.g. <code>rudder_demand</code>), message type (e.g. <code>Float32</code> , <code>Int8</code> or <code>Bool</code>) and the value (e.g. <code>20</code>) are to be specified.
<code>rostopic echo [topic]</code>	It prints to the screen the selected topic in real time.

3.6.15 Communications with Ashore Computer

[ANDERLINI]

The main computer (in this section defined as sender), on board the model, was remotely controlled by another computer ashore (in this section known as receiver). This communication (or data stream) occurred through the secure SSH protocol [Ylonen, 2006]. The operating system of the ashore computer does not matter, as long as it supported SSH and possibly X-Windows System (to provide the possibility of running a GUI of the sender computer on the receiver [Opengroup, 2013]) programs. In this project, the receiving computer ran with Windows 8 with SSH Shell and Exceed (provided under licence by the University of Southampton) for the communication. To avoid the sending computer to changing its IP address every time it was turned on, its Ethernet port IP address was fixed in the `network/interfaces` settings [nixCraft, 2013].

During the self-propulsion tests at QinetiQ a 20 m Ethernet cable enabled communications between the two computers. However this was only temporary. An Acksys [Acksys, 2013] industrial, long-range WIFI router was used during lake tests. The unit was connected via an Ethernet CAT5 cable to the sender and to increase the signal range, an omnidirectional, 8 dB aerial was used and located externally from the Pelicase to reduce its disturbance on the compass. Initially, the antenna had been positioned inside the cover during the lake testing, mainly due to lack of time, resulting in poor signal range and loss of connection to the model. The aerial is now external of the cover, but it is strongly advised to perform other tests of its actual range in the near future, as these issues were not encountered on Delphin2 [Stenson et al., 2011], which mounts the same WIFI unit. Indeed, the poor signal could have been caused by the extension cable (in that case a complete reconfiguration of the Pelicase would be necessary) or of the WIFI port on the receiver (doubtful, as it has one of the newest 802.11 a/b/g/n chips on the market).

In order to avoid future collisions, it is strongly advised to write an extra node (hence, still running in ROS) on the sender to constantly ping the receiver (hence, this computer also needs to be specified a fixed WIFI IP address). Whenever the signal is lost, i.e. the packets of data do not come back, the program sends a negative Boolean operator to the high level nodes (e.g. smach states) and the heading controller and subsequently calls the stop function from the high-level library to ensure that the master Arduino receives the command to stop.

3.7 Miscellaneous

[NEALE & THORNTON]

There were several miscellaneous areas of construction required for functional and aesthetic properties of the model. These included the weight platforms, the watertight cover and the superstructure.

In order to obtain the correct VCG and to allow easy access to the weights for trimming, two weight platforms were built into the hull. The aft platform was located 10 cm below the cover and extended 66 cm forward from the aft bulkhead of the model. The forward platform was located 8 cm below the cover (to ensure there was enough room below for the batteries) and extended 60 cm back from the forward bulkhead. The platforms were constructed from MDF and were supported along their length with several MDF transverse crossbeams.

It was crucial that the operating system of the model, particularly the electronics, were protected from water. Therefore a watertight cover was designed and built for the model. The cover was constructed from 2 mm thick aluminium sheet. This was considered to be sufficient, and did not add too much weight high up in the model, which would negatively effect the VCG. Steel bolts were used to attach the aluminium plate to the top lip of the model hull. A rubber backed washer was placed between the bolt and the plate to ensure the cover was watertight. A penny washer was inserted between the nut and the hull to protect the hull. A strip of neoprene was fixed around the edge of the cover which was compressed as the cover was bolted down, providing a watertight seal around the length of the perimeter.

Four hatches were cut into the cover to enable easy access into the model. Hatch covers, slightly overlapping the holes, were cut from the aluminium. These were fixed to the main cover through the use of latches. A neoprene seal was fixed around the edge of each hatch cover, inside the latch fixings, in order to ensure these covers were also water tight.

Finally, for aesthetic effect, the cover was painted and a University of Southampton logo placed on one of the central hatch covers.

The superstructure was based on a 1/60 model scale of the Virtue Ice Class ships' superstructure. Disregarding obvious simplifications (such as the lack of railings, etc.) the model superstructure portrays geometric similarity to the full scale ship, with the addition of an astern extension to house and protect the rudder servo.

The reason for constructing a superstructure for use during model tests comes down to both aesthetics and the resulting wind resistance, which could play a significant role in the manoeuvring operations during tests. It also provides a suitable platform for additional instrumentation and antennas.

The basic drag equation can be implemented to calculate the wind resistance of the superstructure for the apparent head-wind created by the vessel moving forwards:

$$F = \frac{1}{2}\rho V^2 AC_D \quad (3.7.1)$$

where ρ is the density of air (1.225 kg/m^3), V is the speed of the vessel (1.029 m/s), A is the frontal area (calculated to be 0.110733 m^2) and the coefficient of drag was taken to be 0.7 - according to Owens and Palo [1982]. Thus $F = 0.05 \text{ N}$ which adds roughly 0.75% to the estimated resistance of the model (estimated in Table 2.3.5).

It was constructed based on a modular assembly, cut from blue Styrofoam using a 2-D hot wire cutter, and secured to the deck using Velcro strips to allow for easy removal and access to the hull via the stern hatch. To improve durability, the finished superstructure was coated in a GFRP grade epoxy resin.

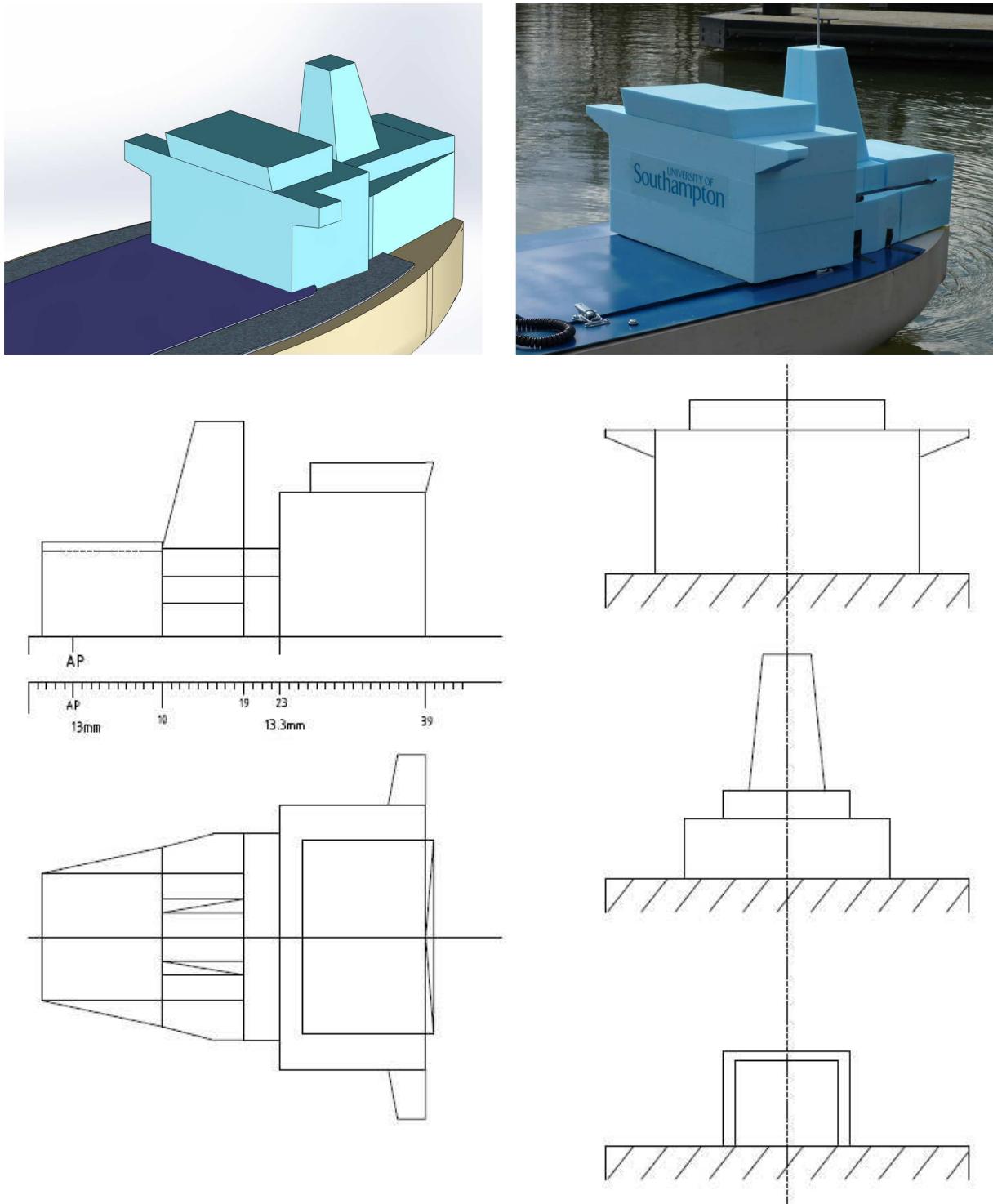


Figure 3.7.1: Superstructure CAD and model.

3.8 Weights And Trim

[NEALE]

3.8.1 Loading Conditions

[NEALE]

In order to scale the results of the model experiments correctly, the displacement, vertical centre of gravity (VCG) and trim of the model were required to correspond to the full scale ship. Two loading conditions were considered: a ballast arrival condition and a loaded departure condition. These were considered to be the extremes of loading that the ship would experience during normal operation. A full scale deadweight of 39,999 tonnes was specified as the fully loaded condition. This was lower than the scantling load (maximum capability of the vessel) of 47,084 tonnes but was considered to be a more realistic loading condition. Testing at these two conditions was to enable the effects of any efficiency improvements to be analysed over a wider band of the ships normal operating conditions. The ship particulars for each loading condition were obtained from the trim and stability booklet for the class and scaled down to model scale (Table 3.8.1). Adjustments were made to account for the model being tested in fresh water, assuming the densities of fresh and salt water to be $1.000t/m^3$ and $1.025t/m^3$ respectively.

Table 3.8.1: Particulars at full and model scale for loaded and ballast conditions.

	Loaded Departure			Ballast Arrival				
	Full Scale	Model		Full Scale	Model			
Scale	60.00	1.00		60.00	1.00			
Water Density	1.025	1.000		1.025	1.000			
Deadweight	39999	MT	180.66	kg	19803	MT	89.44	kg
Deadweight LCG	5.50	m	0.0917	m	14.60	m	0.2433	m
Deadweight VCG	10.89	m	0.1815	m	5.96	m	0.0993	m
Lightship	9970	MT	45.03	kg	9970	MT	45.03	kg
Lightship LCG	-10.00	m	-0.1667	m	-10.00	m	-0.1667	m
Lightship VCG	11.80	m	0.1967	m	11.80	m	0.1967	m
Displacement	49969	MT	225.70	kg	29773	MT	134.47	kg
LCG	2.41	m	0.0402	m	6.36	m	0.1060	m
VCG	11.07	m	0.1845	m	7.92	m	0.1320	m
LCB	4.40	m	0.0733	m	6.35	m	0.1058	m
LCF	-2.28	m	-0.0380	m	4.96	m	0.0827	m
Draught fwd	10.21	m	0.1702	m	6.91	m	0.1152	m
Draught aft	11.84	m	0.1973	m	6.91	m	0.1152	m
Draught mean	11.02	m	0.1837	m	6.91	m	0.1152	m
MCT	615.10	MT-m	2.8477	kg-m	477.40	MT-m	2.2102	kg-m
Trim by stern	-1.62	m	-0.0270	m	0.00	m	0.0000	m

The supplied draughts were located at the perpendiculars. For practicality they were converted to the draughts at the draught marks which were located at section 1 and section 9. Assuming the draught varied linearly between these points:

$$T_{marks} = T_{mean} + (T_{perpendiculars} - T_{mean}) \cdot 0.8 \quad (3.8.1)$$

where T = draught.

Although the full scale ship characteristics were available from the vessel's trim and stability booklet, the actual characteristics of the model hull were unknown. Therefore the model hull was weighed to

Table 3.8.2: Model scale draught mark readings.

	Loaded	Departure	Ballast	Arrival
Draught fwd	0.1152	m	0.1729	m
Draught aft	0.1152	m	0.1945	m

obtain the displacement and longitudinal centre of gravity (LCG); additionally an inclining experiment was performed to obtain the VCG (described in detail later in this section).

In order to obtain the displacement of the hull, the model was emptied of all objects excluding the power train chassis, and supported at Station 1 and Station 10. Weighing scales were used to take a measurement at each location, with 14.68 kg measured at Station 1 and 14.61 kg measured at Station 10. This corresponded to a total displacement of 29.29 kg and an LCG of -0.141m.

The LCG was calculated as follows:

$$LCG = \frac{\sum_{i=1}^n m_i x_i}{\sum_{i=1}^n m_i} = \frac{(14.68 \times 2.9 \times 0.4) + (14.61 \times 2.8 \times -0.5)}{29.29} = -0.141m \quad (3.8.2)$$

A weight survey was conducted in order to obtain a lightship estimate for the model. The results, presented in Table 3.8.8 indicated a lightship displacement of 98.06 kg. The deadweight comprised of a number of lead weights; lead was chosen for its high density, minimizing the volume inside the model that the weights would consume. Additionally in contrast with steel, it is non-ferrous and therefore would not interfere with the magnetometer. An initial estimate for the weight positions was generated using the measured displacements and LCG. The VCG of the hull was estimated from the 3D model to be 0.131 m. However, when placed in the tank it was clear that the model was trimming excessively by the stern. This was most likely due to the hydrostatic properties of the model not matching values in the trim and stability booklet. The model itself was known not to be an exact replica of the full scale ship. Additionally although the trim and stability book was written for the Virtue Class, it was known that there were differences between ships within the class. It was also possible that errors could have been introduced in the measurement of the model's LCG. It was decided to match the trim to full scale, rather than the LCG; this was achieved by making a correction to the hull LCG for each loading condition. The LCG of the hull was therefore corrected to 0.282 m for the ballast condition and 0.52 m for the loaded condition.

3.8.2 Inclining Experiment

[NEALE]

In order to calculate the VCG of the model hull an inclining experiment was performed, the results of which are presented in Table 3.8.3. The experiment was conducted in the Lamont Tank at the University of Southampton. This was a sheltered environment free from wind, current and wave effects. Additionally it was not necessary to use any mooring lines. The temperature of the water was measured as 8.5°C. This remained constant over the length of the experiment and the effect on viscosity was considered negligible.

The model was loaded to ballast condition. Two batteries had yet to be installed and were replaced temporarily with steel weights. Additional missing lightship weights such as the rudder and rudder stock were noted. A table of weights and corresponding VCG's is presented in Table 3.8.3.

An inclining rig was built, which consisted of a vertical wooden beam from which a plumb weight was suspended (Figure 3.8.1). The wooden beam was fixed at its base to a rectangular box filled with oil, designed to dampen the motions of the pendulum. The length of the pendulum was 0.799m; this was

considered a satisfactory length as 4-6 m is recommended for a full scale ship [F.Morishita, 1996-1997]. The length and width of the box were 0.07m and 0.54m respectively. A metal ruler was fixed to the side of the box in order to allow the positions of the pendulum to be measured consistently. The rig was then secured temporarily to the hull, at midships, using strips of Velcro. A singular pendulum was used due to the size of the model; to reduce any chance of error the results were plotted during the experiment to allow any anomalies to be identified.

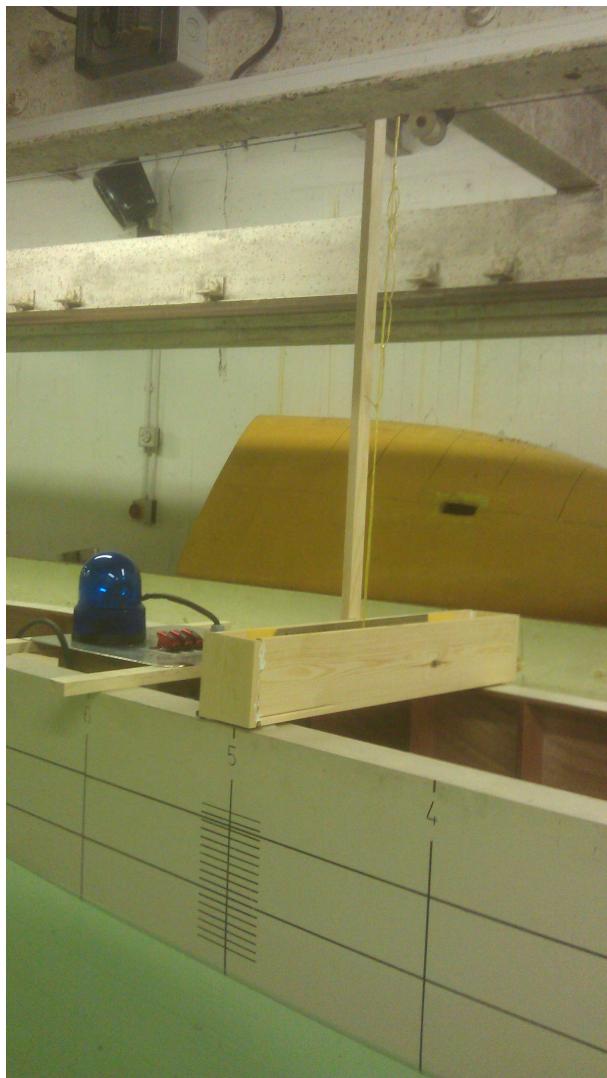


Figure 3.8.1: The inclining rig.

The test weights consisted of four plastic coated 2.1 kg lead weights. The weights were initially positioned with two port and two starboard, as far outboard as possible, just aft of midships. The positions were marked to allow the weights to be replaced in the same positions. The port and starboard weights were placed directly opposite to each other, to prevent any change in LCG or trim during the experiment.

Prior to the experiment being conducted the draught at the draught marks were recorded by eye. The draughts at the forward marks, midship marks and aft marks were all found to be about 0.115 m, corresponding to a level trim. Additionally a spirit level was used to ensure that there was not an initial list, and that the pendulum measured zero deflection at this position.

During the experiment the inclining weights were shifted methodologically, as described in Table 3.8.4. The model was inclined first to port, then to starboard, and finally back to its initial position. After each weight shift the deflection of the pendulum was recorded.

From each pendulum deflection the heel angle was calculated as follows:

$$\tan\phi = \frac{y}{l} \quad (3.8.3)$$

where y = pendulum deflection (m), l = pendulum length (m), and ϕ = heel angle ($^{\circ}$).

A plot of the tangent of the heel angle against the heeling moment (Fig 3.8.2) was created. A linear

Table 3.8.3: Table of Weights for Inclining Experiment.

Item	Weight (kg)	VCG (m)
Hull	29.29	Unknown
Motor	0.69	0.070
Shaft and bearings	0.75	0.064
Electronics Box	2.77	0.444
Thrust Block	0.42	0.053
Cabling	1.05	0.310
Propeller	0.06	0.060
Battery	13.34	0.137
Battery	13.38	0.137
Inclining Rig	2.57	0.423
Small Plastic Coated Weight (Inclining Weight)	2.10	0.030
Small Plastic Coated Weight (Inclining Weight)	2.10	0.030
Small Plastic Coated Weight (Inclining Weight)	2.10	0.030
Small Plastic Coated Weight (Inclining Weight)	2.10	0.030
Small Plastic Coated Weight	2.10	0.030
Small Plastic Coated Weight	2.06	0.030
Small Plastic Coated Weight	2.04	0.030
Small Plastic Coated Weight	2.04	0.030
Small Plastic Coated Weight	2.10	0.274
Small Plastic Coated Weight	2.10	0.274
Large Plastic Coated Weight	4.93	0.252
Large Plastic Coated Weight	4.93	0.252
Large Plastic Coated Weight	4.93	0.287
Large Plastic Coated Weight	4.94	0.287
Small Lead Weight	2.33	0.040
Small Lead Weight	2.35	0.040
Small Lead Weight	2.35	0.040
Small Lead Weight	2.36	0.040
Steel Weight	5.08	0.045
Steel Weight	5.08	0.075
Steel Weight	5.1	0.105
Steel Weight	5.07	0.135
Small Steel Weight	1.03	0.155
Small Steel Weight	0.93	0.165
Small Lead Weight	2.41	0.266
Lead Block	4.68	0.274
1 kilo weight	1.00	0.279
Total	109.69	

regression analysis was used to fit a line to the plot, the slope of which was taken as the average of:

$$\frac{wd}{\tan\phi} \quad (3.8.4)$$

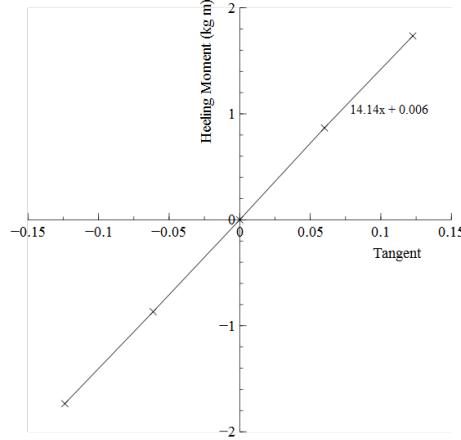


Figure 3.8.2: Plot of the tangent heel angle against the heeling moment for the ballast condition inclining experiment.

The transverse metacentric height (GM) was then calculated from the heeling moment, heel angle and displacement:

$$GM = \frac{wd}{\Delta \tan\phi} \quad (3.8.5)$$

A correction was made to the VCG in order to take into account the free surface effect of the oil in the inclining rig:

$$GM \text{ Correction} = \frac{\text{free surface moment}}{\text{displacement}} = \frac{\frac{lb^3}{12}\rho}{\Delta} \quad (3.8.6)$$

where l = oil box length (m), b = oil box breadth (m) and ρ = oil density (taken to be 0.91 tonne/m³).

The VCG of the model could then be calculated:

$$VCG_{model} = KM - GM \quad (3.8.7)$$

where the value of KM was interpolated from hydrostatic tables in the trim and stability booklet and taken to be 0.2521 m.

Finally the VCG of the hull was calculated as follows:

$$VCG_{hull} = \frac{VCG_{model} \cdot Displacement - \sum(Mass_{item} \cdot VCG_{item})}{Mass_{hull}}. \quad (3.8.8)$$

The results of the experiment are presented in Table 3.8.4.

The VCG of the hull was found to be 0.1765 m. The experiment was also repeated at full load. During the full load experiment a leak allowed some water into the bottom of the hull in the aft most compartment; the extra weight was estimated from the volume to be 2.16 kg. This water was also included in the free surface correction. The VCG of the hull was calculated to be 0.1770 m; which was an increase of 0.5 mm or 0.28%. The two results were averaged to obtain a hull VCG of 0.1768 m.

Table 3.8.4: Inclining Experiment Results.

Reading	1	2	3	4	5	6	7	8	9
Direction of Shifts	0	S →P	S →P	P →S	P →S	P →S	P →S	S →P	S →P
Pendulum Reading (m)	0.000	0.048	0.098	0.050	0.000	-0.049	-0.099	-0.051	0.000
Tan ϕ	0.000	0.060	0.123	0.063	0.000	-0.061	-0.124	-0.064	0.000
Shifting Weight (kg)	2.10	2.10	2.10	2.10	2.10	2.10	2.10	2.10	2.10
Heeling Moment (kg m)	0.000	0.867	1.735	0.867	0.000	-0.867	-1.735	-0.867	0.000
Displacement (kg)						144.660			
Tan ϕ /Heeling Moment						14.140			
GM (m)						0.097			
Free surface moment (kg m)						0.836			
Free surface correction (m)						0.006			
GM Corrected (m)						0.091			
VCG_Model (m)						0.161			
VCG_Hull (m)						0.177			

Although the results support each other there were areas of the experiment which could have introduced errors. The draught was recorded by eye from the draught marks which were in 1cm increments. This made it difficult to obtain precise measurements. Although oil was used to dampen the motions of the pendulum, some movement was inevitable; although the repeatability of the results inferred confidence. The amount of water that leaked into the hull during the fully loaded experiment was an estimation. Furthermore the results relied heavily on the KM values interpolated from the trim and stability booklet. This was considered to be the greatest source of error; as discussed previously the model's geometry was not considered to be an exact replica of the full scale hull. Overall, however, the results were considered to be of much greater accuracy than the SolidWorks prediction and the weight positions were adjusted accordingly.

3.8.3 Trim Conditions

[NEALE]

The model was to be tested at different trims in order to investigate the effects on efficiency. A loading computer, supplied by BP, was used to explore realistic trims for the two loading conditions. The loading computer was based on the vessel 'British Security.' As discussed previously different ships within the class were known to differ from the trim and stability booklet; it was observed that the tank volumes stored in the loading computer differed from the expected volumes. However, the computer was considered to be acceptable to obtain an approximation of reasonable trims.

The loading condition was generated by specifying constant weights (such as crew effects and stores), the loads in the cargo tanks and loads in additional tanks including water ballast. Additionally the densities of the cargo oil and the water could be specified; these were set to 0.855 and 1.025 T/M³ respectively. The volume limitations of the cargo oil tanks were between 7 and 98%. The hydrostatics, intact stability, longitudinal strength and damage stability could then be assessed.

Each loading condition had to fulfil certain criteria. The propeller emergence was not to be greater than 10% of the propeller diameter. The draught forwards was not to be less than 5.9 m with No.1 and 2 ballast tanks full; it was not to be less than 6.6 m forward with No.1 and 2 ballast tanks empty. Additionally regulations regarding the shear force, bending moment, intact stability and damage stability had to be fulfilled.

Water ballast was shifted in each loading condition until the criteria limits were reached. The full scale trim limits are presented in Table 3.8.5.

The trims chosen to test at were -1.62 m (default), 0 m and -3 m in the loaded condition, and 0 m

Table 3.8.5: Full scale trimming limits obtained from the loading computer.

	Max. Trim by the Stern (m)	Max. Trim by the Bow (m)
Ballast	-1.796	1.881
Loaded	-8.39	2.275

(default) and -2 m in the ballast condition. For practicality these trims were converted to model scale and the draughts at the draught marks were calculated using the principle of similar triangles:

Initially the trim angle for each condition was calculated:

$$\theta = \frac{\text{Trim}}{\text{LBP}} \quad (3.8.9)$$

where θ = trim angle.

Next the draught at the LCF was calculated. This was found from the mean draught obtained from the trim and stability book for the default condition as follows:

$$T_{LCF} = T_{mean} - LCF \cdot \tan(\theta) \quad (3.8.10)$$

where T_{LCF} = draught at the LCF, T_{mean} = draught at midships, and LCF is the position of the LCF in relation to midships. As this was the point the model would trim around it was assumed to remain constant for each trim condition. The draughts at the forward and aft perpendiculars were calculated:

$$T_{FP} = T_{LCF} + (LBP/2 + LCF) \cdot \tan(\theta) \quad (3.8.11)$$

$$T_{AP} = T_{LCF} - (LBP/2 - LCF) \cdot \tan(\theta) \quad (3.8.12)$$

These were converted to the draughts at the draughts marks using the method described previously. Lastly the mean draught for each trim condition was calculated:

$$T_{Mean} = T_{LCF} + LCF \cdot \tan(\theta) \quad (3.8.13)$$

The draughts for each trim condition are presented in Table 3.8.6:

Table 3.8.6: Trim Condition Particulars

	Loaded Depature		Ballast Arrival		
	Default Trim	Trim 1	Trim 2	Default Trim	Trim 1
Full Scale Trim (m)	-1.620	-3.000	0.000	0.000	-2.000
Model Trim (m)	-0.027	-0.050	0.000	0.000	-0.033
Trim Angle (degrees)	-0.533	-0.988	0.000	0.000	-0.659
T_{LCF} (m)	0.183	0.183	0.183	0.115	0.115
T_{FP} (m)	0.170	0.159	0.183	0.115	0.098
T_{AP} (m)	0.197	0.209	0.183	0.115	0.131
T_{Fmarks} (m)	0.173	0.164	0.183	0.115	0.101
T_{Amarks} (m)	0.194	0.204	0.183	0.115	0.128
T_{mean} (m)	0.184	0.184	0.183	0.115	0.114

3.8.4 Weight Positioning

[NEALE]

Once the required displacement, VCG, LCG and trims for the two loading conditions had been obtained, the ballast weights could be positioned. As described later in the design and build section, two weight platforms were built inside the model. This allowed weights to be positioned higher in the model in order to obtain the correct VCG; additionally it provided an accessible open space allowing for the movement of weights in order to obtain the required trims. Prior to the positioning of the weights a lightship survey had been conducted (see Figure 3.8.8). Weights were then positioned to achieve the required properties. Where possible, weights used in both the ballast and full load cases were kept in the same positions in order to reduced weight movements between different loading conditions. The weight positions for the ballast and full load conditions are presented in Tables 3.8.9 and 3.8.10 respectively. The final model scale loading conditions are presented in table 3.8.7. Although there are some differences between the required and actual conditions, they were considered small enough to have little effect on the scaled results.

Table 3.8.7: Final model scale loading conditions.

	Actual			Required		
	Displacement (kg)	LCG (m)	VCG (m)	Displacement (kg)	LCG (m)	VCG (m)
Ballast	135.534	0.111	0.135	134.47	0.106	0.132
Loaded	226.674	0.041	0.182	225.696	0.040	0.185

Table 3.8.8: Lightship Weights Table.

Item	Weight (kg)	LCG (m)	VCG (m)
Hull inc chassis	27.50	0.28	0.177
Rudder	0.18	-0.04	0.120
Rudder Stock	0.12	-1.45	0.180
Propeller	0.06	-1.40	0.060
Motor	0.69	-0.44	0.070
Dynamometer	0.69	-0.81	0.068
Shaft and bearings	0.75	-0.98	0.064
Electronics Box	2.77	-0.43	0.444
Thrust Block	0.42	-1.03	0.053
Rudder servo	0.25	-1.40	0.350
Battery	13.34	0.80	0.137
Battery	13.38	0.80	0.137
Battery	13.38	0.20	0.102
Battery	13.38	0.20	0.102
Cover	7.63	-0.04	0.334
Bilge Pump	0.20	-0.06	0.050
Cabling	1.05	-0.43	0.310
Towing Tank Connector	0.20	0.04	0.058
Laptop	1.00	0.05	0.000
Acrylic	1.07	-0.45	0.120
Total	98.06		

3.8.5 Moment of Inertia

[NEALE]

For sea-keeping analysis the model's moment of inertia would need to be obtained. Although the moment of inertia was not calculated during this study, the method to obtain it was investigated.

To obtain the pitch moment of inertia the model would be suspended from a frame. It would then be

Table 3.8.9: Weight Positions - Ballast Condition.

Item	Weight (kg)	LCG (m)	VCG (m)
Small Plastic Coated Weight	2.10	-0.25	0.030
Small Plastic Coated Weight	2.10	-0.25	0.030
Small Plastic Coated Weight	2.10	-0.35	0.030
Small Plastic Coated Weight	2.10	-0.35	0.030
Small Plastic Coated Weight	2.10	-0.45	0.030
Small Plastic Coated Weight	2.10	-0.45	0.030
Small Plastic Coated Weight	2.10	-0.55	0.030
Small Plastic Coated Weight	2.06	-0.55	0.030
Small Plastic Coated Weight	2.04	-0.65	0.030
Small Plastic Coated Weight	2.04	-0.65	0.030
Large Plastic Coated Weight	4.93	-0.15	0.028
Large Plastic Coated Weight	4.93	-0.15	0.028
Large Plastic Coated Weight	4.98	-0.80	0.252
Total	35.68		

oscillated, with T , the time taken to complete a specified number of oscillations measured [Lloyd, 1998]. From this the radius of gyration k_5 could be calculated:

$$k_5 = \sqrt{\frac{(mh + m_F h_F)gT^2}{4\pi^2 m} - h^2 - \frac{I_F}{m}} \quad (3.8.14)$$

where m = model mass, m_F = mass of frame, h = distance from centre of gravity of model to pivot point, h_F = distance from centre of gravity of frame to pivot point and I_F = moment of inertia of the frame (obtained by oscillating the frame without the model attached)

Therefore the moment of inertia, using the parallel axis theorem, would be:

$$I = mk_5^2 + mh^2 \quad (3.8.15)$$

The roll radius of gyration would be found in a similar manner. The yaw moment of inertia k_6 may be obtained by suspending the model from two wires at the bow and stern. The yaw would be calculated [Lloyd, 1998]:

$$k_6 = \frac{T x_R}{2\pi} \sqrt{\frac{g}{h}} \quad (3.8.16)$$

where x_R = the longitudinal distance between the centre of gravity of the model and the points of suspension.

The moment of inertia would then be:

$$I = mk_6^2 \quad (3.8.17)$$

The required moment of inertia may be scaled from the full scale ship as follows:

$$I_M = \frac{I_S}{R^5} \quad (3.8.18)$$

where I_M = model scale moment of inertia, I_S = full scale moment of inertia, R = scale factor (in this case 60)

With the required moment of inertia known, the weights could be re-positioned equally closer or further away from the centre of gravity until the correct moment of inertia was achieved. Unfortunately the full scale moment of inertia was unobtainable which meant that accurate scaling was not a possibility.

Table 3.8.10: Weight Positions - Full Load.

Item	Weight (kg)	LCG (m)	VCG (m)
Lead Roll	0.00	0.775	0.294
Lead Roll	6.46	0.775	0.294
Lead Roll	6.57	0.775	0.294
Lead Roll	6.60	0.775	0.294
Lead Roll	6.40	0.625	0.294
Lead Roll	6.51	0.625	0.294
Small Lead Weight	2.41	0.95	0.269
Small Lead Weight	2.33	0.95	0.269
Small Lead Weight	2.35	-0.25	0.03
Small Lead Weight	2.35	-0.25	0.03
Small Lead Weight	2.36	-0.75	0.03
Small Lead Weight	2.28	-0.75	0.03
Small Plastic Coated Weight	2.10	0.55	0.07
Small Plastic Coated Weight	2.10	0.55	0.07
Small Plastic Coated Weight	2.10	0.55	0.03
Small Plastic Coated Weight	2.10	0.55	0.03
Small Plastic Coated Weight	2.10	-0.35	0.03
Small Plastic Coated Weight	2.10	-0.35	0.03
Small Plastic Coated Weight	2.10	-0.55	0.07
Small Plastic Coated Weight	2.06	-0.55	0.07
Small Plastic Coated Weight	2.04	-0.55	0.03
Small Plastic Coated Weight	2.04	-0.55	0.03
Large Plastic Coated Weight	4.93	-0.15	0.0275
Large Plastic Coated Weight	4.96	-0.15	0.0275
Large Plastic Coated Weight	4.93	-0.925	0.2515
Large Plastic Coated Weight	4.93	-0.925	0.2515
Large Plastic Coated Weight	4.93	-1.075	0.2515
Large Plastic Coated Weight	4.93	-1.075	0.2515
Large Plastic Coated Weight	4.93	-1.225	0.2515
Large Plastic Coated Weight	4.94	-1.225	0.2515
Large Plastic Coated Weight	4.98	-0.8	0.2515
Large Plastic Coated Weight	4.98	-0.8	0.2515
Large Plastic Coated Weight	4.96	-0.8	0.2865
Large Plastic Coated Weight	4.96	-0.8	0.2865
Total	126.82		

WAVE BUOYS

The need for a means of measuring the conditions of the test environment has been identified. This chapter of the report focuses on the design, build and testing of a suitable device. Although the wave buoy and the model were not finalized in time to be used in parallel, the buoy has been tested independently. This chapter begins with an introduction to the methodology; followed by the design and build of the buoys and their hardware; the data processing of the collected sensor values; and finally the results - compared against a fixed wave probe and against specialist software.

4.1 Introduction

[THORNTON]

The requirement and ability to measure ocean conditions around the world is used in a wide range of applications such as global sea forecasts, engineering design of ships and structures and coastal management. As a result, gaining knowledge of the constantly shifting sea states has led to the growing commercial demand for various methods to be deployed for measuring these ocean waves. According to Stewart [2009] the more commonly used techniques include observations made at sea, satellite altimeters, accelerometers mounted on buoys, wave gauge spatial arrays and synthetic aperture radars (SAR) on satellites. Other techniques include the use of seismometers, upward facing acoustic Doppler current profilers (ADCP) [Terray et al., 1999], pressure transducers [Bishop and M.A.Donelan, 1987] and various radars such as wave radars on ships and structures, and scanning-beam microwave radars on aircraft and satellites [P.L.Baker, 1985].

Wind-generated ocean waves consist of wave contributions spreading in different directions with varying wavelengths, and are described using directional wave spectra. An example of a directional spectrum is presented in Figure 4.1.1. These spectra are the key to wave modelling, allowing the consequences of interactions between the waves and other structures to be calculated, be it the response of a ship or forces on an offshore structure [Hauser et al., 2003]. During open water model tests, the ability to measure and scale the waves interacting with the model is important for determining the full-scale responses of the ship during various manoeuvres or conditions. With the emphasis placed on the importance of being able to scale the environment in open water model tests, it was decided that possibilities for measuring these small-scale waves were to be explored.

This project models a full scale ship that operates the majority of its time in the North Atlantic during Beaufort scale 4-6 conditions. Table 4.1.1 [MET Office, 2013] shows the full scale wave conditions with the corresponding scaled-down waves that the model should be tested in. Data for mean zero crossing periods measured by the North Atlantic weather station *India* was used from Draper [1986] to determine the desired model scale periods. Mean zero crossing periods of 5.5-13.5 seconds were measured, pertaining to model scale wave periods of 0.71-1.74 seconds, based on Froude scaling [Lloyd, 1998]. As a result, the ability to accurately measure waves in the range of 1-10 cm height and frequency of 0.5-1.7 Hz is necessary, based on a 1/60 scale.

Limited by a relatively small budget and time-frame, and after assessing different methods (Section 1.5.5), the most feasible method for measuring waves on this scale in various open water environments would be to implement a surface contouring wave buoy. Mounted with a 9 degree of freedom (DOF)

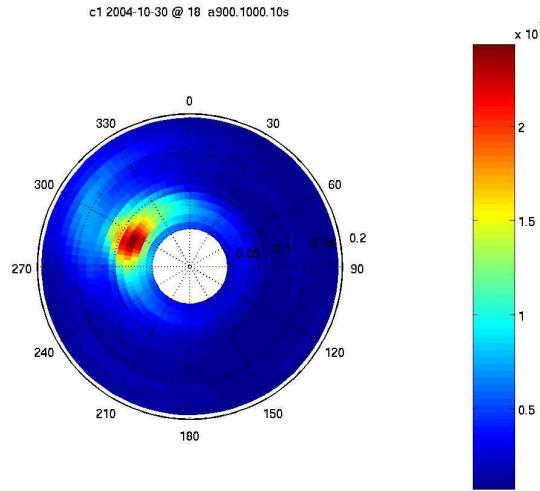


Figure 4.1.1: Example directional wave spectrum [Argus Beach Monitoring Stations].

Table 4.1.1: Desired wave heights for model-scale testing based on the Beaufort scale.

Beaufort Number	Wind Speed m/s	Probable Wave Height m	Max Wave Height m	Model Prob. Wave Height cm	Model Max. Wave Height cm
4	5.5-7.9	1.0	1.5	1.67	2.50
5	8.0-10.7	2.0	2.5	3.33	4.17
6	10.8-13.8	3.0	4.0	5.00	6.67
7	13.9-17.1	4.0	5.5	6.67	9.17

sensor pack, the buoy would be capable of measuring heave, pitch and roll responses to determine the directional wave spectrum.

By developing a system that uses small, surface-contouring wave buoys it allows for easy set-up in almost every location, with the only varying factor being the mooring length (if used). Alternatively, the buoy could be launched upwind of the test site, and allowed to drift freely downwind. It would be a small, lightweight and easily transportable solution that could acquire and transmit data simultaneously to the shore for real-time analysis.

A number of commercially offered wave buoy solutions were discussed in Section 1.5.5, such as the Wavetector, Triaxys and Datawell. However, a major limiting factor of these was the price, which greatly exceeded the available budget. It was apparent that none of the existing commercial wave buoys would be suitable to measure the small-scale waves needed during model test runs, as all are designed to be deployed in the open ocean. With these factors in mind, it was clear that there was a need to develop a wave buoy with the capability of measuring the small-scale sea states during the open water experiments.

Initial plans were to develop and test a suitable wave buoy design, then produce a number of them (approximately 4) in order to set up a grid in which the ship's tests would be conducted. It became clear, however, that this was not actually necessary and would introduce further complications when analysing the wave data, as well as greater costs. By developing a single wave buoy, focus could be aimed on improving the quality and accuracy of data acquired by spending more time and money on the construction and sensors used.

4.2 Design

[HAWKES, NEALE & THORNTON]

The wave buoy design consisted of three main areas: the selection of the sensor, the development of an electronics system to process or log the sensor data, and the design and construction of the wave buoy hull which would carry the sensor.

4.2.1 Sensor Requirements

[NEALE]

A basic wave survey of a local lake was conducted through visual observation in order to obtain an approximation of likely wave conditions. An average frequency of 1.43 Hz and an average amplitude 0.03 m was observed.

A 9-degree-of-freedom (9-DOF) inertial measurement unit (IMU) was used to measure the motions of the wave buoy. The sensor, an ATAVRSBIN2, comprised of a tri-axis micro-electro-mechanical system (MEMS) accelerometer (KXTF9-1026), a tri-axis MEMS gyroscope (IMU-3000) and a tri-axis magnetometer (HMC5883L). The accelerometer measured the linear accelerations of the buoy, which could be integrated to obtain heave motions. The gyroscope was used to measure angular accelerations which could be integrated to obtain pitch and roll. The purpose of the magnetometer was to correct for the gyroscope drift (cumulative integration error) and to provide a reference heading for the buoy. Magnetometers were not fast enough to capture these motions by themselves

The accelerometer incorporated user selectable ranges of $\pm 2g$, $\pm 4g$, or $\pm 8g$; the gyroscope had selectable ranges of ± 250 , ± 500 , ± 1000 or ± 2000 degrees per second; and the magnetometer had ranges between $\pm 0.9\text{Ga}$ and $\pm 7.9\text{Ga}$. Since the accelerations due to the waves were not accurately known, the variety of ranges ensured that sensitivity could be matched during experimentation.

For reference, the suitable range settings were $\pm 2g$, $\pm 500^\circ/\text{second}$ and $\pm 0.9\text{Ga}$.

4.2.2 Electronics

[HAWKES]

Asides from the sensor, a number of other features were added to the wave buoy - including GPS, a microSD card, long-range radio communications, short-range bluetooth communications, a lithium-ion battery and various status LEDs.

The whole system was operated by an MCU - another Arduino Pro Mini, but this time the 8Mhz/3.3V flavour. This voltage was chosen as the principal voltage for the system because the sensor ran at this voltage. Power was supplied from a 3.7 V lithium-ion battery via the MCUs on-board 3.3 V voltage regulator. The sensor connected to the MCU using the I2C protocol.

A 1 GB microSD card was connected to the MCU using the SPI bus and was originally used for data logging, with the intention that when the buoy was retrieved the data could be transmitted via bluetooth. However, the microSD card was plagued with problems which often corrupted the data. Furthermore, it took 29 ms to write each sensor measurement to the card which limited the sample rate. With some more time both of these problems should be resolvable, and the microSD card would then be useful again.

The radio and Bluetooth modules created a wireless serial link between a radio unit connected to a shore computer, or any device fitted with Bluetooth, respectively. They were connected to the serial receive (RX) and transmit (TX) lines appropriately, with a switch to toggle between Bluetooth and radio communications. Jumper switches were also provided to disconnect both units, to prevent interference when trying to program the MCU in the conventional wired fashion.

The chosen radio module was a high-power 2.4 GHz XBee module, with an external antenna for additional range. The receiving module connected to the shore computer was identical, and used an

XBee-USB adapter to connect to the computer. Every effort was made to maximize the range of the buoy transmission. Line of sight was required for the radio communications, but this was not a problem when testing on water.

To avoid using the microSD card the serial communication rate was pushed to the limit, which was found to be a baud rate of 57600 (7.2 KB/second). At this speed the serial transmission of each sensor reading took approximately 10 ms.

At this communication speed, the range of the radio units were tested. The test site was too small to notice any problems in communication, despite being over 1km long.

The Bluetooth module was a ‘Bluetooth Mate Silver’ which advertised a range of approximately 20 m. The Bluetooth module was tested at twice the baud rate of the radio module, although it should be able to go higher with some updated firmware. The performance of the radio module had been drastically underestimated - initially it was only to be used to send commands to the buoy such as ‘start logging’ and ‘stop logging’. As such, the Bluetooth brought only a small improvement to the communications rate, by cutting the transmission time from 10 ms to 5 ms.

The GPS was an ‘Ultimate GPS Breakout Board’ from Sparkfun, connected to the MCU via two input/output pins for serial communications (RX & TX) and one digital pin that could toggle the GPS in and out of standby mode. The GPS was used to record the location of the wave buoy, and could also provide a reference time for the data logging. This extra information was optional, since the time was also recorded on the shore-computer, but would be useful if the buoy was ever operating independently (*i.e.* out of range, and logging to the SD card). Ironically, the GPS would only work when the wireless communications were idle or disabled, since the interference caused instantly destroyed the satellite link to the GPS.

A successful compromise was to allow the GPS to get a satellite fix and then report to the shore computer when it had located itself. Since the buoy would not move significantly during the tests, the GPS could be switched off while data was transmitted. In the end, this procedure was abandoned for simplicity, and the location was not recorded - however, the GPS could certainly be used in the future.

Three status LEDs were also used - one connected to the MCU which could be turned on and off from the code; one connected to the status indicator of the communications modules; and one connected to the GPS status indicator. Each LED used an 83R resistor to limit the current to the LEDs.

At the early stages of the design, the enclosure for the electronics was still in question - but it was expected that the electronics would be out of sight. Therefore, a piezo buzzer was also implemented that could be controlled from the MCU as an additional status indicator.

The complete set of hardware for the wave buoy is shown in Figure 4.2.1.

Unlike the electronics for the ship, which had no constraints over size and weight, the wave buoy electronics had to be as small as possible to fit on the buoy. The aim was to fit the system into a watertight junction box with a footprint of approximately 10cm×10cm. In order to do this it was necessary to construct the wave buoy on a printed circuit board (PCB).

The PCB was designed with components on both sides, to keep the system as small as possible. Items were also stacked by placing components such as the sensor on top of the MCU. Figure 4.2.2 shows the component silk screen, the top-side routing, bottom-side routing and completed PCB.

The complete electronics unit was mounted in an IP68-rated junction box with a clear plastic cover - allowing visual inspection of the electronics and status lights. Images of the final product are shown in Figure 4.2.3. The electronics were held in place using polymorph, a thermoplastic which becomes pliable at about 60 degrees celsius, but sets hard. The polymorph was heated using boiling water and the electronics system was pressed into it. This prevented the electronics, particularly the sensor, from

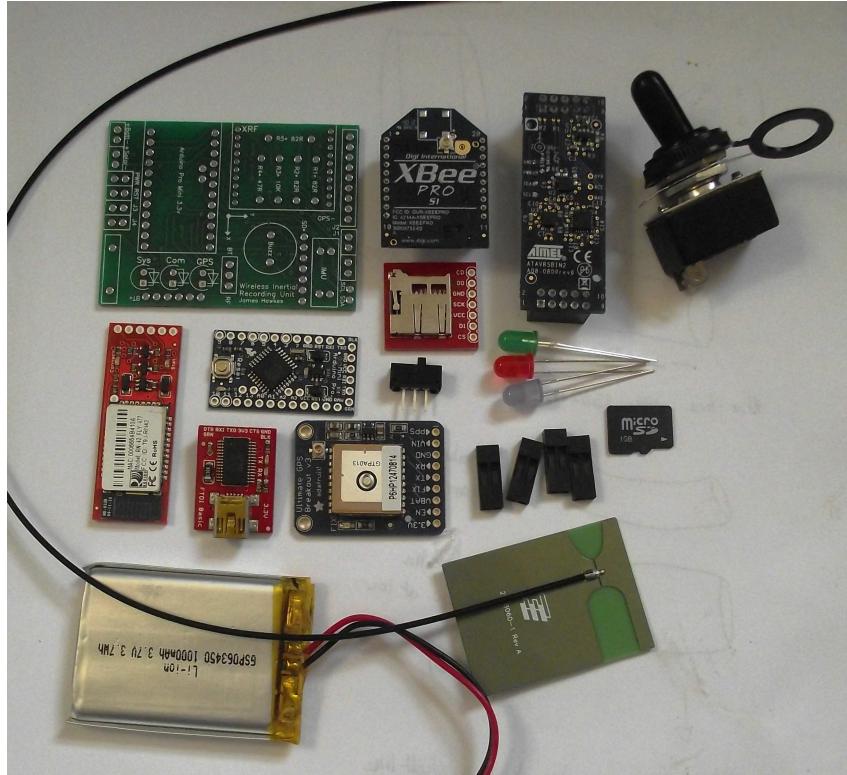


Figure 4.2.1: All the electrical components which formed the wave buoy.

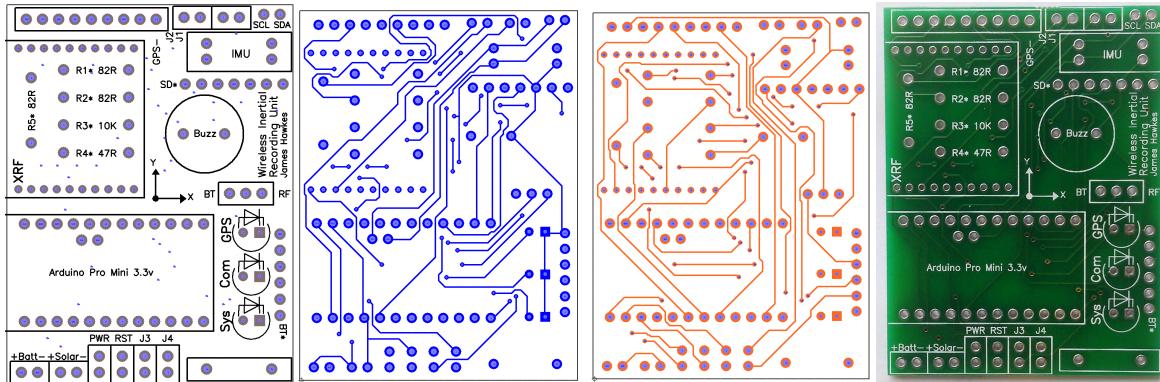


Figure 4.2.2: The PCB design, showing the silk screen, top-side routing, bottom-side routing and completed PCB.

moving about.

Overall, the additional funding for the wave buoy allowed for a neat construction onto a PCB, and the use of very powerful wireless modules. There were a number of software and interference issues which could be rectified in the future, to make full use of the system.

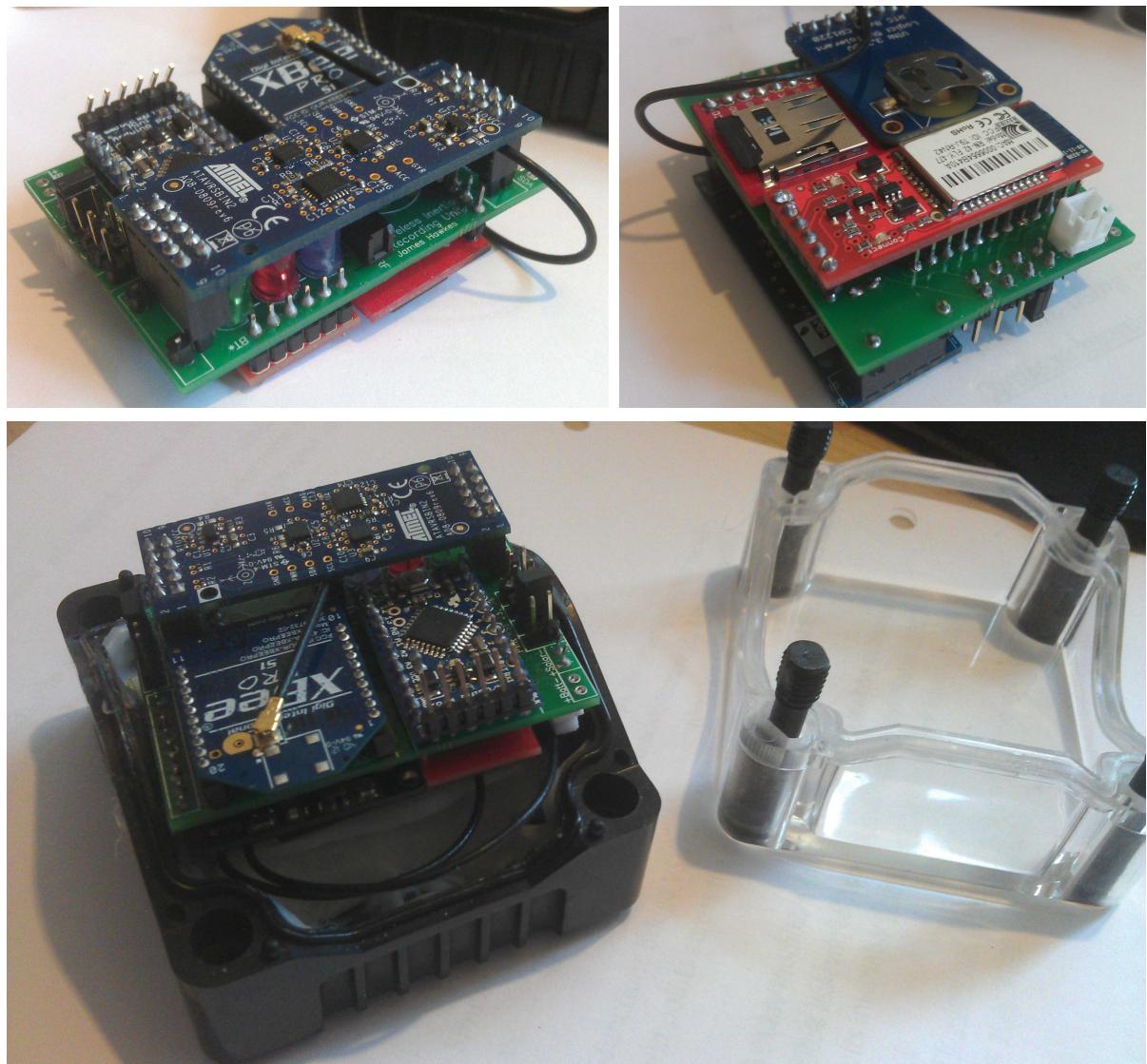


Figure 4.2.3: The complete wave buoy electronics module. The polymorph is just visible under the electronics unit in the lower picture.

4.2.3 Buoys

[THORNTON]

Based on preliminary findings, it was decided that proceeding with the development and construction of a single wave buoy system was the most suitable technique to implement for wave measurements during open water model tests.

Berteaux [1976] depicts a schematic representation of the thought process and phases behind designing a buoy system, shown in figure 4.2.4.

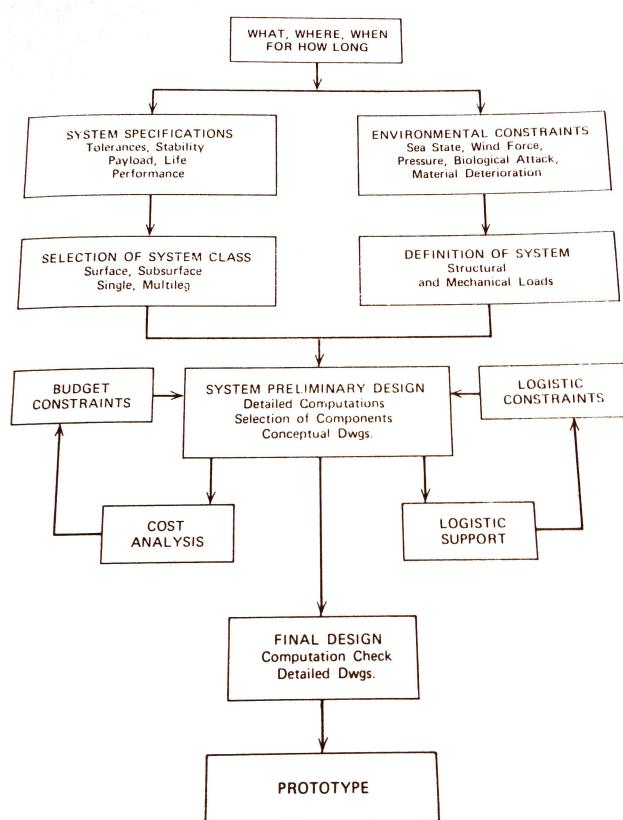


Figure 4.2.4: Buoy design process. [Berteaux, 1976].

As previously stated, surface contouring buoys have a tendency to closely follow the waves and hence would serve as an ideal platform for an instrumented wave buoy. The desired waves for open water model tests were to be in the order of 1-10cm significant wave height, with a mean zero crossing period of roughly 0.7-1.1 seconds. In order to accurately contour the wave surface, the buoys' diameter should be significantly less than the wavelength – about $\frac{1}{4}$ to $\frac{1}{3}$ of the wavelength. Due to practical limitations, in particular the sensors' weight and dimensions, this was not completely realistic, and as a result it was decided to construct a range of prototype wave buoys with varying diameters and shapes to test the response of each in varying seas.

A discus shape was opted for, as previous experiments [Sundar et al., 2008] using this set-up had proven successful in gaining accurate measurements in similar waves to those desired. The buoys' response in waves is largely affected by the underwater profile, as with any other floating structure, hence 2 different hull shapes were to be investigated – a curved edge v. a straight edge with a horizontal level base. Furthermore, it was decided that 2 different hull diameters were to be investigated, allowing for comparisons to be made between buoy diameter and wave profile (shown in figure 4.2.5).

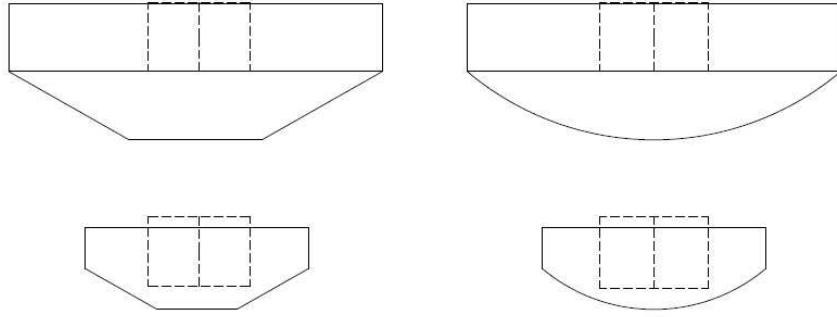


Figure 4.2.5: CAD profiles of buoy hulls.

The dimensions of the larger diameter buoys were based on those found in [Sundar et al., 2008], with a diameter of 275 mm. The smaller buoys were scaled down from the larger designs, with the limiting factor being the sensor package. A direct scaling approach was chosen so as to allow for through comparisons between the 2 diameters, as the profile beneath the waterline remained the same. As the sensor package had not been finalised by this point, assumptions were made that the unit would weigh 500-600g, and an iterative approach was taken in carrying out displacement calculations for the buoys at their desired waterline. Based on assumptions for the sensor units' dimensions and weight, the smallest diameter for the small buoy hull, whilst remaining at the desired waterline, was 165 mm. In reality the sensor unit weighed closer to half our estimation due to changes in decisions such as battery choice; however the difference in design due to this would have been minimal so no alterations were made. Tables 4.2.1 and 4.2.2 show the total weight and dimensions of the buoys, and the weight breakdown respectively.

Table 4.2.1: Buoy weights and dimensions.

Buoy	Weight (g)	Diameter (mm)		Depth (mm)*	
		Top	Bottom	Top	Bottom
Big - Curved (BC)	949.39	275	0	50	50
Big - Straight (BS)	953.23	275	98.75	50	50
Small - Curved (SC)	253.77	165	0	30	30
Small - Straight (SS)	252.53	165	59.25	30	30

*‘Top’ refers to cylindrical upper section, ‘Bottom’ refers to base of shaped lower region

Table 4.2.2: Breakdown of component weights.

Component	Weight (g)
Ballast	545.7
Sensors	219.91
BC Styrofoam	183.78
BS Styrofoam	187.62
SC Styrofoam	33.86
SS Styrofoam	32.62

A key feature of the design was accessibility to the sensor unit. This was due to the decision being made to build one unit that could be easily transferred between buoy prototypes. As a result, the buoys are designed with the top open, so as to allow for the sensor unit to be easily fitted in the center. This also allowed for the transferral of the ballast between the larger buoys, providing the need for a single weight cut from a 10 mm thick steel plate.

As the 4 buoys were intended as prototypes to be tested in controlled environments such as a towing

tank or over short time periods in a lake, they were constructed out of blue Styrofoam. This material was chosen as it was relatively cheap, would provide ample buoyancy and is easy to shape with the use of simple hand tools. It is, however, very susceptible to damage from knocks and bumps so care had to be taken when handling or storing. Cutting the initial buoy shapes was done using a CAD driven 2D hot-wire cutter, with finishes being made using fine grit wet-dry sand paper. Due to the thickness of available Styrofoam sheets, the large buoys were cut from 2 sections and attached using 2-part epoxy. The smaller buoys were cut from a single piece of Styrofoam. Due to the closed cell nature of Styrofoam, and the short times used during testing, water ingress was not deemed to be a major issue. However, the finished buoys were coated in GFRP grade epoxy resin to ensure this was minimised, as well as improve durability.

The design concepts were modelled in OrcaFlex, a leading program for the dynamic analysis of offshore marine systems, for validation and to provide a basis for assessment in a range of sea states. This was done as it allowed various guaranteed sea states to be modelled in a controlled environment, such as extreme cases that may rarely occur in reality, and the buoy responses could be accurately measured in each to examine the design feasibility. It also allowed for different mooring line systems to be implemented, assessing the coupling effects that these had on the buoys' responses in different conditions. Results such as the mooring line tension can be displayed, which is a quantity that would prove extremely difficult to accurately obtain during tests in reality. Design choices could then be made with a degree of certainty and measurements taken from the actual buoy responses could be validated against the model results.

One such design decision was the choice of mooring line implemented. OrcaFlex was used to model 5 different mooring line arrangements, arriving at the decision to use the arrangement as seen in 4.2.6. Mooring systems that were modelled included: straight 2.5 mm diameter, nylon line; continuous line of 0.4 m³ volume floats; and 3 catenary-styles with various combinations of nylon line, floats and chain. Mooring systems were modelled in waves of 10 cm amplitude and 0.7 s zero crossing period to simulate more extreme possible conditions. The straight mooring line and final catenary arrangement proved successful, however the other 3 systems failed and resulted in the buoy capsizing due to sudden 'jerks' as the mooring became taut. As a result, both the straight line and catenary systems were explored further, the results of which are shown in Section 4.4.

The mooring arrangement chosen for tank testing was based on the successful catenary system in OrcaFlex as this provided a gentle damping effect whilst still allowing an ample scope for the buoy. This can be seen in Figure 4.2.7 along with the four hulls. Due to the small forces the buoy would be subject to 35 lb test monofilament fishing line was chosen as the basis of the mooring line, with the addition of 0.5 m of small zinc coated chain links above the float to provide some damping. A small float key-ring was used for buoyancy to hold the line off the bottom, and also to contribute to the damping effect during more extreme cases where the mooring line became taut, as its natural response would be to resume its vertical position. A 2 kg lead dive belt weight was ample mass for the anchor. This arrangement could easily be adapted for any depth by simply lengthening or shortening the fishing line sections. The mooring line was attached to the hull via a carabiner to allow for easy transitions between buoys without retrieving the mooring line. A small metal loop-eye was secured into the base of each buoy using 2 part epoxy (seen in Figure 4.2.7). Due to their prototype nature and the small forces they were predicted to experience, this was a suitable solution.

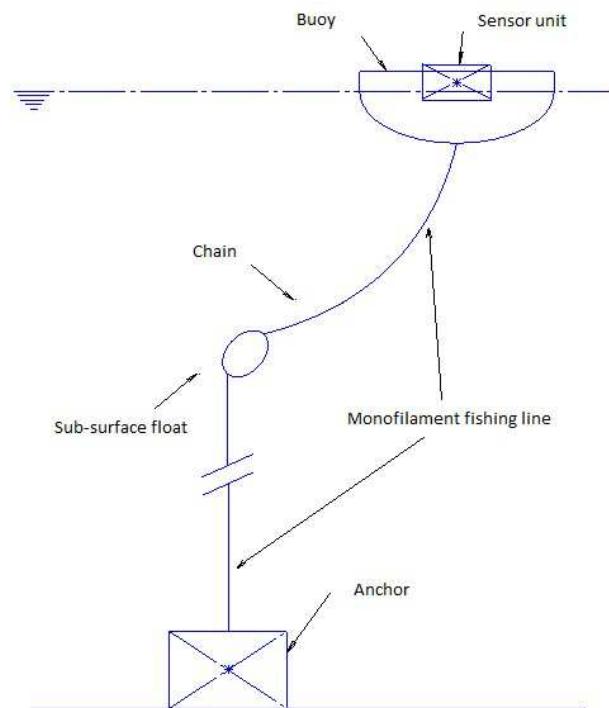


Figure 4.2.6: Mooring system for the buoys, showing the catenary arrangement.



Figure 4.2.7: [Left] Styrofoam buoy hulls. [Right] Buoy system in the Lamont tank where the catenary mooring arrangement and sensor unit are visible.

4.3 Data Processing

[NEALE & HAWKES]

This section covers the process of converting the raw sensor data into directional wave spectra. There are two parts to this process: firstly, the sensor data must be converted to a dataset of heave, pitch and roll motions. Secondly, this data is used to create a directional wave spectra by performing cross-spectral Fourier analysis.

Throughout this process, there is a vast amount of data to be handled involving huge 3D datasets, and it quickly became apparent that a computer program should be developed to perform the analysis. Also, in the first half of the data processing, which involved complex 3D filters, it was useful to assess the performance visually, using rotating 3D graphics.

The development of the software will be tackled at the end of this section.

4.3.1 IMU Processing

[HAWKES]

The IMU gives values for acceleration and magnetic field strength along the x -, y - and z -axis, as well as the angular velocity about these axis.

The data was sampled at a rate of 50 Hz (20 ms period) to ensure everything ran smoothly on the buoy and on the shore computer. The magnetometer and gyroscope could be considered almost noise-free, since the real-time plotting of the sensor values produced smooth curves. However, the accelerometer was very noisy, so a moving-average low-pass filter was added to the sensor data to remove this noise. A 5-point moving average was used, which produced the best results by visual observation.

This reduced the effective sample rate to 10 Hz. As such, the fastest motions that could be measured would be at a frequency of 5 Hz (by the Nyquist principle). This was acceptable for the wave buoy application, but the sample rate could be increased depending on the power of the shore computer.

Before these sensors can be used, a calibration process also has to be applied. The calibration for the gyroscope and accelerometer is simple. The sensor was laid flat and still and the zero-offset measured, and then subtracted from all future readings.

For the magnetometer, the process is less simple. For an ideal magnetometer which is rotated in three-dimensions (such that it covers as many 3D headings as possible) the resultant x -, y and z -values should scribe a perfect sphere centred on the origin [Renaudin et al., 2010].

However, there are two forms of deformation that often occur. The first is hard iron deformation, caused by magnetic flux distortions which retain a constant position (distance and orientation) relative to the magnetometer. In this case, hard iron deformation could arise due to electrical noise or ferrous materials on the buoy - such as steel ballast weights. Hard iron calibration should be performed for each buoy or mounting system.

The other form of deformation is soft iron deformation. This causes the ideal sphere to be squashed into a spheroid. Soft iron deformation arises due to external magnetic flux, whose orientation relative to the sensor is not constant - although the strength of (distance from) the distortion is assumed constant. Soft iron deformation is not usually corrected for, since most systems containing a magnetometer are mobile and sources of soft iron deformation move around too much. However, for devices such as the wave buoy, which rotates in space but mostly does not translate - soft iron calibration should be performed in every location they are used.

In most cases, the soft iron deformation should be small since the wave buoy will be floating far from any interference. However, it was also intended to test the wave buoy in the University of Southampton Lamont Towing Tank which was surrounded by large amounts of iron. The towing tank is notorious for

rendering magnetometers useless; however, since the buoy would be stationary inside the tank (albeit rotating), a hard iron calibration should be enough to maintain use of the magnetometer.

In order to find the equation of the ellipsoid, and hence correct for hard and soft iron deformation, 12 variables must be found, such that [Renaudin et al., 2010]:

$$\begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix} \begin{bmatrix} m_{x,raw} - b_x \\ m_{y,raw} - b_y \\ m_{z,raw} - b_z \end{bmatrix} \quad (4.3.1)$$

where the hard-iron calibration deformation was represented by b_x, b_y, b_z and the soft-iron calibration by the constants a_{1-9} . Note that the ellipsoid would not necessarily be aligned to the principal axis. Also note, that the radius of the ideal sphere represents the Earth's magnetic field strength, but since all magnetometer values were normalized there was no need to quantify this.

No library could be found that performed ellipsoid fitting in the chosen programming language (C++) so it was decided to transpose a similar code from MATLAB, using an open-source library of matrix transformation functions. The code was based on an eigenvalue method developed by the Qingde and Griffiths [2004].

Another option would have been to use a least-squares fitting technique as employed by other codes [Renaudin et al., 2010]. Unofficial sources have reported that the code's create almost identical data, so the method by Qingde and Griffiths [2004] was used, since it was easier to transpose.

Figure 4.3.1 shows an example of quite extreme hard- and soft-iron deformation. For reference, the calibration constants for the buoy in the towing tank are presented here:

$$\begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \begin{bmatrix} 4.72296 & 0.204366 & -0.205252 \\ 0.204366 & 5.00365 & 0.114621 \\ -0.205252 & 0.114621 & 5.47836 \end{bmatrix} \begin{bmatrix} m_{x,raw} + 0.0213084 \\ m_{y,raw} - 0.0547494 \\ m_{z,raw} - 0.027601 \end{bmatrix} \quad (4.3.2)$$

The scale is significantly different to the images in Figure 4.3.1, which were created from raw sensor values; but the principal is the same. There are significant components of all three raw sensor values ($m_{x,raw}, m_{y,raw}, m_{z,raw}$) appearing in the final values indicating that the calibration was very valuable, although the hard-iron (offset) calibration was negligible - probably because the buoy carried little in the way of ferrous material.

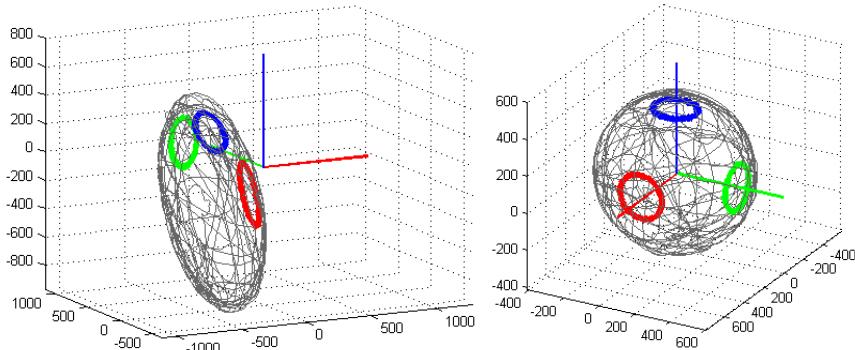


Figure 4.3.1: An example of quite extreme hard- and soft-iron deformation and the successful calibration by fitting the spheroid to a sphere centred on the origin. The scale is in terms of raw, digital sensor values. Images courtesy of Sebastian O.H. Madgwick.

After calibration, the processing of the sensor motions could begin. This closely followed a detailed report by Madgwick et al. [2011]; and the open-source sub-routines which perform the IMU calculations were also taken from Madgwick et al..

The normalized magnetometer values ${}^S\hat{m}_{t=0}$ (where t is the timestep and S refers to the sensor frame of reference) could be used to align the the buoy to a starting orientation, ${}^S_E\mathbf{q}_{t=0}$. Where S_E implies that the quaternion \mathbf{q} refers to the rotation of the sensor frame relative to the earth frame of reference. The entire process is performed in quaternion notation - quaternions are 4-dimensional complex numbers which describe the 3-dimensional orientation of an object.

There are two methods for updating the orientation as the timestep advances. The first is through integrating the angular velocity, where the angular velocity measurements ω_x , ω_y and ω_z are expressed in rad/s and form a vector ${}^S\bar{\omega} = [0, \omega_x, \omega_y, \omega_z]$. A quaternion derivative equal to the rate of change of the orientation quaternion can be calculated as (using the tensor product, \otimes):

$${}^S_E\dot{\mathbf{q}}_{\omega,t} = \frac{1}{2} {}^S_E\hat{\mathbf{q}} \otimes {}^S\bar{\omega}_t \quad (4.3.3)$$

which can be numerically integrated to find the new orientation, from the previous timestep, ${}^S_E\hat{\mathbf{q}}_{est,t-1}$.

$${}^S_E\mathbf{q}_{\omega,t} = {}^S_E\hat{\mathbf{q}}_{est,t-1} + {}^S_E\dot{\mathbf{q}}_{\omega,t} \Delta t \quad (4.3.4)$$

where the subscript ω indicates that this value was found from angular rates.

The second method, called the vector method, uses a gradient descent optimization to find the compound orientation from the magnetometer and accelerometer. The magnetometer can only provide orientation relative to the magnetic field lines at the sensor location, and cannot not discern rotation about the field line - which in the UK, has an inclination of about $\approx 70^\circ$. Thus, the accelerometer must be used to provide another direction vector (average gravitational force) which combines to provide the complete orientation. In this method, the orientation does not drift due to cumulative integration error, which the previous method suffered from.

The gradient descent optimization uses the following algorithm, which increments k until convergence is reached; μ is the optimization step size which would normally be calculated on a per-optimization-step basis.

$${}^S_E\mathbf{q}_{k+1} = {}^S_E\hat{\mathbf{q}}_k - \mu \frac{\nabla f}{||\nabla f||} \quad (4.3.5)$$

In this application it was possible to let each iteration become one timestep so long as $\mu = \mu_t$ was equal or greater to the physical rate of change of orientation. Subscript ∇ indicates the orientation was calculated using the vector method.

$${}^S_E\mathbf{q}_{\nabla,t} = {}^S_E\hat{\mathbf{q}}_{est,t-1} - \mu_t \frac{\nabla f}{||\nabla f||} \quad (4.3.6)$$

In both cases ∇f is the gradient function, defined as

$$\nabla f = \mathbf{J}_{g,b}^T({}^S_E\hat{\mathbf{q}}_{est,t-1}, {}^E\hat{\mathbf{b}})\mathbf{f}_{g,b}({}^S_E\hat{\mathbf{q}}_{est,t-1}, {}^S\hat{\mathbf{a}}, {}^E\hat{\mathbf{b}}, {}^S\hat{\mathbf{m}}). \quad (4.3.7)$$

The objective function $\mathbf{f}_{g,b}$ is a function of the previous time-step orientation quaternion ${}^S_E\hat{\mathbf{q}}_{est,t-1} = [q_1, q_2, q_3, q_4]$, the normalized accelerometer data ${}^S\hat{\mathbf{a}} = [0, a_x, a_y, a_z]$, the Earth's magnetic field vector

at the sensor location ${}^E\hat{\mathbf{b}} = [0, b_x, 0, b_z]$ and the magnetometer data ${}^S\hat{\mathbf{m}} = [0, m_x, m_y, m_z]$. Thus

$$\mathbf{f}_{g,b}({}^S_E\hat{\mathbf{q}}_{est,t-1}, {}^S\hat{\mathbf{a}}, {}^E\hat{\mathbf{b}}, {}^S\hat{\mathbf{m}}) = \begin{bmatrix} 2b_x(0.5 - q_3^2 - q_4^2) + 2b_z(q_2q_4 - q_1q_3) - m_x \\ 2b_x(q_2q_3 - q_1q_4) + 2b_z(q_1q_2 + q_3q_4) - m_y \\ 2b_x(q_1q_3 + q_2q_4) + 2b_z(0.5 - q_2^2 - q_3^2) - m_z \end{bmatrix}. \quad (4.3.8)$$

The transposed Jacobian function $\mathbf{J}_{g,b}^T$ is a function of the previous timestep orientation and the Earth's magnetic field vector only.

$$\mathbf{J}_{g,b}^T({}^S_E\hat{\mathbf{q}}_{est,t-1}, {}^E\hat{\mathbf{b}}) = \begin{bmatrix} -2b_zq_3 & 2b_zq_4 & -4b_xq_3 - 2b_zq_1 & -4b_xq_4 + 2b_zq_2 \\ -2b_xq_4 + 2b_zq_2 & 2b_xq_3 + 2b_zq_1 & 2b_xq_2 + 2b_zq_4 & -2b_xq_1 + 2b_zq_3 \\ 2b - xq_3 & 2b_xq_4 - 4b_zq_2 & 2b_xq_1 - 4b_zq_3 & 2b_xq_2 \end{bmatrix} \quad (4.3.9)$$

The gradient function has been broken down into programmable sections, which allowed the orientation from the vector method to be computed. The orientation from the angular velocity could also be calculated, giving two different methods for calculation.

The first method gives instant and accurate step-changes between iterations, but is subject to cumulative error as the orientation is calculated using 'dead-reckoning' techniques. The second method produces an orientation that could be maintained, since it uses the magnetometer's absolute orientation - but this method is slow to respond, since the time-stepping is also the optimization routine.

In order to make the most of both methods a filter fusion algorithm can be applied, which attempts to combine the signals to create an accurate and reliable orientation.

$${}^S_E\mathbf{q}_{est,t} = \frac{\beta\Delta t}{\mu_t} \left(-\mu_t \frac{\nabla f}{\|\nabla f\|} \right) + {}^S_E\dot{\mathbf{q}}_{est,t-1} + {}^S_E\dot{\mathbf{q}}_{\omega,t}\Delta t \quad (4.3.10)$$

The value of β was chosen as 0.05 by observation of the 3D graphics. A higher value of β represents greater reliance on the magnetometer to correct the orientation from the 'dead-reckoning' gyroscope calculation. If β was too low, the sensor would appear to over-shoot during rotation, and if it was too high the 3D animation would rotate too slowly due to the slow descent algorithm.

The initial value for ${}^S_E\mathbf{q}_0$ did not matter, since the buoy would settle to the absolute orientation within a few seconds. The initial value was, therefore, taken as the null quaternion ${}^S_E\mathbf{q}_0 = [1, 0, 0, 0]$ which represents zero rotation, rather than implementing a separate algorithm to use just the magnetometer and accelerometer data at initialization.

Using the quaternion for each timestep, the absolute values of pitch and roll of the sensor in the Earth's reference system can be found, by converting to Euler angles of pitch and roll. This effectively gives the North-South and East-West tilt of the buoy - *regardless of the buoys rotation*.

This is required for the calculation of the directional wave spectra in second half of the wave buoy data processing.

The other value required for the second half of the processing is the heave of the buoy - which can be found by performing double integration on the accelerometer data. The accelerometer values were resolved to vertical using quaternion vector rotation from ${}^S_E\mathbf{q}_{est,t}$, since components of a_x and a_y would contribute to the vertical acceleration, as well as a_z , when the buoy was pitching and rolling respectively.

Visually, when β was equal to zero, the animation of the buoy would rotate indefinitely due to the gyroscope drift - which was caused by the cumulative error of integration. For orientation purposes, this could be compensated by compounding the orientation with the absolute values from the magnetometer - however, the accelerometer suffers from even worse cumulative error and a creative method had to be

employed to remove it, since the GPS was not fast or accurate enough to compensate for it.

Accelerometers are notorious for having huge cumulative error, partly because they have more physical error than other sensors, such as gyroscopes. Also, the component of gravity (g) limits the resolution of the results, since small oscillations in a_z are carried on a large $1g$ signal. Finally, the accelerations must be integrated twice creating numerical error, and far more importantly, integrating all the physical errors. Even the most expensive accelerometers (thousands of pounds) only guarantee error-free position tracking for 100ms or so; whereas the wave buoy data samples could be up to several minutes.

Firstly, the extent of the cumulative error was found. Figure 4.3.2 shows a typical wave dataset lasting 24 seconds. In this 24 seconds the cumulative error reached 2.7 kilometres, though the signal still carried the wave oscillations (around 8cm height) on this increasing value.

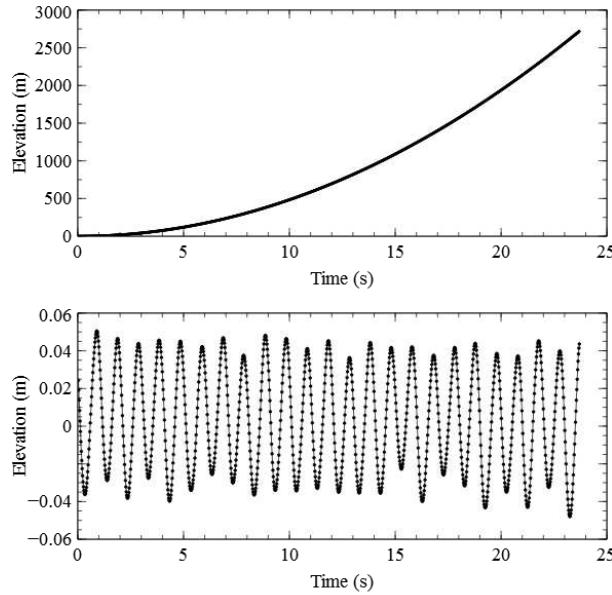


Figure 4.3.2: The top graph shows the typical cumulative error acting on the accelerometer, the small wave oscillations are still present in this signal, but not visible due to the large-scale drift. The lower graph shows this signal after the high-pass filter was implemented.

It was recognized that this cumulative error represented a low-frequency drift; though its frequency varied over time, and over different data sets. This low frequency could be removed by performing a high-pass filter on the data. This was done by taking a moving average and deducting it from the signal. The time-scale of the moving average dictated the time-constant of the high-pass filter, which was typically a 25-point (hence half-second or 2 Hz) high-pass filter. Figure 4.3.2 also shows the same signal after this process was performed.

This process relies on the fact that the average sea height should be constant, and the average of all the data. This provides a reference location which is not available for most accelerometer applications.

Using the methods discussed in this section, the heave, pitch and roll can be used to create the directional and non-directional wave spectra.

4.3.2 Wave Spectra Processing

[NEALE]

Unlike the controlled environment of a towing tank or wave basin, the wave systems occurring on a sea or lake are always irregular, and therefore it is not possible to obtain wave characteristics without data manipulation. Therefore, a statistical analysis was performed in order to obtain the characteristics of the

wave system. These characteristics included the significant wave height, mean frequency and mean wave direction.

It is possible to define the surface of a sea or lake as an infinite number of sinusoidal waves, each with a different amplitude, frequency, phase and direction. The data obtained from the wave buoys was broken down into these component waves through the use of a Fourier Transform. This also transformed the data from the time domain into the frequency domain [Lloyd, 1998].

The frequency interval was defined as $\frac{1}{T}$ where T is equal to the total length of the data record in seconds. This value was also the minimum frequency that could be measured as its period was equal to the length of the data record.

The maximum frequency able to be analysed was defined by the Nyquist principle. This principle states that the maximum frequency measured should be no greater than half the sampling frequency. This is because at least two points per wave are required to sufficiently define the peaks and troughs.

In order to calculate the significant wave height and mean frequency a Fourier analysis was performed on the vertical (heave) displacements. As a result a set of complex wave elevations and corresponding frequencies were obtained. The magnitude of the complex wave elevations were taken. These values were then converted into a wave amplitude energy density spectrum which allowed the relative importance of each component wave to be identified. This was achieved using the following energy density equation:

$$S_\zeta(\omega_n) = \frac{\zeta_n^2}{2\delta\omega} \quad (4.3.11)$$

where ζ = wave amplitude (m) and ω = frequency (Hz).

It can be shown that the variance of the wave amplitude (m_0) is equal to the area under its energy spectrum [Lloyd, 1998]. Furthermore the variances of the velocity (m_2) and acceleration (m_4), also known as spectral moments, are equal to the areas under the velocity and acceleration spectra respectively. Overall it can be shown that:

$$m_n = \int_0^\infty \omega^n S_\zeta(\omega) d\omega \quad (4.3.12)$$

The mean frequency and period were obtained by calculating the centre of area of the spectrum as follows:

$$\bar{\omega} = \frac{m_1}{m_0} \quad (4.3.13)$$

$$\bar{T} = \frac{m_0}{m_1} \quad (4.3.14)$$

The bandwidth describes to what extent the energy is spread out over different frequencies. An energy spectrum with a narrow bandwidth will have the majority of its energy concentrated over a small band of frequencies. Conversely an energy spectrum with a wide bandwidth will have its energy spread over a wide band of frequencies; this indicates the occurrence of a large number of troughs above the datum level and vice versa. It may be calculated as follows:

$$\epsilon = \sqrt{1 - \frac{m_2^2}{m_0 m_4}} \quad (4.3.15)$$

The significant wave height could then be calculated:

$$\bar{H}_{\frac{1}{3}} = 4.00\sqrt{m_0}\sqrt{1 - \frac{\epsilon^2}{2}} \quad (4.3.16)$$

This value was divided by 2 to obtain the significant wave amplitude.

To verify the result a zero crossing analysis was performed on a small section of data. This analysis defines a wave as the portion of data between two upwards crossings of the datum line [Manly Hydraulics Laboratory, 2000]. The significant wave height is then defined as the average of the greatest 33% of the peaks and troughs. Using the same data set, it was found that the significant wave heights calculated using the spectrum method and the zero crossing method were 0.0085 m and 0.0106 m respectively. An increase of 25% was observed using the zero crossing method, however this was expected due to the data set containing peaks below the datum line and vice versa.

The wave amplitude energy density spectrum could also be described as a non-directional spectrum. To obtain the mean wave direction a directional spectrum was calculated by multiplying the non-directional energy spectrum with a directional spreading function [National Data Buoy Center, 1996]:

$$S(\omega, \theta) = S_\zeta(\omega)D(\omega, \theta) \quad (4.3.17)$$

where D, the directional spreading function is calculated as follows:

$$D(\omega, \theta) = \frac{1}{\pi} \left(\frac{1}{2} + r_1 \cos(\theta - \theta_1) + r_2 \cos(2(\theta - \theta_2)) \right) \quad (4.3.18)$$

The variables in the directional spreading function equation are:

$$r_1 = \frac{1}{a_0} (a_1^2 + b_1^2)^{\frac{1}{2}} \quad (4.3.19)$$

$$r_2 = \frac{1}{a_0} (a_2^2 + b_2^2)^{\frac{1}{2}} \quad (4.3.20)$$

$$\theta_1 = \text{Mean wave direction} = \tan^{-1} \left(\frac{b_1}{a_1} \right) \quad (4.3.21)$$

$$\theta_2 = \text{Principle wave direction} = \tan^{-1} \left(\frac{b_2}{a_2} \right) \quad (4.3.22)$$

The following equations will adhere to particular notation:

1 = wave elevation (heave displacement), 2 = east-west wave slope, 3 = north-south wave slope

Variables a_0, a_1, a_2, b_1 and b_2 occurring in Equations (4.3.19) to (4.3.22) are known as the Longuet-Higgins directional parameters [National Data Buoy Center, 1996]. They are calculated as follows:

$$a_0 = \frac{C_{11}}{\pi} \quad (4.3.23)$$

$$a_1 = \frac{Q_{12}}{k\pi} \quad (4.3.24)$$

$$a_2 = \frac{C_{22} - C_{33}}{k^2\pi} \quad (4.3.25)$$

$$b_1 = \frac{Q_{13}}{k\pi} \quad (4.3.26)$$

$$b_2 = \frac{2C_{23}}{k^2\pi} \quad (4.3.27)$$

C_{xy} and Q_{xy} are co-spectral and quadrature spectral density functions respectively. A Fourier analysis is performed on the data for the wave elevations [1], east-west wave slopes [2] and north-south wave slopes [3]. The values of C_{xy} and Q_{xy} can then be calculated as follows:

$$C_{xy} = \frac{Re[X]Re[Y] + Im[X]Im[Y]}{2\delta\omega} \quad (4.3.28)$$

$$Q_{xy} = \frac{Im[X]Re[Y] - Re[X]Im[Y]}{2\delta\omega} \quad (4.3.29)$$

The wave number, k, may be calculated using the following equation, derived from linear wave theory [National Data Buoy Center, 1996].

$$k^2 C_{11} = (C_{22} + C_{33}) \quad (4.3.30)$$

Finally the directional energy spectrum may be plotted as shown in Figure 4.5.11. The mean wave direction may be found by taking the frequency of the maximum value of the wave amplitude energy density spectrum and calculating the corresponding value of θ_1 .

4.3.3 Software

[HAWKES]

A computer program was developed to deal with the hardware-software interface, to plot a 3D visualization of the wave buoy and to process all the data - including calibration, IMU and spectral processing. The program was called WABDAP - Wave Buoy Directional Analysis Program.

The program was created in C++ using the Qt libraries for open-source cross-platform GUI distribution, meaning that the program will compile on Microsoft Windows, Apple OSX, all Linux platforms and smartphone platforms such as Android or iOS. Though some modifications should be made before attempting to distribute this program on to a mobile device [Nokia, 2013].

The program makes uses of asynchronous multi-threading. One thread handles the **graphical user interface** (GUI), which contains the 3D graphics, live plotting of the sensor data and other user interface functions. Amongst other things, this thread is responsible for spawning other threads.

The **serial communications** thread uses an open-source library called qserialport, which controls the connection to the serial port on which the XBee module is connected. It is responsible for reading the sensor values emitted by the sensor to all the other threads that require it. For example, the serial thread sends sensor values (with thread-safety) to the GUI thread, so that it can update the live-sensor plots (Figure 4.3.3).

This thread could be improved to include Bluetooth communications, particularly if it were to be used on platforms with built-in Bluetooth - such as mobile devices.

A third thread, called the **IMU** thread, handles the IMU processing using the sensor measurements emitted from the serial thread. This processing is quite intensive, so benefits from running in a separate thread. This thread emits the quaternion to other threads, including the GUI thread which uses it to update the 3D graphics.

The 3D graphics consists of an OpenGL 2.0 animation which takes the quaternion from the IMU calculations to display the orientation of the buoy. The buoy was represented by an animation of a die, rather than modelling the actual buoy shape in OpenGL. The die was chosen because its faces could easily

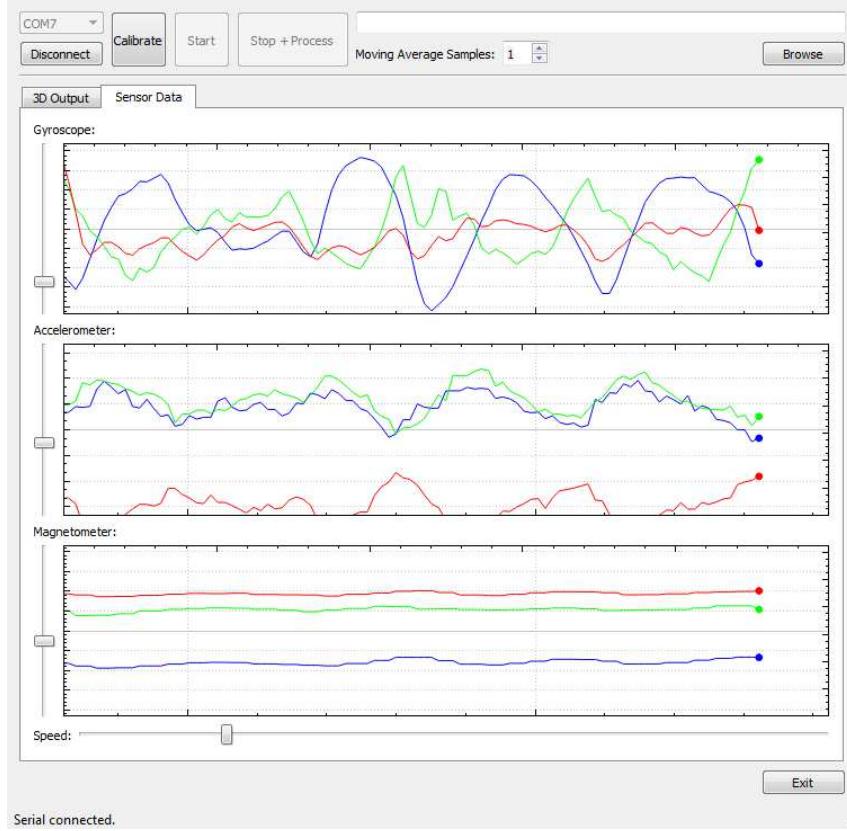


Figure 4.3.3: The serial thread sends sensor data to the main thread for plotting. The scale of the x and y axis can be changed using the sliders, to amplify the signals or change the time-history that is visible. Each dotted vertical line marks one second, and the vertical axes are the non-dimensional raw digital outputs from the sensor. These graphs were used to ensure the sensors were working and to tune the low-pass filter, which could be changed via the ‘Moving Average Samples’ option on the toolbar. Clearly, the accelerometer was very noisy when the filter was turned off.

be identified - as opposed to using a plain cube which would have been confusing. Figure 4.3.4 shows the 3D animation of the buoy within WABDAP.

Another thread, called the **calibration** thread, is spawned when the calibration process is started. Whilst the calibration process is running, this thread handles all the necessary processing for calibration, including the computationally expensive ellipsoid fitting algorithm. This thread emits the calibration constants to the other threads before destroying itself after the calibration is complete. The GUI for this thread is shown in Figure 4.3.5.

A final thread, called the logger, is created when the logging process begins. During the logging it collects data from the serial and IMU threads, but after logging it processes all the logged data to obtain the directional spectra. This thread takes a long time, particularly because it must write all the results to a text file, including a very large 3D dataset, and benefits from being in its own thread so that the program can continue running.

Note that, due to the object-orientated and threaded nature of the program, it can begin logging new data whilst still processing the previous dataset.

This program runs well on quad-core systems since four threads are usually running (the calibration and logging threads would not normally be run at the same time), though also performed well enough on a dual-core laptop. The laptop was a 1.2Ghz dual core laptop, and struggled to perform some of the

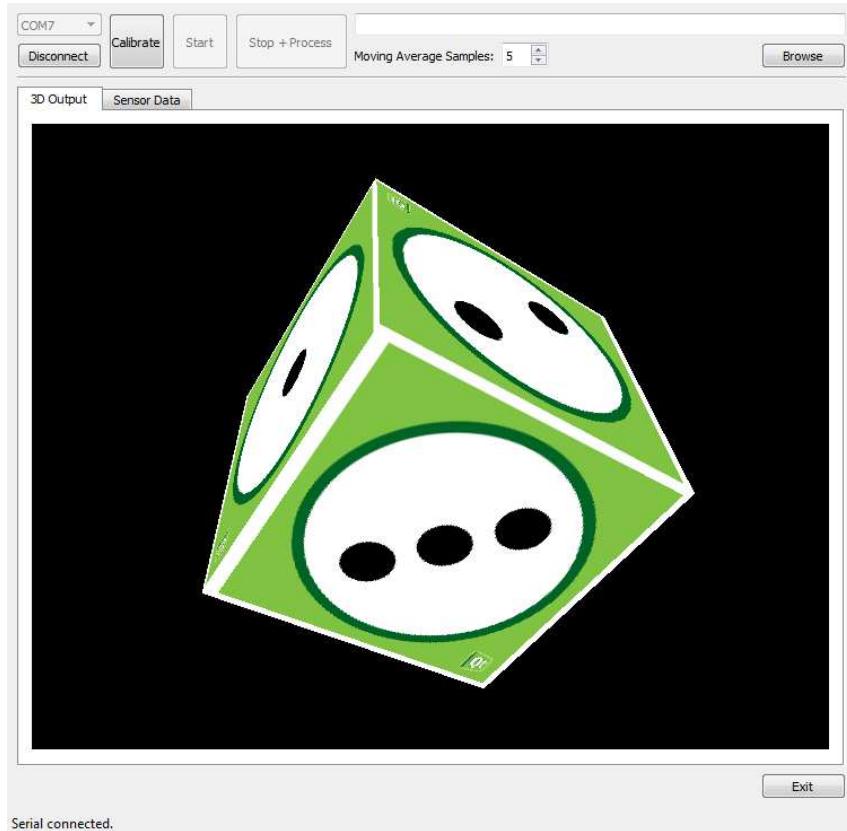


Figure 4.3.4: The 3D animation created in WABDAP, illustrating the real-time buoy orientation. This was necessary, since it was the best means to check the filter fusion algorithm.

logging thread calculations without slowing the user interface, since the processor was saturated, but was still usable.

In the future, it would be best to allow post-processing of the logged data, rather than instantly calculating the spectra - as this would reduce the demand on the shore computer. However, there was no opportunity to build a post-processing script due to time-constraints. This also meant that no attempt was made to try permutations of the data processing - such as changing the high-pass filter frequency to improve the results of the accelerometer.

Overall, the program performed well and the time-consuming process of plotting 3D graphics and developing a GUI was definitely worthwhile, as it allowed a huge amount of testing to be performed quickly and efficiently with real-time results.

4.4 Testing

[THORNTON]

The 4 buoy designs were both modelled in OrcaFlex and tested in the University of Southampton's Lamont Towing Tank. In this section, the procedures and resulting dynamic responses of each buoy are discussed. From here-in, each prototype buoy; big-straight, big-curved, small-straight and small-curved; will be referred to as BS, BC, SS and SC respectively.

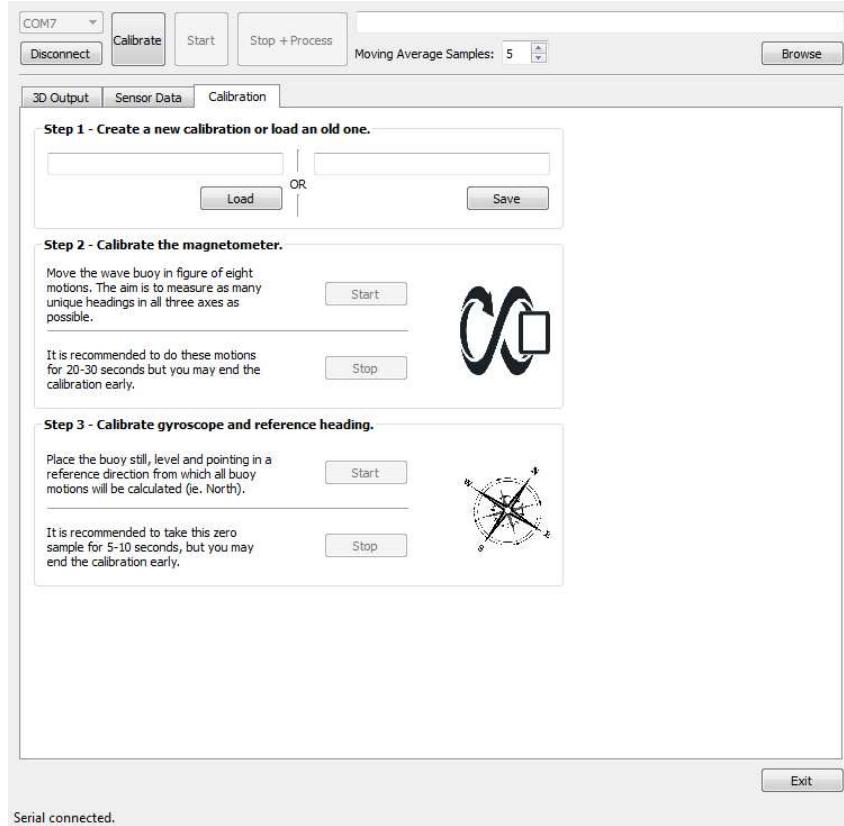


Figure 4.3.5: This tab instructs the user through the calibration process and communicates to the calibration thread via the GUI thread. The user can save calibration results, so that the process does not have to be re-run each time the program is used.

4.4.1 OrcaFlex

[THORNTON]

As previously mentioned, the 4 buoy prototypes were modelled in OrcaFlex to analyse their dynamic responses in various controlled sea states and validate their design and capability to measure the desired waves. They were constructed using OrcaFlex's 6D Spar buoy feature, which models axis-symmetric buoys with all 6 degrees of freedom; that is, 3 translational (X, Y, Z) and 3 rotational (1, 2, 3) [Orcina, 2013]. Buoys are modelled with both mass and moments of inertia; hence a range of forces and moments can be represented. Important features for this report include its ability to model: buoyancy, added mass, damping and drag, accounting for the submerged proportion of the buoy; slamming forces; connection loads for attached lines; and translational responses, i.e. measure heave.

Spar buoys are intended particularly for use where surface piercing effects are important, such as for a surface contouring wave buoy. To allow for custom geometries, they are modelled as a series of cylinders built up along the local z-axis, specifying the number of cylinders and the height and diameter of each. Figure 4.4.1 shows the composition of each prototype buoy modelled in OrcaFlex.

Each buoy was drawn to scale in 2D in AutoCAD, using 45 cylinders to define the hull curvature, as this was deemed to provide suitably accurate hull geometry. For the 2 larger buoys, BS and BC, these cylinders had a height of 1 mm, and for the smaller buoys, SS and SC, a height of 0.6 mm. The upper section of the buoy was modelled as a single cylinder as the walls are vertical. These drawings were then used to determine the offsets which were used in OrcaFlex as the 6D buoy diameters. Offsets were also determined for other components such as the sensor unit and ballast weight. The offsets were

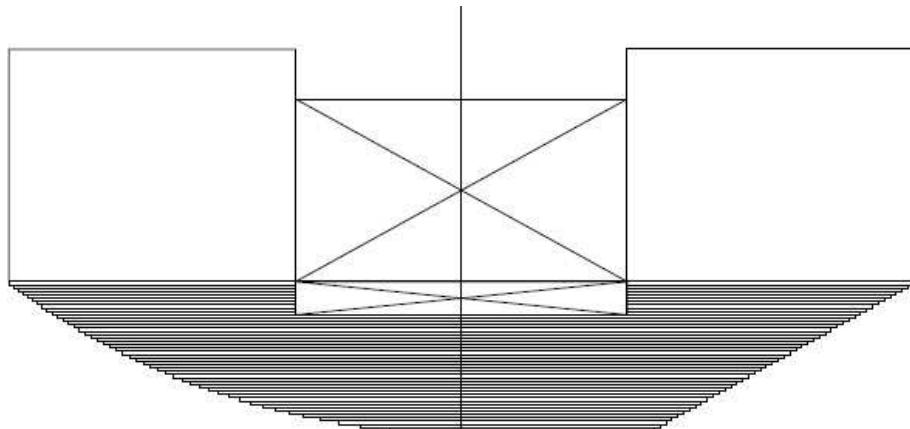


Figure 4.4.1: An example of the cylinder breakdown of two buoys modelled in OrcaFlex [BC - left; BS - right].

first copied into an Excel spread sheet where calculations were carried out to determine moments of inertia, centroids and vertical centres of gravity (VCG) for each component which were then input into the OrcaFlex simulations (Table 4.4.1).

Table 4.4.1: Summary of calculations for modelling in Orcaflex.

Buoy	\sum Moment (kgm)	VCG (m)	Moments of Inertia (m^4)	
			I _{xx} , I _{yy}	I _{zz}
BS	0.0610	0.0640	0.002489	0.002799
BC	0.0574	0.0604	0.002487	0.002804
SS	0.0128	0.0507	0.000513	0.000490
SC	0.0125	0.0604	0.000516	0.000498

Variations in mooring line arrangements were only explored using BS simulations, as the main purpose of this was to finalise an arrangement for testing in the Lamont Tank. All simulations otherwise were run using a straight mooring line for ease, modelled as a 2.5 mm diameter nylon line due to its elastic properties. A scope of 2:1 was used, i.e. a 4 m long nylon line was used for water depths of 2 m, which appeared sufficient for the test conditions as the mooring line rarely became taut during simulations.

4.4.2 Lamont Tank

[THORNTON]

Once the wave buoys were constructed, further validation tests were conducted in the University's Lamont Towing Tank. Wave data was transmitted wirelessly off the buoys during tests to an onshore laptop. A wave probe was used to measure the waves generated during experiments, allowing for comparisons to be drawn between the actual waves and the measurements taken from the buoys' responses. The experimental set up is presented in Figure 4.4.2.

The wave probe used was a rigid, twin-sting resistance design connected to an onshore amplifier box, or Churchill box. The amplified signal output was then sent to an analogue-to-digital converter, which was obtained by the shore based digital-to-analogue converter and displayed on a laptop. The wave probe was calibrated using the voltage readings over 43 mm, which gave a range of 3.975 V. This gave a value of 0.0924 V/mm, which was used in determining the gain for required wave heights.

Waves were generated using the SEASIM Wave maker, which consists of 3 simultaneously oscillating panels driven by a compressed air system. It is programmed using the Hexadecimal system, such that

for example, to generate a wave with a 1 second period would require programming 64 into the SEASIM computer. The amplitude of the generated waves is then controlled via a gain potentiometer by setting the dial manually.

Before conducting the experiments, it was necessary to determine roughly what gain would relate to the desired significant wave height being generated, through measurements using a ruler against the inside of the tank and measuring the voltage display. Though not an exact science by any means, it was determined that for a wave height of roughly 6 cm and 10 cm, a gain setting of 25 and 40 were to be used, respectively. This coincided with probe readings of 5.54 V for 6 cm waves, and 9.244V for 10 cm waves. The gain on the wave probe was then set to its maximum resolution over the 10 cm range. The buoy system and wave probe were set transversely in line in the tank in an attempt to measure the same point of a wave at simultaneous periods in time, though due to the nature of the catenary mooring arrangement, this was not always the case.

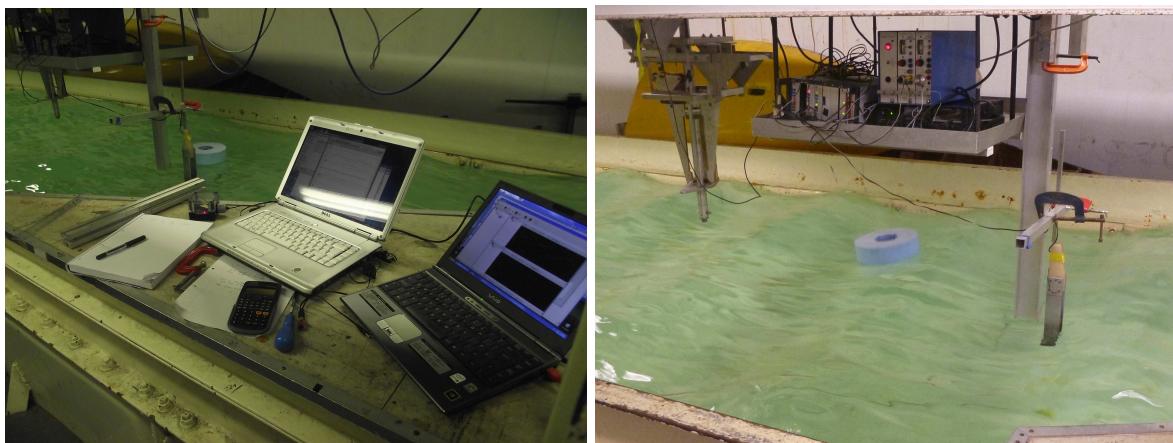


Figure 4.4.2: Experimental setup for the Lamont tank tests. Laptops on the left and right were used to record buoy and probe measurements respectively. Buoy responses were wirelessly transmitted to a transceiver connected to the left laptop. On the right, the typical drift of the buoy during an experiment can be seen - the buoy was moored in-line with the wave probe.

4.5 Results

[THORNTON]

4.5.1 Dean Stream Waves

Figure 4.5.1 shows a series of graphs plotting displacement against time for sea elevation and buoy responses modelled in OrcaFlex, using a Dean-stream wave with a height of 6 cm and period of 1 second (indicated as H6T1). All 4 wave buoy simulations were run over a 60 second time interval, of which a 15 second sample is displayed for clarity between the 2 data sets. Generally, all buoys show accurate responses to these waves, indicating that they are closely contouring the wave surface, providing validation for their design. It is noted that towards the beginning of the simulation the buoys' heave response is slightly premature to the advancing waves, but synchronises later on. This horizontal translation in buoy response is particularly clear for the case of the SS and SC buoys, which could be due to easier excitation as a result of their lower mass. It may also be due to the buoys' leading edge coming in contact with the following wave face before the point in time at which the wave elevation is measured; hence the buoy will begin to rise earlier. This appears to occur in other results as well, with the offset being a function of the buoy hull diameter.

In Figure 4.5.2 results obtained experimentally in the Lamont Tank are plotted for the same scale waves (height=6 cm, period=1 s). The graphs illustrate measurements of sea surface elevation obtained from the wave probe, and buoy elevation obtained from the on-board sensor unit. The actual measured responses of all 4 buoys vary quite considerably in comparison with the OrcaFlex models. Measurements obtained from the buoys indicate roughly 100% higher elevations at the wave peaks and 25% lower troughs versus the actual measured waves for BS, BC and SS data. Elevations measured by the SC buoy indicate wave peaks up to 125% higher than those measured by the wave probe. Measurements at the wave troughs are somewhat higher as well. This phenomenon may be explained by the combination of the buoys' buoyancy and momentum as it reaches the peak, effectively launching the buoy upwards over the peak. This would be amplified as a result of the relatively steep faces associated with the generated waves shared with the low mass of the buoy. Likewise, lower trough measurements may be as a result of the buoys' momentum as it travels down the wave face, submerging beneath the design waterline as it reaches the trough.

In some of the results, there is evidence to suggest that the high-pass filtering was not successful, since they showed a considerable drift which made them unusable. This could have been improved by fine-tuning the filter frequency for each trial.

The overshoot at the peaks of the wave could also be due to the error of the accelerometer, because it was probably drifting during the course of one wave due to the cumulative integration of error. For this accelerometer, the drift seems to be upwards (Figure 4.3.2), which would explain the relatively high peaks. If this is the case, then these results could only be improved by using a better accelerometer, which would drift less over the course of a single wave.

Despite these errors in elevation measurements, the buoys prove to provide reliable measurements with regards to frequency response, indicating they do in fact contour the waves as seen in the OrcaFlex simulations. This is true for all 4 buoys in seas with waves of frequency 1 Hz and height of 6 cm.

Simulations were also run in OrcaFlex to measure the modelled response of the 4 buoy prototypes in Dean-stream waves with a height of 6cm and a period of 0.7 s. The buoy responses from each simulation in this sea state are shown in Figure 4.5.3. For the 2 larger buoys, BC and BS, it is clear that the buoy response does not necessarily contour the waves, proving that this diameter may be too large for the shorter wave periods. For both buoys there is a clear phase lag in the frequency response, falling behind

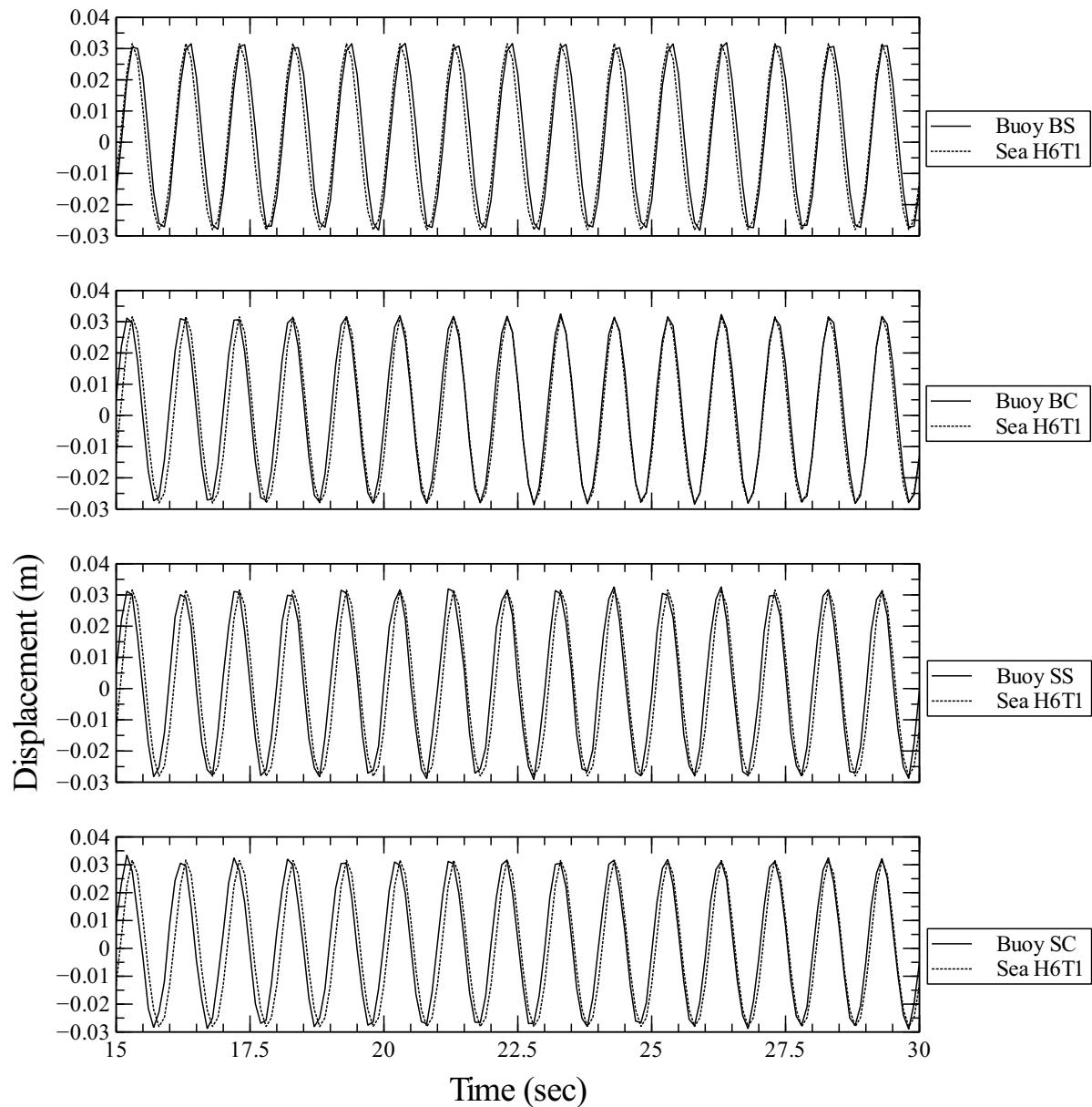


Figure 4.5.1: Results showing elevation vs. time for all buoys in OrcaFlex: Dean stream waves, height 6 cm, period 1 second.

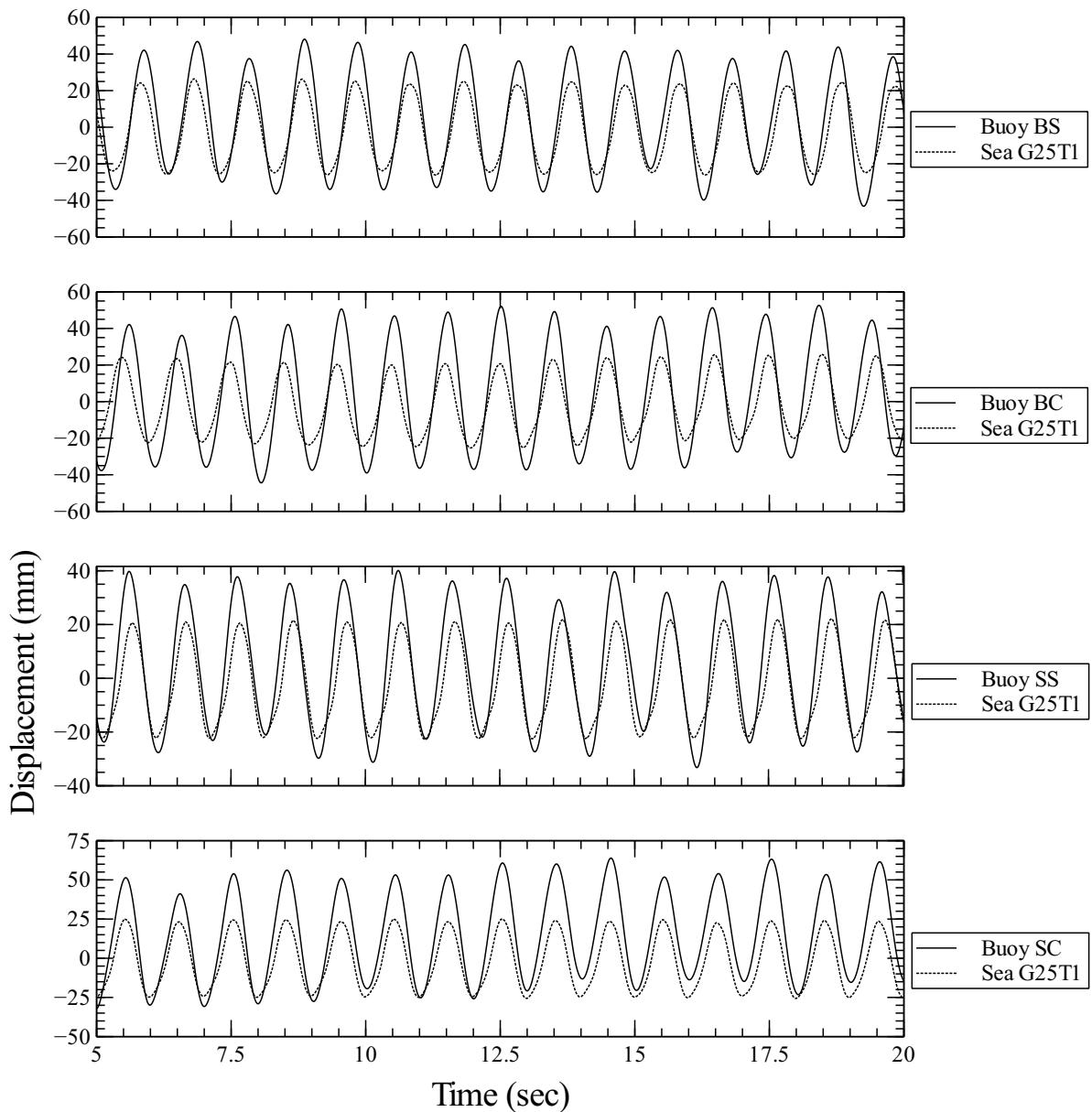


Figure 4.5.2: Results showing elevation vs. time for all buoys in the Lamont tank: Dean stream waves, height ≈ 6 cm, period 1 second.

by a full wavelength within 20 seconds. The heave response however, still remains fairly accurate, with buoy measurements recording wave height of roughly 6-8 mm greater than the actual sea elevation, or roughly 10%. As previously discussed, this error may be due to a combination of the buoys' buoyancy and momentum, the viscous effects of the water and accelerometer drift.

The smaller buoys, SS and SC, show a more linear response to the shorter wave periods, as expected due to their smaller diameter and hence increased ability to contour shorter waves. They both show accurate frequency responses, in particular the SS buoy. Buoy measurements for wave amplitude show both higher peaks and troughs. The higher troughs may be as a result of the buoy hull not being able to completely contour the surface at the trough, due to the sudden change in wave face orientation associated with the low zero crossing period. The lower mass will also decrease the momentum and hence the amount by which the buoy submerges when it hits the trough. This is an indication that a further decrease in hull diameter may be necessary to fully contour waves of this nature, though as previously discussed this is largely limited by the sensors used and hence is not plausible. Buoy response data for the SC hull shows a constant offset in the time domain throughout the simulation, which can be accounted for by the rising of the buoys' leading edge prior to the wave measurement.

Figure 4.5.4 shows the response of all buoys in the Lamont tank subject to waves with a zero mean crossing period of 0.7 seconds and a gain setting of 25, resulting in wave heights between 20 mm and 60 mm. This is obviously quite a variance which indicates the inconsistency associated with the method used to generate waves. The response of the buoys in comparison to the OrcaFlex simulations shows similar traits to the comparison between simulations and tests run with a 1 s period - frequency response is accurate whereas the heave response is considerably greater for the buoy. Results for the BS and BC buoys do not show the phase lag which was apparent in the OrcaFlex simulations, but rather stay in-phase throughout the experiment. Heave response for the SC buoy appears to be most accurate, however, suffers from problems due to accelerometer drift.

4.5.2 JONSWAP Waves

[THORNTON]

Figure 4.5.5 shows the OrcaFlex modelled response of each buoy prototype in a JONSWAP (Joint North Sea Wave Project) spectrum with wave height of 6 cm and zero mean crossing period of 3 seconds. For the most part, responses of each buoy design prove accurate in measuring waves produced in this spectrum. Similarly to results found in the Dean-stream waves, areas where the buoys' responses vary most from the sea surface elevation are around sharp peaks and troughs.

The most accurate representation of the wave spectra used in figure 4.5.5 is that of the BS buoy. When compared to the other 3 buoys, the response data follows the sea elevation closer throughout the simulation. The waves modelled have relatively long wavelengths and low amplitudes which the stable BS hull handles itself to.

4.5.3 Spectral Densities

[THORNTON]

OrcaFlex was used to produce spectral density curves for both sea and buoy responses in the JONSWAP spectra. For height 6 cm and period 3 seconds, all 4 buoy responses showed to contour the waves closely, hence spectral density curves for the buoys were virtually identical to those of the waves. Figure 4.5.6 shows an example of this for the BS hull in a JONSWAP spectrum with height 6 cm and period 3 s.

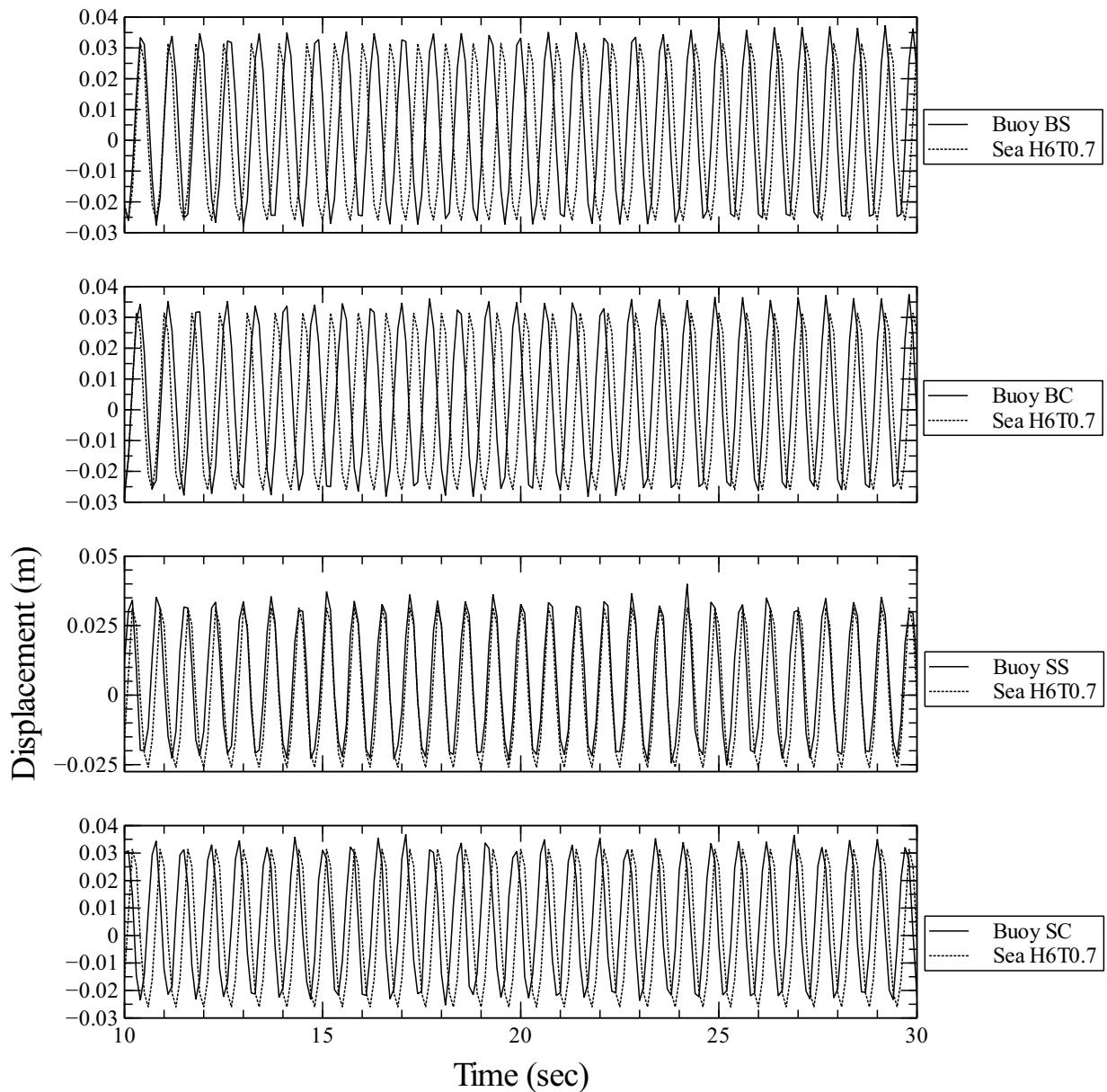


Figure 4.5.3: Results showing elevation vs. time for all buoys in OrcaFlex: Dean stream waves, height 6 cm, period 0.7 seconds.

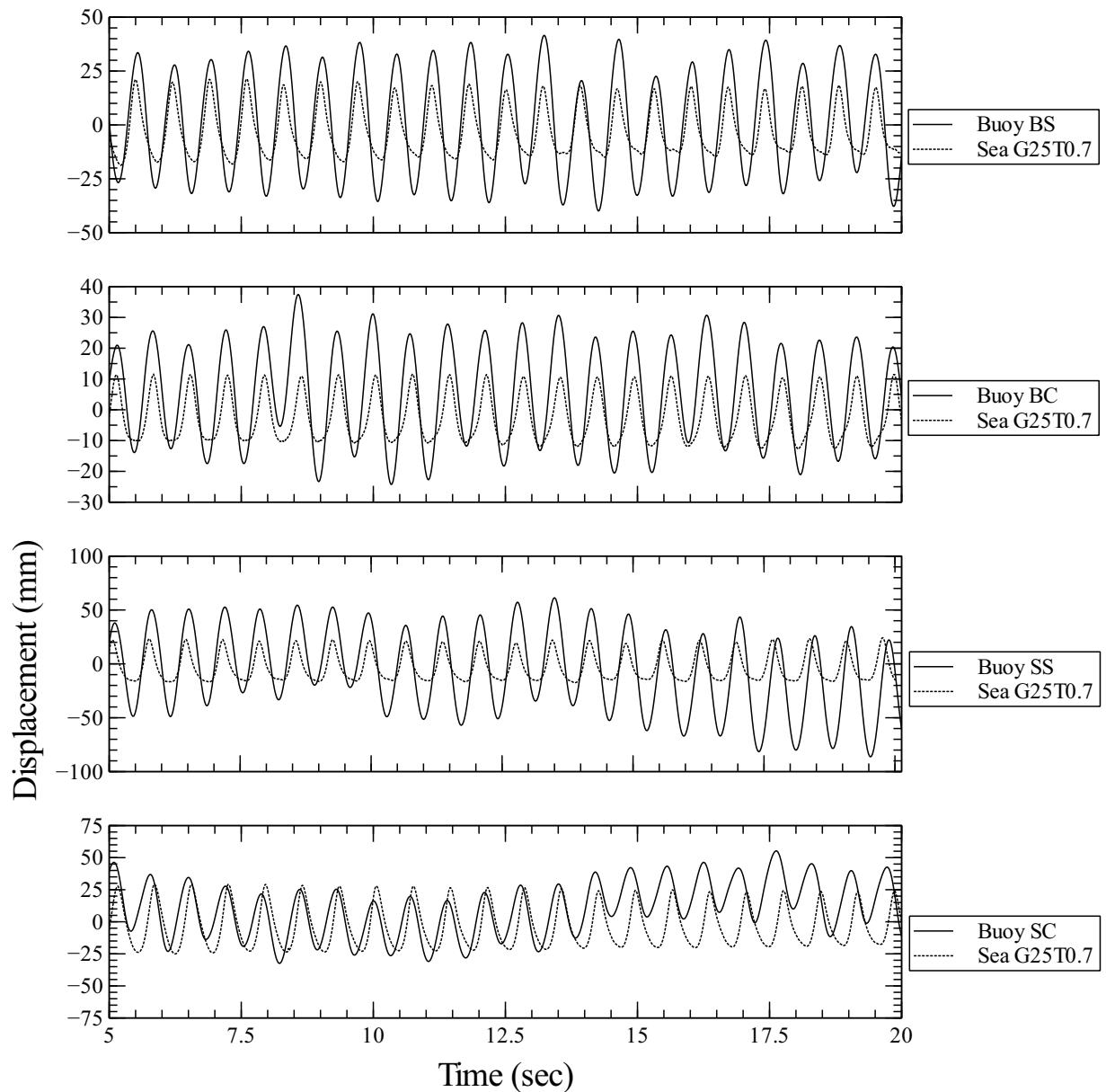


Figure 4.5.4: Results showing elevation vs. time for all buoys in the Lamont tank: Dean stream waves, height ≈ 6 cm, period 0.7 seconds.

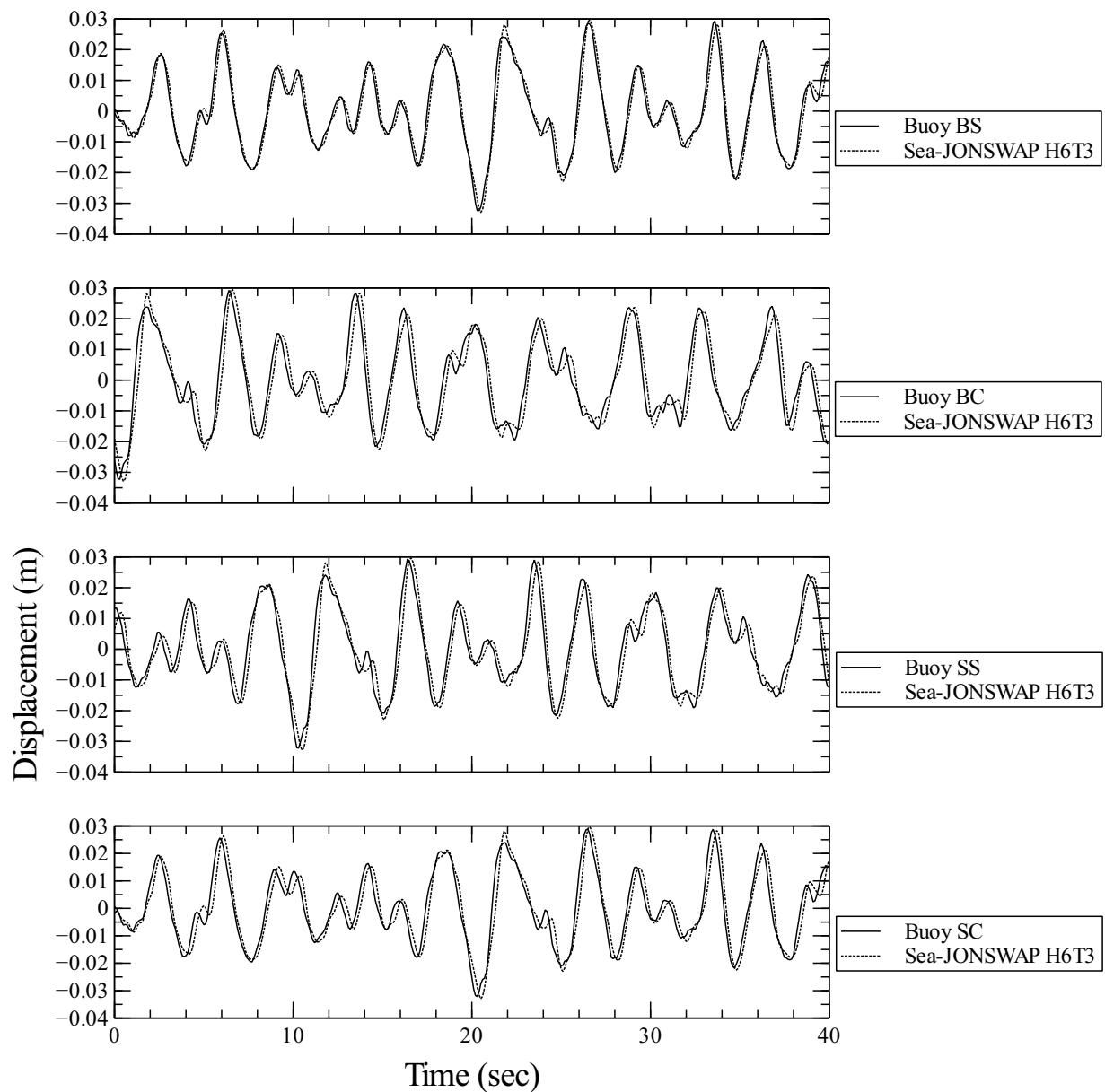


Figure 4.5.5: Results showing elevation vs. time for all buoys in OrcaFlex: JONSWAP spectrum, height 6cm, period 3 seconds.

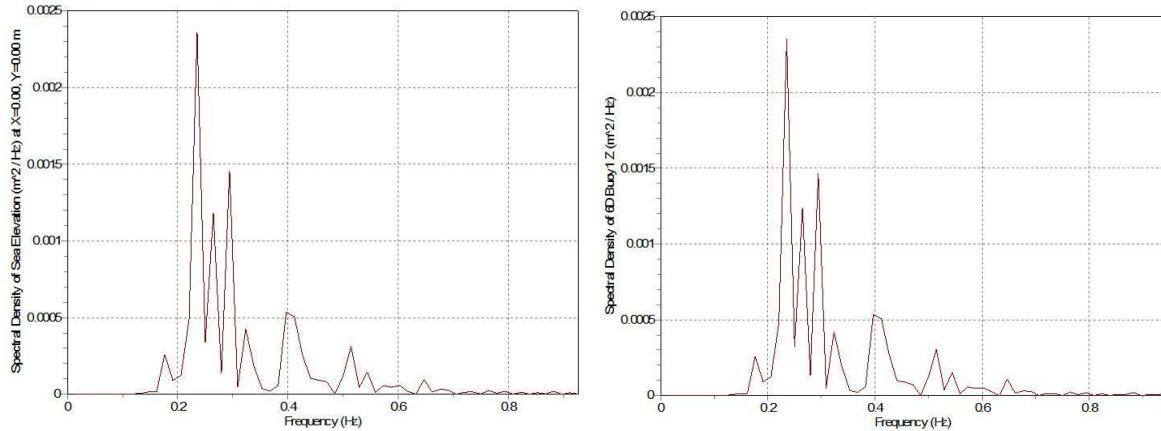


Figure 4.5.6: Results showing spectral densities for the sea elevation [left] and BS buoy [right]: JONSWAP spectrum, height 6 cm, period 3 seconds.

4.5.4 Extreme Tank Conditions

[THORNTON]

Figure 4.5.7 shows a sample of 4 experiments conducted in the Lamont Tank under more ‘extreme’ wave conditions to further investigate the design limitations. Graphs 1 and 2 show the heave response of the BS and SS buoys, respectively, in unidirectional waves with a gain of 40 (roughly 10 cm) and zero mean crossing period of 1 second. This is at the upper end of the desired scale waves that the model should be subject to in open water tests, as it scales up to waves of roughly 6 m. The same conditions were modelled in OrcaFlex and showed accurate responses to both frequency and height; however results from the tank vary somewhat to these. Waves of this nature result in relatively steep faces with sharp peaks and troughs, hence it was expected that the SS buoy would contour the waves more accurately than the BS hull, as confirmed in graph 2. Responses from both show relatively accurate wave frequency and height measurements, however the main region of error is found at the wave troughs, measuring troughs of up to 50 mm deeper for the BS hull and 25 mm deeper for the SS hull. As previously explained, this may be due to the close nature of the waves, amplified by the increased wave height at the same frequency compared to figure 4.5.2 resulting in greater errors in the readings. Accounting for the larger mass of the BS buoy and hence greater momentum, the deeper troughs measured by the BS hull compared with the SS hull can be explained.

Graph 3 shows the measured buoy response of the SC hull in unidirectional waves with a gain setting of 25 and zero crossing period of 0.5 seconds. This combination programmed into the SEASIM wave maker resulted in waves of height roughly 40 mm, which appeared to be closest to those experienced during free running tests at Timsbury Lake. Contrary to results in virtually all other sea states, the buoy response measured lower wave amplitudes than those measured by the wave probe. Average wave height measured by the buoy was in the region of 30 mm - 25% lower than the actual wave height. This may be due to the nature of the waves resembling closer to ‘ripples’ and despite their small height, the high frequency makes it difficult for even the small diameter buoys to contour accurately.

Graph 4 shows the SC hull in unidirectional waves with a mean zero crossing period of 1.3 s and a gain of 25 which generated waves of roughly 6-8 cm height. Frequency response of the buoy hull matches that of the waves, however there is a clear difference in amplitude particularly around the troughs.

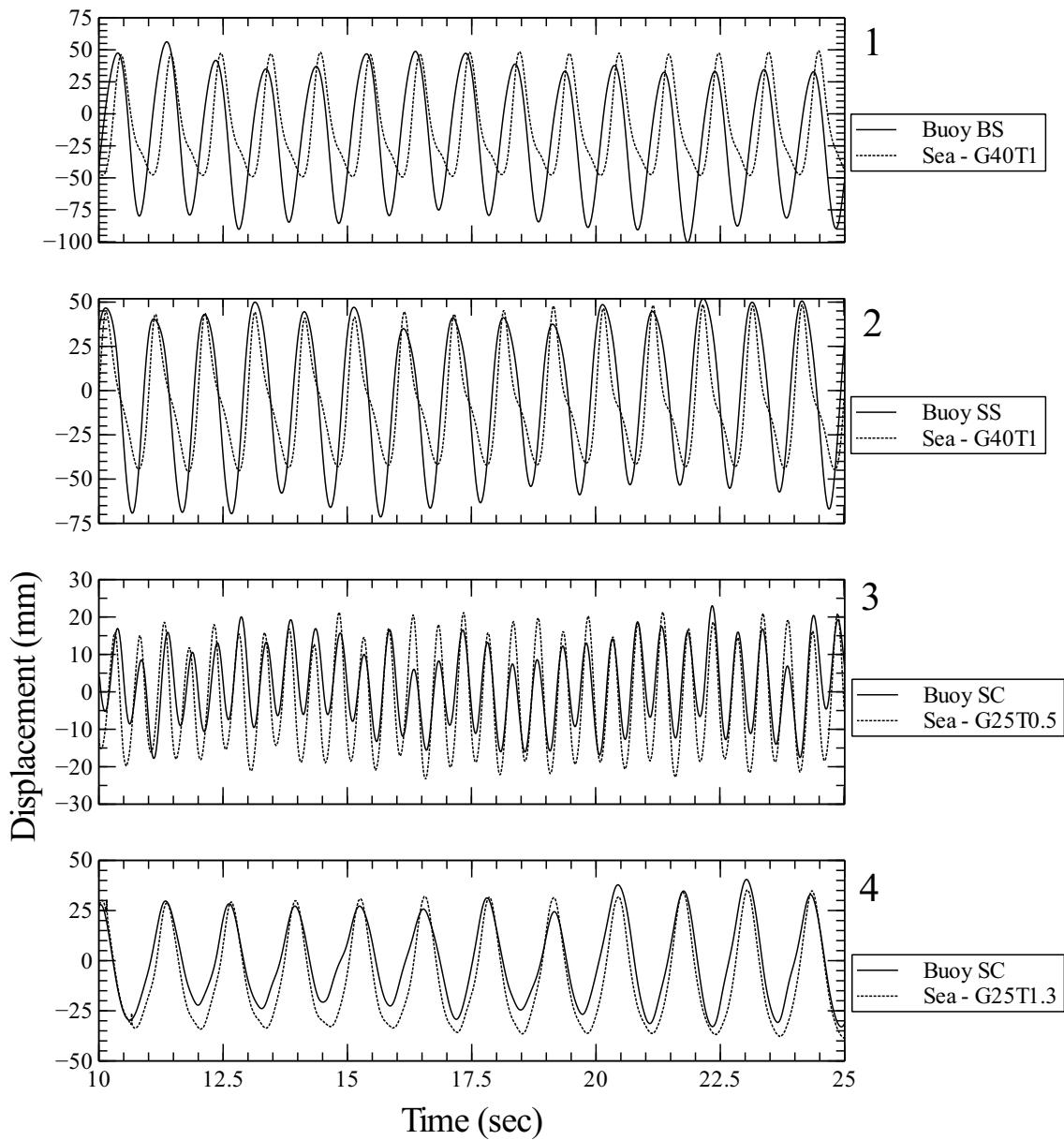


Figure 4.5.7: Results showing elevation vs. time in the Lamont tank: extreme wave conditions.

4.5.5 Mooring Tension

[THORNTON]

A comparison between measurements of effective tension in various mooring-line arrangements have been made in Figure 4.5.9 using OrcaFlex. Modelling the most extreme chosen Dean-stream waves, that is with a height of 10 cm and period 0.7 seconds, graph 1 shows the effective tension in the straight nylon mooring line for each buoy hull design. Peaks indicate where the line became taught as the buoy drifted from its equilibrium position. Values for effective tension generally fluctuate up to 1.5-2 N. The higher peaks indicate where the buoy actually capsized, resulting in values of up to 13 N which can be treated as anomalies for the purpose of this comparison. These values are considerably low, indicating a simple mooring arrangement such as a straight nylon line with a scope of 2:1 would be feasible in the desired range of waves. The catenary alternative can be seen in Figure 4.5.8.

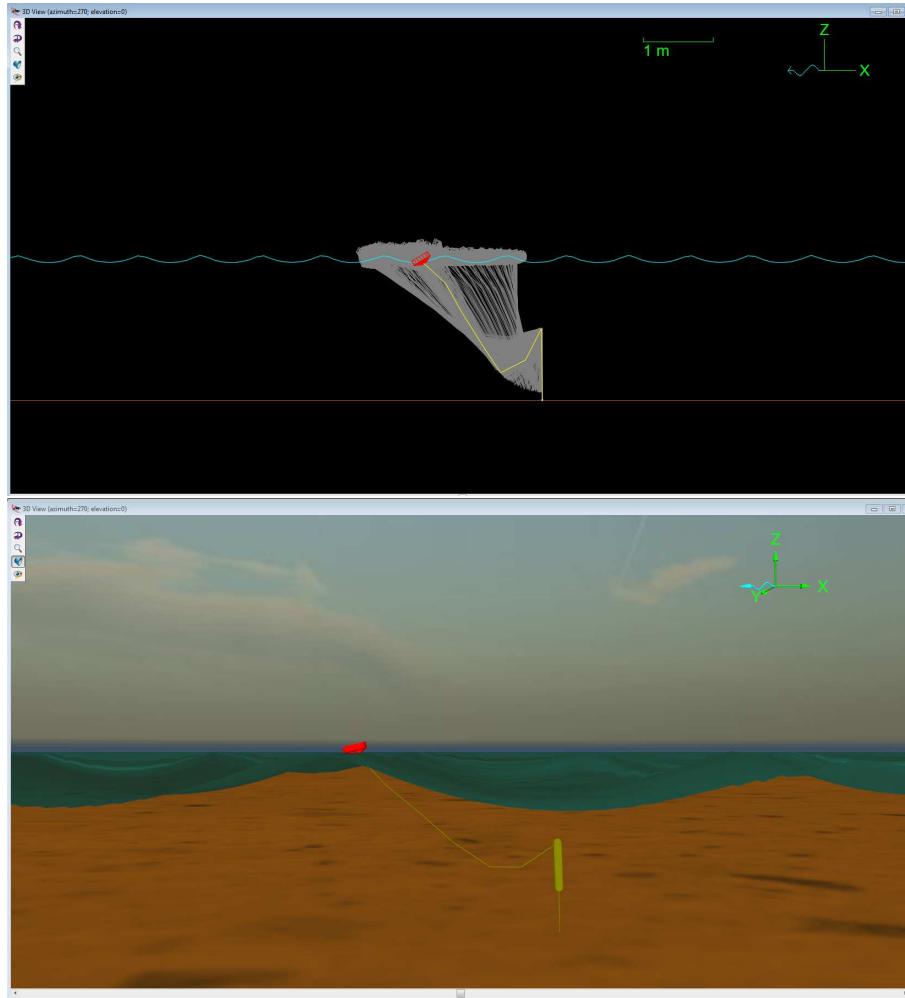


Figure 4.5.8: Screenshots from OrcaFlex simulating the catenary mooring arrangement. Top shows a 2D response history, and bottom shows a rendered 3D view of the simulation implementing it's shaded graphics mode.

Graph 2 shows a comparison between effective line tension for both the straight and catenary mooring arrangements, modelled in OrcaFlex using the BS buoy hull and in waves of height 10 cm and mean zero crossing period of 0.7 seconds. The catenary arrangement shows greater tension throughout the simulation, generally fluctuating between 0 and 4-5 N versus the 1-1.5 N values of the straight mooring. However, there appears to be a more uniform distribution of loads across time, with less extreme peaks

and no sign of capsizing. The gradual increase in tension around the peaks indicates a steady build up as the arrangement becomes taut, a representation of the damping effects of the chain and sub-surface float, as opposed to the sudden capsizing portrayed in the straight line arrangement. Though it appears to experience greater tensions, 4 N is still fairly insignificant and the catenary mooring will result in a more stable buoy response with less excitation due to the steady applied load. The risks of capsizing or sudden jerks in the buoy response are also reduced with the implementation of this arrangement which is shown in Figure 4.5.8..

4.5.6 WABDAP Results

[HAWKES & NEALE]

The results from the wave analysis program, for a period of 0.7 second and 25 gain, are presented here in terms of the spectral density functions produced for the Dean stream waves in the Lamont tank. Ideally, the buoys should be tested in a real environment so that multi-directional and multi-frequency spectra are created. Nonetheless, the results of uniform frequency uni-directional waves should validate much of the buoy's functionality - in terms of the electronics and data processing.

Figure 4.5.10 shows the 2D spectrum created by WABDAP, with a very strong peak at the principal frequency $\frac{1}{0.7} = 1.43\text{Hz}$. There is some low-frequency noise which should have been removed by the high-pass filter, but the reason for this could not be identified. Future work, particularly with a less erroneous accelerometer, should investigate this low-frequency noise.

Figure 4.5.11 shows the 3D spectrum created by WABDAP, there is a clear peak at 1.43 Hz once again. There is also a good indication of the direction of the waves, with most of the spreading probably occurring due to magnetometer problems in the tank. The sensor moved around on the mooring, which could have caused large errors in the heading calculations because there was so much magnetic interference in the towing tank.

The results also show negative peaks perpendicular to the directional peaks. This is a quirk of the data processing, since negative spectral energy is impossible. National Data Buoy Center [1996] suggests using a weighting function to remove this, although it was chosen not to. In this report, weighting functions were not applied either, since the graph shows all the necessary information without the added complexity and unreliability of a weighting function.

Notably, the data processing routine can distinguish the polarity of the wave direction, *i.e.* the buoy can calculate whether the waves are approaching from the North or South, not just whether they travel along the N-S line. This is handled by the cross-spectra processing, although there is some noise (physical or otherwise) contributing to a slight energy peak in the polar opposite direction.

Due to time constraints and lack of windy conditions, it was not possible to test the wave buoys in a real environment. The results from the towing tank give confidence that the buoy would perform well in irregular waves, varying in both frequency and amplitude - but this cannot be proven.

4.5.7 Other Results

[THORNTON]

OrcaFlex simulations and Lamont Tank tests were also carried out to investigate the response of each buoy in waves with a height of 10 cm and mean zero crossing periods 0.7 s and 1 s. The response of each was found to follow similar trends to those seen in the results for the series of 6 cm waves.

Simulations were also run to explore each response in a JONSWAP spectrum with a height of 6 cm and mean zero crossing period of 1 s. Heave responses for all buoys was erratic due to the intense energy and profile of this spectrum, resulting in inconclusive measurements.

4.5.8 Sources of Error

[THORNTON]

There are a few areas where experimental errors may be attributed to in tests conducted in the towing tank. The main source of this error may be due to the wave maker itself. The belt and pulley system used to control the 3 simultaneously oscillating panels was prone to slipping, resulting in the panels falling out of phase and generating irregular waves. This was particularly an issue when generating larger, shorter waves. Resetting the panels' zero-orientations manually in between experiments was done to minimize this.

It was discovered that the wave heights varied across the different frequencies, despite programming the same gain setting. This is the nature of the wave maker and may only be controlled by determining a gain setting that generates the correct waves before every experiment. With the time restrictions in available tank bookings, this was not feasible hence the error was adopted.

Another cause is the interference of waves reflecting back from the beach or from the tank walls. This interference would alter the unidirectional wave profile, affecting both measured buoy and probe responses. To minimize this, the tank was allowed to settle in between experiments and experiments were started after a short ramp up time (roughly 5-10 s) and concluded after 40-60 s to reduce the overall testing time.

The buoy and wave probe were positioned transversely in the tank, though this did not remain constant throughout experiments due to the nature of the catenary mooring and the tendency for the buoy to drift, particularly in higher wave amplitudes. This difference resulted in waves being measured at different points in time by the probe and buoy, allowing for dissipation or further interference effects to take place.

Bishop and M.A.Donelan [1987] discusses the meniscus effect as a source of error when measuring waves with surface-piercing gauges. The meniscus error accounts for surface tension and contaminations which may cause the water surface fluctuations to vary less on the probe than in the surrounding waves. For wave heights of roughly 6 cm, it is shown to contribute an error of up to 5%.

4.5.9 Summary

[THORNTON]

A range of OrcaFlex simulations and tank experiments have been conducted in order to test and validate a variety of contouring surface buoy designs for the application of an instrumented wave buoy. Four hull geometries were investigated, exploring the effects of 2 different hull diameters and 2 underwater profiles. Tests and simulations were conducted with various wave heights and zero mean crossing periods in the range of those desired during open water scaled model tests. In general, heave responses from all buoys were representative of a buoy accurately contouring the wave profiles. As expected, the larger diameter hulls performed most accurately in larger, longer waves; whereas the smaller diameter hulls were able to follow the smaller, shorter wave profiles more accurately. Despite this, there was evidence that a further decrease in hull diameter was necessary to fully contour the shorter waves, particularly in the region of the troughs. This will always be the case, however, and the only way to fully contour such sharp changes in wave face orientation is to model a single point on the wave, which in reality is not feasible. There was also evidence that the straight-edged profile provided more accurate buoy responses, which may be due to a more stable response in the water. Various mooring systems were also modelled, determining that the catenary arrangement specified was the most suitable.

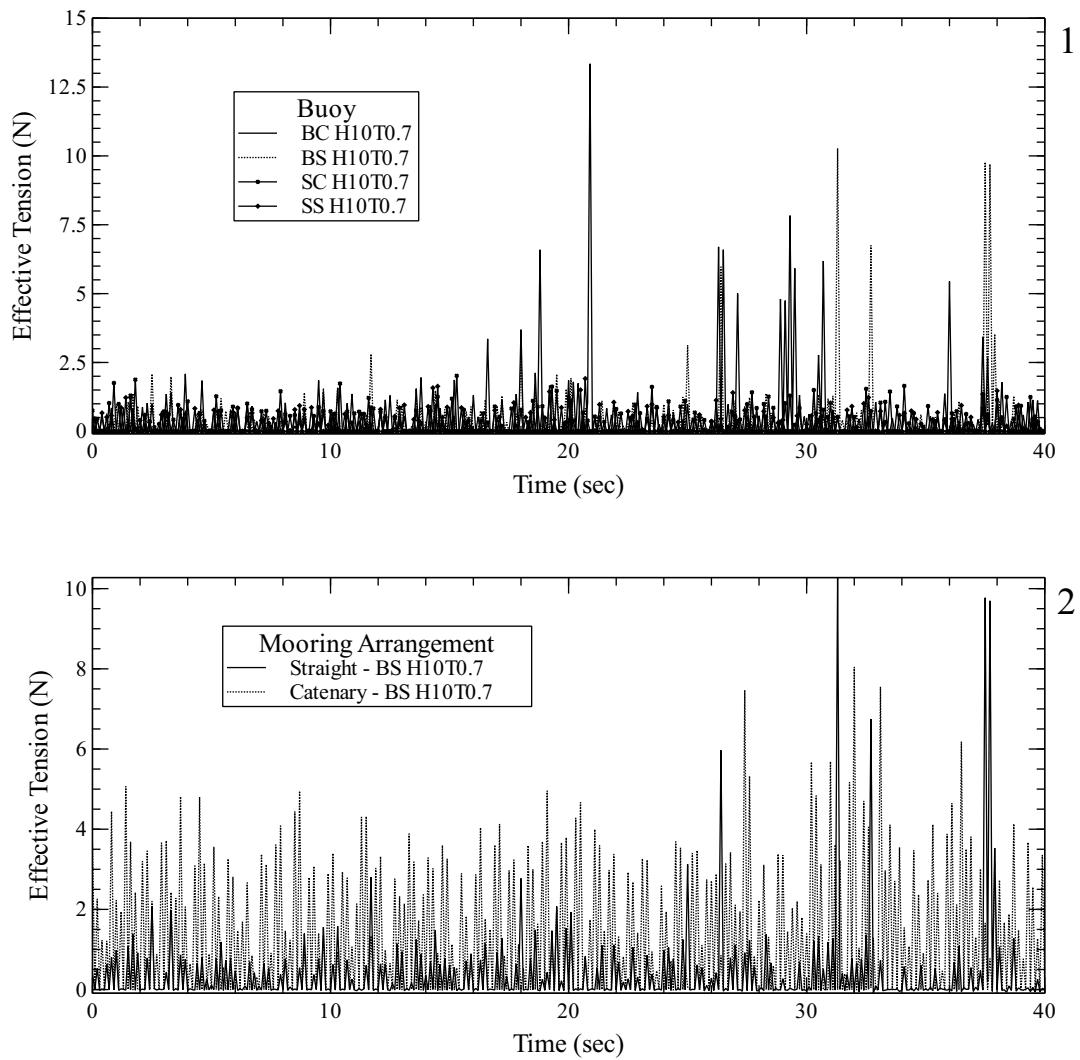


Figure 4.5.9: Results showing effective mooring line tension during OrcaFlex simulations. [1] compares different buoys using the straight mooring system; [2] compares different mooring systems using the BS buoy. All comparisons made in extreme wave conditions: height 10cm, period 0.7 seconds.

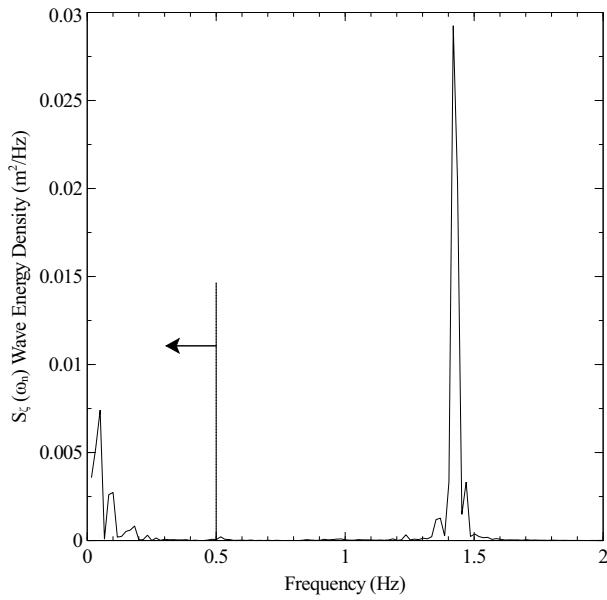


Figure 4.5.10: 2D spectrum from WABDAP for the SC buoy in 0.7 s period, 6 cm height waves. There is very little frequency spreading, which was expected.

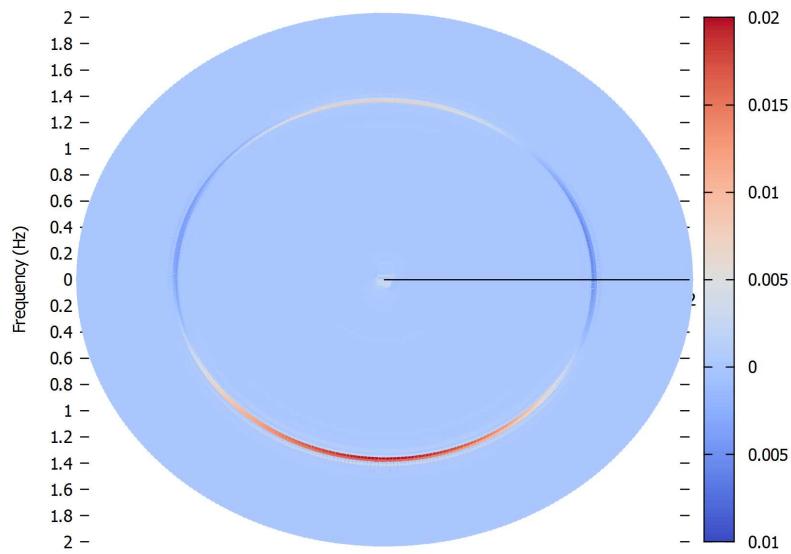


Figure 4.5.11: 3D spectrum from WABDAP for the SC buoy in 0.7 s period, 6 cm height waves. There is very little frequency spreading, which was expected, and there are promising directional results. The contour colouring is the wave energy density in terms of $\text{m}^2/\text{Hz}/\text{degree}$.

TESTING

This chapter covers the testing procedure for the model. The first phase of testing was performed in a towing tank - allowing the collection of reliable data for the vessel in calm water. The second phase was performed on a lake, from which comparisons could be made back to the towing tank tests. The lake testing also allowed the performance of the system to be assessed, in terms of its ability to perform autonomous tasks and measure its powering and manoeuvring characteristics. The results and evaluation from these tests will be performed in the following chapter, with the exception of the bollard pull tests and shaft efficiency tests. These tests were performed mostly to facilitate calibration of the sensors, which can be considered part of the testing process.

5.1 Towing Tank Tests: QinetiQ

[LE]

Towing tank tests were undertaken in order to obtain the resistance of the model and calculate the self-propulsion point in a controlled environment. Furthermore it was an opportunity to investigate an efficiency improvement method in the form of a trim optimisation study. The experiments were performed at the QinetiQ towing tank in Haslar, which is 270 m long, 12.2 m wide and 5.5 m deep. Although the tank was equipped with wave-making facilities, a towing rig with enough sensitivity to measure the small motions of the model, but also the strength to withstand the waves, was not available. Therefore tests were performed in calm water only.

5.1.1 Test matrix

[LE]

To ensure the experiments were conducted efficiently, a detailed test matrix was created prior to the event. The tank was available for testing for 7.5 hours each day. Of that, 2 hours were dedicated to setting up and packing away the model. Each run was known to take approximately 15 minutes between starting and allowing the water in the tank to return to a calm state. The length of the tank allowed for 3 different model speeds to be investigated in one run. However, one speed per run was recommended in order to obtain a more accurate result and reduce the complexity of the process.

The tests required were:

- Naked-hull tests in ballast and full-load conditions. Different trims were to be investigated subject to time constraints.
- Self-propulsion tests in ballast and full-load conditions. Different trims were also to be investigated subject to time constraints.

Additionally, at least one bollard-pull test was to be conducted to calibrate the model's thrust sensor by comparing it to the carriage dynamometer. The results from the bollard-pull experiments were used in conjunction with dry testing of the shaft without the propeller to estimate the shaft efficiency from the motor current.

The model was much smaller than those usually tested in the towing tank. Therefore there was some uncertainty as to whether the testing equipment would be sensitive enough to measure differences in

resistance, particularly in the ballast condition and in different trims. Alternative bows were prepared which could be tested if no reasonable results were obtained; this was not a preferred option as the bows were primarily designed for performance in waves.

The testing plan is detailed in Tables 5.1.1 to 5.1.4. The following notation is used: NH = naked-hull tests, SP = self-propulsion test, CG = contingency, BL = ballast condition, FL = full-load condition, DT = default trim and T1, T2, T3 = different trims.

It should be noted that, prior to the test, as can be seen in the test matrix, only one default trim was planned to be tested for the ballast condition because it was thought that the carriage dynamometer would not be able to measure small force. However, it was found during the test that the dynamometer was capable of doing so. Therefore, the number of trims in full-load condition were reduced and in exchange, an additional trim condition was added for the ballast condition.

Table 5.1.1: First half of the first day.

Type	NH						
Condition	BL_{DT}	FL_{DT}	FL_{DT}	FL_{DT}	FL_{DT}	FL_{T1}	FL_{T1}
Velocity/RPM	7.97	5.40	6.69	7.97	8.49	6.69	7.97
No of test	2	1	1	2	1	1	2

Table 5.1.2: Second half of the first day.

Type	NH	NH	NH	NH	SP	SP	SP
Condition	FL_{T1}	FL_{T2}	FL_{T2}	FL_{T2}	FL_{DT}	FL_{DT}	FL_{DT}
Velocity/RPM	8.49	6.69	7.97	8.49	900	1048	1200
No of test	1	1	2	1	1	2	1

Table 5.1.3: First half of the second day.

Type	NH	NH	NH	SP	SP	SP	SP
Condition	FL_{T3}	FL_{T3}	FL_{T3}	FL_{T2}	FL_{T2}	FL_{T2}	FL_{T3}
Velocity/RPM	6.69	7.97	8.49	900	1048	1200	900
No of test	1	2	1	1	2	1	1

Table 5.1.4: Second half of the second day.

Type	SP	SP	SP	SP	SP	CG	CG
Condition	FL_{T3}	FL_{T3}	BL_{DT}	BL_{DT}	BL_{DT}		
Velocity/RPM	1048	1200	900	1048	1200		
No of test	2	1	1	2	1		

5.1.2 Bollard Pull

[MOZDEN]

A bollard pull can be used to determine the actual thrust delivered by the propeller by measuring the pulling force of a craft. It is typically used at full scale for rating tugboats when they are built as it can be used to compare the performance of different models. However a similar procedure was performed in the towing tank to calibrate the on-board thrust sensor and ensure it produced accurate thrust readings across the full operating range of the motor.

The towing tank procedure equated the raw integers produced by the load cell on-board the model (M_{raw}) to the thrust value given by the carriage tow post (T_c) for a series of propeller velocities up to the maximum speed of the motor. The calibration could assume the carriage tow post was accurate; as one of the leading towing tanks in the country, QinetiQ ensure their rig is extremely accurate by regular checks. It was therefore possible to match the model's system to the carriage using the relationship in Equation 5.1.1, where m and c are the gradient and the y-intercept of Figure 3.4.8.

$$M_{raw} = mT_c + c \quad (5.1.1)$$

The hull was affixed to the QinetiQ tow post using their single post attachment on the thrust line at the LCB. It was towed partway down the tank so that wave reflections did not interfere with the tests and once the level had settled the propeller was run at various RPMs for 30 seconds each; the average RPM and M_{raw} were recorded from the ship and the average thrust was recorded from QinetiQ's thrust rig. This was performed for full and ballast conditions; Figure 5.1.1 and Tables 5.1.6 and 5.1.7 show the results of each run, with table 5.1.5 showing the final calibration coefficients for the system. Based on the applied calibration coefficients, the thrust system has a very high accuracy of $\pm 5\%$.

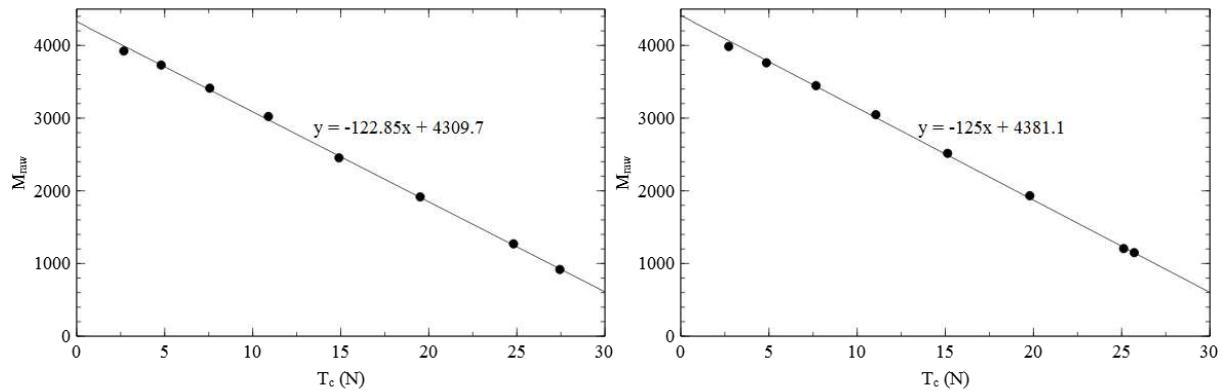


Figure 5.1.1: Calibration graph for full load [left] and ballast condition [right].

Table 5.1.5: QinetiQ thrust calibration coefficients.

	Full	Ballast	Average
m	-122.85	-125	-123.925
c	4309.7	4381.1	4345.4

The maximum value for the digital signal M_{raw} is 4095 (12-bit), and it can be appreciated from 5.1.1 that the lower thrust values were out of this range. This was because the gain was set too high, as mentioned in Section 3.4.5. For future tests it was important to reduce the gain to bring the zero-point back into range, so that the zero-offset could be tracked throughout the day and used as a datum before

Table 5.1.6: Qinetiq bollard pull data (full load, level trim).

Averaged RPM	Motor Current	RPM ²	M _{raw}	T _c (N)	T _m (N)	% Errors
0.00	0.0	0.00	4053.79	0.00	0.00	0.00
503.55	2.23	253557.9	3923.16	2.68	3.41	21.34
674.61	2.33	455098.9	3730.27	4.80	4.96	3.30
833.73	2.74	695103.3	3411.91	7.56	7.53	-0.36
1000.48	3.68	1000956.5	3023.04	10.89	10.67	-2.06
1155.71	4.22	1335670.3	2453.63	14.90	15.27	2.39
1330.06	4.19	1769056.2	1916.68	19.51	19.60	0.45
1493.63	4.21	2230918.1	1270.34	24.81	24.81	0.02
1565.86	4.21	2451907.4	917.64	27.44	27.66	0.80

Table 5.1.7: Qinetiq bollard pull data (ballast load, level trim).

Programmed RPM	Averaged RPM	Motor Current	RPM ²	M _{raw}	T _c (N)	T _m (N)	% Errors
0	0.00	0.0	0.00	4040.00	0.00	0.00	0.00
514 (300)	510.84	2.18	260956.0	3985.24	2.72	2.91	6.41
686 (400)	681.94	2.32	465041.5	3760.68	4.86	4.72	-3.00
857 (500)	847.92	2.64	718971.1	3446.70	7.67	7.25	-5.76
1029 (600)	1012.30	3.43	1024748.5	3047.58	11.06	10.47	-5.61
1200 (700)	1179.03	4.21	1390112.7	2515.71	15.13	14.76	-2.48
1371 (800)	1336.34	4.21	1785813.6	1932.67	19.79	19.47	-1.65
1543 (900)	1511.43	4.21	2284417.8	1205.20	25.11	25.34	0.91
1714 (1000)	1501.68	4.21	2255039.4	1149.37	25.71	25.79	0.31

each test.

To determine the quality of the data, the thrust relationship could be used. As shown by Equation 2.1.8, thrust has a linear relationship to RPM². By plotting this relationship it was possible to show that K_T did not vary through the runs; density and propeller diameter were fixed values, hence thrust coefficient was the only potential variable which could be affected by errors (Figure 5.1.2).

$$T = K_T \cdot \rho \cdot n^2 \cdot D^4 \quad (2.1.8)$$

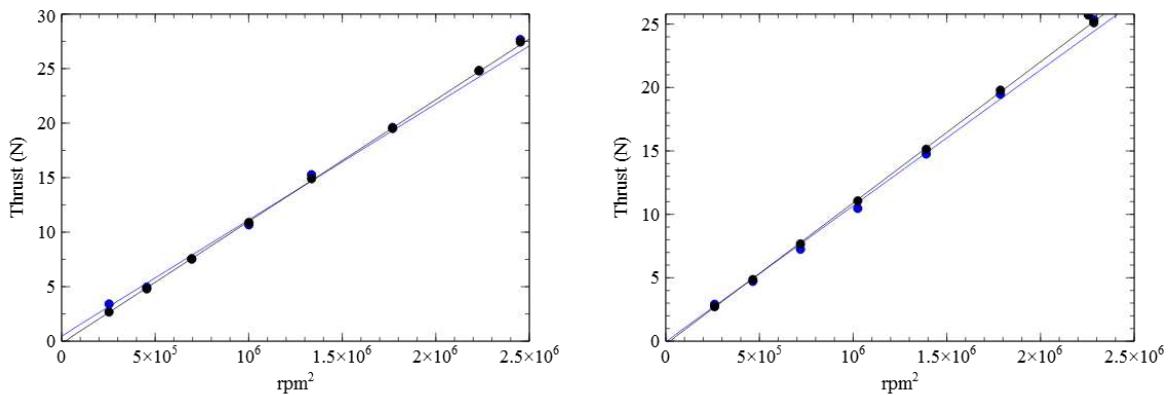


Figure 5.1.2: Thrust relationship for full load [left] and ballast condition [right]. Black lines are the carriage readings, blue lines are from the model thrust block.

5.1.3 Shaft Efficiency

[HAWKES]

In the absence of the torque dynamometer it was possible to measure the torque supplied to the shaft by using the input motor current. Since, in all DC motors, voltage is proportional to rotational speed and current is proportional to torque, the current could be directly related to the shaft torque. The manufacturer supplied a torque constant of 16.4 mNm/A, which was effectively the *motor efficiency*, since it related input power to output power.

Using the measured value for motor current, the shaft torque could then be found. Unfortunately this was not the same as propeller torque (which was the desired variable) but a technique was developed to find the *shaft efficiency*, which would allow the propeller torque to be found from the shaft torque.

When the propeller was detached, the motor and shaft could still be run, and it took a substantial amount of current (and therefore shaft torque) in order to do this. All of this power was being used to turn the shaft against its bearings. When the propeller was attached, the increase in measured torque should be entirely due to the propeller - therefore, the shaft efficiency could be found.

This was not an entirely true assumption, since the shaft efficiency will depend on many factors and could change when the propeller was added, but was nevertheless a good method to estimate the shaft efficiency.

The motor current was measured with the propeller attached during the bollard pull, at various RPMs, and out of the water with no propeller attached. There was no need to convert to torque using the motors torque constant, since the motor efficiency was a common factor to both measurements. Figure 5.1.3 shows the results of the experiment. Note that the current sensor's gain was optimized for normal operating conditions and that the bollard pull results exceeded these considerably since the motor was given up to full power (150 W) - so only the lower end of the bollard pull was used.

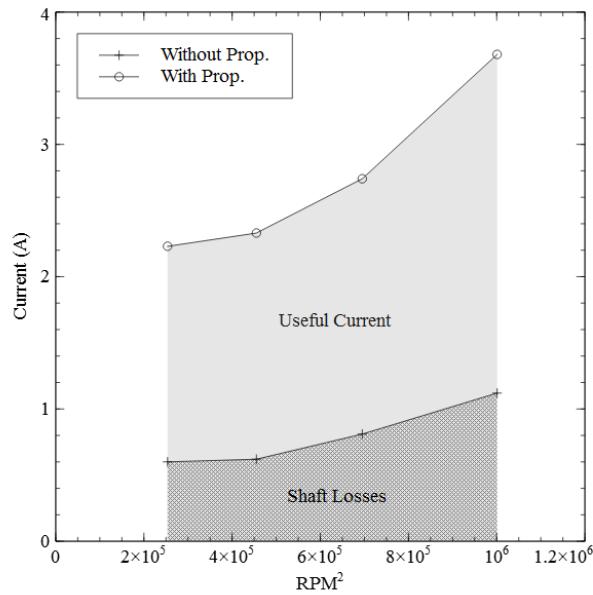


Figure 5.1.3: A plot of the current consumed by the motor with the propeller attached and submerged, and with the propeller detached.

Table 5.1.8 shows the calculations performed to find the shaft efficiency. As it turns out, the percentage of power absorbed by the shaft stayed roughly constant and averaged to $\approx 29\%$. This gave a shaft efficiency of 71%.

This is quite low compared to a full-size ship, which would aim for at least 95% efficiency but this was to be expected since the shaft had to rotate much faster than at full scale. At least, at model scale, the shaft efficiency could be measured - at full scale this would be very difficult and costly.

Using this information, the propeller torque could be found from the motor current by using the motor's torque constant and by applying the shaft efficiency.

Table 5.1.8: Measured values of motor current during the bollard pull with the propeller attached, and with the propeller removed.

RPM ²	With Prop (A)	Without Prop (A)	Ratio
253557.9	2.23	0.60	0.269
455098.9	2.33	0.62	0.266
695103.3	2.74	0.81	0.296
1000956.5	3.68	1.12	0.304
Average:			0.286

5.1.4 Naked Hull

[LE]

The aim of this test was to determine the naked-hull resistance of the model which could be scaled up to estimate the naked-hull resistance of the full scale ship and the effective power requirement P_E . Also, the naked-hull resistance of the ship R_{Ts} will be used to determine propeller speed in the self-propulsion test.

There are two main methods to obtain the resistance of the ship by model tests, one is called the traditional approach (or Froude method) and the other is called form factor approach (or Hughes' method), which was adopted by the International Towing Tank Conference according to Molland et al. [2011].

The form factor approach requires testing the model with very low Froude number, however the carriage dynamometer of the QinetiQ tank is unable to provide accurate results with low drag. Therefore the form factor approach was not chosen to be the first choice. As a result, the traditional approach was applied for all four conditions to calculate the ship's resistance and the Hughes' method was only used for the full load condition for the purpose of comparison.

The most important speed was the speed at operation condition: 15.5 knots, corresponding to the model running at 1.029 m/s. The maximum speed of the tanker is 16.5 knots which corresponded to a model testing speed: 1.096 m/s. An option for the lower speed (13 knots) was tested, to account for the tanker operating at a lower speed in the future in case it needs to save energy when the fuel's price is increasing. Other small speed tests were conducted in full load condition to determine the form factor according to Hughes' method.

5.1.5 Self-Propulsion Tests [LE]

Similar to the naked-hull tests, self-propulsion tests were conducted with 4 different conditions. In each condition, due to time constraints, the model was towed at only one speed (1.029 m/s) but with different propeller revolutions.

The purpose of this test was to find the self-propulsion point of the model and the vessel. The model self-propulsion point was the propeller RPM at which the thrust generated by the propeller is equal to the resistance of the underwater body. The ship's self-propulsion point accounts for the skin friction correction force. The ship self-propulsion point was used to determine quasi-propulsive efficiency whilst the model self-propulsion point provided a starting point for the lake tests.

For each test, propeller RPM, strain-gauge thrust, motor current, and the carriage dynamometer force was measured. The strain-gauge thrust was then calibrated according to the formula derived from the bollard pull to find the real thrust generated by the propeller. The torque was derived from the motor current by multiplying it by the torque constant, gear ratio and shaft efficiency.

$$Q = \text{motor current} \times 0.0164 \times 3.5 \times 0.71 \quad Nm \quad (5.1.2)$$

Molland et al. [2011] mentioned that if the total resistance complied with Froude's law then the ship self-propulsion point would be the same as the model self-propulsion point. However, the skin friction resistance coefficients of model and ship are different. In order to determine the ship self-propulsion point, the skin friction correction force must be calculated and offset on the diagram shown in Figure 5.1.4.

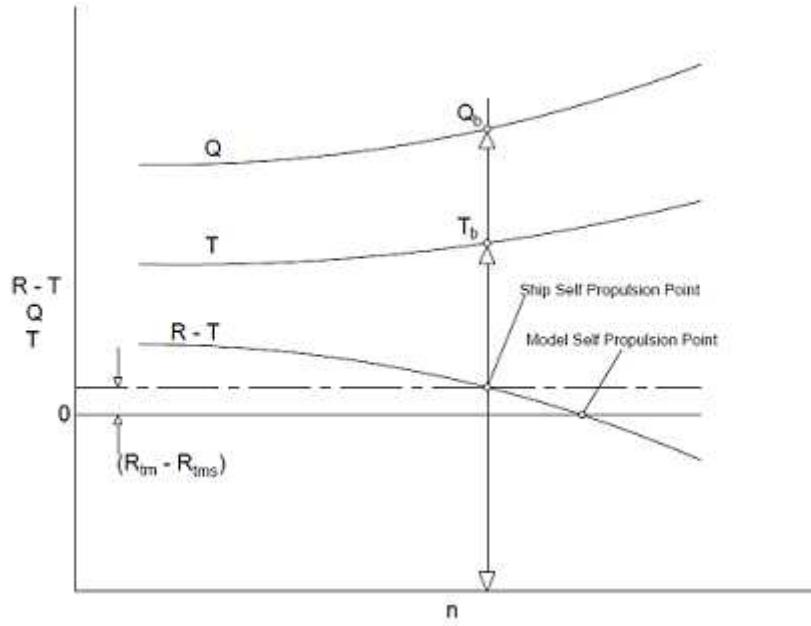


Figure 5.1.4: Diagrammatic representation of a self-propulsion test result [Molland et al., 2011].

According to ITTC [2002b], the skin friction correction force takes into account the difference in skin friction coefficients between the model and the full scale ship.

$$\Delta C_F = R_{Tm} - R_{Tms} = \rho_f \times WSA_m \times V_m \times (C_{Tm} - C_{Ts}) \quad (5.1.3)$$

$$= R_{Tm} - \frac{\rho_f}{\rho_s} \times \frac{R_{Ts}}{\lambda^3} \quad (5.1.4)$$

where R_{Tm} and R_{Ts} can be found in the table of naked-hull test results.

For instance, in the FL_{DT} condition, the offset amount should be found from the chart (figure 5.1.4) as

$$R - T = \Delta_{CF} = 6.09 - \frac{1000}{1025} \times \frac{739008}{60^3} = 2.752(N) \quad (5.1.5)$$

5.1.6 Summary

[LE]

The tank tests were successful as all the key tests were completed. Fewer trim conditions were conducted than initially planned but this was due to the resistance change for different trims being insignificant.

- Two bollard pulls were performed in ballast and full-load conditions for the thrust calibration.
- Naked-hull tests were conducted for both ballast and full-load conditions in the default and alternative trim states. The length of each run was long enough for the data to be trusted. In each condition, at least two runs were conducted at the velocity corresponding to the operation speed 15.5 knots. In full-load condition, several runs were conducted at lower speed to provide information for finding form factor.
- Self-propulsion tests were conducted for all 4 conditions. The model was towed at 1.029 m/s only but with at least three different propeller revolutions. Quasi-propulsive efficiencies for all conditions were derived and compared. The model self-propulsion point was also found for the use in the lake tests.
- Had there been more time, more tests should have been conducted for towing model with different speeds, for instance 1.096 m/s, so that a correct propeller revolution for a certain model speed could be predicted.

5.2 Lake Testing: Timsbury Lake

[ANDERLINI]

Lake testing was performed at Timsbury Lake in order to judge the effectiveness of the developed model in measuring its own powering and manoeuvring characteristics, which could be compared to the full-scale ship. In addition, different bow designs, produced as part of a separate individual project at the University of Southampton by Robert Cooke, will be used to prove the versatility of the model as a testing platform.

The first two days of testing were dedicated to solving numerous issues (e.g. communications loss, compass problems, GPS issues, software errors, etc.), as expected for a novel system. Although no useful data could be sampled during this time, the time spent problem-solving paid off, as the third day of testing went smoothly - allowing the collection of a very large amount of useful measurements. As only one effective day of testing was available, it was decided to abandon the changes in trim since preliminary results from the towing tank testing showed no noticeable efficiency improvements.

Similarly, only the full load condition was used, as this would save the time required for un-ballasting and re-ballasting to and from the ballast condition - the two rudder would be tested instead. It was possible to dedicate two hours, at the end of the tests, to trial two different bow designs in straight run tests, as these were expected to show significant changes in thrust.

The types of tests performed with the tanker model in its original configurations were designed to estimate its resistance and powering characteristics, with a comparison to the self-propulsion tests at QinetiQ, and to assess its controllability. For this reason, straight run, circle and zigzag tests were performed. In all cases, three speeds were tested: one close to the expected model self-propulsion point after the tank testing at QinetiQ (Section 6.3.2) (1000 RPM), one lower and one higher. This was done in order to cover a large range of model powering with a significant sea margin, so that the desired operational speed could be met also in quite severe head and following seas, where the expected speed is lower and higher respectively.

In order to account for the changing weather conditions (particularly wind and waves strength and direction), multiple runs were performed at each motor setting (four initially later reduced to two due to time constraints) and in opposite directions.

Before beginning the actual test matrix, it was necessary to perform another bollard-pull test in order to test new gain settings for the thrust sensor.

5.2.1 Bollard Pull

[MOZDEN]

The bollard pull was repeated prior to the lake testing, in order to confirm the accuracy of the system. The performance of the thrust sensor was improved between the QinetiQ tests and Timsbury Lake tests by reducing the gain (see Section 5.1.2), the this meant the thrust sensor had to be recalibrated.

The ship was tested in the University of Southampton Lamont Towing Tank. As at QinetiQ, it was affixed to the tow post on the thrust line at the LCB and the propeller was run at 100 rpm increments up to 1500 rpm for 30 seconds each; the average shaft rotational speed, and M_{raw} were recorded from the ship and average thrust was taken from the carriage system. The towing tank carriage did not directly produce thrust readings; instead it gave a voltage reading (V_c) which changed according to the load. It was calibrated by measuring the reading at 0 N and then with a 10 N applied load. The relationship was assumed to be linear so a correction equation $T_c = mV_c + c$ could be used (where m and c are calibration constants). Table 5.2.1 and Figure 5.2.1 show the results to give the relationship for the carriage thrust data.

Table 5.2.1: Lamont tank carriage calibration coefficients.

Applied load (N)	Carriage reading (V)
0	2.70
10	-2.15

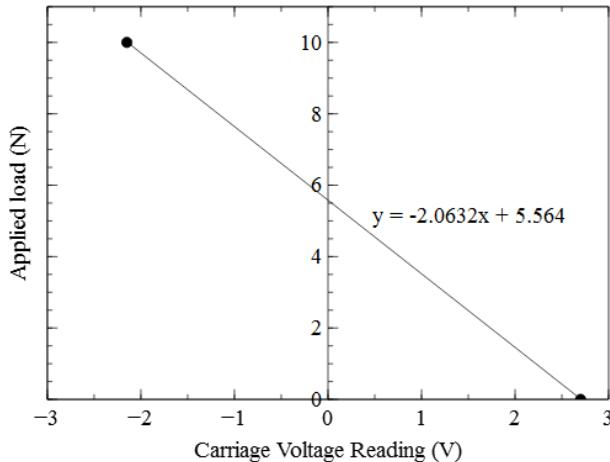


Figure 5.2.1: Carriage calibration graph to convert tow post voltage reading to thrust data.

The bollard pull was performed for the full operating range of the motor, full details in Table 5.2.2, and the calibration graph and thrust relationship are in Figure 5.2.2. The thrust relationship shows a strong linear relationship, confirming the accuracy of the set up proven in the QinetiQ bollard pull. Table 5.2.3 gives the final calibration coefficients for the on-board thrust system.

Table 5.2.2: Lamont tank bollard pull data.

Programmed rpm	Averaged rpm	rpm^2	Raw data average				
			M_{raw}	V_c	T_c (N)	T_m (N)	% Errors
0	0.00	0.0	3488.36	0.00	0.00	0.00	0.00
200	202.61	41051.6	3648.13	2.540	0.32	0.44	26.08
300	302.38	91431.5	3641.57	2.318	0.78	0.52	-49.66
400	399.73	159781.3	3583.24	2.005	1.43	1.27	-12.06
500	499.25	249248.3	3488.31	1.591	2.28	2.50	8.60
600	600.27	360325.8	3406.25	1.025	3.45	3.55	2.91
700	699.59	489428.1	3319.25	0.479	4.58	4.67	2.11
800	800.18	640285.8	3215.35	-0.216	6.01	6.01	0.06
900	899.36	808857.4	3100.68	-0.993	7.61	7.49	-1.65
1000	1000.75	1001505.6	2920.79	-1.968	9.62	9.81	1.85
1100	1099.91	1209806.8	2741.32	-2.868	11.48	12.12	5.27
1200	1199.86	1439662.6	2584.43	-3.921	13.65	14.14	3.44
1300	1299.58	1688918.1	2296.97	-5.281	16.46	17.84	7.75
1400	1348.23	1817718.5	2172.67	-5.719	17.36	19.44	10.70
1500	1340.87	1797936.1	2191.05	-5.715	17.35	19.21	9.64

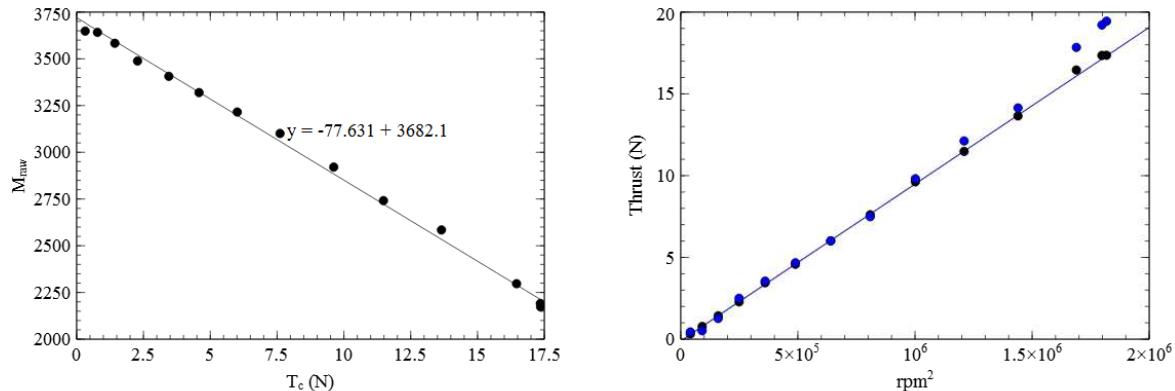


Figure 5.2.2: Calibration graph for Lamont bollard pull [left] thrust relationship [right], black lines are the carriage readings and blue lines are from the model thrust block.

Table 5.2.3: Lamont tank thrust calibration coefficients.

Full	
m	-77.631
c	3682.1

5.2.2 Straight Line Tests

[ANDERLINI]

Straight line tests were performed to assess the resistance and powering characteristics of the model. As noted with the self-propulsion towing tank tests performed at QinetiQ, during free running experiments, it was impossible to ensure that the model travels at the desired operational speed, as it was only possible to specify the predicted motor power through the propeller RPM. Therefore, these tests were performed at three motor settings: one corresponding to the propeller RPM of the predicted model self-propulsion point (Section 6.3), approximately 1000 RPM, one lower (750 RPM) and one higher (1250 RPM) to span a wide range of motor powering to be sure that the operational speed would be included even with a considerable sea margin.

The straight line tests were approximately 80 m along the lake length close to its centreline. As mentioned, multiple tests were performed to increase the data range available considering the unpredictability of the testing environment. These involved runs in different directions so that different wind effects could be observed.

In order to prove the effectiveness of the model as a testing platform as well as to assess possible gains in efficiency in waves, the straight run tests were repeated with two different bow designs, produced separately as part of an individual project at the University of Southampton.

5.2.3 Manoeuvring Trials

[ANDERLINI]

Standard manoeuvring trials have been defined by the [ITTC, 2002a] over the years to assess the stability or control characteristics of full-scale ships or models. In this project, considering the consistent stability shown by the model during both the self-propulsion (with no rudder) and free-running tests (with two different rudders, where the only corrections were due to wind effects), controlled manoeuvres were performed, namely the ‘circle’ and ‘zigzag’ [ITTC, 2002a]. Although both manoeuvres are typically performed in calm water, the selected testing days did have some wind, which may have biased the results (e.g. some ellipsoidal circles). As data for the full-scale ship was available for these runs (kindly provided by BP), it was possible to compare the model and full-scale ship non-dimensional results in order to analyse scaling effects.

Circle Manoeuvre

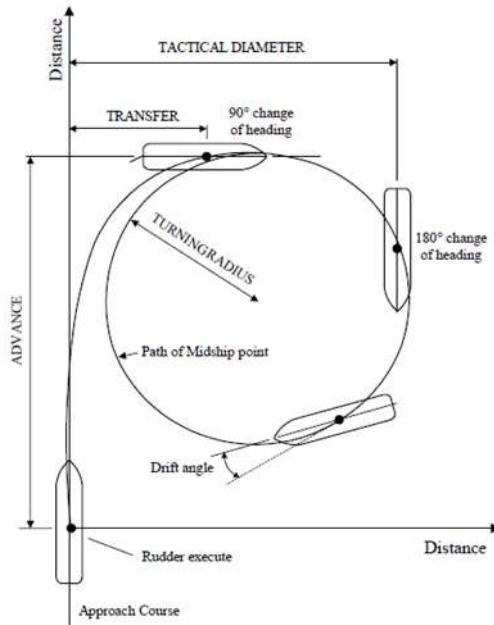


Figure 5.2.3: Typical ship path during a circle manoeuvre and main particulars to be recorded. This figure has been provided by BP Shipping.

The circle manoeuvre is initiated with a straight run at constant speed. Under constant throttle setting conditions, a selected constant rudder angle is applied [Bishop and Parkinson, 1976]. The trial consists of determining the subsequent path of the vessel during a complete 360° change of heading [Bishop and Parkinson, 1976]. The trial is repeated for different throttle settings and rudder angles and typically performed on both sides [Bishop and Parkinson, 1976].

During the circle manoeuvre it is typical to record the following details (shown in Figure 5.2.3) [Bishop and Parkinson, 1976]:

- Tactical diameter;
- Advance;
- Transfer;

- Time to change heading by 360° ;
- Speed ahead during steady conditions;
- Drift angle during steady conditions;
- Angle of heel during steady conditions.

This data is typically compared with those of successful designs. Hence, in theory, it would be possible to compare the full-scale and model ships non-dimensional results to assess how scaling affects the recorded properties. However, during the lake testing in this project, the rudder angle employed for the full-scale trials was not matched due to the limited time-scale. It is strongly suggested to modify the software so as to perform the manoeuvre with a rudder angle of 35° in the future.

If steady-state conditions are assumed during a circle manoeuvre, it is possible to estimate the Nomoto coefficients in Equation 3.6.1 in section 3.6.6 [Journée and Pinkster, 2002]. However, due to the unreliability and unsteadiness of the data relating to these trials (see Section 6.4.3, these coefficients, necessary for tuning the PID controller of the model, were derived from the zigzag manoeuvres.

Zigzag Manoeuvre

After an initial straight run when the model reaches a constant speed, at constant throttle setting the rudder is set to a predefined value to starboard “as quickly and as smoothly as possible” [Bishop and Parkinson, 1976]. As a consequence the ship changes heading and the rudder angle is kept constant until the difference in change in heading and rudder angle is 0 (i.e. if the rudder angle is 20° , the heading must change by 20° from the original course) [Bishop and Parkinson, 1976]. At this point, the rudder angle is reversed to port “as quickly and as smoothly” as possible and the new rudder angle is kept while the ship changes heading again to the other side [Bishop and Parkinson, 1976]. Once the change in heading, as compared to the reference heading during the straight run, has again changed by the magnitude of the new rudder angle, the rudder angle is again reversed “as quickly and as smoothly as possible” [Bishop and Parkinson, 1976]. The whole procedure has been repeated two times during the model tests in order to achieve a steady periodic motion (this has not been the case during full-scale trials). In general, the accepted rudder angles are 20° - 20° , 10° - 10° and 20° - 10° [ITTC, 2002a], where the first value refers to the rudder angle demand and the second one to the desired change in heading. In addition, this process is generally repeated for different throttle settings. From the periodic part of the curve, the following quantities are measured [Bishop and Parkinson, 1976]:

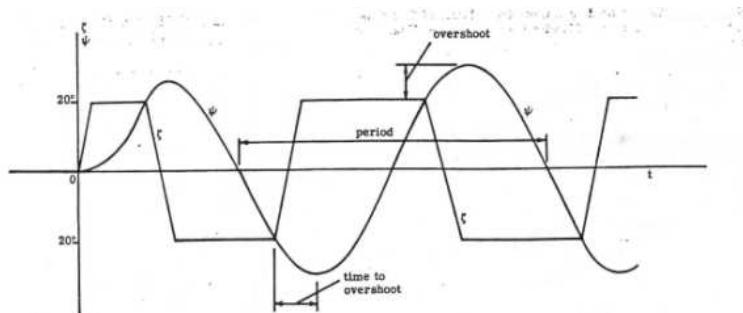


Figure 5.2.4: Typical plot of heading and rudder angles during a zigzag manoeuvre. This figure has been taken from Bishop and Parkinson [1976].

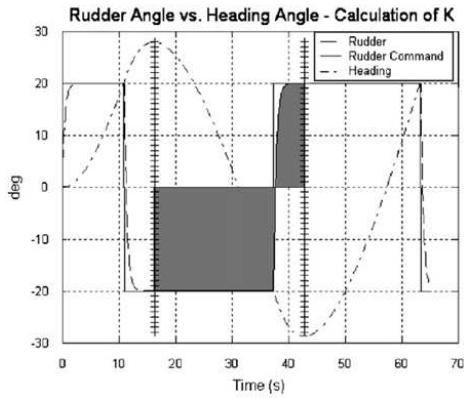


Figure 5.2.5: Derivation of K from a sample 20°-20° zigzag manoeuvre, taken from Moreira et al. [2007].

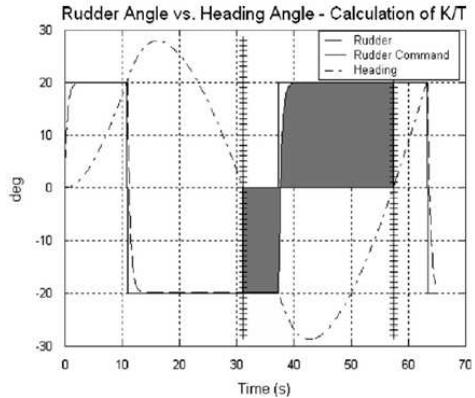


Figure 5.2.6: Derivation of K/T from a sample 20°-20° zigzag manoeuvre, taken from Moreira et al. [2007].

- Overshoot;
- Period;
- Time to overshoot.

The zigzag manoeuvring trial has been used by Nomoto in the 1960s to evaluate the K' and T' indices [Bishop and Parkinson, 1976]. Indeed, from the integration of Equation 3.6.1 in Section 3.6.6 it is possible to show that [Moreira et al., 2007]:

$$K = -\frac{\psi_1 - \psi_2}{\int_{t_1}^{t_2} \delta dt} \quad (5.2.1)$$

$$K = \frac{r_3 - r_4}{\int_{t_3}^{t_4} \delta dt} \quad (5.2.2)$$

In Equation 5.2.1, ψ_1 is the first overshoot angle and ψ_2 the second overshoot angle (negative), as shown in Figure 5.2.5 (taken from Moreira et al. [2007]). Similarly, the integral of the rudder angle is also presented. This integral has been approximated numerically with the trapezium rule [Imp]. In Equation 5.2.2, r_3 is the rate of turn at the point of zero heading after the first overshoot angle, while r_4 is the rate of turn at zero heading after the second overshoot angle, as shown in Figure 5.2.6 (taken from Moreira et al. [2007]). These have been approximated with a first-order-accurate forward finite difference [Rung, 2012]. The trapezium rule has been used to estimate the integral of the rudder angle, which can be seen in the figure, in this case as well.

The Nomoto coefficients are fundamental in the derivation of an optimal heading controller, as described in Section 3.6.6. Additionally, they may be used to assess the stability and manoeuvrability of a ship, by comparing the values with successful designs [Fossen, 2011] [Journée and Pinkster, 2002], just like the other parameters measured during the circle and zigzag manoeuvres [Bishop and Parkinson, 1976]. For this reason, the non-dimensional values of the Nomoto coefficients ($K' = KL/\bar{U}$ and $T' = T\bar{U}/L$ [Journée and Pinkster, 2002] from Section 3.6.6) were adopted to enable a comparison between the full-scale and the model vessels. T' is expected to rise with displacement (thus, for the full-scale ship). Furthermore, a reasonable standard for course-changing qualities was set by Norrbin as $K'/T' > 0.6$ [Fossen, 2011], while $K'/T' > 0.4$ seems to be sufficient for tankers.

5.2.4 Summary

[HAWKES]

The lake testing consisted of two days of trouble-shooting, which was to be expected; and one day of useful testing. The main culprits for the problems were the wireless link to the model, the compass and the GPS.

The compass caused issues due to calibration and its position in the model. It was dramatically affected by the power cables in the pelicase. This was not easy to discover, but fortunately easy to solve - by re-routing some cables.

The wireless link caused problems by restricting the communication range to about 20m. This was 'solved' by taking a laptop on the support boat and staying close to the model, which was not ideal and may have interfered with results. The wireless link should be thoroughly investigated before future testing, since even with the aerial above-deck the range was very limited.

The GPS also caused problems, it failed to get a satellite fix for almost an hour and it appears that the data from the GPS was unreliable - both in terms of position and speed. Presumably the GPS did not have a good fix during the tests, even when it was placed above-deck. Either the lake is in a poor location for GPS tracking or the device was not performing well due to interference or other factors.

This problem with the GPS meant that the circle manoeuvre data was of poor quality. Fortunately, the zigzag results were promising and produced very sensible mathematical models of the ship motions so that a controller could be designed.

A good array of straight-line tests were performed, which allowed comparisons to the towing-tank data to be carried out; as well as comparisons between different bow sections.

The weather conditions were mild but unpredictable. For most of the three days the lake was flat calm, with occasional gusts creating small wave patterns. Unfortunately the wave buoys were not completed in time for the lake testing. An attempt was made to measure the significant wave height using a ruler and a video camera, but the conditions were so changeable that the results could not be lined up to the vessel's behaviour.

The average wind-speed for the testing day was measured to be approximately 9.2 m/s using an on-shore anemometer, but this was also very changeable and there were some manoeuvres that were performed in significantly higher (or lower) wind-speeds.

As explained, multiple runs were performed to average-out these anomalies, but in the future it would be best to test the model alongside the wave buoys and an anemometer (mounted on either the vessel or the buoys); and it should also be possible to fully automate the entire test matrix allowing much more repetition (*i.e.* 20 runs per test, rather than 2 or 4).

RESULTS AND EVALUATION

In this chapter the results of the testing are discussed. This does not just include the results of the various manoeuvres and trials, but also the results of how the system performed as a test platform. This chapter will begin with a discussion on the system and sensor performance, which will help to evaluate the remainder of the results. The results are then split into the powering results, which includes all the naked hull, self-propulsion and efficiency improvement tests; and manoeuvring results including a rudder comparison.

6.1 System Performance

[HAWKES]

In this section, the aim is to analyse the performance of the system. The intention was that the model could perform autonomous testing on a lake, and measure its own characteristics in terms of powering and manoeuvring as it did so.

In general, the model was highly able and managed to perform all the tests required both in the towing tank environment and the open-water environment. However, it also had some significant problems and reliability issues which will be discussed here.

6.1.1 Power Train

[HAWKES & MOZDEN]

The motor performed well, capable of providing much more than the required RPM and torque. The power train system, mounted on its linear rails appeared effective at measuring thrust in a reliable way, as demonstrated by the highly repeatable bollard pull experiments.

The only limitation was that the thrust bearing only transferred a forwards thrust; this meant using the propeller in reverse was ineffective. Future developments should include a multi-directional thrust bearing to allow reverse power to be transferred efficiently to the ship.

6.1.2 Torque Dynamometer

[HAWKES]

The torque dynamometer was not ready in time for either the tank testing or lake testing. This was due to a delay in the manufacture process and problems with the dynamometer wiring. It is hoped that in the future this device could be fitted and calibrated using some of the data obtained from the tank testing.

The current sensor performed well, and allowed the torque to be calculated for the self-propulsion tests; so the project could continue as planned.

6.1.3 Electronics

[HAWKES]

The electronics performed reliably, with the only problem occurring due to a faulty connection to the motor - which probably arose during transit. About 30 minutes of tank testing was lost due to this problem, which was easy to fix but difficult to locate - initially it was believed to be a software problem.

There were also some problems with the sensors implemented, due to human error, but this will be covered in Section 6.2.

The battery life was not explicitly tested, since there was no opportunity to leave the entire system powered on in continuous operation for the required amount of time; however, the battery life was definitely adequate.

As mentioned earlier however, the main reason for using multiple batteries was to prevent the load voltage dropping too low when the motor was powered up. During the QinetiQ towing tank self-propulsion tests, two batteries were used since the tow post arrangement took up too much space to allow all four batteries to be used. During the bollard pull, when the batteries were pushed to their limits, the battery voltage dropped to 10.6V even with two batteries; limiting the maximum RPM. Fortunately the voltage was still high enough to easily pass the self-propulsion point. However, during non-critical testing one battery was used, and the voltage dropped as low as 9V - suggesting the drop to be (approximately) a linear relationship to the number of batteries. Also, for these tests the autonomy unit was powering itself from the shore, significantly reducing the load on the batteries.

Four batteries were connected for the lake trials and during testing on the final day the voltage did not drop below 11.4 V. This was much more acceptable, especially considering that this was now powering all of the autonomy system.

6.1.4 Autonomy System

[HAWKES]

The autonomy system, which includes the autonomy computer and its peripherals was fine for its data-logging task in the towing tank, but had some problems associated with the wireless transmission when performing lake testing.

The wireless link was unreliable and did not meet the required transmission range. This could have been due to interference, which is quite unlikely since the wireless signal was very high power - the wireless unit was more likely to *cause* interference problems rather than be affected by them. The workaround was to position the support boat close to the model at all times, with the 'shore'-based laptop on-board. This worked, but was not ideal since the support boat often interfered with the model. In the future, it is recommended that the wireless antenna be placed on a mast above the vessel to ensure maximum range and reduce its interference to other sensors. Another improvement would be to add an external antenna (probably requiring an external wireless card) to the shore laptop, giving it a much needed boost in power.

6.2 Sensor Performance

[HAWKES]

Another important factor was the reliability and accuracy of the sensors employed on the vessel, on which many of the results depended. This can be analysed statistically by taking an example of the data from the lake testing. Table 6.2.1 shows a number of different straight-line runs that the model undertook on the final day of testing. For each run, the minimum, maximum, mean and standard deviation of the results are presented.

The skew and kurtosis (flatness) of the results were also calculated for a few runs, but these gave very little additional information (*i.e.* $S \approx 0$, $K \approx 0$) and were omitted from the table.

RPM stayed very close to the mean, which in turn stayed very close to the target value (750, 1000 or 1250), with the minimum and maximum RPM typically within 5% of the average value. Standard deviation was also low, but increased slightly at higher speeds. This distribution was most likely due to the actual RPM fluctuations rather than a poor sensor - since the optical encoder was very precise. The MCU was timing 30 counts of the encoder, to the nearest microsecond. At 1000 RPM, the elapsed time for 30 counts would be 1.7 ms, giving a resolution of 0.05% for the RPM. At 1250 RPM this rose to 0.07%. However, the PWM amplifier had a limited resolution of 127 steps (0.7% or 15 RPM), due to its 7-bit logic, between full RPM and zero RPM.

The motor module would have been correcting the RPM so fast between the two closest ‘steps’ to the required RPM, that the motor would have experienced a signal that appeared to be within the two voltage steps. However, the feedback loop may have experienced some delay, either through hardware or software lag, causing an occasional overshoot.

Further investigation could be done, by taking a Fourier transform of a typical signal to identify any significant frequencies. Unfortunately, no significant low frequencies that could indicate such a lag, were visible - as shown in figure 6.2.1.

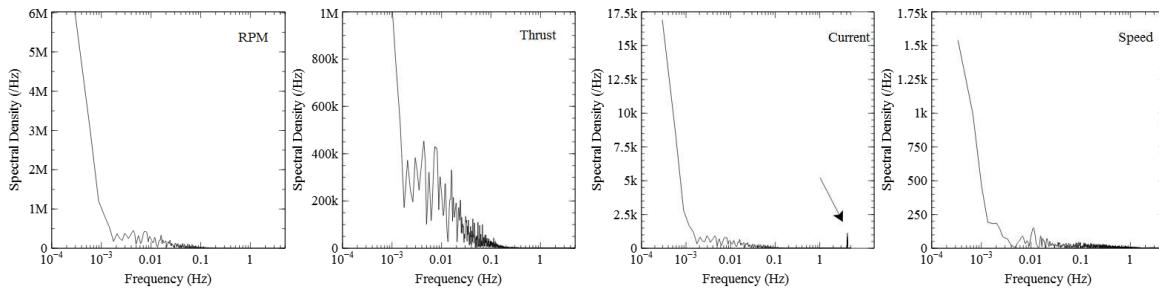


Figure 6.2.1: Fourier series of a typical RPM, thrust, current and speed data set.

However, the effective sample rate was just 10 Hz, so only frequencies up to 5 Hz could be measured (according to the Nyquist principle). There were no noticeable peaks in the spectral density function up to 5 Hz, in fact, there was virtually no spectral energy beyond 0.1 Hz. The Fourier transform showed that this RPM signal was very low noise since most of the spectral energy was focused at zero frequency (*i.e.* the constant mean).

The **Motor Current** also showed good results with a standard deviation between 1-1.2%. The current supplied to the motor was not controlled, it simply drew as much as it needed to maintain the required RPM (which was set via the voltage). This meant it was an externally driven signal and was likely to experience environmental noise.

The motor current was also sampled at 10 Hz by the autonomy unit, despite being sampled at a much higher frequency on the MCU. The moving average performed on the MCU acts as a low-pass filter,

removing all frequencies higher than 10 Hz.

For this signal, it would have been interesting to log the data as it was recorded (at approximately 250 Hz in this case), as this should show the low-pass filtering created by the R-C filter.

The motor RPM in the trial used in Figure 6.2.1 was 1000 RPM, or 16.67 revolutions per second. This was out of the measurable range, but being a four-bladed propeller the blade frequency would be 4.17 Hz - encouragingly there was a small peak at this location as marked by the arrow.

The motor voltage followed a very similar pattern, which - combined with the motor current results gave motor power: $P_M = I_M V_M$. The standard deviation for motor power was slightly higher, since it was a combination of two sensor signals.

Thrust was expected to be less reliable than the RPM and the various currents and voltages measured around the vessel. This was because the strain gauge would be sensitive to physical vibrations of the vessel - caused by the motor, shaft, propeller or other sources.

As shown in Figure 6.2.1, the signal was certainly noisier but this noise occurred at very low frequencies - below the propeller or blade frequency. In fact, most of the oscillation in the measurements was on a time-scale of at least 10 seconds. This was probably due to a time-lag in the strain gauge response, caused by the friction of the linear rails. This friction and time-lag may be the reason that the blade frequency was not visible: since the friction acts as a damper to the system it could have created a low-pass filter suppressing this frequency.

The **GPS**, which provided speed data for these runs, was clearly not functioning properly since the range of speed values was very large - and the model clearly did not undergo such large speed changes. The standard deviation was also up to 35% for some cases, indicating that the GPS had a very poor satellite fix. The Fourier series in Figure 6.2.1 does not give much insight into the signal since the GPS was only updating very slowly.

The combination of RPM, thrust, current (hence torque) and speed can be used to evaluate the results; and are the four components of Equation 2.1.1 which gives the propulsive efficiency:

$$\eta_T = \frac{T_E u}{Qn}$$

Of these four values, the thrust, torque and RPM could be found reliably; but the speed was far from accurate. This will be taken into account in the following sections, which discuss the results of the powering and manoeuvring experiments.

Table 6.2.1: A sample of straight-line lake-tests showing mean, max, min and standard deviation (%)

	Run	Speed	Thrust	Motor Current	Motor Power	Propeller RPM
Mean	1	0.637	5.123	2.139	9.75	750.43
Max	1	1.019	6.738	2.315	10.76	776.76
Min	1	0.139	3.183	1.998	9.03	721.24
Deviation (%)	1	24.5%	18.7%	1.1%	2.3%	2.7%
Mean	2	0.758	5.058	2.092	9.50	750.29
Max	2	1.286	6.429	2.270	10.36	784.81
Min	2	0.288	3.247	1.971	8.81	726.69
Deviation (%)	2	29.7%	19.0%	1.1%	2.1%	2.6%
Mean	3	0.640	5.141	2.117	9.62	750.31
Max	3	0.931	6.494	2.248	10.37	777.70
Min	3	0.370	3.247	1.996	9.01	728.54
Deviation (%)	3	19.1%	18.1%	1.1%	2.0%	2.4%
Mean	4	0.602	5.099	2.116	9.64	750.43
Max	4	1.111	6.506	2.240	10.27	780.88
Min	4	0.123	3.325	1.997	8.98	725.36
Deviation (%)	4	34.6%	18.6%	1.2%	2.0%	2.4%
Mean	5	0.744	8.377	2.221	13.65	1000.49
Max	5	1.214	10.757	2.558	15.67	1026.97
Min	5	0.437	6.919	2.008	12.14	976.01
Deviation (%)	5	25.5%	9.4%	1.1%	4.1%	4.7%
Mean	6	0.865	8.247	2.398	14.91	1000.85
Max	6	1.219	9.572	2.651	16.01	1029.69
Min	6	0.530	7.060	2.223	13.70	973.66
Deviation (%)	6	19.3%	8.0%	1.0%	2.8%	2.9%
Mean	7	0.918	8.108	2.378	14.78	1000.85
Max	7	1.307	9.559	2.588	16.08	1032.33
Min	7	0.592	6.816	2.203	13.56	977.58
Deviation (%)	7	17.1%	7.9%	1.1%	2.5%	3.0%
Mean	8	0.745	8.269	2.262	14.11	1000.52
Max	8	1.132	10.422	2.629	16.42	1030.91
Min	8	0.314	6.970	1.978	12.23	973.74
Deviation (%)	8	20.1%	8.6%	1.1%	5.3%	5.7%
Mean	9	1.067	13.048	3.226	25.93	1250.10
Max	9	1.260	17.675	3.809	31.08	1284.20
Min	9	0.520	10.422	2.839	22.58	1215.18
Deviation (%)	9	15.6%	11.6%	1.0%	6.1%	6.6%
Mean	10	1.015	13.248	3.408	27.41	1250.24
Max	10	1.353	18.744	4.004	33.25	1288.13
Min	10	0.602	10.371	3.035	24.14	1210.14
Deviation (%)	10	19.2%	12.6%	1.2%	5.1%	5.5%
Mean	11	1.071	13.080	3.345	27.01	1250.27
Max	11	1.312	17.984	3.855	31.61	1286.92
Min	11	0.710	10.397	2.959	23.64	1213.23
Deviation (%)	11	13.5%	12.1%	1.1%	5.1%	5.6%
Mean	12	1.030	13.241	3.320	26.90	1249.94
Max	12	1.338	17.069	3.641	30.03	1284.42
Min	12	0.658	10.564	3.085	24.88	1214.60
Deviation (%)	12	15.0%	11.8%	1.1%	2.7%	3.1%

6.3 Resistance and Propulsion

[LE]

6.3.1 QinetiQ: Naked-hull tests

In each condition, the model was towed at different speeds and drag was measured from the carriage dynamometer. For each speed, several runs were conducted and the average value of the drag was taken. Drag measured from the dynamometer was considered as the total resistance of the model.

Full Load Condition

Traditional Approach: The Froude method was used to derive total resistance of the ship and hence the power requirements (see Table 6.3.1). The following formulae and assumptions were used:

$$C_{Tm} = \frac{Drag}{\rho_f v^2 WSA} \quad (6.3.1)$$

$$C_F = \frac{0.075}{[\log_{10} Re - 2]^2} \quad (6.3.2)$$

The water temperature at the beginning of the tests was 12.3° , falling to 12.1° by the end. Therefore, the change in density of fresh water (ρ_f) due to temperature was considered negligible. The dynamic viscosities of the fresh water and salt water were taken to be 1.19×10^{-6} Ns/m² and 1.14×10^{-6} Ns/m² respectively.

Table 6.3.1: Effective power requirements of the model for different speeds for FL_{DT} condition.

V[knots]	Drag [N]	C_{Tm}	C_{Fm}	C_R	C_{Fs}	C_{Ts}	R_{Ts} [N]	P_E [kW]
7.50	1.48	5.40E-03	4.47E-03	9.29E-04	1.65E-03	2.57E-03	156297	603
10.50	2.87	5.34E-03	4.17E-03	1.18E-03	1.58E-03	2.75E-03	327459	1768
13.00	4.23	5.14E-03	3.99E-03	1.15E-03	1.53E-03	2.68E-03	488984	3269
15.49	6.09	5.20E-03	3.85E-03	1.35E-03	1.50E-03	2.85E-03	739008	5890
16.50	7.06	5.32E-03	3.81E-03	1.51E-03	1.49E-03	3.00E-03	882368	7491

Form Factor Approach: The model was run at very low Froude number and according to Molland et al. [2011]

$$C_T = (1 + k)C_F + AF_n^4 \quad (6.3.3)$$

$$C_T/C_F = (1 + k)AF_n^4/C_F \quad (6.3.4)$$

The detailed calculation is presented in Table 6.3.2.

Table 6.3.2: Form factor derivation.

F_n	C_T/C_F	F_n^4/C_F
0.093368	1.21E+00	1.70E-02
0.130677	1.28E+00	7.00E-02

The form factor was calculated to be $k = 1.18$ which is significantly lower than preliminary prediction of all three methods including Watanabe's empirical formula. With this form factor

$$C_w = C_{Tm} - (1 + k)C_{Fm} = 6.56E-04 \quad (6.3.5)$$

and

$$C_{Ts} = C_w + (1 + k)C_{Fs} = 2.43E-03 \quad (6.3.6)$$

This results in an effective power requirement of 5016 kW, which is 14% lower than the traditional method. The form factor from Watanabe's empirical formula gave an effective power requirement of 4191 kW. The predicted form factor (i.e 1.425) and the form factor taken from Collison et al. [2012] (i.e. 1.388) would give negative wave resistance coefficient. Hence, they have been omitted. The results from the different method will be discussed further in the self-propulsion section.

6.3.2 QinetiQ: Self-Propulsion Tests

[LE]

Shaft revolution (RPM), thrust and torque were measured in the experiments and calibrated according to an empirical formula. The procedure of calibration was discussed previously. Thrust and torque coefficients are derived as follows:

$$K_{Tb} = \frac{T_b}{\rho n^2 D^4} \quad (6.3.7)$$

$$K_{Qb} = \frac{Q_b}{\rho n^2 D^5} \quad (6.3.8)$$

Data in this section is taken for the case of the resistance calculated by the traditional method ($k = 1$). For the resistance calculated by form factor approach, results will be shown in Table 6.3.5.

Full Load Condition

The skin friction correction force for the full load condition was calculated earlier as:

$$\Delta_{C_F} = R - T = 2.75(N) \quad (6.3.9)$$

Using Table 6.3.3, the required thrust, torque and hence K_T and K_Q can be determined by interpolating at the point where $R - T = 2.75(N)$. The results can be seen in Table 6.3.4 where $rpm = 805$, $K_{Tb} = 0.2471$ and $K_{Qb} = 0.0516$. The ship's self-propulsion point (RPM) can also be observed from Figure 6.3.1.

Table 6.3.3: Typical results for the FL_{DT} condition.

RPS	RPM	T_b (N)	Q_b (Nm)	R-T (N)	K_{Tb}	K_{Qb}
6.99	419	2.43	0.067748	5.86	0.497033	0.13861
11.31	679	3.34	0.087265	4.03	0.261115	0.068171
12.71	762	3.97	0.089381	3.26	0.246058	0.055345
14.09	846	4.93	0.095153	2.26	0.248194	0.047914
19.49	1169	10.34	0.145706	-2.96	0.272286	0.038366

Table 6.3.4: Interpolation value at self-propulsion point.

R-T	RPS	RPM	T	Q	K_T	K_Q
3.26	12.71	762	3.97	0.089381	0.2461	0.055345
2.26	14.09	846	4.93	0.095153	0.2482	0.047914
2.75	13.41	805	4.46	0.0923	0.2471	0.0516

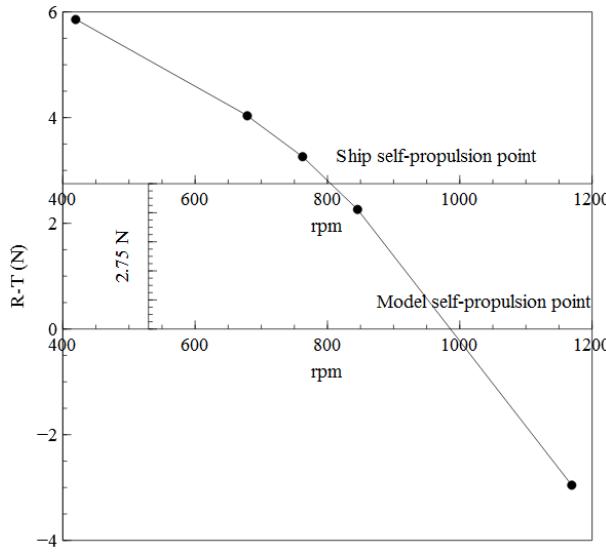


Figure 6.3.1: Self-propulsion test results of the FL_{DT} condition with skin friction correction force $\Delta_{C_F} = 2.75$.

Thrust Identity Analysis

Using thrust identity analysis, the thrust generated by the propeller in open-water condition was assumed to be identical to the thrust behind the hull $K_{To} = K_{Tb}$. J_o , K_{Qo} and η_O were found from the propeller's open-water chart to be: $J_o = 0.398$, $K_{Qo} = 0.034$ and $\eta_O = 0.465$

$$J_b = \frac{V_s}{nD} = \frac{1.029}{13.41 \times 0.1} = 0.767 \quad (6.3.10)$$

$$w_T = 1 - \frac{J_o}{J_b} = 1 - \frac{0.398}{0.767} = 0.481 \quad (6.3.11)$$

$$t = 1 - \frac{R}{T_b} = 1 - \frac{2.75}{4.46} = 0.383 \quad (6.3.12)$$

where R is the towed model resistance corresponding to the ship self-propulsion point.

$$\eta_R = \frac{K_{Qo}}{K_{Qb}} = \frac{0.0344}{0.0516} = 0.667 \quad (6.3.13)$$

$$QPC = \eta_D = \eta_O \eta_H \eta_R = \eta_O \times \frac{1-t}{1-w_T} \times \eta_R = 0.465 \times \frac{1-0.383}{1-0.481} \times 0.667 = 0.369 \quad (6.3.14)$$

or more directly, identically (the difference was only because of the rounding process):

$$QPC = \eta_D = \frac{R \times V_s}{2\pi n Q} = \frac{2.75 \times 1.029}{2\pi \times 13.41 \times 0.0923} = 0.364 \quad (6.3.15)$$

It can be seen in Table 6.3.5 that the Hughes' method with form factor calculated according to Watanabe's formula gave the most reasonable answer as the power requirement recorded in the sea's trial was only 8047 kW. However, using the form factor approach, the values of wake fraction, thrust deduction and relative rotative efficiency become too extreme, for instance, thrust deduction has an unusual negative value. Therefore, it was decided to use the traditional method to analyse other loading conditions and the large power requirement was attributed to measurement errors.

Table 6.3.5: Propulsive efficiency η_D and delivered power P_D for three different form factors.

k	1	1.18	1.35
R_{Ts} (N)	739008	629330	525746
R_{Tm} (N)	6.09	6.09	6.09
Δ_{CF} (N)	2.752	3.247	3.715
n_m (rps)	13.413	12.726	12.077
n_s (rps)	1.732	1.643	1.559
J_b	0.767	0.809	0.852
K_{Qb}	0.092	0.089	0.087
$K_{Tb} = K_{To}$	0.247	0.246	0.245
J_o	0.398	0.401	0.403
K_{Qo}	0.034	0.034	0.034
η_o	0.465	0.468	0.471
ω_T	0.481	0.505	0.527
t	0.383	0.185	-0.050
η_R	0.667	0.620	0.582
η_H	1.190	1.644	2.218
η_D	0.369	0.477	0.607
$P_E(kW)$	5890	5016	4191
$P_D(kW)$	15966	10512	6903

Model's Thrust Deduction and Wake Fraction Calculation

Model's Thrust Deduction: Data from 6.3.1 can also be used to plot figure 6.3.2. It can be seen in that figure, the thrust required to propel the model in calm water is $T_m = 7.365$. Therefore, thrust deduction of the model is:

$$t_m = 1 - R_{Tm}/T_m = 1 - 6.09/7.365 = 0.173 \quad (6.3.16)$$

Another way to calculate the required thrust to propel the model in calm water at 1.029 m/s is to add the thrust measured by thrust block and thrust measured by carriage dynamometer from Table 6.3.3. Except for the first run, the sums of two values are between 7.19 and 7.39 and the average value is 7.3.

Model's wake fraction: According to self propulsion test, at 980 RPM (i.e. RPS = 16.3), the model was expected to run at $V_m = 1.029$ m/s and the propeller generated 6.75 N. Using the thrust identity method, from the open-water chart, the propeller was expected to produce 7.365 N (i.e. $K_{Tb} = K_{To} = 0.265$ when $J_o = \frac{V_a}{nD} = 0.349$ hence $V_a = 0.57$) Thus, model's wake fraction was $w_{Tm} = 1 - V_a/V_s = 0.446$.

Model's Propulsive Efficiency: Model's propulsive efficiency can be calculated based on the model self propulsion point

$$\eta_{Dm} = \frac{R_m \times V_m}{2\pi n Q} = \frac{6.09 \times 1.029}{2\pi \times 980/60 \times 0.107} = 0.57 \quad (6.3.17)$$

6.3.3 QinetiQ: Comparison of Different Loading Conditions

[LE]

Table 6.3.6 extracted the power requirements of the model with 4 different conditions at 1.029 m/s (corresponding to normal operation speed of the full scale ship) to give the direct comparison (detail is presented in Tables 6.3.7, 6.3.8 and 6.3.9). It was derived from the table that power requirement for

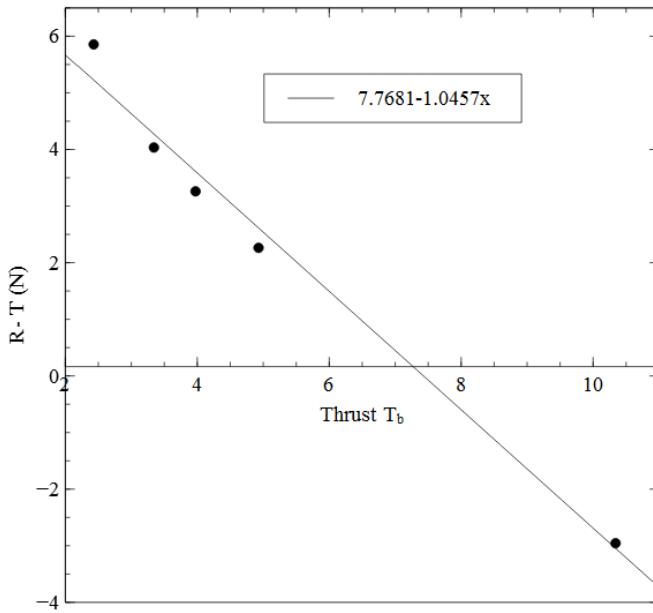


Figure 6.3.2: Self-propulsion test results for the FL_{DT} condition.

the ship in FL_{T1} condition is 0.56% lower than in default trim FL_{DT} . Power requirement in BL_{T1} was 2.94% worse than the default trim condition.

The results for full-load are certainly with the error margins of the QinetiQ dynamometer, since the drag force was only ≈ 6 N, so are not conclusive. For the ballast case, it appears that the default trim case was better than the alternative. A more comprehensive trim study could be performed on lakes in the future, to find any trends that may be emerging.

Table 6.3.6: Effective power requirements of the model with different conditions at 1.029 m/s corresponding to 15.5 knots.

Condition	Drag [N]	C_{Tm}	C_{Fm}	C_R	C_{Fs}	C_{Ts}	R_{Ts} [N]	P_E [kW]
FL_{DT}	6.1	5.20E-03	3.85E-03	1.35E-03	1.50E-03	2.85E-03	739008	5890
FL_{T1}	6.1	5.19E-03	3.85E-03	1.33E-03	1.50E-03	2.84E-03	734846	5857
BL_{DT}	5.2	4.46E-03	3.85E-03	6.06E-04	1.50E-03	2.11E-03	427717	3409
BL_{T1}	5.3	4.52E-03	3.85E-03	6.68E-04	1.50E-03	2.17E-03	440284	3509

Figure 6.3.3 shows the effective power requirements of the ship at different conditions and different speeds.

- The figure reflects the correct relationship between effective power and velocity i.e. a third-order power curve.
- The power requirement curve for the full-load case is offset from the ballast-condition, as expected.
- In full-load condition, the different trim FL_{T1} is slightly better than the default case at operation speed but worse in other speeds.
- In ballast condition, the effective power requirement of the default trim is slightly lower than the alternative trim.

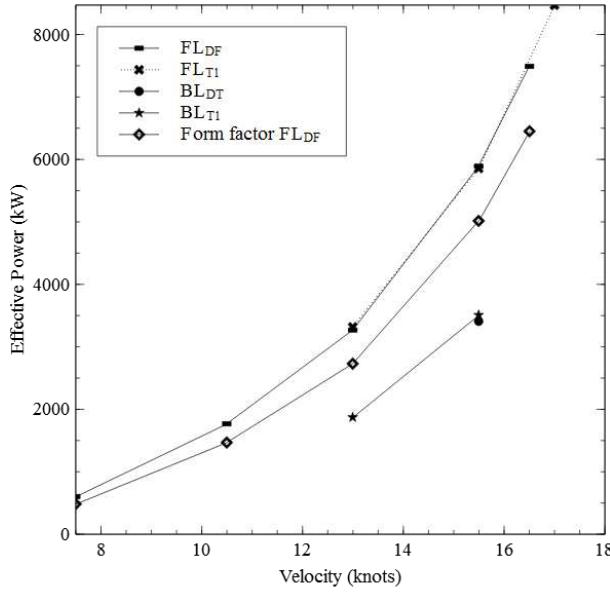


Figure 6.3.3: Experiment results for the naked-hull tests.

Tables 6.3.7, 6.3.8 and 6.3.9 showed the naked-hull tests results of other conditions and the equivalent effective power requirements derived using traditional method. Tables 6.3.10-6.3.12 show similar results for the self-propulsion tests.

Table 6.3.7: Effective power requirements of the model for different speeds for FL_{T1} condition.

V[knots]	Drag [N]	C_{Tm}	C_{Fm}	C_R	C_{Fs}	C_{Ts}	R_{Ts} [N]	P_E [kW]
13.00	4.263725	5.18E-03	3.99E-03	1.19E-03	1.53E-03	2.72E-03	496451	3319
15.49	6.071205	5.19E-03	3.85E-03	1.33E-03	1.50E-03	2.84E-03	734846	5857
17.00	7.614326	5.40E-03	3.78E-03	1.62E-03	1.48E-03	3.10E-03	968664	8471

Table 6.3.8: Effective power requirements of the model for different speeds for BL_{DT} condition.

V[knots]	Drag [N]	C_{Tm}	C_{Fm}	C_R	C_{Fs}	C_{Ts}	R_{Ts} [N]	P_E [kW]
15.49	5.22	4.46E-03	3.85E-03	6.06E-04	1.50E-03	2.11E-03	427717	3409

Table 6.3.9: Effective power requirements of the model for different speeds for BL_{T1} condition.

V[knots]	Drag [N]	C_{Tm}	C_{Fm}	C_R	C_{Fs}	C_{Ts}	R_{Ts} [N]	P_E [kW]
13.00	3.637684	4.42E-03	3.99E-03	4.28E-04	1.53E-03	1.96E-03	280123	1873
15.49	5.292515	4.52E-03	3.85E-03	6.68E-04	1.50E-03	2.17E-03	440284	3509
19.43	9.938916	5.40E-03	3.69E-03	1.72E-03	1.46E-03	3.18E-03	1012779	10120

Table 6.3.10: Typical results for the FL_{T1} condition.

rps	RPM	T_b (N)	Q_b (Nm)	R-T (N)	K_{Tb}	K_{Qb}
11.35	681	3.12	0.087346	4.01	0.242385	0.06784
12.72	763	3.67	0.089296	3.23	0.226992	0.055156
14.17	850	4.67	0.092943	2.31	0.232877	0.046304
19.75	1185	10.41	0.138131	-2.89	0.266967	0.035421

Table 6.3.11: Typical results for the BL_{DT} condition.

rps	rpm	T_b (N)	Q_b (Nm)	R-T (N)	K_{Tb}	K_{Qb}
11.37	682	3.56	0.08736	2.89	0.275141	0.067531
12.75	765	4.43	0.091514	2.03	0.272572	0.056332
14.34	860	5.61	0.097906	1.01	0.272892	0.047632
16.97	1018	7.92	0.115287	-1.27	0.275011	0.040049

Table 6.3.12: Typical results for the BL_{T1} condition.

RPS	RPM	T_b (N)	Q_b (Nm)	R-T (N)	K_{Tb}	K_{Qb}
11.40	684	3.24	0.087534	2.91	0.249249	0.067393
12.77	766	4.02	0.089996	2.12	0.24623	0.055153
14.14	848	5.05	0.094403	1.20	0.252487	0.047223
16.70	1002	7.40	0.111131	-1.10	0.265364	0.039827

Table 6.3.13: Calculation propulsive efficiency η_D .

	FL_{DT}	FL_{T1}	BL_{DT}	BL_{T1}
R_{Ts} (N)	702237	734846	678730	694785
R_{Tm} (N)	6.090	6.071	5.220	5.290
ΔC_F (N)	2.752	2.752	2.154	2.152
n_m (rps)	13.413	13.475	12.551	12.712
n_s (rps)	1.732	1.740	1.620	1.641
J_b	0.767	0.764	0.820	0.810
K_{Qb}	0.052	0.051	0.058	0.056
$K_{Tb} = K_{To}$	0.247	0.230	0.273	0.246
J_o	0.398	0.446	0.329	0.390
K_{Qo}	0.034	0.032	0.037	0.035
η_o	0.465	0.511	0.389	0.457
ω_T	0.481	0.416	0.599	0.518
t	0.383	0.341	0.499	0.459
η_R	0.667	0.634	0.640	0.620
η_H	1.190	1.128	1.248	1.123
η_D	0.369	0.365	0.310	0.318

6.3.4 QinetiQ: Discussion

[LE]

1. The towing tank tests of the GDP group last year concluded that the drag of the model in ballast condition is approximately 5.15 N (the results varied from 5.04 to 5.25) [Collison et al., 2012]. The results from QinetiQ tank tests agree with those experiment data i.e. the average resistance for ballast condition is 5.22 N.
2. According to BSRA thrust deduction regression, the thrust deduction in full load condition should be

$$t = -0.2064 + 0.3246 \times C_B^2 - 2.1504 C_B (LCB/L_{BP}) + 0.1705 (B/\nabla^{(1/3)}) + 0.1504 (P/D) = 0.23 \quad (6.3.18)$$

where P/D = 0.85 and LCB = 3.48 (from Trim and Stability Book). However, according to table 6.3.13 in full load condition, ship's thrust deduction is $t = 0.381$, which is nearly double the prediction.

Also, according to British Ship Research Association (BSRA) wake data regression [Molland et al.,

2011]

$$w_T = -0.0458 + 0.3745C_B^2 + 0.1590D_w = 0.8635F_n + 1.4773F_n^2 = 0.222 \quad (6.3.19)$$

Meanwhile, the wake fraction of the ship was found about 0.45 for the full-load condition and 0.6 for the ballast condition. The reason could be the error in thrust measurement, which had direct effect on determining K_{To} and other consequences.

3. It can be derived from Table 6.3.13 that effective wake fraction of the ship in the ballast condition is 24.5% higher than in the full-load condition. The fact that effective wake fraction in ballast condition is higher than full-load condition agrees with the experiments conducted by Van Manen [1959] and the statement in Molland et al. [2011] that the wake fraction in the ballast condition tends to be 5-15% larger than the wake fraction in the loaded condition. It should be noted the wake fraction calculated in Table 6.3.13 is effective wake fraction which takes into consideration the interaction of the propeller and the hull, and presence of waves. If only nominal wake is considered then conversely in full-load condition, there is a larger wetted surface area, which would cause a higher wake velocity and result in a higher wake fraction than in ballast condition.
4. The hull efficiencies are approximately 1.2, which is of normal value between 1.1 to 1.3 . Relative rotative efficiency was expected to be approximate 1 but it was roughly 0.7. This could be caused by the error in torque measurement or a by-product of the high rotational speed of the propeller. Open-water efficiency is slightly lower than expected but those values are acceptable as it can be between 0.35 to 0.75.
5. It can be seen in Table 6.3.13 that in full-load condition, the propulsive efficiency of the default trim is better than the alternative trim whilst in ballast condition, the alternative trim gives 4.5% increase in propulsive efficiency.
6. In the preliminary calculations, it was assumed that thrust deduction of the model is similar to the value of the ship, that is 0.2. It was found that the model's thrust deduction is $t_m = 0.17$, which is very close to the expected value. More impressively, the wake fraction estimated using wind tunnel by GDP's group last year was 0.4386 [Collison et al., 2012]. The results in self-propulsion test indicated that the model's wake fraction is 0.446. Thus, the theoretical prediction is only 1.6% below the experimental result.
7. It can be seen in Figure 6.3.4 that with the same propeller RPM, the propeller generated different thrusts in different loading conditions. It should be noted that, with the same propeller (same diameter), same revolution, the difference in thrust is proportional to the difference in thrust coefficient. From the open water chart, the higher thrust coefficient means the lower the advanced velocity (towards the left of the chart). Therefore, the higher the thrust, the lower the advanced velocity and if the tests were conducted at the same speed (as in self-propulsion tests) then the thrust has direct relationship with the wake fraction i.e. the higher the thrust, the higher the wake fraction. This trend can be observed in Figure 6.3.4 and Table 6.3.13.

6.3.5 Lake Tests (Straight Run)

[LE]

Only full-load condition was applied for the model in the lake tests but with different bows: a normal bow, a first alternative bow and a second alternative bow.

The thrust block was recalibrated after the self-propulsion tests. Therefore, a new formula is used to calibrate the thrust from the thrust measurement of the thrust block. Torque is still calculated as a function of motor current. The power input is the product of angular speed and torque. The power output is supposed to be the product of model resistance and model velocity. However, it is impractical to either directly measure model resistance in the lake or indirectly find thrust deduction of model in the lake. Therefore, the power output is considered as the product of thrust and velocity. The efficiency of the power system of model can be calculated as following:

$$\eta = \frac{P_{out}}{P_{in}} = \frac{T \times V}{2\pi n Q} = \frac{T \times V}{2\pi n Q} \quad (6.3.20)$$

The electrical efficiency η_e of the motor can be calculated as the ratio between the power input (the actual power used for torque) and the motor power (the energy drained from the battery).

$$\eta_e = \frac{P_{in}}{\text{Motor Power}} \quad (6.3.21)$$

Propulsive Efficiency of the Model

Similar to the part in tank-testing self-propulsion tests, the efficiency of the propulsion system can also be derived using the thrust identity method.

Thrust Deduction: The resistance of the model is assumed to be the same of resistance in calm wave. This is a big assumption but there is no other available way to measure resistance of hull in waves.

$$t = 1 - R/T_b = 1 - 6.09/12.79 = 0.523 \quad (6.3.22)$$

Wake fraction: The model's advance velocity according to Table 6.3.15 is:

$$V_a = J_o \times nD = 0.2364 \times 1231.79/60 \times 0.1 = 0.4853 \quad (6.3.23)$$

$$w_T = 1 - V_a/V = 0.528 \quad (6.3.24)$$

Hull Efficiency:

$$\eta_H = \frac{1-t}{1-w_T} = 1.01 \quad (6.3.25)$$

Relative Rotative Efficiency:

$$\eta_R = Q_o/Q_b = \frac{\rho n^2 D^5}{Q_b} = 1.315 \quad (6.3.26)$$

Propulsive Efficiency:

$$\eta_D = \eta_H \eta_o \eta_R = 1.01 \times 1.315 \times 0.275643 = 0.366 \quad (6.3.27)$$

Or more quickly

$$QPC = \eta_D = \frac{R \times V_s}{2\pi n Q} = \frac{6.09 \times 1.029}{2\pi \times 1231.79/60 \times 0.13} = 0.37. \quad (6.3.28)$$

The model's propulsive efficiency in calm water was calculated as 0.57. Therefore, the efficiency of the model in waves is only 65% the efficiency of the model in calm water.

Table 6.3.14: Interpolation for the performance of the model at 1.029 m/s.

Speed	Thrust	M. Current	Torque	M.Power	Prop Rev	P_T	P_{in}
m/s	N	A	Nm	W	RPM	W	W
0.659	5.11	2.12	0.09	9.63	750.36	3.37	6.68
0.818	8.25	2.31	0.09	14.36	1000.68	6.75	9.75
1.046	13.15	3.32	0.13	26.81	1250.14	13.76	17.49
1.029	12.79	3.25	0.13	25.90	1231.79	13.16	16.84

Table 6.3.15: Data from open-water chart.

rpm	rps	T_b	$K_{Tb} = K_{To}$	J_o	K_{Qo}	η_o
1231.79	20.52983	12.794	0.297	0.236429	0.040557	0.275643

Effect of Waves on the Performance of the Model

According to the self-propulsion tests from the towing tank, it was predicted that at 1000 RPM, the model would run at 1.029 m/s with its propeller generating a thrust of 7.3 N. It could also be derived that the torque that it would generate at 1000 RPM is 0.107 Nm and as the result, the power it required for that speed is 11 W.

However, in fact, to achieve 1.029 m/s in lake, it requires 16.84 W (Table 6.3.4). It means the waves add 53.4% to the power requirement of the model in calm water. The deflection of the rudder (due to the adjustments required by the heading controller) may have also contributed to the increase in the resistance and power requirements. Additionally, the unreliability of the GPS sensor (as described in Section 6.2) means the velocity data should be treated with caution.

It can also be seen in Table 6.3.4 that at 1000 RPM, the propeller produced a bigger thrust of 8.2N but the model could only move at 0.82 m/s. The increase in thrust could be explained by the effect of waves on the resistance of the hull. However, the errors in the GPS reading mean that any performance results cannot be relied upon.

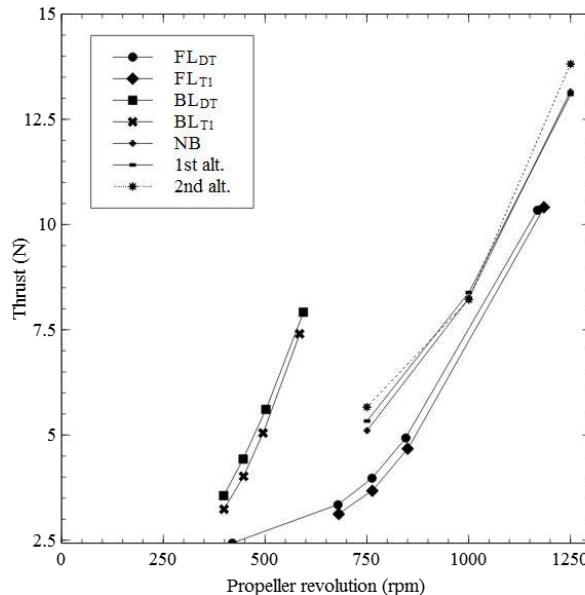


Figure 6.3.4: Thrust generated by propeller versus propeller revolutions.

Effect of Waves on the Full-Scale Ship Resistance

[LE]

From table 6.3.4 it was shown that the thrust requirement for the model to achieve 1.029 m/s is 12.79 N. In order to estimate the resistance of model, it is necessary to know the thrust deduction. However, thrust deduction cannot be measured hence either of two assumptions should be made.

- Assuming that the thrust deduction in waves is the same as the thrust deduction in self-propulsion test then the resistance of model in waves is:

$$\text{Drag} = \text{Thrust} \times (1 - t) = 12.79 \times (1 - 0.383) = 7.89N$$

- Assuming that the thrust deduction in waves is the same as in BRSA regression formula.

$$\text{Drag} = \text{Thrust} \times (1 - t) = 12.79 \times (1 - 0.23) = 9.85N$$

In both cases, the ship's resistance can be calculated using traditional method. Results are shown in Table 6.3.16. In the first case, model's drag in waves is 29.6% higher than the calm water case and the full-scale ship is 53.97% higher. In the second case, model's drag in waves is 61.7% higher and the resistance of full-scale ship in waves is 112.6% higher. Therefore, it can be concluded that the first assumption (which is the assumption that thrust deduction from the self-propulsion test is the same as in waves) gives more reasonable results. For both cases, the QPC in waves was assumed to be the same as in calm water. A similar procedure but with form factor approach resulted in Table 6.3.17.

Table 6.3.16: Delivered power requirement for ship in waves calculated by traditional method.

Type	Drag [N]	C_{Tm}	C_{Fm}	C_R	C_{Fs}	C_{Ts}	$R_{Ts}[N]$	P_E [kW]	P_D [kW]
CW	6.09	5.20E-03	3.85E-03	1.35E-03	1.50E-03	2.85E-03	739008	5890	15963
1st	7.8913	6.74E-03	3.85E-03	2.89E-03	1.50E-03	4.39E-03	1137815	9069	24577
2nd	9.8483	8.41E-03	3.85E-03	4.56E-03	1.50E-03	6.06E-03	1571095	12523	33937

Table 6.3.17: Delivered power requirement for ship in waves calculated by form factor approach.

Type	Drag [N]	C_{Tm}	C_{Fm}	C_W	C_{Fs}	C_{Ts}	$R_{Ts}[N]$	P_E [kW]	P_D [kW]
CW	6.09	5.20E-03	3.85E-03	1.13E-06	1.50E-03	2.03E-03	525746	4191	6881
1st	7.8913	6.74E-03	3.85E-03	1.54E-03	1.50E-03	3.57E-03	924554	7369	12101
2nd	9.8483	8.41E-03	3.85E-03	3.21E-03	1.50E-03	5.24E-03	1357834	10823	17771

It can be seen that the lowest estimation of power requirement in waves using form factor method is still higher than the real power consumption recorded in the sea trial. The errors were possibly resulted from measurement error in the naked-hull tests, thrust measurement errors in the self-propulsion tests or incorrect form factor prediction. Error from the GPS probably accounted for a lot of error in these results - most likely the GPS significantly underestimated the speed of the vessel, which affects all of the results.

Propulsive Performance of the Model with Different Bows

Table 6.3.18 shows the direct comparison among the performance of the three bows. The model with these bows were tested with the same propeller revolutions and their performances were recorded. The propeller revolutions were set by the model's microcontroller and that is why they are nearly identical for the three cases.

It can be seen in Table 6.3.18 that the first alternative bow has significantly less power requirement at high speeds i.e. it takes less power whilst runs faster. The same trend can be observed from Figure 6.3.5.

However, it should be remembered that the speed of the model recorded from GPS is highly unreliable. Therefore, in this case the method is of more value than the real results.

Given that the speed is correct, from the column of propulsive efficiency, it can be derived that the first alternative bow has the best efficiency when the propeller runs at 1000 RPM and 1250 RPM whilst the second alternative bow has the highest efficiency when the propeller runs at 750 RPM.

Provided that the resistance of the hull with different bows in waves are identical, it did make sense in the case at 1000 RPM, the propeller in the model with bow 1 generated the highest thrust and achieved the highest speed. The reason could be simply explained as this bow directs more flow to the propeller and increases propeller efficiency. However, at 1250 RPM, the propeller in the model with bow 1 generated the least thrust but still achieved the highest speed. One could argue that the reason was possibly the tests were conducted when there were very strong wind. However, Table 6.3.20 showed that even in the run supposedly against the wind (9th run), the model still achieve a higher speed (1.08 m/s) with lower thrust (13.13 N) than cases of NB and Bow 2. The tracks the model ran in these cases were checked and they did not show any unusual problem. Therefore, this was not caused by the wind. The only possible reason could be that the model with this bow has significant lower resistance in waves than other two bows, or again due to the errors of the GPS system.

Table 6.3.18: Direct comparison between bows using mean values.

Bow	Speed	Thrust	M. Current	Torque	M.Power	Prop Rev	P_{out}	P_{in}	η	η_e
	m/s	N	A	Nm	W	RPM	W	W		
NB	0.66	5.11	2.12	0.09	9.63	750.36	3.37	6.68	50%	69%
Bow 1	0.66	5.33	2.12	0.09	9.68	750.27	3.54	6.69	53%	69%
Bow 2	0.68	5.66	2.22	0.09	10.27	749.92	3.83	7.01	55%	68%
NB	0.82	8.25	2.31	0.09	14.36	1000.68	6.75	9.75	69%	68%
Bow 1	0.91	8.39	2.40	0.10	14.91	1000.09	7.59	10.09	75%	68%
Bow 2	0.85	8.22	2.63	0.11	16.44	1000.32	6.97	11.08	63%	67%
NB	1.05	13.15	3.32	0.13	26.81	1250.14	13.76	17.49	79%	65%
Bow 1	1.17	13.09	3.26	0.13	26.46	1250.07	15.30	17.13	89%	65%
Bow 2	1.04	13.81	3.57	0.14	28.81	1250.27	14.35	18.77	76%	65%

Table 6.3.19: Performance of the model with normal bow.

Run	Speed	Thrust	M. Current	Torque	M.Power	Prop Rev	P_{out}	P_{in}	η	η_e
1	0.64	5.12	2.14	0.086	9.747	750.4	3.26	6.75	48%	69%
2	0.76	5.06	2.09	0.084	9.502	750.3	3.83	6.60	58%	70%
3	0.64	5.14	2.12	0.085	9.623	750.3	3.29	6.68	49%	69%
4	0.60	5.10	2.12	0.085	9.643	750.4	3.07	6.68	46%	69%
5	0.74	8.38	2.22	0.089	13.65	1000.5	6.23	9.35	67%	68%
6	0.87	8.25	2.40	0.096	14.907	1000.8	7.13	10.10	71%	68%
7	0.92	8.11	2.38	0.096	14.779	1000.8	7.44	10.01	74%	68%
8	0.75	8.27	2.26	0.091	14.11	1000.5	6.16	9.52	65%	67%
9	1.07	13.05	3.23	0.130	25.933	1250.1	13.92	16.97	82%	65%
10	1.02	13.25	3.41	0.137	27.407	1250.2	13.45	17.93	75%	65%
11	1.07	13.08	3.35	0.134	27.009	1250.3	14.01	17.60	80%	65%
12	1.03	13.24	3.32	0.133	26.902	1249.9	13.64	17.46	78%	65%

Table 6.3.20: Performance of the model with 1st alternative bow.

Run	Speed	Thrust	M. Current	Torque	M.Power	Prop Rev	P_{out}	P_{in}	η	η_e
	m/s	N	A	Nm	W	RPM	W	W		
1	0.65	5.33	2.14	0.086	9.726	750.4	3.45	6.75	51%	69%
2	0.68	5.35	2.12	0.085	9.726	750.1	3.66	6.70	55%	69%
3	0.66	5.32	2.10	0.084	9.584	750.3	3.53	6.62	53%	69%
4	0.89	8.47	2.35	0.094	14.631	999.9	7.55	9.89	76%	68%
5	0.89	8.35	2.37	0.095	14.707	1000.2	7.41	9.98	74%	68%
6	0.92	8.43	2.42	0.097	15.108	1000.6	7.75	10.21	76%	68%
7	0.92	8.29	2.44	0.098	15.21	999.7	7.66	10.27	75%	68%
8	1.21	13.06	3.31	0.133	26.865	1250.2	15.84	17.42	91%	65%
9	1.08	13.13	3.24	0.130	26.348	1250.2	14.12	17.06	83%	65%
10	1.22	13.08	3.22	0.129	26.163	1249.8	15.93	16.92	94%	65%

Table 6.3.21: Performance of the model with 2nd alternative bow.

Run	Speed	Thrust	M. Current	Torque	M.Power	Prop Rev	P_{out}	P_{in}	η_D	η_e
	m/s	N	A	Nm	W	RPM	W	W		
1	0.87	7.83	2.73	0.109	16.971	1000.1	6.80	11.47	59%	68%
2	0.83	8.60	2.54	0.102	15.915	1000.6	7.11	10.70	66%	67%
3	1.04	13.99	3.62	0.145	29.097	1250.2	14.55	19.02	77%	65%
4	1.04	13.64	3.52	0.142	28.518	1250.3	14.14	18.53	76%	65%
5	0.71	5.72	2.22	0.089	10.3	749.8	4.07	7.00	58%	68%
6	0.63	5.61	2.22	0.089	10.27	749.8	3.53	7.01	50%	68%
7	0.69	5.65	2.23	0.089	10.241	750.1	3.89	7.03	55%	69%

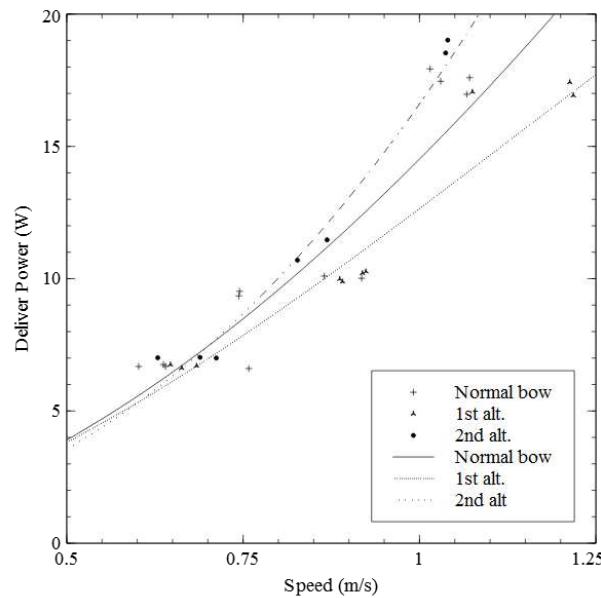


Figure 6.3.5: Powering requirement for different bows at different speed.

6.3.6 Error Analysis

[LE]

- The wetted surface area (WSA) could be slightly different for the default trim and the alternative trim in both full load condition and ballast condition. However, it was decided that the difference between the two trim is negligible. The wetted surface area in full load condition and ballast condition were determined by interpolating data from the "Trim and stability book" provided by

the ship owner.

- Sensitivity of carriage dynamometer: It was acknowledged that the carriage dynamometer is not sensitive in the range of force that is smaller than 10 N. However, all experiment results are within this range, which would subject to a certain level of inaccuracy.
- Thrust block calibration: in the thrust calibration section, it was showed that the results of the two bollard pulls were slightly different and the calibration formula takes into consideration both results. Also, after the tests in the QinetiQ, the thrust block was recalibrated so a new formula was used to calibrate thrust measured in the lake test. This could have contributed to the error when comparing results from QinetiQ and lake tests.
- Torque measurement: torque was derived by the function of battery current, which could be inaccurate. In the future research, a torque dynamometer should be used instead.
- GPS system and measurement of speed: there were big errors in measurement of speed (see Section 6.2), which made a large percentage of the results very unreliable. However, the method to analyse data is still valid. Had the GPS results been more accurate, the results would have made more sense.

6.3.7 Future Testing

[LE]

- The self-propulsion tests can be conducted with more speeds other than the operating speed, for instance the corresponding speed for 16.5 knots and 13 knots. Because the naked-hull tests for the full-load condition have already been done for those speeds, the QPC of the ship at those speeds can be calculated.
Moreover, a relationship between propeller revolution and model/ship's speed can be established after these tests. Those data can be used to compare performances of the model in waves with different propeller revolutions.
- In this research, the propulsive efficiency of the ship in waves was considered unchanged when the model moves into the seaway. However, in reality, this efficiency in calm water is not the same as in the lake. Several method could be used to estimate this change.
 - The first method could be based on the ratio between propulsive efficiency of the model in waves and the model in calm water.
 - The second method is more ambitious and based on the idea of self-propulsion tests. The idea of self-propulsion tests is to find the ship self-propulsion point and the efficiency of the propeller when the model moving at 1.029 m/s but with the thrust less by an amount equals skin friction correction force. In lake, there is no carriage but a simpler method could be used: a big fan can be put on top of the model which can generate the thrust exactly equal the skin friction correction force. With the presence of the fan, the propeller revolution that helps the model achieve the correct speed in waves would be the same as the propeller revolution that helps the ship achieve the corresponding speed at sea. Similarly, the propulsive efficiency of the model at that point would be the same as the ship. The limitation of this method is how to make a fan that can actually generate approximately 3N force.

6.4 Manoeuvring

[ANDERLINI]

6.4.1 Full-Scale Ship Results

Some of the most important particulars for two circle manoeuvres performed on the full-scale ship (one to port and one to starboard) have been reported in Table 6.4.1, in conjunction with the average value, which refers to the magnitude only (hence, only the absolute values). The advance, transfer and tactical diameter have been reported in non-dimensional format (divided by the ship length) for non-disclosure purposes.

Table 6.4.1: Main registered particulars during two circle manoeuvres for the full scale ship. The average value has been calculated as well.

Registered Data	Port	Starboard	Average
Advance	3.132	3.341	3.237
Transfer	-0.899	1.108	1.003
Tactical Diameter	-2.837	3.211	3.024
Rudder angle (°)	-35.200	34.400	34.800
Initial Speed (kts))	15.600	15.300	-
Final Speed (kts)	4.600	4.800	-
Time to change heading by 360° (s)	411	418	-
Initial Wind Speed (kts)	0	0	0
Initial Wind Dir. (°)	0	0	0

For the zigzag trials, some data is available for the 10°-10° and 20°-20° manoeuvres. However, the latter is incomplete, with only the first bend represented. Therefore, it has been discarded, as it could not be used to derive the Nomoto coefficients. The manoeuvre has been performed with an initial speed of 15.1 knots reducing to approximately 13 knots, where the average velocity is 14 knots. The first overshoot has a registered value of 8.8° and the second one of 20°. In addition, the Nomoto coefficients have been calculated as $K = -0.0075$ rad/s and $T = 18.3835$ s (corresponding to $K' = -0.1801$ and $T' = 0.7609$ and thus $|K'/T'| = 0.2367$) as described in Section 5.2.3.

6.4.2 Model Ship Results

[ANDERLINI]

All results within this section related to the standard bow (as per the BP tanker) and the larger rudder (scaled by keeping the equality of the lift coefficient) (see Section 3.5).

Circle Manoeuvre

The motor setting and rudder angle demand used for the circle manoeuvring trials can be seen in Table 6.4.2. In addition, some relevant comments that have been logged during the runs are reported as well. The data that has been measured during the trials can be seen in Tables 6.4.3 and 6.4.4. The standard deviation is expressed as a percentage of the average value. From the analysis of the data, it has been possible to calculate the mean turning radius (Table 6.4.2) as the average distance of all points belonging to the curve from a centre, whose coordinates (selection based on common sense) can also be seen in Table 6.4.2. The standard deviation (expressed as a percentage of the average value) in this distance is reported in the table as well. Additionally, it is possible to see the non-dimensional tactical diameter, which has been approximated as twice the value of the steady turning radius because of the poor quality of the position data, as explained in Section 6.2. For the same reason, no attempt has been made to estimate the advance and the transfer.

Figures 6.4.1 and 6.4.3 show the model path recorded by the GPS during runs number 3 and 9. Similarly, Figures 6.4.2 and 6.4.4 present the heading recorded by the compass during the same runs (3 and 9). It can be noted that the heading has a strong discontinuity at a certain point: this is due to the fact that the limiting range for the heading angle is 0° - 360° .

Table 6.4.2: Motor setting, rudder demand and general comments for the circle manoeuvres.

Run	Prop demand (RPM)	Angle demand ($^\circ$)	Comments
1	750	25	
2	750	25	
3	1000	20	
4	1000	20	
5	1000	25	
6	1000	25	waves from rib
7	1000	30	
8	1000	30	
9	1250	25	
10	1250	25	

Table 6.4.3: Total time length, thrust, speed and motor voltage for the circle manoeuvring trials. For the last three variables, both the average and the percentage standard deviation are expressed.

Run	Total Time length (s)	Speed (m/s)		Thrust (N)		Motor Voltage (V)	
		Average	Standard deviation (%)	Average	Standard deviation (%)	Average	Standard deviation (%)
1	85.5	0.4761	39.72%	5.3703	1.96%	4.6013	1.35%
2	86	0.3878	39.14%	5.3677	1.98%	4.5746	1.35%
3	71.4	0.593	24.92%	8.8895	1.75%	6.2303	1.46%
4	73	0.6013	35.37%	8.9900	1.77%	6.2818	1.28%
5	63.2	0.5542	27.95%	9.0441	1.86%	6.2495	1.67%
6	66.3	0.5652	30.75%	9.1342	1.91%	6.3054	1.37%
7	59.8	0.5618	36.83%	9.1471	2.05%	6.3036	1.39%
8	58.3	0.509	33.54%	9.2695	1.99%	6.3139	1.34%
9	52.2	0.7199	27.42%	14.7055	5.10%	8.0851	1.13%
10	50.7	0.7431	29.47%	14.4465	5.10%	8.1295	1.04%

Zigzag Manoeuvre

The motor setting, the rudder angle demand and some subjective comments on the environmental conditions during the zigzag manoeuvring trials can be seen in Table 6.4.2. Furthermore, the table contains also the average rudder angle and the 1^{st} , 2^{nd} and 3^{rd} overshoot angles. The data that has been measured during the trials can be seen in Tables 6.4.2 and 6.4.2. The standard deviation is expressed as a percentage of the average value.

The Nomoto coefficients have been calculated for every run, as shown in Table 6.4.2. If trials 5 and 6 are excluded, since their results are very dubious, the average values for the non-dimensional Nomoto coefficients are $K' = -0.7952$ and $T' = 1.1597$, with a standard deviation, expressed as a percentage of the mean, of 5.62% and 25.32% respectively. Furthermore, their ratio is $K'/T' = -0.6857$.

Table 6.4.4: Average motor current and power, propeller revolutions and rudder angle measured during the circle manoeuvres. Additionally, the standard deviation of the data is included as well.

Run	Motor Current (A)		Motor Power (W)		Propeller revs (RPM)		Rudder Angle ($^{\circ}$)	
	Average	Standard deviation (%)	Average	Standard deviation (%)	Average	Standard deviation (%)	Average	Standard deviation (%)
1	2.1444	2.27%	9.8674	2.54%	750.2514	1.12%	24.6835	13.97%
2	2.1743	2.21%	9.9446	2.60%	750.2253	1.10%	24.3827	15.52%
3	2.6276	2.23%	16.3667	2.55%	1.0002	0.96%	19.1633	19.45%
4	2.6522	2.56%	16.6624	2.69%	0.9994	0.96%	19.2528	20.49%
5	2.649	2.24%	16.5559	2.96%	1.0005	0.90%	23.9727	15.79%
6	2.6591	2.47%	16.7637	2.72%	1.0001	0.95%	23.6542	14.76%
7	2.6775	3.22%	16.8801	3.56%	1.0004	0.97%	27.6761	14.52%
8	2.7104	2.83%	17.1155	3.12%	1.0001	0.98%	29.022	10.47%
9	3.8998	1.47%	31.5361	2.08%	1.2507	0.93%	24.0034	14.42%
10	3.8463	2.39%	31.2721	3.02%	1.2503	0.86%	24.2361	16.07%

Table 6.4.5: Average turning radius and its standard deviation, coordinates of the centres of the circles based on a reference system whose origin is an arbitrary point on the lake and non-dimensional tactical diameter.

Run	R_C (m)	Standard deviation	X_C (m)	Y_C (m)	D'
1	6.4705	43.29%	3	-28	4.4624
2	5.5316	29.46%	17	-17.5	3.8149
3	7.228	29.91%	29	-1	4.9848
4	8.177	44.34%	18	14	5.6393
5	5.3952	7.89%	14	24.5	3.7208
6	6.9159	19.38%	10	32	4.7696
7	5.899	49.75%	1	-12	4.0683
8	4.9496	44.51%	5	-23	3.4135
9	6.7922	65.27%	15	12.5	4.6843
10	5.7633	54.88%	20	-4	3.9747

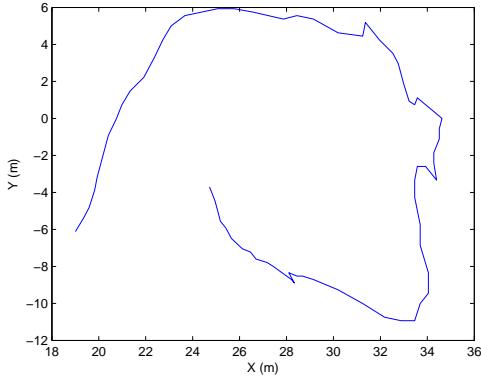


Figure 6.4.1: Path of the model during the circle manoeuvre number 3. The origin of the system of reference is an arbitrary point on the lake.

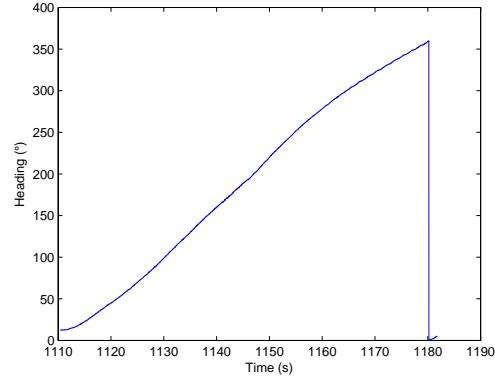


Figure 6.4.2: Change in heading angle with time during circle manoeuvre number 3.

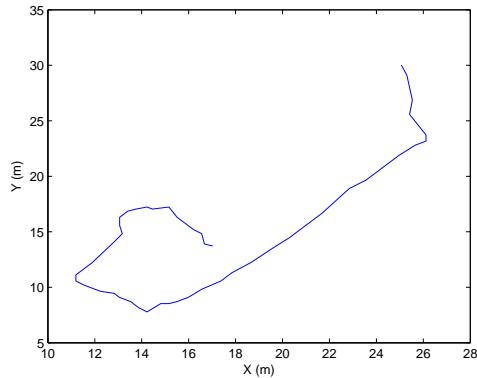


Figure 6.4.3: Path of the model during the circle manoeuvre number 9. The origin of the system of reference is an arbitrary point on the lake.

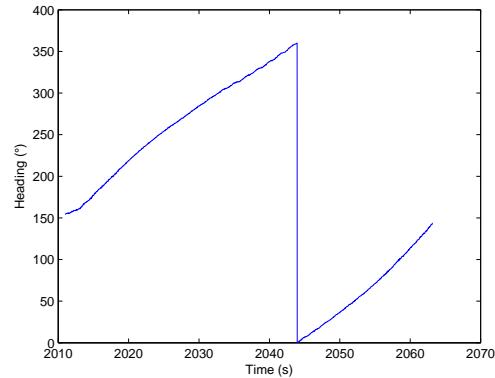


Figure 6.4.4: Change in heading angle with time during circle manoeuvre number 9.

In Figure 6.4.2, it is possible to see the heading and the requested and actual rudder angle during the zigzag trial number 3. The heading has been offset in order to obtain 0° initial heading angle.

Table 6.4.6: Motor setting, rudder demand, general comments, average rudder angle, 1st, 2nd and 3rd overshoot angles recorded during the zigzag manoeuvring trials.

Run	Prop demand (RPM)	Angle demand (°)	Comments	Average Rudder Angle (°)	1 st Over-shoot (°)	2 nd Over-shoot (°)	3 rd Over-shoot (°)
1	1000	20-20	slight breeze	1.6261	5.3	9	-
2	1000	20-20	slight breeze	-3.8286	6.5	8.2	6.5
3	750	20-20	less breeze	1.9867	5.9	9.3	-
4	1250	20-20		-3.3651	7.2	6.5	8.3
5	1250	20-20	head waves	-4.0481	7	10.6	5.1
6	750	20-20	momentum from previous test	-4.2218	8.7	6	6.1

Table 6.4.7: Total time-length, average thrust, speed and motor voltage for the zigzag trials. Furthermore, the standard deviation of the last three variables has been computed as well.

Run	Total Time length (s)	Speed (m/s)		Thrust (N)		Motor Voltage (V)	
		Average	Standard deviation (%)	Average	Standard deviation (%)	Average	Standard deviation (%)
1	31.1	0.807	13.88%	8.5327	1.70%	6.2095	1.43%
2	44	0.7965	14.68%	8.7684	1.81%	6.261	1.19%
3	31.9	0.6978	18.21%	8.6666	1.78%	6.2691	1.28%
4	38.7	0.8736	13.34%	13.7200	4.85%	8.0165	1.54%
5	36.7	0.8889	12.40%	13.9480	5.11%	8.0783	1.17%
6	57.7	0.6397	19.07%	5.6176	1.78%	4.6048	1.13%

Table 6.4.8: Average motor current and power and propeller revolutions measured during the zigzag manoeuvres. Additionally, the standard deviation of the data is included as well.

Run	Motor Current (A)		Motor Power (W)		Propeller revs (RPM)	
	Average	Standard deviation (%)	Average	Standard deviation (%)	Average	Standard deviation (%)
1	2.5336	3.23%	15.7365	3.75%	1000.5	0.87%
2	2.5431	3.13%	15.924	3.44%	1000.1	0.95%
3	2.5623	3.20%	16.0627	3.58%	1000.1	0.92%
4	3.6536	3.81%	29.2949	4.74%	1250.9	0.98%
5	3.6442	4.48%	29.4534	5.33%	1250.1	0.95%
6	2.165	2.10%	9.9701	2.65%	750.2	1.04%

Table 6.4.9: Dimensional and non-dimensional Nomoto coefficients calculated from the measurements of the zigzag manoeuvres.

Run	K (rad/s)	K'	T (s)	T'	$ K'/T' $
1	-0.2065	5.3042	-0.7421	1.4760	0.5027
2	-0.2175	4.9424	-0.7919	1.3575	0.5834
3	-0.2083	4.5497	-0.8657	1.0948	0.7908
4	-0.2353	2.3594	-0.7811	0.7107	1.0990
5	-0.2795	-50.7436	-0.9119	-15.5538	-0.0586
6	-0.1696	12.9552	-0.7689	2.8577	0.2690

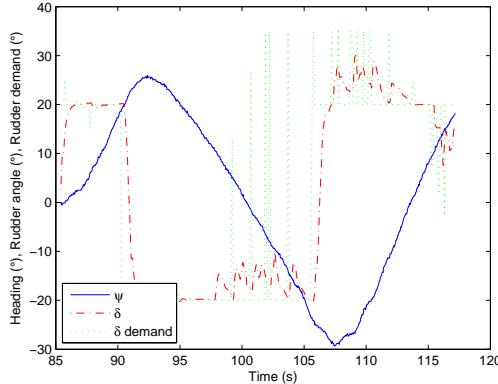


Figure 6.4.5: Heading, rudder angle demand and actual rudder angle against time during the zigzag trial number 3.

6.4.3 Discussion

[ANDERLINI]

Error Analysis

As the environmental conditions were far from ideal during model testing, large margin of errors are to be expected. Indeed, during most of the runs there was some wind, thus affecting the motion of the model due to both the aerodynamic force and the waves generated. Additionally, the wind direction was changing about every 10 minutes. Furthermore, the apparent wind and wave direction was influenced during every run by the model direction, which is particularly true during the circle manoeuvre. For instance, this is clear from the experimental data for the zigzag manoeuvring trials. In fact, while in most of the runs, the first overshoot is always smaller than the second one as expected (the explanation

follows in the next subsection), this is not the case for run 4 and 6 in Section 6.4.2. In this case, although no comments have been logged, it is clear that a slight breeze (and resulting waves) coming from approximately 0°-90° on the port side of the model (as compared with the heading during the straight run) must have altered its course. Another example can be seen in Figure 6.4.4 in Section 6.4.2, which shows the plot of the change in heading angle with time for circle manoeuvre number 9. If it is compared with Figure 6.4.2 (same graph, but relating to run number 3), it is clear that it presents a small hump between time 2020 and 2030 s and a hollow between 2030 and 2050 s. This clearly shows that the rate of turn of the model has been influenced by wind effects or by the waves from the supporting rib (see next paragraph). This problem clearly requires a much more reliable solution than an incomplete logging based on subjective perceptions (as adopted during lake testing in the absence of other equipment), if an accurate error analysis is to be performed. The wave buoys produced as part of this project (see Section 4.1) do offer the possibility of decreasing the uncertainty in the environmental conditions for future tests. However, it is still recommended that an anemometer should also be installed on the model itself, which would also allow an estimate of the aerodynamic drag force, by registering the relative wind direction and speed. These solutions, which enable an accurate analysis of the testing environmental conditions, offer the possibility of a more thorough understanding of the results data, so that possible corrections could safely be applied.

Another issue that has greatly influenced the tests results is the presence of a rib travelling close to the model (Section 6.2), which would cause a wave pattern much more significant than the ship model one, thus sometimes altering its course and resistance. For example, it is believed that the kink in the model path during a circle manoeuvre shown in Figure 6.4.1 in Section 6.4.2 has been caused by the rib causing a change in course to the model either by direct hit (which sometimes did happen unfortunately) or indirectly through its wave system. This issue can be easily solved in the future by locating the WIFI aerial on top of the model (see Section 3.6.15) so that a much wider signal range can be achieved (as suggested in Section 6.2). This would result in the possibility of controlling the model from ashore. Nevertheless, it is likely that a support boat will be still required to operate at a distance from the model so that it can intervene to avoid collisions with other boats and or with the banks in case the signal is lost.

A more accurate analysis of the results has proven the reliability of the whole system, with the exception of two points, which represent two unexpected major issues. Indeed, the data for the thrust, motor voltage, current and power and propeller revolutions present very low standard deviations (always less than 5.5%), as it can be seen from Tables 6.4.3, 6.4.4, 6.4.2 and 6.4.2 in Section 6.4.2. This shows that steady-state conditions have in fact been achieved during the runs, in particular concerning the circle manoeuvres. Conversely, the speed (and position) and rudder angle readings do not behave as well.

The reason why the rudder angle presents so high a standard deviation is clear from the analysis of the rudder angle demand log file. This is exemplified by Figure 6.4.2 in Section 6.4.2, which shows the plot of the rudder angle demand and actual rudder angle against time for a zigzag manoeuvre. The same oscillations in the rudder demand value are presented by all other log files from the zigzag and circle manoeuvring trials. The fact that these changes in rudder angle demand occur every 0.5 s is a strong hint for the cause of this behaviour: the heading controller. Indeed, it is known that this node was running in the background during the tests. However, it was erroneously thought that its lower publishing rate would result in its being overridden by the smach states nodes. Hence, the rudder angle results suffer only from a software problem, which can be easily corrected (already implemented) by publishing an additional flag (Boolean topic) whenever the zigzag and circle manoeuvres nodes are running, which is picked up (subscribed to) by the heading controller node. The heading controller will enter the while-loop

(which also results in the publishing of the rudder angle demand) only if the flag of the manoeuvring nodes is off (or 0). Indeed, as it can be seen from Figure 6.4.2 in Section 6.4.2, the close resemblance of the curves corresponding to the rudder angle demand and its actual value proves the effectiveness of the rudder responsiveness and control algorithm.

The issue concerning the position (x - and y -position) and speed data of the model is more serious. The unreliability of these results was already picked up from the straight run tests (Section 6.2) and is exemplified by Figure 6.4.3 and by the high standard deviation of the model speed in Tables 6.4.3. In particular, the graph of the x - and y -coordinates presents a model path that is much more discontinuous than the one that was observed in practice. In addition, the GPS signal was lost during the high speed runs. A similar even must have occurred during the circle manoeuvres at 1250 RPM, as nothing like Figure 6.4.3 in Section 6.4.2 was observed in practice. Hence, the cause for this behaviour is clearly the GPS unit and in particular its acquisition rate (too slow, the sampling period should be at least 0.1 s) and its accuracy (rated at 10 m, definitely too large a distance for these manoeuvres). However, this behaviour was not observed on the AUV Delphin2, which mounts the same GPS unit [Stenson et al., 2011]. It is believed that the main cause for the poor data quality experienced on the ASV is due to the location of GPS unit under the aluminium cover, which must have partly shielded its signal. However, the fact that the poorest data quality being observed during the high speed runs, with also two cases of loss of signal (during straight runs), seems to suggest the unsuitability of this unit for the higher model speeds, which could not be met by the AUV. For this reason, the University of Southampton has already acquired a new GPS unit with a faster sampling rate to substitute the current one so as to achieve greater measurements reliability and precision. Conversely, the compass unit has proven to be surprisingly accurate.

Finally, a clear human error has occurred during the zigzag trial number 3, where instead of launching the mission file for a zigzag manoeuvre with a propeller revolutions demand of 750 RPM, the one corresponding to a request of 1000 RPM must actually have been run. This is evident from Table 6.4.2 in Section 6.4.2, where the desired propeller revolutions demand has been left for clarity.

Data Analysis

Circle manoeuvring trial Unfortunately, since none of the model trials has used the same rudder angle as the full-scale ship, it is not possible to perform a direct comparison with the full-scale ship results presented in Section 6.4.1. In the future, such tests should be performed by slightly modifying the node dedicated to the circle manoeuvre.

Considering the results from the model trials (Tables 6.4.3, 6.4.4 and 6.4.2 in Section 6.4.2), the following conclusions may be drawn:

- The average model speed rises with propeller revolutions for constant turning rudder angle as expected, thus resulting in a decrease in the time duration of the circle manoeuvres.
- For constant motor setting, the model speed shrinks for increasing rudder angle. However, the total time of each run also decreases because of the smaller turning radii associated with higher rudder angles, which result in less distance to be covered.
- The (non-dimensional) turning radius does indeed show a reduction for higher rudder angles for fixed motor setting and for larger power for constant rudder angle. Additionally, it seems that a change in rudder angle of 5° is more influential on the average turning radius than a change of 250

RPM in propeller revolutions, although no conclusions can be drawn with confidence due to the unreliability of the position data as explained above.

- The thrust augments significantly with motor power and model speed as expected. For constant motor setting, the thrust increases with rudder angle despite the decrease in model speed. This is mainly due to the rise in resistance due to the drag and induced drag forces of the rudder (significantly augmenting due to stall). As a consequence, motor power increases as well in order to provide the required thrust force.

Zigzag manoeuvring trial Considering the zigzag manoeuvres data, contained in Tables 6.4.2 and 6.4.2 in Section 6.4.2, it can be seen that the first overshoot angle is smaller than the second one for all runs, with the exception of runs 4 and 6 due to wind effects as described in Section 6.4.2. This behaviour is also shown by the full-scale ship as shown in Section 6.4.1. The reason for this is expected to be the larger turning and sideways momentum the ship possesses after the first run, since it has to achieve a larger change in heading angle (double in fact as compared to the first bend because the ship has started from an initial straight run). A possible proof for this hypothesis is the fact that the full-scale ship (with larger inertia (mass) and speed and therefore momentum) presents a much larger difference between the second and the first overshoot angle, despite its rudder angle demand being only 10° as compared with the 20° used in the model testing. It is more difficult to predict the reason why the third overshoot angle is less than the second one (except for runs 4 and 6, which were biased by the wind effects). It is believed that this is due to the smaller model speed once the steady turning condition has been achieved, which results in a smaller momentum. However, it is difficult to prove this point in the absence of a fourth overshoot. As a consequence the node dedicated to the zigzag manoeuvre should be enlarged with the addition of a further while-loop in order to have the model perform a further curve. Another possibility for this behaviour would be a lateral bias of the model (having a stronger sideways force to port than towards starboard, possibly also due to waves or wind), but this has not been observed in practice.

Considering the model data only, it can be seen from Tables 6.4.2 and 6.4.2 in Section 6.4.2 that the overshoot angles are larger for larger model speeds (1250 RPM, run 5, as compared with 1000 RPM, corresponding to runs 1, 2 and 3). Unfortunately, no conclusions can be drawn from the runs with an expected propeller revolutions demand of 750 RPM. Indeed, run 2 has in fact been performed at 1000 RPM due to human errors, while run 6 should be discarded like run 4 due to the wind effects, which have corrupted the overall quality of the results. Furthermore, the thrust increases with motor power as predicted, which proves the efficacy of the strain gauge system of the thrust block.

Nomoto coefficients From Table 6.4.2, it is possible to see the relatively small standard deviation of the Nomoto coefficients obtained for the model. As aforementioned, the two main exception are runs 5 and 6, although the reasons for their discrepancy are different. The inconsistency of the data of trial 6 as compared with the other runs is due to the wind and/or waves generated acting on the rib during this manoeuvre, as described above, whose effects have been exacerbated by the slow model speed. For the same reason, the results from run 4 should have been discarded, but they have been kept so as to have some data from a throttle setting other than 1000 RPM, since their quality is not excessively poor, possibly due to the higher speed, which must have dwarfed the other negative effects. The reason why T' is negative in run 5 is probably the negative turning rate at zero heading after the second overshoot. This may have been caused by waves or purely by the rudder angle demand changing near that point because of the issue in the heading controller.

Comparing the Nomoto coefficients of the full-scale ship and the model, it is possible to see that the model results in a more controllable ship. Indeed, while its K'/T' ratio is above the suggested minimum (0.6, see Fossen [2011]), for the full-scale ship is actually lower than the one deemed acceptable for large tankers [Fossen, 2011]. However, the full-scale ship data should be treated with caution, as it belongs to one manoeuvre only, executed with 40 knots of wind. For the same reason, more tests should be performed with the model in the future with different throttle settings and rudder angles. However, the results obtained so far are very sensible, enabling a good description of the ship motion, as shown in Section 3.6.6, thus proving the validity of the selected methodology.

6.5 Rudder Comparison

[ANDERLINI]

6.5.1 Results

Some manoeuvring trials have been performed with the smaller rudder (geometrically scaled, see Section 3.5) in order to compare the behaviour of the two rudders and possibly estimate which one resembles closest the manoeuvring characteristics of the full-scale ship. Although both circle and zigzag manoeuvres have been performed, most of the data was of limited accuracy due to the strong winds encountered during this part of the testing. In addition, the analysis of circle manoeuvres was deemed unnecessary after the very poor reliability of the position data was found (Section 6.4.3).

The relevant data from the zigzag manoeuvres has been reported in Tables 6.5.1, 6.5.1, 6.5.1 and 6.5.1. It can be noted that this analysis is the same as per Section 6.4.2. In addition, in Table 6.5.1, the proportional gain for a P-controller has been reported for both manoeuvres so that it can be compared with the values obtained for the full-scale ship and the model with the other rudder, as can be seen in Section 3.6.6.

Table 6.5.1: Motor setting, rudder demand, general comments, average rudder angle, 1st, 2nd and 3rd overshoot angles recorded during the zigzag manoeuvring trials.

Run	Prop demand (RPM)	Angle demand (°)	Comments	Average Rudder Angle (°)	1 st overshoot (°)	2 nd overshoot (°)	3 rd overshoot (°)
1	1000	20-20	head waves	-2.341	5.8	6.4	4.9
2	1000	20-20	following waves	-3.734	7.2	6.8	7

Table 6.5.2: Total time-length, average thrust, speed and motor voltage for the zigzag trials. Furthermore, the standard deviation of the last three variables has been computed as well.

Run	Total Time (s)	Speed (m/s)		Thrust (N)		Motor Voltage (V)	
		Average	Std. Dev. (%)	Average	Std. Dev. (%)	Average	Std. Dev. (%)
1	51.4	0.6136	27.57%	8.7607	0.73%	6.3015	1.06%
2	47.7	0.7547	11.90%	8.4180	0.69%	6.2573	1.01%

Table 6.5.3: Average motor current and power and propeller revolutions measured during the zigzag manoeuvres. Additionally, the standard deviation of the data is included as well.

Run	Motor Current (A)		Motor Power (W)		Propeller revs (RPM)	
	Average	Std. Dev. (%)	Average	Std. Dev. (%)	Average	Std. Dev. (%)
1	2.4807	3.46%	15.6347	3.88%	1000.5	0.001%
2	2.4292	3.49%	15.2006	3.77%	1000.5	0.001%

6.5.2 Data Analysis

[ANDERLINI]

The error analysis should be the same as per Section 6.4.3. Nevertheless, it is interesting to notice the smaller standard deviations for all variables (except for the speed due to the issues with the GPS), which

Table 6.5.4: Dimensional and non-dimensional Nomoto coefficients calculated from the measurements of the zigzag manoeuvres. The gain for a P controller obtained from the Nomoto equation with these indices is reported as well according to Section 3.6.6.

Run	K (rad/s)	K'	T (s)	T'	$ K'/T' $	K_P
1	-0.1871	4.3899	-0.8843	0.9288	0.9520	0.812
2	-0.2215	4.2524	-0.8511	1.1067	0.7691	0.708

is probably due to milder environmental conditions (wind and waves) and the smaller influence of the rudder (which was operated by the heading controller as described in Section 6.4.3).

Comparing the results of Table 6.5.1 with those of Table 3.6.6 in Section 3.6.6, it is possible to notice that the smaller rudder requires a larger proportional gain in a P-controller for both manoeuvres. This is as expected, as the shorter chord results in a smaller lift force and the smaller area in a smaller distributed sideways load and therefore a decrease in manoeuvrability. The even larger gain required by the full-scale ship would show that the use of the smaller rudder would be more appropriate. Nevertheless, this is believed not to be the case, as the data available for the full-scale ship is limited and thought to be biased by strong side winds (see Section 6.4.3).

In addition, if Table 6.4.2 and Table 6.5.1 are compared, it is possible to see that the smaller rudder results in a larger $|K'/T'|$ than the larger one, whose was already greater than the suggested minimum and for that of the full-scale ship as well [Fossen, 2011]. This does not make sense and requires future repeat tests in order to understand the cause for this behaviour.

6.6 Summary of Results

[ANDERLINI & LE]

As part of this project extensive model testing has been performed both to study the influence of EIs such as changes in trim and in bow designs for different loading configurations and to prove the efficiency and flexibility of the ASV as a testing platform.

Initially, estimates for the naked hull resistance for a range of speeds have been derived during towing tank testing at QinetiQ, in Haslar, where a sufficient amount of data was measured for each run. These experiments have been performed in two different trim conditions at full-load and two trim conditions in ballast loading conditions. The results for the ballast condition and default trim are in good agreement with the experimental measurements performed by Collison et al. [2012] in the Solent University tank.

On the second day of testing at QinetiQ, it has been possible to perform self-propulsion tests at the scaled operational speed for two different trim conditions in both ballast and full loading. From these tests, the wake fraction, thrust deduction and relative rotative efficiency have been derived reliably for each condition, so that the respective propulsive efficiency could be determined. Hence, it has been found that while the change in trim results in an increase in powering in the full-loading condition, it causes a beneficial 4.5% reduction in propulsive efficiency in ballast. Additionally, from the tests, it has been possible to determine the model self-propulsion point at the operational speed, which has been implemented in the control software in order to achieve such a velocity.

Although the self-propulsion tests have also been employed to validate the thrust readings measured by the thrust block, it was necessary to first calibrate the system through bollard-pull tests. Indeed, the results could be used to determine the relationship of strain in the thrust block sensor with the thrust of the model. Although bollard-pull tests have been performed at QinetiQ in both full and ballast loading conditions, it has been necessary to repeat them in the Lamont towing tank at the University of Southampton due to an offset in the value of the shaft revolutions at QinetiQ.

In the final stages of the project, free-running tests have been performed at Timsbury Lake to show the potential of the ASV as a self-measuring platform. Initially, straight runs have been executed for a range of throttle settings around the self-propulsion point in order to assess the influence of the real environment, i.e. wind and especially waves, on the powering requirements of the model. Furthermore, two additional bow designs, which have been produced as part of a separate individual project at the University of Southampton by Robert Cooke, have been tested as well to assess their efficiency in waves and to prove the flexibility of the ASV. Although one of the alternative bows appears to present reduced powering requirements, these results should be treated with caution due to the very poor signal the GPS experienced at the lake. Additionally, the rudder could have increased the model resistance while adjusting to the course the model was required to follow. Nevertheless, the quality of the measurements from the low-level systems has been of extremely high quality (standard deviations lower than 6%).

During lake testing, circle and zigzag manoeuvring trials have also been performed to assess the manoeuvring capabilities of the model with the rudder scaled according to the lift coefficient. Different throttle settings and rudder angles have been employed to assess their impact on the tactical diameter for circle manoeuvres and the overshoot for zigzag tests. On the one hand, the circle manoeuvring results should be treated with great caution to the very poor quality of the position data (GPS). On the other hand, the zigzag trials provided reliable data, which was also used in the development of a simple mathematical model of the motions of the ASV. This in turn has been employed to produce a PID heading controller for future use. Additionally, by applying this method and repeating the manoeuvres for the geometrically scaled rudder, it was possible to observe the lower control costs associated with the larger rudder (scaled with the lift coefficient).

CONCLUSION

This project set about to develop an autonomous self-propulsion model, capable of self-measuring small-scale changes in powering and manoeuvring characteristics in an open-water environment. To assess the success of the project this aim can be broken down, just as at the beginning of the report.

The model's **self-propulsion** system was completely replaced, using a new motor and a linear-rail concept. The motor was perfectly capable of powering the model and was not limited in terms of rotational speed or power.

A detailed study into the rudder system, which forms part of the self-propulsion system, was performed - showing interesting comparisons between a geometrically-scaled rudder and a lift-coefficient-scaled rudder. The mechanism itself had no issues, electrically or mechanically, and provides a means of replacing rudders quickly. The process of rapid-prototyping rudders has been experienced and documented, and it would be a simple matter to build and test multiple different rudders on the model in the future.

The electronics which handled the powering and control of the motor and rudder were also successful. Results showed very constant motor RPM and high repeatability between experiments which was important.

In terms of producing a **self-measuring** model, there were mixed results. Unfortunately, the torque dynamometer had wiring issues, which could not be resolved due to time constraints. However, if this was fixed in the future then it should be a simple process to implement the amplification, Bluetooth communications and integration to the autonomy unit - since all of this has been tackled in the project already, for different sensors.

The project could continue as intended, using the motor current sensor to provide the torque measurements. There was no opportunity to thoroughly calibrate the current sensor to the shaft torque, as it was always intended to use the torque dynamometer for the final torque measurements - however, using the manufacturer's information it was possible to obtain reasonable results for the propeller torque.

The current sensor, like all the low-level logic sensors such as the RPM, thrust, voltage, temperature and rotation sensors performed excellently, with very high resolution, accuracy and the potential for fast sample rates. Although the electronics which implement the amplification and sampling are complex, the end result is a system that is easy to use; and certainly a match for commercial signal processing units.

The principle of a sliding power train appears to have worked well as a concept for effective thrust measurements, and could set the precedence for a range of investigations into more advanced linear sliding mechanisms with reduced damping. The results from the various bollard pulls and self-propulsion tests give a good indication that the thrust sensor and linear sliders were functioning reliably.

There were some issues with the high-level sensors, particularly with the GPS, which connected directly to the autonomy unit; although they were good enough to test the relevant functionality as a proof of concept. In the future, a better GPS could be added with minimal effort.

The **autonomy** system performed well after some teething problems, which were to be expected. Although there is plenty of scope for improvement, there is no denying the fact that the autonomy unit, combined with the rest of the system, managed to perform repeatable powering and manoeuvring tests with minimal human interference - and log data as it did so.

The model was tested in an **open water** environment, in which it experienced waves and wind -

although on a smaller scale than would be necessary for proper wave-affected trials. The wave buoys which were required to measure the waves were not complete in time for testing in parallel with the model, but this was of little concern since both parts of the project were tested independently. The small waves that did occur during testing of the vessel were merely a small inconvenience rather than a measurable variable of the tests. Asides from verification of the wave buoys in a natural wave environment and possibly fitting a better accelerometer, the wave buoys are ready to be used alongside the vessel in a more pronounced seaway.

Overall, the project can be considered a success since it fulfilled each part of the aim in one way or another, though there are significant improvements to be made to most of these parts (*e.g.* by completing the torque dynamometer and fixing the GPS).

Results were obtained successfully, and changes in trim and bow sections have been evaluated for their changes in powering performance. Two rudders were also tested, revealing trends in the manoeuvring performance. For the most part, reliable conclusions were difficult to draw due to small problems with the GPS which could easily be fixed in the future.

Also, a significant amount of work must be done to make the system more accessible to future users; although the system *is* capable of performing the necessary trials to test efficiency improvements, it really requires an expert in ROS and a good understanding of the entire system to use it effectively.

7.1 Group Management

[HAWKES]

The project was very diverse, consisting of many sections which depended on each other, and some sections (such as the wave buoys) which were very independent. Where possible, at least two people were assigned to each section so that there was some redundancy. In reality though, it was simply impossible for two people to learn everything autonomy-related, electronics-related or OrcaFlex-related (etc.) because of the ambitious project aim, and the human resources available.

There were a number of major time-consuming administrative roles which were filled by members of the group:

- James Hawkes was the group leader;
- Kimberley Neale was the treasurer;
- Hieu Le was the BP correspondent;
- Enrico Anderlini was the QinetiQ correspondent;
- Heather Crossley was the Timsbury Lake correspondent.

7.2 Budget Summary

[NEALE]

This project was jointly funded by the University of Southampton and BP Shipping. An initial budget estimate was created at the start of the project whereupon it was discovered that extra funds would be needed to be raised in order to develop the wave buoys. This extra funding was obtained through an elevator pitch to the university.

The greatest overspend was on the model which was due mainly to unexpected costs associated with the electronics. The costs of transporting the model and printing the report were also greater than anticipated. However, other areas of the project were under-budget. Originally it was planned to test a number of retro-fit devices, however as the complexities associated with creating the testing platform

Table 7.2.1: Summary of funds and expenditures

	Available Funds (£)	
University	860.00	
Elevator Pitch	500.00	
Sponsor	3848.64	
Total	5208.64	
	Expenditure (£)	
	Predicted	Actual
Model	1019.91	1732.16
Retro-fits	145.00	0.00
Wave Buoys	722.76	549.84
Facilities	2140.00	2040.00
Transport	380.00	516.05
Printing	76.00	200.00
Contingency	724.97	-
Total	5208.64	5038.05

became apparent, it was decided that this would be outside the scope of the project. At the start of the project it was also planned to create a matrix of wave buoys, however due to design considerations it was decided to create a singular buoy which incorporated a greater number of features. Finally the cost of facilities was also slightly under-budget, as use of Timsbury Lake for model testing was offered free of charge. A contingency of £724.97 had been included in the original budget; of this £554.38 was spent. This put the final expenditure for the project £170.59 under-budget. This money can be used to develop the model in the future.

FURTHER WORK

There are a number of areas which should be considered for further work. This section will outline some general ideas, as well as identify further work which could be performed to certain sections of the model.

- There are a number of 3D models of the model's hull, including a NURBs model and a 3D polylines solid model, but neither is very accurate because they are just constructed from a table of offsets. It is recommended that a 3D scan of the model be performed in order to facilitate accurate numerical studies of the hull.
- Currently, a comprehensive knowledge of ROS and a reasonable knowledge of the electronics system is required to run the model. In order to make the vessel user-friendly for the average student a comprehensive user guide could be created. Ideally, more time should be spent on developing the system so that it requires less user-input or detailed knowledge of how it works.

8.0.1 Power Train

- The motor, electronics, sensors and autonomy unit all support reversing the propeller however the power train does not. This problem can be rectified by installing a twin thrust-bearing in order to allow the thrust block to transmit reverse load.

8.0.2 Torque Dynamometer

- Clearly, the torque dynamometer should be completed and then calibrated using some form of external calibration rig. A suitable method for attaching the electronics onto the holder needs to be developed which will hold components securely whilst spinning at high RPM yet will still allow them to be easily removed.

8.0.3 Electronics

- The reference voltage inputs of the various op-amps should be controlled via the output pin of the relevant MCU to allow dynamic compensation of zero-offset errors on the signals. Commands could then be written which allow automatic zero-ing of all the sensors prior to a test routine.
- Similarly, the gain should be remotely controlled by using a resistor relay, allowing gain to be selected from software - rather than from rotary potentiometers spread around the vessel.
- In order to improve the functionality and versatility of the vessel, there are many additional sensors which could be added to the model such as IMUs, echo sounders, anemometers or webcams. Additionally, hardware could be added to make the vessel such as bow thrusters or electronic anchor systems, so long as the realism of the vessel is not sacrificed.

8.0.4 Autonomy System

- As mentioned in section 3.6, more accurate GPS and compass sensors with higher sampling rates should be installed.
- Concerns were raised regarding the WIFI signal range, so investigations into better solutions are necessary. For example, a long-range WIFI transceiver could be installed on the receiving computer, or the current WIFI dongle on the model may be substituted with a more powerful serial transmission unit like those installed in the wave buoys (4.2.2).
- To avoid future collisions, the system should shut down in the instance of signal loss. A simple program on the Beagleboard could continuously ping the receiving computer. If the signal is lost, the ping does not come back, so program shuts down all nodes (except the Arduino and the Arduino converter nodes) via a bash command on the terminal. It should publish a value of 0 for the motor setting and rudder angle demand topics, as described in section 3.6.15.
- The whole control system could be made much more user-friendly with the addition of a GUI (Graphical User Interface). This should be developed for the receiving computer and in its simplest form should call the required nodes and launch files on the terminal of the sending computer via the SSH connection. It should be able to specify direct values for specific topics to allow manual control of the ship. If a more complicated, fancier GUI is desired, it is possible to use Gazebo [ROS.org, 2013], but this requires a Linux operating system and ROS installation on the receiving computer. This library enables the development of a three-dimensional environment, where the position and orientation of the model are constantly updated using topics published from the compass and the GPS. It is possible to specify the desired path of the model (through transit points) for obstacle avoidance [Pickem et al., 2012].
- It would be interesting to develop different types of controllers to compare their robustness and efficiency with the current PID controller. This could be done through free running tests where the model is run in a predefined path, e.g. a square, and the overshoot, the settling time and the rise time of the response are compared, as well as the energy costs of each solution. If the manoeuvring coefficients are estimated experimentally (very costly, especially for planar motion mechanisms [Lewis, 1989]) or numerically (standard in commercial software, such as Star-CCM+ [CD-adapco, 2013], or non-commercial solutions [Arakia et al., 2012]), it is possible to obtain a state-space model from the linearised equations of motions of the ASV. This could be used to design suitable controllers based on the requirements for overshoot, rise and settling times. Furthermore, a LQG controller could be derived as well where the cost parameter should be the electric current. Additionally, an observer could also be implemented. Extensive free running tests should determine the advantages and disadvantages of each procedure and which one provides the best solution.

8.0.5 Weights and Centres

- A rig should be developed, or an existing one used, suitable for measuring the moment of inertia of the hull, especially if sea-keeping tests or more extensive manoeuvring trials are performed.
- An automatic trimming system could be built, which would remotely move weights around the vessel using a stepper motor and linear rail system. This would require an electronics module to be developed to control the system.

8.0.6 Wave Buoys

- Initially, further testing of the buoy should be performed in the actual environment, hopefully revealing more usual directional wave spectra.
- Further work regarding the wave buoy design should focus on the testing of hull geometries in more sea states, particularly focusing on the response spectra in more irregular waves found on a lake or open water. Buoy motions should be validated by a separate form of measurement, such as a high-speed camera, for use in studying the physical responses of various hull geometries in the wave environment. Analysis of the viscous effects of wind resistance on the buoy response due to the profile above the waterline should also be carried out.
- A more comprehensive investigation into various buoy geometries should also be conducted. Other possible shapes such as a ring or sphere, and their responses in various sea spectra should be investigated to find an optimum buoy profile for contouring the desired small scale wave heights and frequencies.
- The use of materials and construction techniques such as vacuum-infused GFRP (glass fiber reinforced polymer) would provide protection from bumps that may hinder a Styrofoam buoy's performance, whilst still remaining relatively lightweight. This would also provide the capability of manufacturing a hollow buoy, allowing for increased space for on-board equipment or a further reduction in buoy diameter to improve the ability to contour wave profiles closer.

8.0.7 Testing

- The model is virtually ready to be used as a test platform for various efficiency improvements. As intended, there is scope for a very large amount of future testing - including a more comprehensive trim study, the testing of more efficiency improvements, and the testing of other modifications to the vessel.

BIBLIOGRAPHY

- A & D Ltd. Measurement principle of load cells, 2012. URL www.aandd.jp/products/weighing/loadcell/introduction/. Accessed on: 05/01/2013.
- Acksys. Acksys, 2013. URL www.acksys.fr/. Accessed on: 21/03/2013.
- Jaap Allema. Intercepting the interceptor, September 2005. URL <http://www.marin.nl/web/News/News-items/Intercepting-the-Interceptor-at-MARIN.htm>.
- Motoki Arakia, Hamid Sadat-Hosseinib, Yugo Sanadaa, Kenji Tanimotoa, Naoya Umedaa, and Frederick Stern. Estimating maneuvering coefficients using system identification methods with experimental, system-based, and cfd free-running trial data. *Ocean Engineering*, 51:63–84, 2012.
- Arduino.cc. Arduino introduction, 2012. URL arduino.cc/en/Guide/Introduction. Accessed on: 12/11/2012.
- G. Baddeley. Gps - nmea sentence information, 2001. URL <http://aprs.gids.nl/nmea/>. Accessed on: 21/03/2013.
- Beagleboard.org. Beagleboard-xm product details, 2011. URL www.beagleboard.org/hardware-xm. Accessed on: 21/03/2013.
- H. O. Berteaux. *Buoy Engineering*. Wiley-Interscience, 1976. ISBN 0-471-07156-0.
- C.T. Bishop and M.A. Donelan. Measuring waves with pressure transducers. *Coastal Engineering*, 1987.
- R.E.D. Bishop and A.G. Parkinson. Directional stability and control of rigid marine vehicles. *Department of Mechanical Engineering, UCL*, 1976.
- R. Bowen, T. Coe, A. Kay, and A. Palmer. The design, build and refit of self propulsion and open water test rigs. *University of Southampton: Group Design Final Report*, 2005.
- Ø. Buhaug, J.J. Corbett, Ø. Endresen, V. Eyring, J. Faber, S. Hanayama, D.S. Lee, H. Markowska A.Z. Mjelde A. Neilssen D. Lee, D. Lindstad, C. Nilsen, J. På lson, J.J. Winebrake, W. Wu, and K. Yoshida. Second imo ghg study 2009. Technical report, International Maritime Organisation (IMO), london, UK, April 2009.
- J.R. Calvert and R.A. Farrar. *An Engineering Data Book*. Palgrave MacMillan, 3rd edition, 2008.
- Canonical. Ubuntu, 2013. URL www.ubuntu.com/. Accessed on: 21/03/2013.
- J Carlton. *Marine Propellers and Propulsion*. Elsevier.Ltd, Oxford, UK,, 2012.
- CD-adapco. Star-CCM+, 2013. URL www.cd-adapco.com/products/star_ccm_plus/. Accessed on: 12/04/2013.

- Robbie Collison, Marion James, Simon Mathieson, Christophe Rident, Charles Scott, and Edward Unwin. Retro-fit solutions for energy efficient shipping. *University of Southampton: Group Design Final Report*, 2012.
- Datawell. Direction waverider gps. 2012.
- Johan H. de Jong. A framework for energy saving devices. Technical report, Maritme Institute Netherlands (MARIN), n.d. URL www.ship-efficiency.org/onTEAM/pdf/06%20de%20Jong_Marin.pdf.
- Dell. Latitude e4200 laptop, 2013. URL www.dell.com/us/business/p/latitude-e4200/pd. Accessed on: 21/03/2013.
- DNV. Fuel saving by trim optimisation. Technical report. URL www.dnv.com/binaries/fuel%20saving%20by%20trim%20optimisation_tcm4-528861.pdf. Accessed on: 05/03/2013.
- D. Drake. Writing udev rules, 2006. URL www.reactivated.net/writing_udev_rules.html. Accessed on: 21/03/2013.
- L. Draper. Wave measurements at north atlantic weather stations india and juliett in the nineteen seventies (india 2 and juliett 2. 1986.
- Mark Drela and Harold Youngren. Harold youngren, 2001. URL www.web.mit.edu/drela/Public/web/xfoil/xfoil_doc.txt. Accessed on: 30/03/2013.
- P F Dunn and T J Mueller. Fluid dynamic and acoustic experiments on turbomachine rotors subject to variable flow conditions. Technical Report 1783-AR1, University of Notre Dame, 1990.
- ENIRAM. Vlcc propulsion energy efficiency study, 2012. URL www.eniram.fi/resources/VLCC-tanker-study-order. Accessed on: 25/02/2013.
- A Falkner. A capacitor-based device for the measurement of shaft torque. *IEEE Transactions on Instrumentation and Measurement*, 45(4):835–838, August 1996.
- F.Morishita. Practical aspects of inclining experiments. 1996-1997.
- E.J. Foeth, R. Eggers, and F.H.H.A Quadvlieg. The efficacy of air-bubble lubrication for decreasing friction resistance. Maritime Institute Netherlands (MARIN), 2010.
- Thor I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley and Sons Ltd., 1st edition, 2011.
- Gene F. Franklin, J. David Powell, and Abbas Emami-Naeini. *Feedback Control of Dynamic Systems*. Pearson Prentice Hall, 2010.
- Fugro OCEANOR. Seawatch wavescan buoy. 2010.
- GlobalSat. Gps receiver, 2013. URL www.globalsat.com.tw/products-page.php?menu=2&gs_en_product_id=2&gs_en_product_cnt_id=76. Accessed on: 21/03/2013.
- Hans Richard Hansen, Tom Dinham-Peren, and Takeo Nojiri. Model and full scale evaluation of a 'propeller boss cap fins' device fitted to an aframax tanker. In *2nd International Symposium on Marine Propulsors*, Hamburg, 2011.

- D. Hauser, K. K. Kahma, H.E.Krogstad, S.Lehner, J.Monbaliu, and L.Wyatt. Measuring and analysing the directional spectrum of ocean waves. 2003.
- Gregory Hitz. Autonomous navigation and data collection for the limnobotics autonomous lake sampling boat. Master's thesis, Swiss Federal Institute of Technology Zurich, 2010.
- Paul Horowitz and Windfield Hill. *The Art of Electronics*. Cambridge University Press, 2 edition, 1989. ISBN 978-0-521-37095-0.
- Tetsuji Hoshino, Akira Oshima, Kazunobu Fujiya, Takao Kuroiwa, Fumio Hayashi, and Etsuo Yamazaki. Development of high-performance stator fin by using advanced panel method. Technical report, Mitsubishi Heavy Industries Ltd., December 2004.
- IMO. Eedi - rational, safe and effective, August 2012a. URL www.imo.org/MediaCentre/HotTopics/GHG/Pages/EEDI.aspx. Accessed on: 12/03/2013.
- IMO. International shipping facts and figures - information resources on trade, safety, security, environment. March 2012b.
- IMO. Sea-going vessels and maritime technology - instructions for planning, carrying out and reporting sea trials. April 2005.
- Independent Maritime Assessment Associates. Wavetector - a small wash/wave measuring buoy. online, 2012. URL <http://homepage.ntlworld.com/john.lewthwaite1/index.htm>. Accessed on: 12/04/2013.
- ITTC. Recommended procedures: Full scale manoeuvring trials procedure. Technical report, International Towing Tank Conference, 2002a.
- ITTC. Recommended procedures: Testing and extrapolation methods, propulsion, performance propulsion test. Technical report, International Towing Tank Conference, 2002b.
- J.M.J. Journée and Jakob Pinkster. *Introduction in Ship Hydromechanics*. Delft University of Technology: Lecture MT519, 2002.
- Gabor Karafiath and Steven C. Fisher. The effect of stern wedges on ship powering performance. *Naval Engineers Journal*, 1987.
- S. Krueger. *Grundlagen der Propulsion*. Technische Universitat Hamburg-Harburg, 2004. in German.
- S. Krueger. *Schiffspropeller*. Technische Universitat Hamburg-Harburg, 2008. in German.
- E. V. Lewis, editor. *Principles of Naval Architecture*. Society of Naval Architects and Marine Engineers, 1989.
- A. R. J. M. Lloyd. *Seakeeping: Ship Behaviour in Rough Weather*. Ellis Horwood Ltd, Chichester, UK, 1998.
- S.O.H. Madgwick, A. J L Harrison, and R. Vaidyanathan. In *Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on Estimation of IMU and MARG orientation using a gradient descent algorithm*, pages 1–7, 2011. doi: 10.1109/ICORR.2011.5975346.
- Manly Hydraulics Laboratory. Analysis of ocean wave data, 2000. URL www.mhl.nsw.gov.au/www/wave_data_stats.htmlx. Accessed on: 03/03/2013.

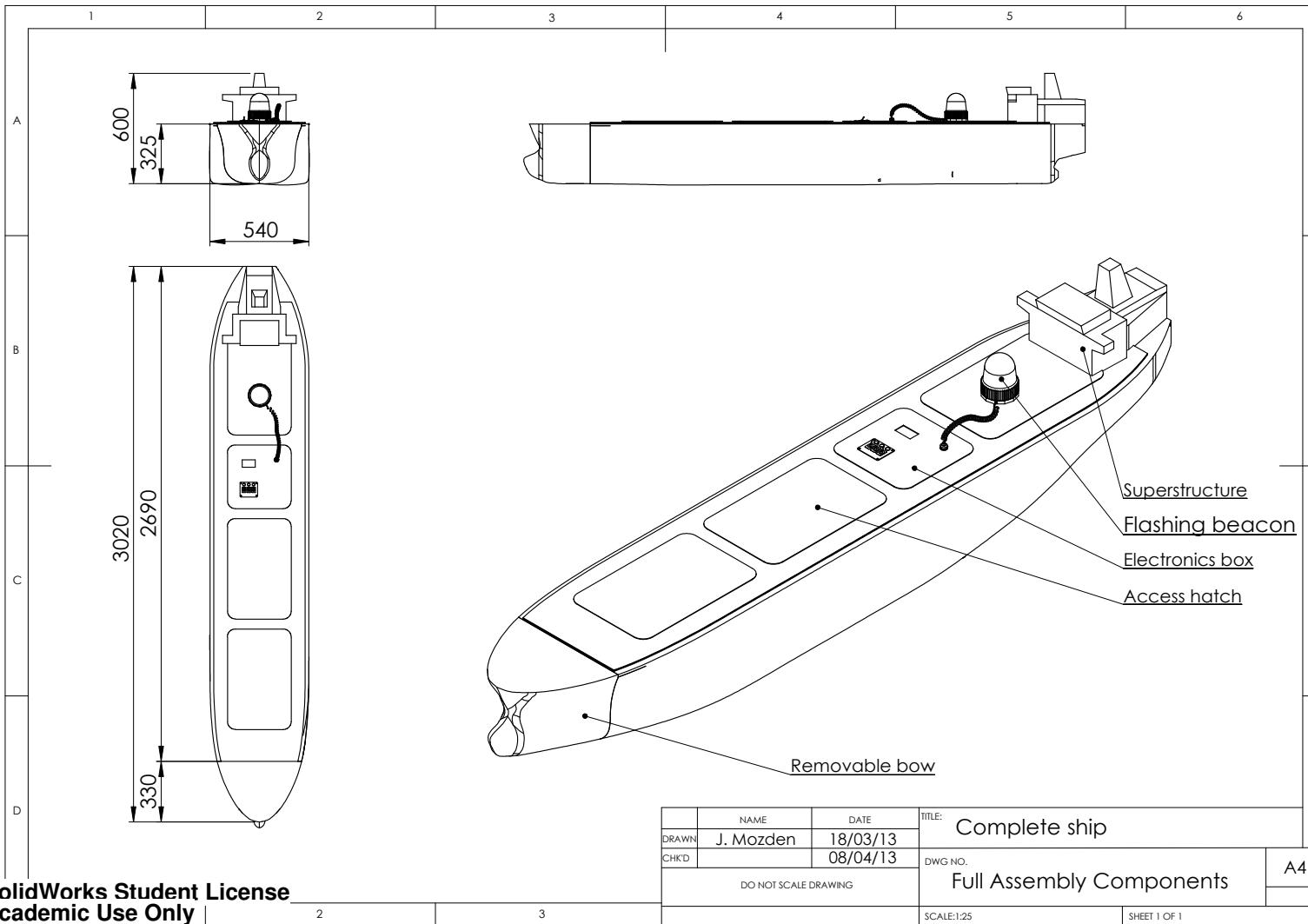
- McPhail and Pebody. Navigation and control of an autonomous underwater vehicle. *Underwater Technology*, 23(1), 1998.
- MET Office, 2013. URL www.metoffice.gov.uk/. Accessed on: 15/03/2013.
- A.F. Molland and S.R. Turnock. The design, construction and calibration of a thrust and torque dynamometer for a wind tunnel propeller model. Technical report, August 1990.
- Anthony Molland, Stephen R. Turnock, and Dominic A. Hudson. *Ship Resistance and Propulsion: Practical Estimation of Propulsive Power*. Cambridge University Press, 2011. ISBN 9780521760522.
- Anthony F. Molland and Stephen R. Turnock. *Marine Rudders and Control Surfaces*. Butterworth-Heinemann, 2007.
- L. Moreira and C. Guedes Soares. Autonomous ship model to perform manoeuvring tests. *Journal of Maritime Research*, VIII(2):29–46, 2011.
- L. Moreira, Thor I. Fossen, and C. Guedes Soares. Path following control system for a tanker ship model. *Ocean Engineering*, 34:2074–2085, 2007.
- Movable Type Scripts. Calculate distance, bearing and more between latitude/longitude points, 2013. URL www.movable-type.co.uk/scripts/latlong.html. Accessed on: 21/03/2013.
- Tiberiu Muntean. Ship propulsion train efficiency sensing. *WARTSILA Technical Journal: in detail*, (2): 34–38, 2008.
- Richard M. Murray. *PID Control*. California Institute of Technology, 2006.
- National Data Buoy Center. Nondirectional and directional wave data analysis procedures. Technical report, Stennis Space Center, January 1996.
- National Data Buoy Center, 2012. URL <http://www.ndbc.noaa.gov/>. Accessed on: 12/04/2013.
- National Maritime Research Institute, Japan. Energy saving by covering a ship with air bubbles, n.d. URL www.nmri.go.jp/turbulence/group/040615Energy_saving_by_microbubbles.pdf.
- A. Neundorf. Welcome to cutecom, 2009. URL <http://cutecom.sourceforge.net/>. Accessed on: 21/03/2013.
- nixCraft. `/etc/network/interfaces` ubuntu linux networking example, 2013. URL www.cyberciti.biz/faq/setting-up-an-network-interfaces-file. Accessed on: 21/03/2013.
- Nokia. Qt project, 2013. URL qt-project.org. Version 5.0.2.
- OceanServer. Digital compass users guide, os5000 series, 2010. URL www.ocean-server.com/download/OS5000_Compass_Manual.pdf. Accessed on: 21/03/2013.
- OpenCFD Ltd. Openfoam-the open source cfd toolbox, 2012. URL www.openfoam.com. Accessed on: 28/10/2012.
- Opengroup. The x-window system, 2013. URL www.opengroup.org/tech/desktop/x. Accessed on: 21/03/2013.
- Orcina. Orcaflex. online, 2013. URL www.orcina.com. Accessed on: 12/04/2013.

- R. Owens and P. Palo. Wind-induced steady loads on ships. 1982.
- A.B. Philips, L.V. Steenson, J. Liu, C.A. Harris, S.M. Sharkh, E. Rogers, S.R. Turnock, and M. Furlong. Delphin2: an over actuated autonomous underwater vehicle for manoeuvring research. *Transactions of the Royal Institution of Naval Architects*, A, 2010.
- Philips Semiconductors. The i2c-bus specification, 2000. Version 2.1.
- Daniel Pickem, David Morioniti, Chris Taylor, Santiago Balestrini-Robinson, and Dimitri Mavris. Captain hindsight: An autonomous surface vessel. Master's thesis, Georgia Institute of Technology, 2012.
- P.L.Baker. Aircraft and satellite measurement of ocean wave directional spectra using scanning-beam microwave radars. *Journal of Geophysical Research*, 90:987–1004, January 1985.
- Kevin Pleiss. Integrating life-cycle solutions for maximum return on investment, May 2011. URL <http://e2s2.ndia.org/schedule/Documents/Abstracts/12474.pdf>.
- L Qingde and J.G. Griffiths. Least squares ellipsoid specific fitting. *Geometric Modeling and Processing*, 2004.
- K. J. Rawson and E. C. Tupper. *Basic Ship Theory Volume 2*. Butterworth-Heinemann, 2001.
- V. Renaudin, M.H. Afzal, and G. Lachapelle. Complete triaxis magnetometer calibration in the magnetic domain. *Journal of Sensors*, 2010.
- Renilson Marine Consulting Pty Ltd. Reducing underwater noise pollution from large commercial vessels. Technical report, March 2009.
- ROS.org. Documentation, 2013. URL www.ros.org/wiki/. Accessed on: 21/03/2013.
- RS AQUA Ltd. Waveradar rex - wave, sea level and air gap sensor.
- Thomas Rung. Numerische thermofluidynamik i. TUHH Lecture Notes, 2012.
- Schneekluth Hydrodynamik. Schneekluth wake equalizing duct - w.e.d. URL www.schneekluth.com/en/index.php.
- Sensor Technology. Optical technology, 2012. URL www.sensors.co.uk/about-us/technology/optical/. Accessed on 05/01/2013.
- SSPA. Energy saving devices, 2011. URL www.hydrodynamics.se/hydrodynamics/energy-saving-devices. Accessed on: 02/03/2012.
- L. V. Steenson, A. P. Phillips, M. E. Furlong, E. Rogers, and S. R. Turnock. Maneuvering of an over-actuated autonomous underwater vehicle using both through-body tunnel thrusters and control surfaces. In *17th International Undersea Untethered Submersible Technology Conference*, Lee, United States, 2011.
- R. Stewart. *Introduction to Physical Oceanography*. Orange Grove Texts Plus, September 2009.
- V. Sundar, S.A. Sannasiraj, and R. Balaji. Buoy oh buoy. *Journal of Ocean Technology*, 2008.
- Jan Øivind Svardal and Friedrich Mewis. Three years of experience with the mewis duct - a contribution to ship efficiency, 2011. URL www.ship-efficiency.org/onTEAM/pdf/04%20Mewis_Svardal_Grieg%20Shipping.pdf. Accessed on: 02/03/2013.

- S. Tatham. Putty: A free telnet/ssh client, Feb 2013. URL www.chiark.greenend.org.uk/~sgtatham/putty/. Accessed on 15/03/2013.
- E. A. Terray, B. Brumley, and B. Strong. Measuring waves and currents with an upward looking adcp. 1999.
- Triaxys Technologies Inc. Triaxys directional wave buoy.
- S. R. Turnock and A.F. Molland. Energy efficiency of ship propulsive systems: rudder-propeller interaction. In *Ship Propulsion Systems Conference*, London, UK, January 2010.
- S.R. Turnock. Course notes: Advances in resistance and propulsion. 2012.
- University of Michigan. Introduction: Pid controller design, 2012. URL www.ctms.engin.umich.edu/CTMS/index.php?example=Introduction§ion=ControlPID. Accessed on: 21/03/2013.
- VAF Instruments. Tt-sense optical thrust and torque measuring systems. product bulletin 663, 2012. URL http://www.vaf.nl/downloadfile/&file=/storage/files/PB-663-GB-1212_TT-Sense.pdf. Accessed on: 17/01/2013.
- W. P. A. Van Lammeren, J. D. Van Manen, and M. W. Oosterveld. The wageningen b-screw series. *Transactions of SNAME*, pages 269–317, 1969.
- J.F. Wellicome, P. Temarel, and A.F. Molland. The design, construction and calibration of a wave buoy for ship model tests in open water. *Ship Science Report*, (11), 1999.
- R. L. Wiegel. *Directional Wave Spectra Applications*. American Society of Civil Engineers, 1982.
- Wolfson Unit. Propeller design program, 2012. URL www.wumtia.soton.ac.uk/software/propeller-design-program. Accessed on: 28/10/2012.
- T. Ylonen. The secure shell (ssh) protocol architecture, 2006. URL www.ietf.org/rfc/rfc4251.txt. Accessed on: 21/03/2013.
- Gert-Jan Zondervan, Jan Holtrop, Jaap Windt, and Tom van Terwisga. On the design and analysis of pre-swirl stators for single and twin screw ships. In *Second International Symposium on Marine Propulsors*, Hamburg, June 2011. Maritime Institute Netherlands (MARIN).

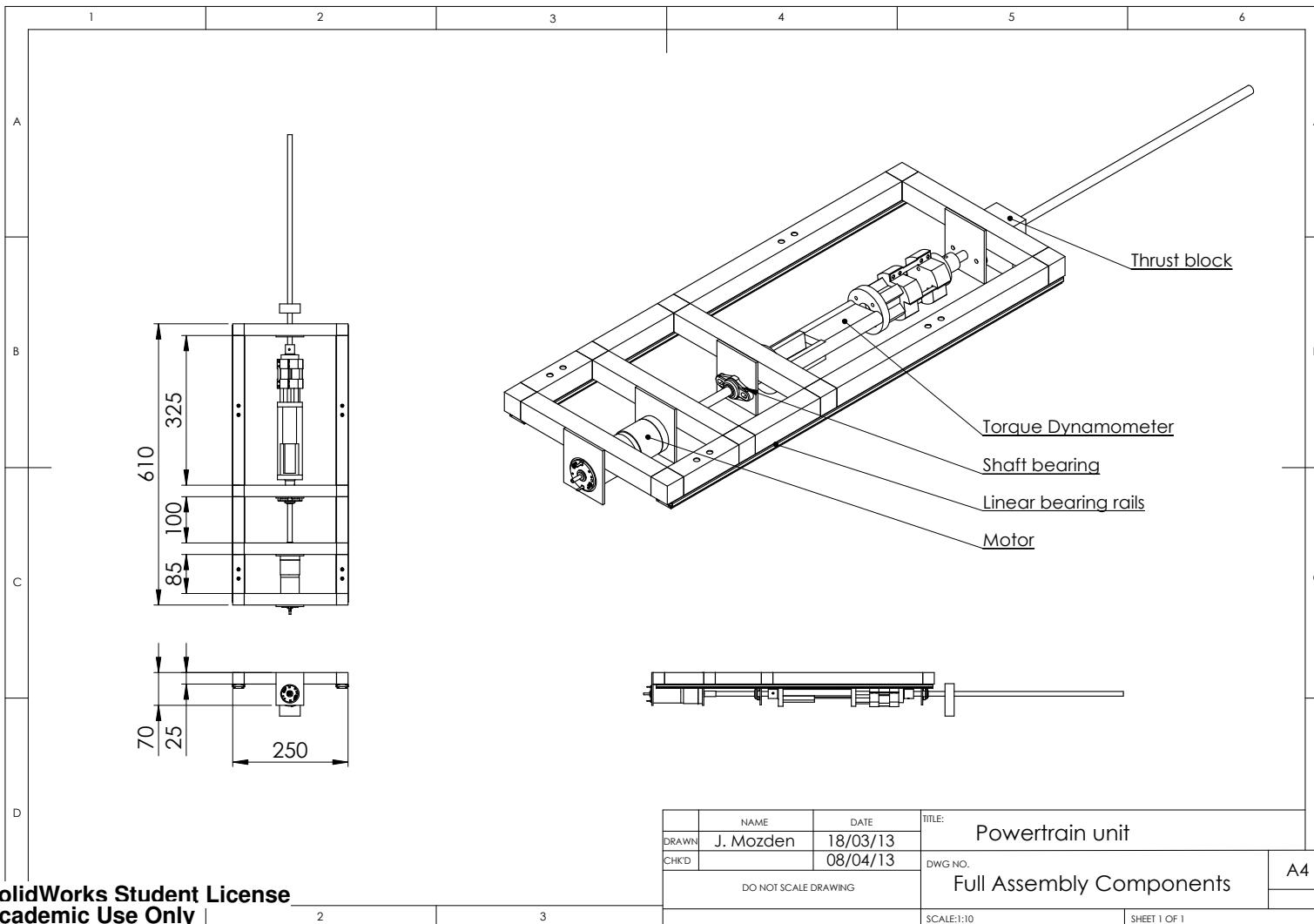
APPENDIX: SYSTEM OVERVIEW

192



APPENDIX: POWER TRAIN

193



SolidWorks Student License
Academic Use Only

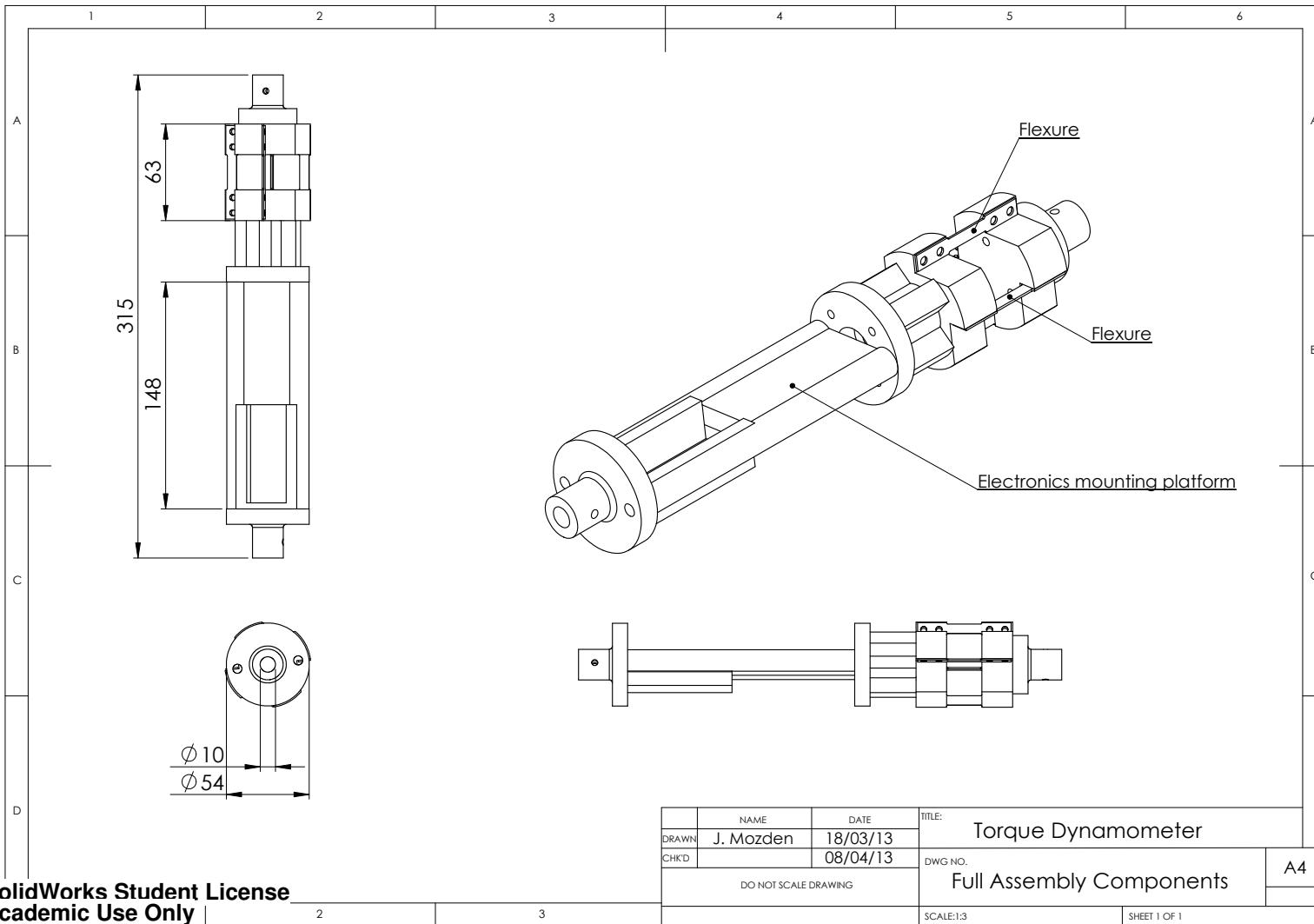
2

3

DRAWN	NAME	DATE	TITLE:	
CHK'D	J. Mozden	18/03/13	Powertrain unit	
DO NOT SCALE DRAWING		DWG NO.		A4
Full Assembly Components				
SCALE:1:10		SHEET 1 OF 1		

APPENDIX: TORQUE DYNAMOMETER

194



SolidWorks Student License
Academic Use Only