Programme Area: Marine

Project: PerAWAT

Title: Implementation Report: Frequency-Domain Model

Abstract:

This deliverable provides details of the model implementation, based on the previously defined methodology in the frequency domain to analyse arrays in regular and irregular waves. Initial results relating to the influence of several design variables (Fundamental Design Concept; array layout; control of the power conversion mechanism of each farm element, etc) are presented. Implications for the further model development are discussed.

Context:

The Performance Assessment of Wave and Tidal Array Systems (PerAWaT) project, launched in October 2009 with £8m of ETI investment. The project delivered validated, commercial software tools capable of significantly reducing the levels of uncertainty associated with predicting the energy yield of major wave and tidal stream energy arrays. It also produced information that will help reduce commercial risk of future large scale wave and tidal array developments.

Disclaimer:

The Energy Technologies Institute is making this document available to use under the Energy Technologies Institute Open Licence for Materials. Please refer to the Energy Technologies Institute website for the terms and conditions of this licence. The Information is licensed 'as is' and the Energy Technologies Institute excludes all representations, warranties, obligations and liabilities in relation to the Information to the maximum extent permitted by law. The Energy Technologies Institute is not liable for any errors or omissions in the Information and shall not be liable for any loss, injury or damage of any kind caused by its use. This exclusion of liability includes, but is not limited to, any direct, indirect, special, incidental, consequential, punitive, or exemplary damages in each case such as loss of revenue, data, anticipated profits, and lost business. The Energy Technologies Institute does not guarantee the continued supply of the Information. Notwithstanding any statement to the contrary contained on the face of this document, the Energy Technologies Institute confirms that the authors of the document have consented to its publication by the Energy Technologies Institute.

ETI MARINE PROGRAMME PROJECT
PerAWaT MA1004
WG1 WP1 D2
IMPLEMENTATION REPORT:
FREQUENCY-DOMAIN MODEL

Client	Energy Technologies Institute
Contact	Geraldine Newton-Cross
Document No	104327/BR/03
Issue	2.0
Classification	Not to be disclosed other than in line with the terms of the Technology Contract
Date	31 March 2011

Author: M Livingstone, E Mackay,
B Child, D McCowen, J Cruz

Checked by: J Cruz

Approved by: R I Rawlinson-Smith

IMPORTANT NOTICE AND DISCLAIMER

1. This report is intended for the use of the Client on whose instructions it has been prepared, and who has entered into a written agreement directly with Garrad Hassan and Partners Limited ("GH"). GH's liability to the Client is set out in that agreement. GH shall have no liability to third parties for any use whatsoever without the express written authority of GH. The report may only be reproduced and circulated in accordance with the Document Classification and associated conditions stipulated in this report, and may not be disclosed in any public offering memorandum without the express written consent of GH.

2. This report has been produced from information relating to dates and periods referred to in this report. The report does not imply that any information is not subject to change.

KEY TO DOCUMENT CLASSIFICATION

Strictly Confidential	:	Recipients only
Private and Confidential	:	For disclosure to individuals directly concerned within the recipient's organisation
Commercial in Confidence	:	Not to be disclosed outside the recipient's organisation
GH only	:	Not to be disclosed to non GH staff
Client's Discretion	:	Distribution at the discretion of the client subject to contractual agreement
Published	:	Available to the general public

REVISION HISTORY

Issue	Issue date	Summary
1.0	05.11.2010	D2 Implementation Report: Frequency-Domain Analysis - Original issue (electronic version only)
2.0	31.03.2011	D2 Implementation Report: Frequency-Domain Analysis - Revised issue, following client feedback (electronic version only)

CONTENTS

COVER PAGE

DISCLAIMER

REVISION HISTORY

LIST OF TABLES

LIST OF FIGURES

EXECUTIVE SUMMARY IX

1 INTRODUCTION 13

1.1 Scope of this document 13

1.2 Purpose of the frequency-domain model 14

1.3 Specific tasks associated with WG1 WP1 D2 14

1.4 WG1 WP1 D2 acceptance criteria 15

2 SOLVING THE EQUATION OF MOTION 16

2.1 Introduction to multi-body modelling 16

2.2 MBCore: The GH multi-body structural code 16

2.3 MBCore: co-ordinate systems 18

2.4 GHTools: vector and matrix Handling support 19

2.5 An introduction to multi-body co-ordinate system transformations 20

2.6 Developing WaveDyn 23

2.7 User input for a multi-body code 24

2.7.1 Parser functionality and the ".IN" file 24

2.7.2 The Parser algorithm: .IN file special text characters and parsing rules 26

2.7.3 Example .IN file 28

2.7.4 The MBCore input parameters 31

2.7.5 .IN file error checking 34

2.8 Frequency-domain analysis using a multi-body code 35

2.9 The time-domain numerical integration process 36

2.10 The role of WaveDyn 40

3 HYDRODYNAMICS MODULE 41

3.1 Key theoretical aspects 41

3.2 Data formats and co-ordinate systems 41

3.3 The multi-body hydrodynamics approach 49

3.4 Force calculation algorithms 51

3.4.1 Excitation force 51

3.4.2 Radiation force 52

3.4.3 Hydrostatic force 53
3.5 Time-domain performance for frequency analysis 54

4 WAVE ANALYSIS MODULE 56
4.1 Current inputs to the simulation 56
4.1.1 Standard models for wave spectra 56
4.1.2 Loading spectra to the wave module: the .SM file 63
4.1.3 Simulation of the sea surface from spectra: the .SEA file 64
4.2 Current status of wave climate parameterisation 67
4.2.1 Partitioning wave spectra 67
4.2.2 Fitting standard models to the frequency spectrum 69
4.2.3 Goodness-of-fit statistics 69
4.2.4 Initial results 69

5 MOORINGS MODULE 73
5.1 Overview of the approach 73
5.2 Mooring design and Dynamics (MDD) 74
5.2.1 Using MDD 75
5.2.2 Key findings - MDD 77
5.3 ROMEO 77
5.3.1 Functionality of ROMEO 77
5.3.2 Key findings - ROMEO 79
5.4 AQWA 79
5.4.1 AQWA-LINE 79
5.4.2 AQWA-NAUT 80
5.4.3 Key findings - AWQA 84
5.5 OrcaFlex 84
5.5.1 Overview 84
5.5.2 Theoretical aspects 84
5.5.3 Key findings - Orcaflex 87
5.6 Development strategy: moorings module 88
5.7 Implementation details 89
5.7.1 Algorithm 89
5.7.2 Coordinate system 91
5.7.3 Integration with existing software architecture 92

6 POWER TAKE-OFF (PTO) AND CONTROL MODULES 95
6.1 The PTO modelling approach 95
6.2 Initial formulation of the PTO module: applying forces to structural elements 96
6.3 Initial formulation of the control module: restriction to the applied PTO forces 97
6.4 Calculation Module Software Architecture 98

7 OPTIMISER MODULE 100
7.1 Functionality 100
7.1.1 Structure 100
7.1.2 Inputs and outputs 101

7.1.3 Extensions 105
7.2 Implementation 107
7.2.1 Current approaches 107
7.2.2 Extensions 108
7.3 Detailed algorithm description and case studies 108
7.3.1 Scenario 1A 109
7.3.2 Scenario 1B 112

8 CASE STUDIES 116
8.1 Definition of the case studies 116
8.2 Preliminary results – single FDCs 120
8.3 Preliminary results – arrays of FDCs 128
8.3.1 Scenario 1A 128
8.3.2 Scenario 1B 133

9 FINAL REMARKS AND NEXT STEPS 137
9.1 Final remarks: frequency-domain implementation report 137
9.2 Next steps 138

REFERENCES 141

NOMENCLATURE 153

KEY TO ACRONYMS 157

LIST OF TABLES

Table 2.1: .IN file special text characters 26

Table 2.2: Accepted .IN file parameter blocks 27

Table 2.3: Sub-blocks to be read from within the BODY block 27

Table 2.4: Sub-blocks to be read from within the MOORINGS block 27

Table 2.5: .IN file BODY and NODES block and mb (MBCore) sub-block parameters. mb sub-block parameters have been given for all body types currently supported by the code. 34

Table 3.1: Hydrodynamics Force Calculation Data .IN File Parameters 45

Table 4.1. Free and fixed parameters for families of unimodal spectra given by Equation 4.5. 58

Table 5.1: Overview of selected moorings modelling packages 74

Table 5.2: Example of a look-up table as used to represent an element of the damping matrix. 90

Table 6.1: PTO sub-block .IN file parameters for the initial, linear PTO model 96

Table 8.1: Definition of the physical properties of case study #1 (point-absorber) 118

Table 8.2: Definition of the physical properties of case study #2 (attenuator) 118

Table 8.3: Irregular wave results for case study#1 (point-absorber) and 121

Table 8.4: Set-ups for Array case studies 128

LIST OF FIGURES

Figure 2.1: A multi-body representation of a heaving float connected to the ground by a single degree of freedom hydraulic piston. 17

Figure 2.2: Global and body-fixed MBCore co-ordinate systems. Note that the body-fixed co-ordinate systems origins are located at the proximal node of each body. 19

Figure 2.3: Rotation of vectors in a fixed reference frame and conversion of vectors between a global (XYZ) and superimposed, rotated, relative frame (*x,y,z*) 21

Figure 2.4: The position vectors involved in the transformation of the position of a point A from a global (X,Y,Z) to a displaced, rotated relative local reference frame (*x,y,z*)........ 22

Figure 2.5: The modular WaveDyn architecture........ 24

Figure 2.6: Example .IN file for a simple point absorber system........ 30

Figure 2.7: The State Derivatives Calculation Process in WVStructure........ 39

Figure 2.8 Summary of GH WaveFarmer's envisaged structure under the PerAWaT programme..... 40

Figure 3.1: Determining the instantaneous body position relative to the hydrodynamic reference plane, `BodyFixedHyOrigin` 47

Figure 3.2: Rotation matrix combinations for determining the instantaneous body rotation relative to the Hydrodynamic reference frame, `BodyFixedHyOrigin` 48

Figure 3.3: Hydrodynamics Force Calculation Module Class Architecture and Key Functions 1

Figure 3.4: Mid-point bisection interpolation method. The first 6 search comparisons have been numbered........ 52

Figure 3.5: Radiation force and displacement time histories for a short section of simulation using MinConvInterval values of 0s and 1s. The simulation model is shown in Figure 2.1. The radiation force is not computed as precisely in the 1s simulation, but the effect on the device response is minimal........ 55

Figure 4.1. JONSWAP spectra for γ = 1-10 and Ochi and Gamma spectra for r = 2-11. Spectral densities have been normalised so that m_0 = 1. In each plot the Bretschneider spectrum is shown in bold........ 59

Figure 4.2. Comparison of Cos2s and wrapped normal directional distributions for σ_l = 20°. 61

Figure 4.3. Comparison of directional spreading with frequency for the swell and wind sea directional distributions proposed by Ewans (1998, 2001)........ 63

Figure 4.4. Upper panel: Time series of measured omnidirectional spectral density (log scale). Lower panel: Time series of peak frequencies of partitions identified by the partitioning algorithm, with colour scale denoting the H_s of each partition........**Error! Bookmark not defined.**

Figure 4.5. Left: occurrence of sea states in bins of width 0.5m by 0.5s. Right: Variability of spectral shapes about mean spectral shape, binned by H_s and T_e.. 70

Figure 4.6. Ratios of the mean deviance, Δ, for various spectral parameterisations.................... 72

Figure 5.1: MDD's main GUI form. 75

Figure 5.2: Typical mooring configuration (design view) in MDD. 76

Figure 5.3 The user interface window for the optional Coupled Cable Dynamics module in AQWA-NAUT. 83

Figure 5.4 Line element representation in Orcaflex, showing the element's frame of reference.......... 85

Figure 5.5: OrcaFlex screenshot showing both wire-frame and shaded 3-D views. 88

Figure 5.6: Graphical representation of the look-up table (Table 5.2) 91

Figure 5.7: The mooring line layout, illustrating the basis of the 92

Figure 6.1: PTO Force Calculation Module Class Architecture and Key Functions. 1

Figure 7.1: Structure of optimisation algorithm. 101

Figure 7.2: Regular grid definition sketch (axially symmetric devices). 104

Figure 7.3: Array layout for Set-up I. 109

Figure 7.4: Array layout for Set-up II. 111

Figure 7.5: Genetic Algorithm reproduction operators 114

Figure 8.1: Graphical representation of the mesh (top) and WAMIT .GDF file (bottom);................ 117

Figure 8.2: 3-D rendering of the case study #2 geometry (attenuator - top) and 3-D mesh of an array of point-absorbers (bottom) 119

Figure 8.3: A multi-body representation of a case study #1 (point-absorber – top).......................... 122

Figure 8.4: Heave RAO – case study #1 (point-absorber) 123

Figure 8.5: PTO force RAO – case study #1 (point-absorber).. 124

Figure 8.6: RCW – case study #1 (point-absorber)... 125

Figure 8.7: Pitch RAO – case study #2 (attenuator); PTO damping: 50MNms (per PTO mode)...... 126

Figure 8.8: RCW – case study #2 (attenuator) ; PTO damping: 50MNms (per PTO mode) 127

Figure 8.9: Power absorbed by each device in Set-up I. 129

Figure 8.10: Total power absorbed in Set-up I. 129

Figure 8.11: Power absorbed by each device in Set-up II... 130

Figure 8.12: Total power absorbed Set-up II... 130

Figure 8.13: Power generated from an array of 4 devices with different PTO damping settings for each device (Coarse step-size). 132

Figure 8.14: Power generated from an array of 4 devices with different PTO damping settings for each device (Fine step-size, not all values shown).... 133

Figure 8.15: Array layout resulting from optimisation. Axes in meters and waves approaching from the left-hand side. 136

EXECUTIVE SUMMARY

The present report (WG1 WP1 D2) describes the current implementation status of the wave energy converter (WEC) numerical modelling software being developed by Garrad Hassan and Partners Ltd. (GH) under the PerAWaT project. The report builds on previous deliverables, namely the Methodology Report (WG1 WP1 D1B), and details the core aspects and key functionalities of the developed code, with emphasis to performance variables related to frequency-domain simulations.

The report is organised in nine sections which give a detailed overview of:

- The scope of the documents and its key objectives – Section 1;
- The nature of the developed code, the core aspects of a multi-body dynamic solver and key implementation details (e.g. coordinate systems) – Section 2;
- The hydrodynamics module and its functionality – Section 3;
- The wave analysis module and its functionality – Section 4;
- The moorings module and its functionality – Section 5;
- The power take-off (PTO) and control modules and their functionality – Section 6;
- The optimiser module and its functionality – Section 7;
- Case studies that outline initial results – Section 8;
- The next steps in terms of the implementation – Section 9.

The report begins by presenting background information regarding the GH multi-body code (Bladed) used as the kernel unit to describe the WECs. Under PerAWaT GH has developed a series of software routines that plug into the Bladed code, allowing the definition of a wide range of WECs. This new package (code named WaveDyn) can be considered a loads calculation tool which allows the reduction of the uncertainty in the estimation of the relevant (single WEC) performance variables and the definition of the parameterisation of the WEC behaviour in order to use this in the optimiser module (code named WaveFarmer).

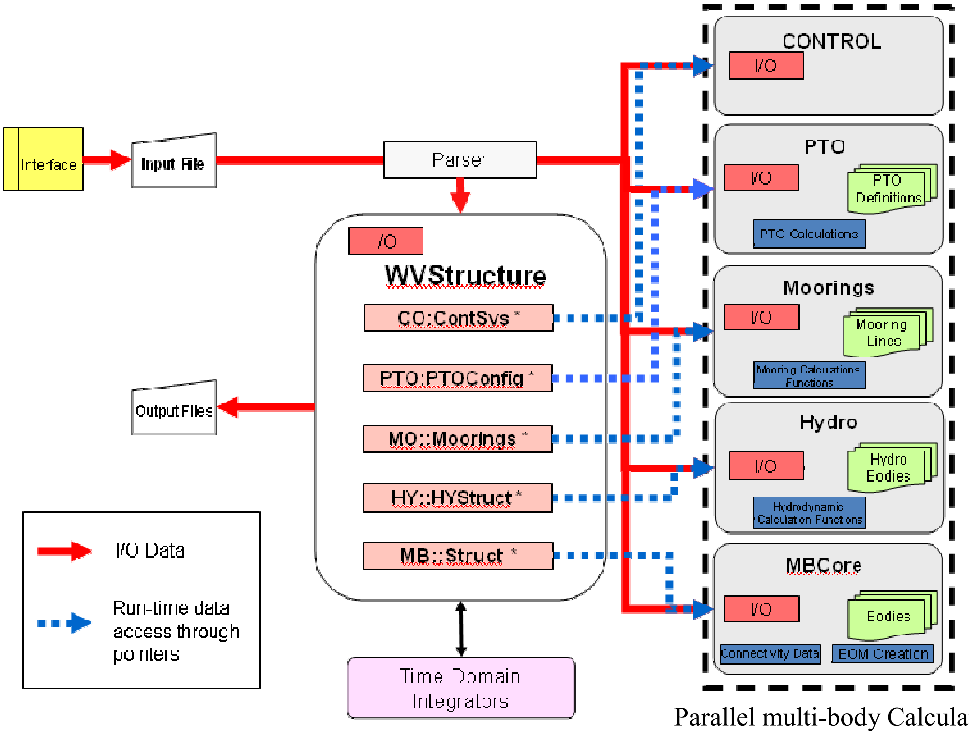
The WaveDyn multi-body modelling approach covers the scope of the ‘FD’, ‘TD’, ‘Basic Controller’, ‘External forces code’ and ‘Wave data loader’ modules originally envisaged for the WaveFarmer code structure in the Methodology Report (WG1 WP1 D1B, Section 1). The ‘Basic Controller’ and ‘External forces code’ blocks now operate as the series of parallel multi-body Calculation Modules shown in the figure on the next page, whilst the FD and TD blocks are represented in the higher level WVStructure co-ordinating code. The wave data loader is one of the I/O routines in the Hydrodynamics Calculation Module. It is envisaged that the Calculation Modules may be developed to give the user the option of replacing the core functionality of each with a DLL interface, however this interface will itself be part of the Calculation Module from an overall software architecture perspective. WaveFarmer as a software package now refers purely to the array design code (and includes the Optimiser and Mapping blocks shown in WG1 WP1 D1B). The WaveFarmer Optimiser has the ability to prescribe and analyse the results from WaveDyn simulations as described in Section 7.

Following WG1 WP1 D1B, a multi-body dynamic solver has been created, providing a means of mathematically describing the structural forces within the WEC structure. All of the non-structural forces applied to the WEC, resulting from the hydrodynamics, PTO or moorings must be calculated separately in code modules parallel to the core code (MBCore). If the flexible nature of the multi-body structural approach, which allows a range of WECs to be defined and simulated, is to be maintained

then a multi-body theme must be propagated throughout these applied force calculations, with the hydrodynamic and mooring forces being calculated on a body-by-body basis and PTO calculations being associated with a particular joint in the multi-body structure.

In this way the complete software formulation for the WEC modelling code may be envisaged as a collection of parallel multi-body Calculation Modules, including MBCore (see the figure in the next page):

- The geometrical/structural definition formulated using the MBCore code.
- A hydrodynamic definition containing the hydrodynamic information and force calculation functions for the wave activated bodies in the system (any wave activated structural body having a parallel hydrodynamic body containing its particular hydrodynamic properties).
- A moorings system built up from multiple mooring line 'bodies', each containing information on the line properties and attachment location on the geometrical structure.
- A PTO system containing information on the PTO properties and force calculation functions for any joint contributing to the energy capture of the device. Each PTO 'body' describes the PTO mechanism operating on a parallel structural joint.
- A control system containing the control algorithm used to control / operate the PTO.



Parallel multi-body Calculation Modules.

The modular WaveDyn architecture

Each of these modules (apart from the MB core) was specifically developed (and will continue to be developed) under PerAWaT. The module functionalities are discussed in greater detail in the corresponding sections of this report.

The applied force Calculation Modules (those containing the hydrodynamic, PTO, Control and moorings multi-body systems) may be queried with body kinematic data at any time to return a set of force vectors to be applied to the structure at specific locations. The distinct functionality of the Calculation Modules means that each may be developed independently, without any direct link to the others, as knowledge increases or demand for particular functionality in the relevant field expands. This flexibility is core to ensure that advanced functionalities and / or new techniques are incorporated at later stages, should the result(s) of the verification and validation exercises lead to the conclusion that it is necessary to adjust the calculation methodology.

The only permitted inter-module communications are between the control and PTO systems. All other simulation data transfers must pass through a higher, coordinating level of code. This code manages the transfer of body kinematic information from the MBCore Calculation Module to the applied force Calculation Modules and conversely, the application of the force data returned by the force calculation Calculation Modules to the MBCore structure. This coordinating code level will also handle communication with the numerical integrator in the time-domain. The complete arrangement is illustrated in the figure above, where the coordinating code level is marked as ‘WVStructure’. WVStructure stores as little information as possible about the WEC itself: the parameters defining the device are read and stored directly in the calculation modules which are simply located during simulation through pointers in the co-ordinating block (marked as the blue dashed lines in the figure).

The frequency-domain model allows a first characterisation of a WEC and / or an array of WECs. The main limitation is associated with the description of the external forces (e.g. mooring force; PTO force), which must remain linear (or be linearised) for the model to be valid. At the expense of a potentially less accurate description of some of these forces, a less computationally intensive model is created. This may allow a first (quick) assessment of the performance of a WEC or an array of WECs, or an optimisation exercise with more variables than the time-domain equivalent (which may be necessary if the time-domain effort is computationally prohibitive).

Although these limitations imply constraints in the majority of the modules listed previously, a fundamental implementation decision led to the definition and creation of a suitable software structure, common to the frequency and time-domain models, and also suitable for coupling with the spectral-domain models developed under WG1 WP2. Synergies with the methodology developed in WG1 WP2 will also allow a spectral-domain representation of the fundamental device concepts (FDCs) via the core modules presented in this report. It is envisaged that such spectral-domain representation will allow the optimisation of large (100+ WECs) arrays of FDCs.

As a direct consequence of sharing the core structure there are considerable synergies between this report (WG1 WP1 D2) and its time-domain equivalent (WG1 WP1 D3 Implementation Report: Time-Domain Model). This will result in the use of some sections of this report in WG1 WP1 D3, with the suitable adaptations and updates. It is anticipated that the major differences will be associated with the development of the PTO and control modules, as in the time-domain the description of the external forces can be nonlinear (and if necessary discontinuous). The limitations of a frequency-domain model and the need to prioritise the developments – e.g. the core structure needed to be in place for any module to work – have meant that the PTO and control modules are those that require the most significant developments when implementing the time-domain approach for WG1 WP1 D3. This is inline with the expectations and the developments to date and provides sufficient functionality to allow implementation of such functionalities to start immediately.

The initial results presented in this report show the baseline functionality of the code under the limitations of a frequency-domain approach. In Section 8 the following aspects are covered in the case studies:

- The ability to define different FDCs with different characteristics;
- The influence of different wave input conditions in the response of the WEC;
- The influence of the array layout in the power absorption characteristics;
- The influence of the control of the PTO of each array element in the power absorption characteristics (by treating the array as a power plant).

Even though the results in Section 8 are preliminary, an effort was made so that these are aligned with the verification scenarios listed in WG0 D1, in particular the array of four point-absorbers. It is expected that this will ensure that comparisons related to initial results from different project partners (GH, UoOx and QUB) can be more easily made. As listed in WG1 WP1 D1B, these cases should not be confused with a set of representative scenarios for which results will be presented in WG1 WP1 D4 and D15 (versions A and B). Finally, the implementation case studies allowed the test of the functionality of each module, and the definition of the immediate next steps (Section 9.2).

1 INTRODUCTION

1.1 Scope of this document

This document describes the implementation of the core modules of the wave energy converter (WEC) and wave farm modelling package in the frequency-domain. The primary objective of the numerical models developed under WG1 is to predict the performance of an array of WECs under different input conditions, multiple design constraints and for several objective functions. The core modules described in this report allow the design and optimisation of arrays of fundamental device concepts (FDCs) previously identified in WG0 D1, within the limitations of the frequency-domain approach.

It should be emphasised that work on the core modules will occur throughout the course of the PerAWaT project. This is clear in the multiple releases and validation stages identified by several key project deliverables. As a consequence the results presented in the implementation reports and those derived from all versions except Beta 2 should be seen as indicative of the model functionalities and not as verified or validated outputs.

The report begins by presenting background information regarding the GH multi-body code (MBCore), which is used to mathematically describe the geometry and structural forces within WEC designs, in a manner that is sufficiently flexible to cope with physically dissimilar FDCs (Section 2). Under PerAWaT GH has developed a series of Calculation Module software routines that operate in parallel with MBCore: these also adopt a multi-body structure and deal with the calculation of hydrodynamic, power take-off (PTO) and mooring forces. Higher level code has also been developed to manage the transfer of information between calculation modules, the input and output functionality and the equation of motion numerical integration process. plug into the Bladed code, allowing the definition of a wide range of WECs. The new software package (named WaveDyn) can be considered a loads calculation tool and is designed to simulate the performance of a single WEC that may be operating in an array environment. Parameterisation of the WEC model and its individual environmental conditions allows WaveDyn to be called by an optimiser module (WaveFarmer) to evaluate the performance of a WEC in any given array position. The role of WaveDyn is described in detail in Section 2.10.

The generic layout of the new software routines and their interaction with the MBCore code is introduced in Section 2, and further detailed in the subsequent sections. In particular the following modules are described in the sections mentioned below:

- Hydrodynamics – Section 3;
- Wave climate – Section 4;
- Moorings – Section 5;
- Power take-off (PTO)and control – Section 6;
- Optimiser – Section 7;

The report concludes with two further sections. Firstly, in Section 8 the functionalities of the developed package are illustrated by presenting results for a series of cases studies. These refer to simulations where single FDCs and arrays of FDCs are analysed and their response to representative input conditions is quantified. Finally, in Section 9 a discussion regarding the next key steps is initiated by addressing the implications of the current implementation status in further numerical model development activities.

1.2 Purpose of the frequency-domain model

The key objectives of the frequency-domain model described in this report are to allow:

1. the modelling of arrays of various FDCs under a wide range of input conditions (within the limitations of a frequency-domain approach);
2. the definition and creation of a suitable software structure, common to the frequency and time-domain models, and also suitable for coupling with the spectral-domain models developed under WG1 WP2.

The frequency-domain model allows a first characterisation of a WEC and / or an array of WECs. The main limitation is associated with the description of the external forces (e.g. mooring force; PTO force), and the structural geometry (dictating the kinematic in response to these), which must remain linear (or be linearised) for the model to be valid. At the expense of a potentially less accurate description of some of these forces, a less computationally intensive model is created. This may allow a first (quick) assessment of the performance of a WEC or an array of WECs, or an optimisation exercise with more variables than the time-domain equivalent (which may be necessary if the time-domain effort is computationally prohibitive). Furthermore, some of the fundamental hydrodynamic properties are frequency-domain variables, thus such models may be considered an intermediate step towards a potentially more precise (and more computationally intensive) time-domain representation of the WEC's systems. Although the above mentioned limitations imply constraints in the majority of the modules listed in Section 1.1, the core code and functionalities will be shared (e.g. the optimiser module will have the same key functionalities, and will be able to load both the frequency and time-domain representations of the WECs). Synergies with the methodology developed in WG1 WP2 will also allow a spectral-domain representation of the FDCs via the core modules presented in this report. It is envisaged that such spectral-domain representation will allow the optimisation of large (100+ WECs) arrays of FDCs.

As a direct consequence of objective 2 (listed above) there are considerable synergies between this report (WG1 WP1 D2) and its time-domain equivalent (WG1 WP1 D3 Implementation Report: Time-Domain Model). This will result in the use of some sections of this report in WG1 WP1 D3, with the suitable adaptations and updates. It is anticipated that the major differences will be associated with the development of the PTO and control modules, as in the time-domain the description of the external forces can be nonlinear (and if necessary discontinuous).

However both implementation reports (frequency and time-domain) should not be confused with a theory or a user manual. Although there are more resemblances with regard to the draft theory manuals (WG1 WP1 D4a and D15a), it should be emphasised that an implementation report, although describing the theoretical principles behind the developed code and its key functionalities, is a document more prone to updates as a result of the ongoing verification and validation effort.

1.3 Specific tasks associated with WG1 WP1 D2

The tasks addressed in this report are:

1. Development of a frequency-domain numerical model aligned with the methodology and objectives defined in WG1 WP1 D1b.

2. Definition of a common software structure to be shared with the time-domain and spectral versions of the wave farm modelling package, including all the associated modules.

3. Compilation of a first series of representative results (limited to frequency-domain simulations at this stage), to illustrate the core functionalities and typical outputs (allowing end-user feedback to be incorporated in the software releases).

1.4 WG1 WP1 D2 acceptance criteria

The acceptance criteria as listed in the Technology Contract and the sections of this report that demonstrate that they have bee met are:

1. The report contains details of the model implementation of the previously defined methodology (WG1 WP1 D1b) in the frequency-domain to analyse arrays of FDCs in regular and irregular waves (such that it could be replicated by a third-party) - Sections 2 through 7.
2. Initial results related to the influence of several design variables (FDC, wave climate, array layout, control of the PTO for each array element, …) are presented - Section 8.
3. Implication for the further model development activities are discussed - Section 9.

2 SOLVING THE EQUATION OF MOTION

2.1 Introduction to multi-body modelling

The imbalance between the vast range of potential numerical modelling applications in many industry sectors and the large amounts of development time required to build bespoke numerical simulation models has led to significant academic interest over the past decade in the development of so-called 'multi-body' modelling tools. The multi-body simulation concept allows a very wide range of engineering systems to be modelled, using a single simulation tool, as a generalised collection of interacting components, or bodies. Each body within a multi-body system contains only information about its own physical properties (and potentially also the nature of external forces applied directly to it) and serves as a building block for the generation of a complete multi-body structure using a generalised technique. A record of the connectivity between adjacent bodies allows forces or prescribed deflections, or indeed any change that affects the system states – the minimum set of variables that wholly describe the system at a given instant in time – to be applied at one body and transmitted through the structure (in a structural sense, the states are the displacements and velocities of all the degrees-of-freedom, DOFs).

In the wave energy industry, emergent WEC designs are yet to converge on a common geometrical or internal system format. It is unclear whether such convergence will occur at all, with multiple very different FDCs now approaching a late pre-commercial development state. With this in mind, any device performance and loading simulation tool designed to service the industry as a whole must operate to some extent in a multi-body format.

2.2 MBCore: The GH multi-body structural code

Developments in GH's wind turbine performance and loading software, *Bladed*, have in recent years stimulated the creation of an in-house structural multi-body formulation. The code, "MBCore" has been through an extensive internal verification and validation procedure and has been developed as a stand-alone library that is discrete from the wind turbine specific *Bladed* modelling functionality, making it ideally suited to an expanded role as the geometrical and internal structural forces formulation in a new WEC design tool. A decision regarding the commercial name of this new tool has been taken, and the tool will be named WaveDyn (from which beta versions will be supplied to the PerAWaT consortium).

The MBCore formulation accommodates both rigid and flexible bodies (although the assumption is that flexibility is limited to small deflections so that a linear modal approach may be adopted), as well as a series of linking bodies and joints designed to introduce specified DOFs in a controlled way. Complete multi-body structures must be built up from a reference, stationary 'Ground' element in a tree-structure (a structure in which just a single path exists from any body to the Ground); closed loops of bodies are not presently permitted (note that in the current implementation, mooring lines forces are applied as external loads, the lines themselves are not modelled in MBCore – see Section 5), however the facility to solve these through iteration may form part of future development work. Note that although all GH Multi-body structures must be built up from a Ground body, a free six DOF joint may immediately succeed this, effectively allowing the subsequent bodies in the tree to be floating.

Connectivity between bodies is defined through the specification of a number of 'nodes'. Every body is connected to at least one node, its 'proximal' node, which is defined as that closest to the Ground body, and all bodies in the middle of a branch also have one or more 'distal' nodes allowing additional bodies to be connected around them. Many bodies (joints and the Rigid Body point masses) occupy no physical space, whilst others, the Rigid Links, are designed specifically for this purpose. The nodes are positioned between bodies and connect them so that a distal node of one body is the proximal node of the body immediately higher up in the tree. The nodes have been labelled a simple heaving buoy multi-body structure presented in Figure 2.1 to illustrate this concept.

The complete set of bodies and the nodes list are used to build a higher level structure, through a Lagrangian formulation, that mathematically represents the system as a whole.

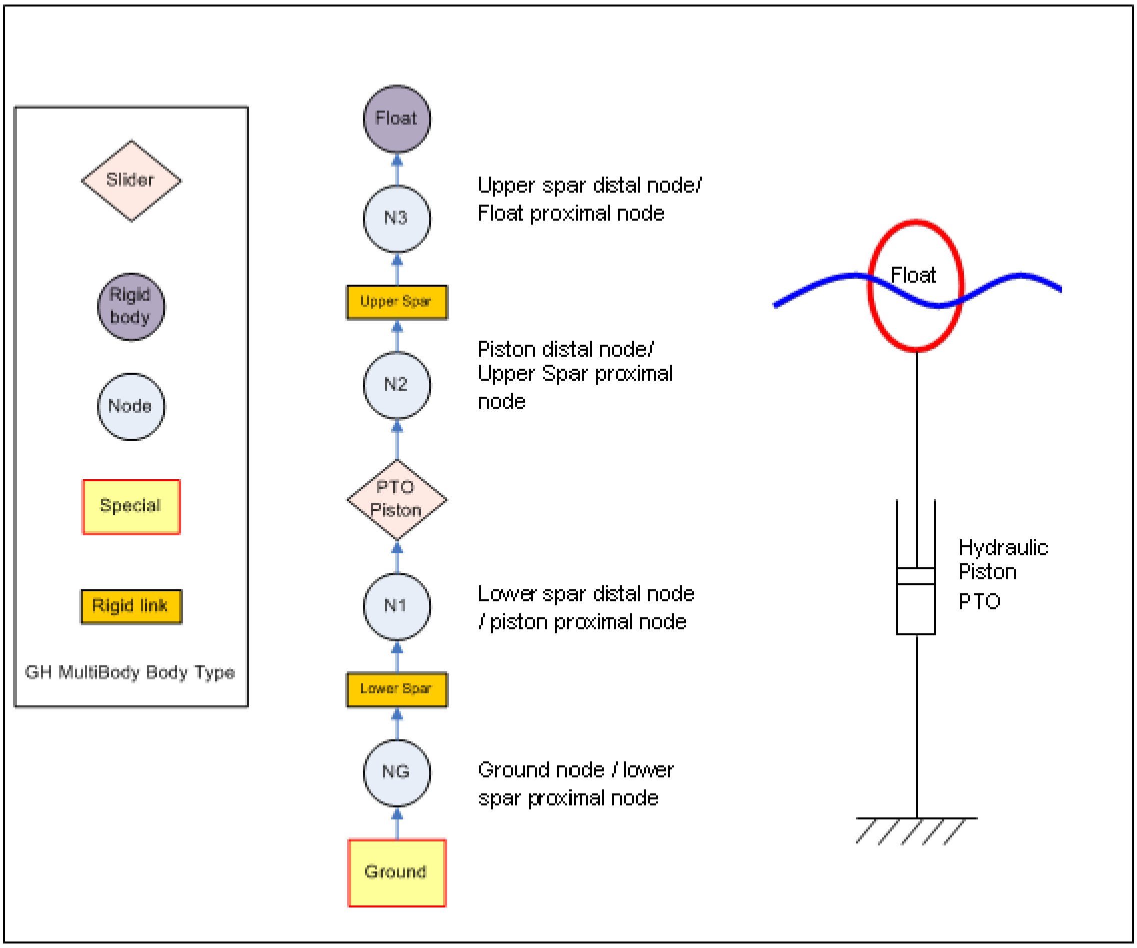

Slider
Rigid body
Node
Special
Rigid link
GH MultiBody Body Type
Float
N3
Upper Spar
N2
PTO Piston
N1
Lower Spar
NG
Ground
Upper spar distal node/ Float proximal node
Piston distal node/ Upper Spar proximal node
Lower spar distal node / piston proximal node
Ground node / lower spar proximal node
Float
Hydraulic Piston
PTO

Figure 2.1: A multi-body representation of a heaving float connected to the ground by a single degree of freedom hydraulic piston.

Multi-body modelling lends itself readily to an object orientated programming structure (in fact the two concepts could be considered synonymous) and each body within the MBCore code is represented computationally as an instance of a defined body class which stipulates which physical attributes the body may have. A specifically designed connectivity map class is used to store the node information which is passed in conjunction with the instances of the body classes to the higher level complete system object. The multi-body system object is manipulated at run-time, allowing forces and prescribed motions calculated by other elements of the code to be applied to specified nodes or freedoms in the system.

The equation of motion itself must be solved by an external numerical integrator, not part of the MBCore code (which is a purely structural representation). However, the multi-body system class incorporates a comprehensive set of functions that allow the integrator and calculation modules to query the structure for the system states (or indeed the position or velocity of any individual body), or request the completion of a state derivative calculation based on a set of integrator supplied state values. In this latter case, the integrator may dictate the nature of the calculation so that that the states supplied may be taken as permanently recorded updates to the structural system state, or treated as intermediate values to be temporarily used and then discarded (as is necessary for many integrator types, such as a Runge-Kutta formulation, which calculate a complete time-step result based on multiple intermediary calculations).

The object orientated MBCore code has been developed in C++, which is open to development using a wide range of commercially available and open source compilers for a range of operating system platforms and is well documented and standardised (note however that many non-standard libraries have been created by the software development community that may not embrace the full compiler and platform flexibility of the standard code). As a lower level code, C++ represents a significant speed advantage over higher level alternatives such as MATLAB (the original choice), but places an additional burden on the developer to write efficient code with rigorous memory management.

2.3 MBCore: co-ordinate systems

The equation of motion for the system, constructed by MBCore, is expressed in what will here (and throughout the development of WaveDyn) be referred to as the global co-ordinate system; a right-hand inertial frame of reference with its origin defined relative to the position of the Ground body (see the ‘location’ parameter in Table 2.5) and the *z*-axis being positive upwards. The direction of the positive global *x*-axis is aligned with south for the purpose of defining the incident wave directions. Each rigid body in the system has its own ‘body-fixed frame of reference’ which is positioned at the body’s proximal node and is used to track the body position as it moves in global space. The body-fixed frame of reference of the ground body has the same orientation as the global co-ordinate system and an origin at the ground body proximal node. The orientation of the body-fixed co-ordinate systems for bodies further up the tree-structure are inherited from the bodies below. Bodies with distal nodes allow the user to specify a relative rotation of their distal node to the proximal node; the distal node is the proximal node of the proceeding body in the tree, so specifying a non-zero rotation here rotates the body-fixed co-ordinate system of the proceeding body. This inheritance system is illustrated in Figure 2.2 below.

The multi-body system may be queried for a position vector describing the global position of a specified body and the instantaneous rotation of its body-fixed frame of reference relative to the global frame at any time.

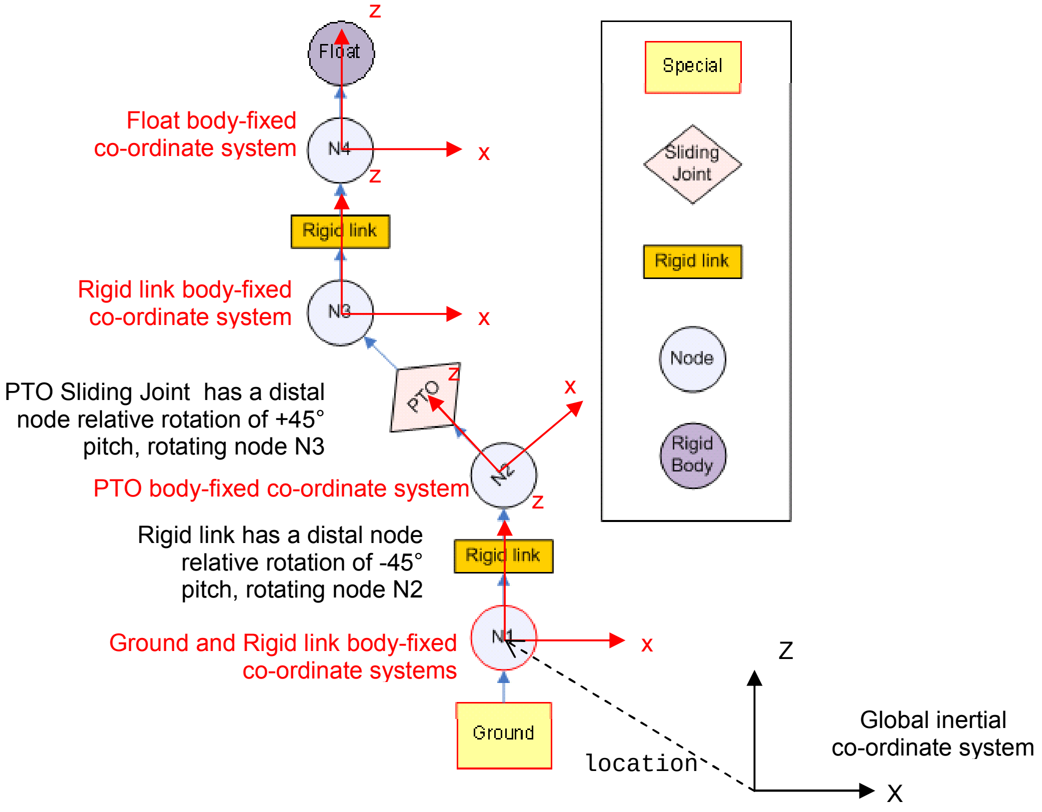


Figure 2.2: Global and body-fixed MBCore co-ordinate systems. Note that the body-fixed co-ordinate systems origins are located at the proximal node of each body.

2.4 GHTools: vector and matrix Handling support

The mathematical calculations within both MBCore and the other parts of the WaveDyn code are most efficiently and clearly described in a matrix/vector format. GH have developed C++ support for vector and matrix handling in a separate library, GHTools, which is based on the LAPACK/BLAS high performance Fortran mathematical libraries (LAPACK, 2010). GHTools provides matrix and vector type classes designed to handle algebraic operations (typically provided as operator overloadings), as well as an 'EulPar' class designed specifically to cope with the handling of Euler parameter scalar-vector quartonians and conversions to/from three parameter rotation matrices describing successive roll, pitch and yaw rotations about the global axes.

For clarity, the Euler parameter quartonian, $\underline{q}$ used within GHTools is defined for a rotation of angle α about a Cartesian 3x1 unit vector $\underline{n}$ as:

$$\underline{q} = \left(q_0, q_1, q_2, q_3\right) \qquad \text{where:} \quad q_0 = \cos\left(\alpha/2\right); \quad q_i = n_i \sin\left(\alpha/2\right) \qquad [2.1]$$

$$\text{So that: } q_0^{\ 2} + q_1^{\ 2} + q_2^{\ 2} + q_3^{\ 2} = 1$$

The same rotation may alternatively be represented by successive rotations of a roll angle ϕ about the global x-axis, a pitch of angle χ about the global y-axis and a yaw of angle θ about the global z-axis, which may be encapsulated in rotation matrix form as:

$$\underline{R} = R_\theta R\chi R_\phi = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\chi & 0 & \sin\chi \\ 0 & 1 & 0 \\ -\sin\chi & 0 & \cos\chi \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix}$$

$$= \begin{pmatrix} \cos\theta\cos\chi & \cos\theta\sin\chi\sin\phi - \sin\theta\cos\phi & \cos\theta\sin\chi\cos\phi + \sin\theta\sin\phi \\ \sin\theta\cos\chi & \sin\theta\sin\chi\sin\phi + \cos\theta\cos\phi & \sin\theta\sin\chi\cos\phi - \cos\theta\sin\phi \\ -\sin\chi & \cos\chi\sin\phi & \cos\chi\cos\phi \end{pmatrix}$$

[2.2]

Note the overall rotation matrix, $\underline{R}$, is nonlinear in terms of the roll, pitch and yaw angles and also that the matrix is orthogonal, so that its inverse is simply its transpose.

2.5 An introduction to multi-body co-ordinate system transformations

It can be confusing to visualise the correct way to apply a rotation matrix as rotations can be thought of in terms of a rotation of a position vector of a point in space to a new position in the same reference frame, or as a transformation in the co-ordinates of a fixed point, from a rotated frame of reference back to the global frame of reference. This is illustrated by the examples given below.

Consider a rotated relative frame of reference rolled through a 90deg angle, but with its origin located in the same place as the underlying global, inertial frame of reference, as shown in Figure 2.3. Point *A* has a position vector d_{gA}= (2,1,1) (*X*,*Y*,*Z*) in global co-ordinates; the $_g$ subscript indicates that the vector values are relative to the global reference frame. Ignoring the local frame for now, in order to rotate point *A* on to another point, *B*, which is has global position vector d_{gB}= (2,-1,1), then it is necessary to multiply the position vector of *A* by the rotation matrix for a 90deg roll:

$$d_{gB} = \underline{R}_\phi d_{gA} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 90 & -\sin 90 \\ 0 & \sin 90 & \cos 90 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix} \qquad [2.3]$$

The same position vector could now be rotated by 90deg about the global y-axis, so that point *A* comes to rest in position *C*, which has global co-ordinates (1,-1,-2) by pre-multiplying the previous result by the pitch rotation matrix:

$$d_{gC} = \underline{R}_\chi d_{gB} = \begin{bmatrix} \cos 90 & 0 & \sin 90 \\ 0 & 1 & 0 \\ -\sin 90 & 0 & \cos 90 \end{bmatrix} \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ -2 \end{bmatrix} \qquad [2.4]$$

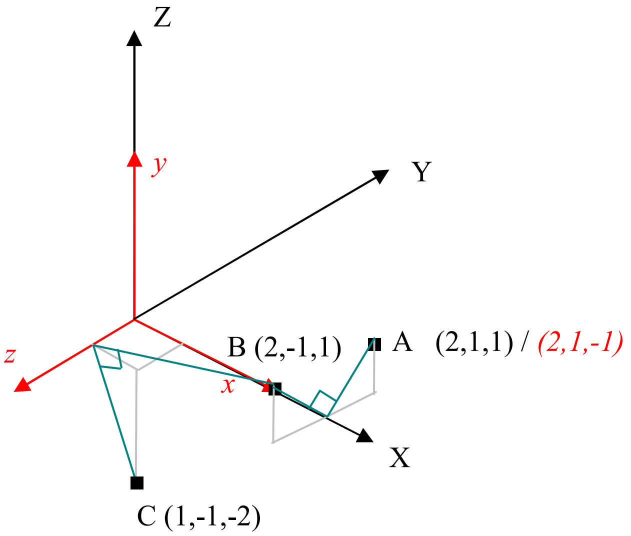


Figure 2.3: Rotation of vectors in a fixed reference frame and conversion of vectors between a global (XYZ) and superimposed, rotated, relative frame (*x,y,z*)

Naturally it is possible to complete the two rotations in a single step by multiplying the position vector of A by the combined rotation matrix $\underline{R} = \underline{R}_\chi \underline{R}_\phi$:

$$d_{gC} = \underline{R}_\chi \underline{R}_\phi d_{gA} \qquad [2.5]$$

In general, a vector can be rotated within a single reference frame by pre-multiplying by the rotation matrix $\underline{R}$ composed from the required roll, pitch and yaw about the stationary co-ordinate system axes.

Now consider the rotated local reference frame shown in red in Figure 2.3. Point A has position d_{bA} = (2,1,-1) in this relative co-ordinate system (notice the subscript is now $_b$). If the objective is to convert from global values, d_{gA} to the relative local frame position, then the local reference frame position vector must be multiplied by the *inverse* (the transpose) of the 90deg roll rotation matrix (recalling that the local frame is orientated at a 90deg roll offset from the global frame):

$$d_{bA} = \underline{R}_\phi^{-1} d_{gA} = \underline{R}_\phi^{T} d_{gA} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix} \qquad [2.6]$$

In general it is possible to say that, to convert a vector from the global reference frame to a rotated, local relative reference frame with the same origin, it is necessary to pre-multiply the vector by the transpose of the rotation matrix comprised of the rotations that a set of axes starting exactly on top of the global axes would have to go though about the global axes to reach the local, relative frame.

Conversely, to convert a vector given in the local, relative frame into global values, it is necessary to pre-multiply by the rotation matrix describing the relative orientation of the local frame:

$$d_b = \underline{R}^{-1} d_g = \underline{R}^T d_g \qquad [2.7]$$

$$d_g = \underline{R} d_b \qquad [2.8]$$

Typically the local reference frame (most likely a body-fixed reference frame) is unlikely to have its origin on top of the global origin, but will rather have a spatial offset described in global co-ordinates by an additional position vector d_{gBF}, as shown in Figure 2.4. This offset can be used to expand the equations 2.7 and 2.8 to a more general form for any relative (to global) reference frame so that:

$$d_b = \underline{R}^T \left(d_g - d_{gBF} \right) = \underline{R}^T d_g - \underline{R}^T d_{gBF} \qquad [2.9]$$

$$d_g = \underline{R} d_b + d_{gBF} \qquad [2.10]$$

Note that it is necessary to apply such a spatial offset only in the case of position vectors (denoted here by the letter d), as the difference in position is lost in the differentiation to velocity vectors (it is assumed that the case of a relative reference frame with a *velocity* offset is not of use in the calculations presented in this report). Similarly, force vectors may be transformed from a relative co-ordinate system with a position offset to the global one by simply using the relationships described by equations 2.7 and 2.8.

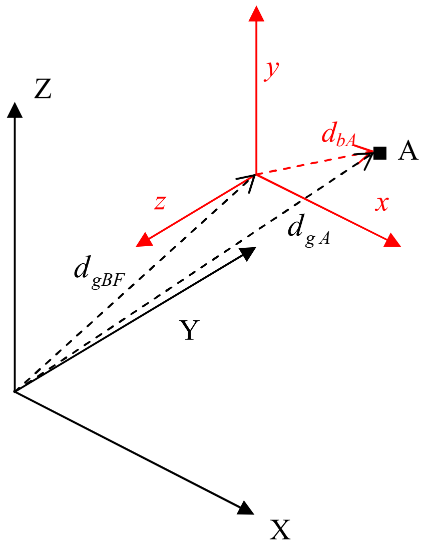


Figure 2.4: The position vectors involved in the transformation of the position of a point A from a global (X,Y,Z) to a displaced, rotated relative local reference frame (*x,y,z*).

2.6 Developing WaveDyn

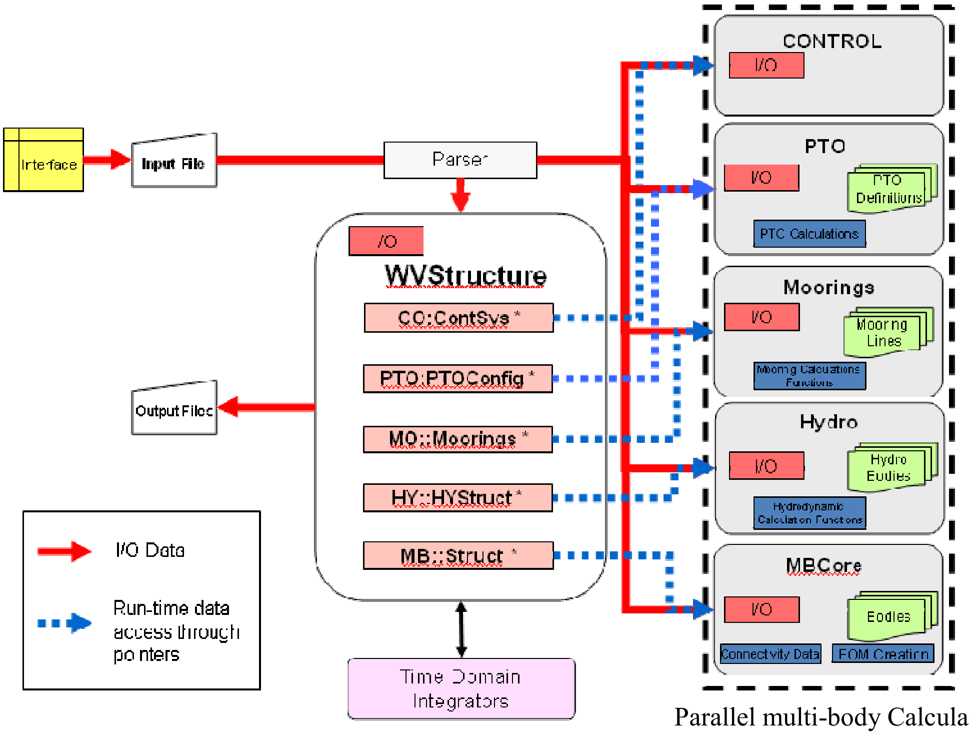
In a wave energy context, the MBCore code provides a means of mathematically describing the structural forces within the WEC structure. In the initial, beta releases of the code, WaveDyn will make used of the rigid-body mechanics provided by MBCore (a complete list of the multi-body components currently supported is provided in Table 2.5).

All of the non-structural forces applied to the WEC, resulting from the hydrodynamics, PTO or moorings must be calculated separately in code modules parallel to MBCore. If the flexible nature of the multi-body structural approach is to be maintained then a multi-body theme must be propagated throughout these applied force calculations, with the hydrodynamic and mooring forces being calculated on a body-by-body basis and PTO calculations being associated with a particular joint in the multi-body structure. In this way the complete software formulation for the WEC modelling code may be envisaged as a collection of parallel multi-body Calculation Modules, including MBCore:

- The geometrical/structural definition formulated using the MBCore code
- A hydrodynamic definition containing the hydrodynamic information and force calculation functions for the wave activated bodies in the system (any wave activated structural body having a parallel hydrodynamic body containing its particular hydrodynamic properties).
- A PTO system containing information on the PTO properties and force calculation functions for any joint contributing to the energy capture of the device. Each PTO 'body' describes the PTO mechanism operating on a parallel structural joint.
- A control system containing the control algorithm used to operate the PTO.
- A moorings system built up from multiple mooring line 'bodies', each containing information on the line properties and attachment location on the geometrical structure.

Each of these modules (apart from the MBCore) were specifically developed (and continue to be developed) under PerAWaT and are discussed in greater detail in subsequent sections of this report. The applied force Calculation Modules (those containing the hydrodynamic, PTO, Control and moorings multi-body systems) may be queried with body kinematic data at any time to return a set of force vectors to be applied to the structure at specific locations. The distinct functionality of the Calculation Modules means that each may be developed independently, without any direct link to the others, as knowledge increases or demand for particular functionality in the relevant field expands. This flexibility is key to ensuring that advanced functionalities and / or new techniques are incorporated at later stages, should the result(s) of the verification and validation exercises lead to the conclusion that it is necessary to adjust the calculation methodology.

The only permitted inter-module communications are between the control and PTO systems. All other simulation data transfers must pass through a higher, coordinating level of code. This code manages the transfer of body kinematic information from the MBCore Calculation Module to the applied force Calculation Modules and conversely, the application of the force data returned by the force calculation Calculation Modules to the MBCore structure. This coordinating code level will also handle communication with the numerical integrator in the time-domain. The complete arrangement is illustrated by Figure 2.5, where the coordinating code level is marked as 'WVStructure'. WVStructure stores as little information as possible about the WEC itself: the parameters defining the device are read and stored directly in the calculation modules which are simply located during simulation through pointers in the co-ordinating block (marked as the blue dashed lines in Figure 2.5). The read process is described in more detail in Section 2.7.



Parallel multi-body Calculation Modules.

Figure 2.5: The modular WaveDyn architecture

2.7 User input for a multi-body code

2.7.1 Parser functionality and the ".IN" file

The flexible, multi-body code architecture must be supported by an equally fluid user input format for the supply of the WEC information. It is envisaged that WaveDyn will in the future be controlled through a graphical user interface (GUI) supporting the interactive construction of a WEC design from a library of body types, each of which having been potentially assigned with hydrodynamic, PTO or moorings definition data. Such an interface will be developed by a dedicated software team and it is desirable to decouple the engineering simulation code from this so that development work leading up to the creation of Beta versions of the code may easily continue prior to the GUI creation, and even subsequently, without any need for GUI support. The decoupling point is required to be language and compiler neutral so as not to restrict the choice of GUI development tools. Further to this, it is necessary for a pre-GUI WaveDyn user to be able to write the simulation code input files directly and, in the future, for competent users to be able to review the GUI output prior to processing in the engineering code. For these reasons, the engineering code has been designed to read all of the relevant WEC information from a specially formatted text file, assigned a ".IN" file extension. The following more detailed list of .IN file requirements were identified at the concept stage:

1. The file should support information for an unknown number of bodies making up the structure.
2. The bodies can be listed in any order.

3. Each body definition includes information for each of the multi-body system modules (MBCore, Hydrodynamics, PTO and potentially control and moorings). The information for each module should be easily identifiable by a simple parser (code designed to read information from text with a specific syntactic form).
4. The body connectivity in the form of a nodes list will need to be read.
5. The .IN file will contain general simulation setup / simulation options and output data requirements information in addition to the body/structure information.
6. Modules for single, structure wide entities such as the control system should also be supported.
7. The file should be readable by a human without extensive knowledge of the parser algorithms, and easy to write in the absence of a GUI.
8. Interface and engineering code version specification information may need to be stored for reference in a 'Header' block.
9. At a later date the file may allow encryption of certain pieces of sensitive information.
10. A special character, sequence of characters or spaces should be available to separate parameter names and parameter values, so that parameter names may contain whitespace.
11. Comment lines not read by the parser should be supported.
12. The .IN file should be easy for a GUI to write.
13. Matrices, arrays of matrices and other large vectors of data should be written in a clear, compact form.

In order to address these requirements, the user supplied input parameters in the completed .IN file format are split into blocks to be processed by the parser. The blocks may be processed in any order, each block having a specific label to control how it is read. These labels are listed in Table 2.2. The majority of blocks will be of the "BODY" type; one of these, for the Ground body, must exist for each multi-body WEC structure. The parameters in each body block are further split into a series of sub-blocks, each of which is associated with one of the parallel Calculation Modules displayed on the right hand side of Figure 2.5 and is again marked with a specific identifier label. This allows to the parse process to operate on two levels in the code: a central, high level parser controls entry and exit from the main body blocks (as well as the other blocks present) and the reading of some general body information required by multiple Calculation Modules; whilst Calculation Module level parser code reads specifically the information relevant to the module where it is located. When the central parser reaches a sub-block it tests to see if the code for the relevant Calculation Module is available (it may be that WaveDyn is being run without e.g. the moorings module, by a user with a seabed mounted device) and if successful transfers parse control to the input/output (I/O) functions within that Calculation Module; this is illustrated by the thick red lines between the I/O elements shown in Figure 2.5. As the Calculation Modules read their sub-block information, they create a new body to store the data in their multi-body system. Control is transferred back to the main parser once the Calculation Module parsers reach their sub-block closing identifier labels. This approach allows additional parameters required by the multi-body system modules to be added to their local I/O routines as they are needed for new calculation code functionality, without any onus on the developer to update the central WaveDyn parser.

As specified in requirements 4-6 above, the body blocks are accompanied in the .IN file by others providing further simulation data of a form that does not readily split into body specific segments. The data is nonetheless read directly to the specific code data structure where it is required for the simulation in a parsing process that may be handled by the high level central parser (in the case of the simulation setup and options parameters), or by a lower level parsing sub-routine located in one of the system modules (as is the case for the controller data, which is read by the Control Calculation Module and the wave data, which is read by a routine in the hydrodynamics Calculation Module).

MBCore in its existing form does not contain I/O support for the construction of the multi-body structural system and developments in this code are presently timed to fit in with the Bladed development programme. Future plans allow MBCore to be developed as a standalone module (and it is already largely written as such, as mentioned previously), but in the meantime WaveDyn relies on an 'MBCore I/O Backpack' interface written to fulfil the function of the Calculation level I/O routines present in the other Calculation Modules. The I/O Backpack code functionality will be added to the MBCore as a whole in the future, bringing it in line with the other WaveDyn Calculation Modules.

The parsing process described above builds up each of the multi-body systems in parallel as the .IN file is read. Final function calls to ready each multi-body system for simulation are made once the end of the .IN file is reached.

2.7.2 The Parser algorithm: .IN file special text characters and parsing rules

The WaveDyn input functions are designed to recognise .IN file block and sub-block identifier labels, as well as other special characters as listed in Table 2.1 below.

The accepted parameter blocks and sub-blocks are listed in Table 2.2 below. Note that data on the 'block' level (rather than sub-block level) is read by the central higher level parser in the majority of cases, however the wave data block is parsed by code in the hydrodynamics module, alongside the <hy >hy sub-blocks and the parsing of the Control data is completed by a parser in the Control Calculation Module as soon as this block is reached.

Symbol	Usage
//	Remainder of the line is a comment
{BLOCK	Start of a new block with identifier label "BLOCK"
}BLOCK	End of the block with identifier label "BLOCK"
<id	Start of a sub-block inside the block – pass parser control to I/O code in the relevant multi-body system Calculation Module.
>id	End of a sub-block. Pass parser control back to the high level central WaveDyn parser.
=	Assignment – splits parameter name and parameter value.
	Separates elements in vector or successive elements on a single row in a matrix (whitespace).
,	Signals new row in a matrix (comma) Note: a new line also has this effect

Table 2.1: .IN file special text characters

Block Name	.IN file identifier label	Parser (At block level)
Body Data	{BODY }BODY	Central WaveDyn parser
Header	{HEADER }HEADER	Central WaveDyn parser
Simulation parameters and settings	{SIMPARAMS }SIMPARAMS	Central WaveDyn parser
Wave Information	{WAVES }WAVES	Hydrodynamics Calculation Module parser
Controller parameters	{CONTROL }CONTROL	Control Calculation Module Parser
Moorings	{MOORING }MOORING	Central WaveDyn parser

Table 2.2: Accepted .IN file parameter blocks

The body and moorings blocks may contain the sub-blocks listed in Table 2.3 and Table 2.4 respectively.

Sub-block names	Sub-block identifier labels	Sub-block level parser location
MBCore Data	<mb >mb	MBCore I/O Backpack
Hydrodynamics Data	<hy >hy	Hydrodynamics Calculation Module
PTO Data	<pto >pto	PTO Calculation Module

Table 2.3: Sub-blocks to be read from within the BODY block

Sub-block names	Sub-block identifier labels	Sub-block level parser location
Mooring Line Lookup Table Data	<lk >lk	Moorings Calculation Module

Table 2.4: Sub-blocks to be read from within the MOORINGS block

All of the IO functions involved in the parsing process additionally abide by the following rules:

- Only matrix data can extend over multiple lines. A new line may indicate a new matrix row (alternatively a comma can be used), the beginning of a completely new two dimensional matrix level in a three dimension set of data. Note that the 3D case is distinguishable from the new row case because the expected size of all matrices to be read is known).
- Comments cannot be placed mid-line (only at the start or end of a line: there is no ‘uncomment’ symbol).
- Each new parameter should start on a new line.
- The {BLOCK, }BLOCK, <id and >id labels should always start on a new line and are case sensitive.
- Parameter names and any string values other than file paths are case sensitive
- Whitespace may appear at the start of a line, so that indentation can be used to make the .IN files easier to read. Additional whitespace is also accepted either side of the assignment operator (“=”) and between successive vector or matrix elements.

2.7.3 Example .IN file

The .IN file describing the simple device shown in Figure 2.1 has been reproduced in Figure 2.6 below. Some of the hydrodynamic data has been omitted for brevity; the aim here is to illustrate the block / sub-block structure of the data input, which is designed to complement the multi-body code formulation. Many of the parameters are also described in more detail in the Sections 3 to 6, which give an overview of the applied force Calculation Modules development. Note that all numerical values assume standard index (SI) units.

```
//.IN file for a simple point absorber
//Note that the indentation is not required, it's just for clarity
{HEADER
  //Nothing; may later contain information such as WaveDyn code version
}HEADER

{SIMPARAMS
  SimName = Test
  OutPath= C:\Projects\Test_Sim
  SimTime = 40
  Timestep = 0.01
  OutputInterval = 5
}SIMPARAMS

{NODES
  Number = 4
  1 = N
  2 = N1
  3 = N2
  4 = N3
}NODES

{BODY
```

```
  //The Ground element. One of these must exist.
  Type = Ground
  Name = Ground
  ProxNode = N
  <mb    //Label indicates MBCore Calculation Module data
    Location = 0.0 0.0 0.0
  >mb
}BODY

{BODY
  //The lower rigid link
  Type = RigidLink
  Name = LinkLow
  ProxNode = N
  <mb
    DistNode = N1
    DistNOffset = 0 0 5
    DistNRelRot = 1.0 0 0 0
    //Comment
  >mb
}BODY

{BODY
  //The upper rigid link
  Type = RigidLink
  Name = LinkHi
  ProxNode = N2
  <mb
    DistNode = N3
    DistNOffset = 0 0 5
    DistNRelRot = 1.0 0 0 0
  >mb
}BODY

{BODY
  //Sliding Joint
  Type = SlidingJoint
  Name = PTOPiston
  ProxNode = N1
  <mb
    DistNode = N2
    SlideFreedom = 0 0 1
    DistNRelRot = 1.0 0 0 0 //Let's stick a comment in just for fun
    StructStiff = 0.0 //Note not PTO stiffness
    StructDamp = 0.0  //Note not PTO Damping
  >mb
  <pto
    Damping = 6.0e6
    Stiffness = 0
  >pto
}BODY

{BODY
  //RigidBody
  //The only body in the system with hydrodynamic information
  Type = RigidBody
  Name = Float
  ProxNode = N3
  <mb
    Mass = 11286000
    CoM-ProxN-Offset = 0 0 0
    InertiaTensor = 1e4 0.0 0.0, 0.0 1e4 0.0, 0.0 0.0 1e4
  >mb
  <hy                        //Hydrodynamics Calculation Module Data
    -Morrison 0+BEM = 1      //Parameters described in Table 3.1.
    AddMassInf = 0 0 0
    AddInerInf =
    1e4 0.0 0.0
    0.0 1e4 0.0
    0.0 0.0 1e4
    ImpResAvailable = 1
     NumRDampFreq = 0   //Read impulse response data directly, not
                        //radiation damping data. If this is non-zero, read
                        //in a radiation damping matrix for a number of
                        //frequencies
    IRFEqualTimeSpace = 0.5
    NumImpResTimes = 61          //Impulse response data at 61 times values
    IRFTimes = <61 vector values> //omitted here for brevity
    //The IRF data given in consecutive matrices
     IRFData  =  <61  lines,  each  containing  6  comma  separated  lists  of  6
               values - so 61 6x6 matrices> //omitted for brevity
    IRFCutTime = 30
    MinConvInterval = 1.0
    //Just one frequency and direction shown here
    NumExciteFreq = 1
    NumExciteDirs = 1
    -Naut 0+Cart = 1
    ExciteFreqs = 0.5
    ExciteDirs = 0.0
    //Amplitude phase combinations for each direction
    //Note these are 6xFreqs matrices - a "," indicates a new row
    Amp = 0.0, 0.0, 2606600, 0.0, 0.0, 0.0
    Phase = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0
    HydStaticStiff = 0.0 0.0 0.0 0.0 0.0 0.0, 0.0 0.0 0.0 0.0 0.0 0.0
     0.0 0.0 3162000 0.0 0.0 0.0, 0.0 0.0 0.0 0.0 0.0 0.0,
     0.0 0.0 0.0 0.0 0.0 0.0, 0.0 0.0 0.0 0.0 0.0 0.0 //A 6x6 matrix
    BodyFixedHyOrigin = 0.0 0.0 10.0 0.0 0.0 0.0
  >hy
}BODY

{WAVES
  ComponentWaves = 1
  SpectrumFile = Waves.SM
  WaterKinematics = -1
}WAVES
```

Figure 2.6: Example .IN file for a simple point absorber system

2.7.4 The MBCore input parameters

The MBCore rigid body types supported by the WaveDyn code (and the Multi-body I/O Backpack), as well as their .IN file input parameters are described in Table 2.5 below. The input parameters read by the other Calculation Modules are described in the proceeding sections of the report.

NODES and BODY block and <mb >mb Sub-block Parameters			
Input Parameter (as presented in the .IN file)	**Data Format**	**Parser Dependencies**	**Description**
{NODE }NODE block parameters			
`Number`	Integer	None, Compulsory	Total number of nodes used to define the structure connectivity.
<List of node numbers (see example in Figure 2.6>	String	Number of node names provided must equal `Number.`	Node names
All bodies – {BODY }BODY block parameters			
`Type`	String Can be `Ground, RigidBody, SlidingJoint, Hinge, RigidLink`.	None, Compulsory	Body type
`Name`	String	None, Compulsory	User defined body name
`ProxNode`	String (must appear in NODES block list)	None, Compulsory	Proximal node name
<mb >mb Sub-block parameters – `Ground` type body			
`Location`	3 Element vector of double precision floating point numbers indicating a global X,Y,Z position.	None, Compulsory	Ground body location. Typically set simply to 0.0 0.0 0.0
<mb >mb Sub-block parameters – `RigidLink` type body			
`DistNode`	String (must appear in NODES	None, Compulsory	Distal node name. A Rigid Link has a proximal node (closest to the

	block list)		Ground) and a distal node – the attachment point of the next body higher up in the tree-structure.
`DistNOffset`	3 Element vector of double precision floating point numbers indicating a body-fixed (x,y,z) position.	None, Compulsory	Relative position of the distal node to the proximal node in body-fixed co-ordinates.
`DistNRelRot`	4 Element vector of double precision floating point numbers making up the Euler parameter quartonian for the rotation in body-fixed co-ordinates.	None, Compulsory (No rotation: `1 0 0 0` may be specified)	The relative rotation of the distal node to the proximal node, in body-fixed co-ordinates. This rotation defines the orientation of the body-fixed co-ordinate system for the next body higher up the tree.
<mb >mb Sub-block parameters – `SlidingJoint` type body			
`DistNode`	String (must appear in NODES block list)	None, Compulsory	Distal node name
`SlideFreedom`	3 Element unit vector of double precision floating point numbers in body-fixed co-ordinates.	None, Compulsory	The unit vector describes the body-fixed direction in which sliding motion is permitted.
`DistNRelRot`	4 Element vector of double precision floating point numbers making up the Euler parameter quartonian for the rotation in body-fixed co-ordinates.	None, Compulsory (No rotation: `1 0 0 0` may be specified)	The relative rotation of the distal node to the proximal node, in body-fixed co-ordinates. This rotation defines the orientation of the body-fixed co-ordinate system for the next body higher up the tree.
`StructStiff`	Double precision floating point number.	None, Compulsory, but may be 0.	Structural sliding stiffness (in addition to any applied by the PTO applied force Calculation Module)
`StructDamp`	Double precision floating point number.	None, Compulsory,	Structural sliding damping (in addition to

		but may be 0.	any applied by the PTO applied force Calculation Module. Energy dissipated through structural damping, rather than a PTO defined damping, is lost and does not contribute to the WEC power output).
<mb >mb Sub-block parameters – `Hinge` type body			
`DistNode`	String (must appear in NODES block list)	None, Compulsory	Distal node name
`HingeAxis`	3 Element unit vector of double precision floating point numbers in body-fixed co-ordinates.	None, Compulsory	The unit vector describes the body-fixed axis about which rotation is permitted.
`DistNRelRot`	4 Element vector of double precision floating point numbers making up the Euler parameter quartonian for the rotation in body-fixed co-ordinates.	None, Compulsory (No rotation: `1 0 0 0` may be specified)	The relative rotation of the distal node to the proximal node, in body-fixed co-ordinates. This rotation defines the orientation of the body-fixed co-ordinate system for the next body higher up the tree.
`StructStiff`	Double precision floating point number.	None, Compulsory, but may be 0.	Structural hinge motion stiffness (in addition to any applied by the PTO applied force Calculation Module)
`StructDamp`	Double precision floating point number.	None, Compulsory, but may be 0.	Structural hinge motion damping (in addition to any applied by the PTO applied force Calculation Module. Energy dissipated through structural damping, rather than a PTO defined damping, is lost and does not contribute to the WEC power output).

<mb >mb Sub-block parameters – `RigidBody` type body			
`Mass`	Double precision floating point number.	None, Compulsory	Physical body mass
`CoM-ProxN-Offset`	3 Element vector (x,y,z) of double precision floating point numbers in body-fixed co-ordinates.	None, Compulsory	Position of body centre of mass relative to the proximal node position.
`InertiaTensor`	3x3 Matrix of double precision floating point numbers in body-fixed co-ordinates.	None, Compulsory	Physical moment of inertia tensor about the body centre of mass.

Table 2.5: .IN file BODY and NODES block and mb (MBCore) sub-block parameters. mb sub-block parameters have been given for all body types currently supported by the code.

2.7.5 .IN file error checking

Error checking in the .IN file reading process occurs in both the central and multi-body system parser levels and every data parsing function confirms the parameters it has read by writing them out to a common verification file, passed around within the code by reference. This file has the extension ".$VE". Specifically, the error checks completed by the parser code are as follows:

- Unidentified blocks or sub-blocks are simply passed over by the parser, although a warning message is displayed.

- Although the blocks and sub-blocks may appear in any order, parameter name – value combinations read by the code within these may not be. In this way, it is possible to ensure that a valid WEC description (if not a true representation of the intended system) is always read before a simulation commences. If the parser fails to find an expected parameter then a message is displayed detailing which parameter could not be found (and what was found instead), before the program is terminated. A user may examine the .$VE file to determine where in the .IN file the parser read to at a later date.

- Failure to read specific vector or matrix elements or supply of an invalid data type for a particular parameter will result in a terminating error (for example a string will not be accepted in place of a double precision number – note however that a string may legitimately incorporate numerical values).

- Nested blocks are not accepted – an attempt to enter a new block whilst the parser is already inside one will result in a terminating error. The same is true for sub-blocks.

- If the end of the .IN file is reached part way through a parameters block, then the program terminates. WaveDyn will also terminate if the end of the .IN file is reached before SIMPARAMS, HEADER and at least1 BODY block has been read. These blocks are compulsory. The WAVES block is compulsory if any hydrodynamic data has been supplied for a body.

2.8 Frequency-domain analysis using a multi-body code

The linear transfer function style of frequency-domain analysis presented in the report WG1 WP1 D1b, Section 3 is only valid for linear (or linearised) systems; the applied forces on the device must be linear (or linearised) and so must be the WEC kinematic response to those forces. The flexible structural modelling approach provided by the MBCore Calculation Module means that even if the applied force definitions are linear, a user may easily define a nonlinear WEC structure for simulation, invalidating a pure frequency response calculation. Indeed any structure designed to incorporate large rotational motions as a result of applied forces that are not themselves acting in a purely rotational sense (as the hydrodynamic forces will not be), will be nonlinear.

If the complete device is linear, then the MBCore structural mass and stiffness matrices may be used directly in a transfer function type calculation of the form (see Equation 3.8, WG1, WP1, D1b):

$$X = \left(-\omega^2\left(M_m + M_r\right) + j\omega\left(B_m + B_r\right) + K_m + K_{hs}\right)^{-1}\left(F_e + F_{ext}\right) \qquad [2.11]$$

where X is a matrix containing the response amplitude for each of the device DOFs; M_m is the structural mass matrix; M_r is a frequency dependent added-mass matrix; B_m contains any structural damping terms; B_r is a frequency dependent radiation damping matrix; K_m is the structural stiffness matrix; K_{hs} is the hydrostatic stiffness matrix; F_e is the complex wave excitation force amplitude and F_{ext} represents the forces applied to the system by the PTO and Moorings.

The exact form of the structural mass, stiffness and damping matrices will depend on the configuration of joints and bodies defined by the user, so that the permitted system degrees of freedom will be taken into account in the calculation.

An expression of the above form may be used to model a nonlinear structure (again under the influence of linear applied forces) that deflects by only a small amount in its nonlinear freedoms about a mean operating point. In this case, the result will be approximate only. It is worth noting that frequency-domain hydrodynamic solvers also typically only calculate hydrodynamic force coefficients for small deflections. Verification studies as part of the PerAWaT programme aim to determine to what extent these linear hydrodynamic approaches are valid, but it may be that these also limit a frequency-domain analysis.

The above discussion leads to three alternative approaches to frequency response analysis:

1. If the structure and applied forces are linear, then a pure frequency-domain expression of the form presented in Equation 2.11 may be used to generate a response spectrum. The global structural mass, damping and stiffness matrices are constant for all kinematic system states (they would not vary with time in a time-domain simulation) and may be extracted from the MBCore structural model. The response amplitude may be non-dimensionalised with incident wave amplitude so that information may be expressed in terms of response amplitude operators (RAO). This approach assumes that a linear hydrodynamic representation is a valid approximation for all of the absolute response amplitudes being considered (the RAO is not

likely to be valid for extreme responses, where hydrodynamic forces are still likely to be nonlinear in reality).

2. The structure is nonlinear, but motion in the nonlinear freedoms is small. The MBCore structural matrices change as the device moves, but those for the mean operating point may be extracted and used for an approximate frequency-domain formulation as long as the hydrodynamic formulation is valid throughout the small range of response amplitudes. Non-dimensional RAO values may still be calculated but their validity is limited by the range of structural responses that may be approximated as linear, in addition to the range of responses over which the hydrodynamic formulation is accurate.
3. It is likely that many nonlinear structures will have some form of time-periodic response to excitation at a single wave frequency. An analysis of response time histories produced by multiple short time-domain simulations conducted over a range of monochromatic regular wave inputs would allow defining features of the periodic response to be plotted as frequency response spectra. The spectra would be dependent on wave amplitude (twice the wave amplitude may not necessarily produce twice the amplitude of device response) and so RAO values could not be calculated.

Despite not being a ‘pure’ frequency-domain approach (the time response is analysed), the third approach listed above may be applied to the widest range of structures (the response spectra produced would simply display a linear scaling with increasing wave amplitude for a device fitting in to the criteria required for approach 1) and so this has been selected as first frequency-domain implementation technique. A check is required to ensure that the time response is periodic, but otherwise no check on the structural definition provided by the user need be made. Frequency spectra calculated using this technique are presented for the case studies described in Section 8. A pure frequency-domain approach as described in points 1 and 2 above, making direct use of the MBCore structural matrices may be implemented as WaveDyn develops, if the need arises to more rapidly simulate the performance of purely linear devices.

2.9 The time-domain numerical integration process

The WaveDyn code structure is designed to accommodate time domain simulation alongside the frequency domain post-processing described above. The time domain aspects of the code will be described in detail in the proceeding PerAWaT deliverables, however an overview of the time-domain integration process is given below as this has a strong influence on the WaveDyn code architecture that has been introduced above.

In the time-domain, the simulation code is designed to operate with explicit integrators, which require the system to return the state derivatives as a function of a supplied set of state values. Some integrator algorithms require multiple state derivative calculations to be completed each simulation time-step, passing the system a different set of *intermediate* state values in each call and using the state derivative values returned by the system to calculate a *final* set of state values for the time-step; these are the values that are saved as the simulation results before the simulation advances to the next step. Other, multi-step integrators may rely on state values for multiple past time-steps for the calculation of future time-step states. Variable time-step integrator algorithms adjust the time-step size dynamically based on the magnitude of the state derivative values, so that periods where the system dynamics are fast can be more accurately captured. The WaveDyn code has been designed to support a range of integrators, potentially requiring this functionality. Implicit integration algorithms are not currently supported.

A single code function in WVStructure, accepting system states as arguments and returning state derivatives, the StateDevs function, is called directly by the integrator, once, or on multiple occaisions each time-step. As an additional argument, the integrator must specify whether the call is based on a intermediate set of state values or a final time-step set (in the intermediate step case, the system force and kinematic data are not saved to the simulation output buffers or stored in any convolution buffers containing previous time-step values). StateDevs co-ordinates a series of operations:

1 In the first, the structural states are passed as updates to the MBCore structural multi-body system.

2 following this, any applied force calculation states are passed to a 'Loading' function which loops through a set of Forces functions, one for each applied force Calculation Module.

3 The Forces functions query MBCore for a required set of body kinematics (a prior query is made to the force Calculation Module to determine what this should be) and pass these on to their applied force Calculation Modules along with any relevant force calculation states (which may for example be an accumulator charge in the PTO Calculation Module).

4 The Calculation Modules return a set of forces and locations on the structure where these should be applied in a specifically designed applied force data transfer class object. The applied force Calculation Modules also return any state derivative values for the internal force calculation states.

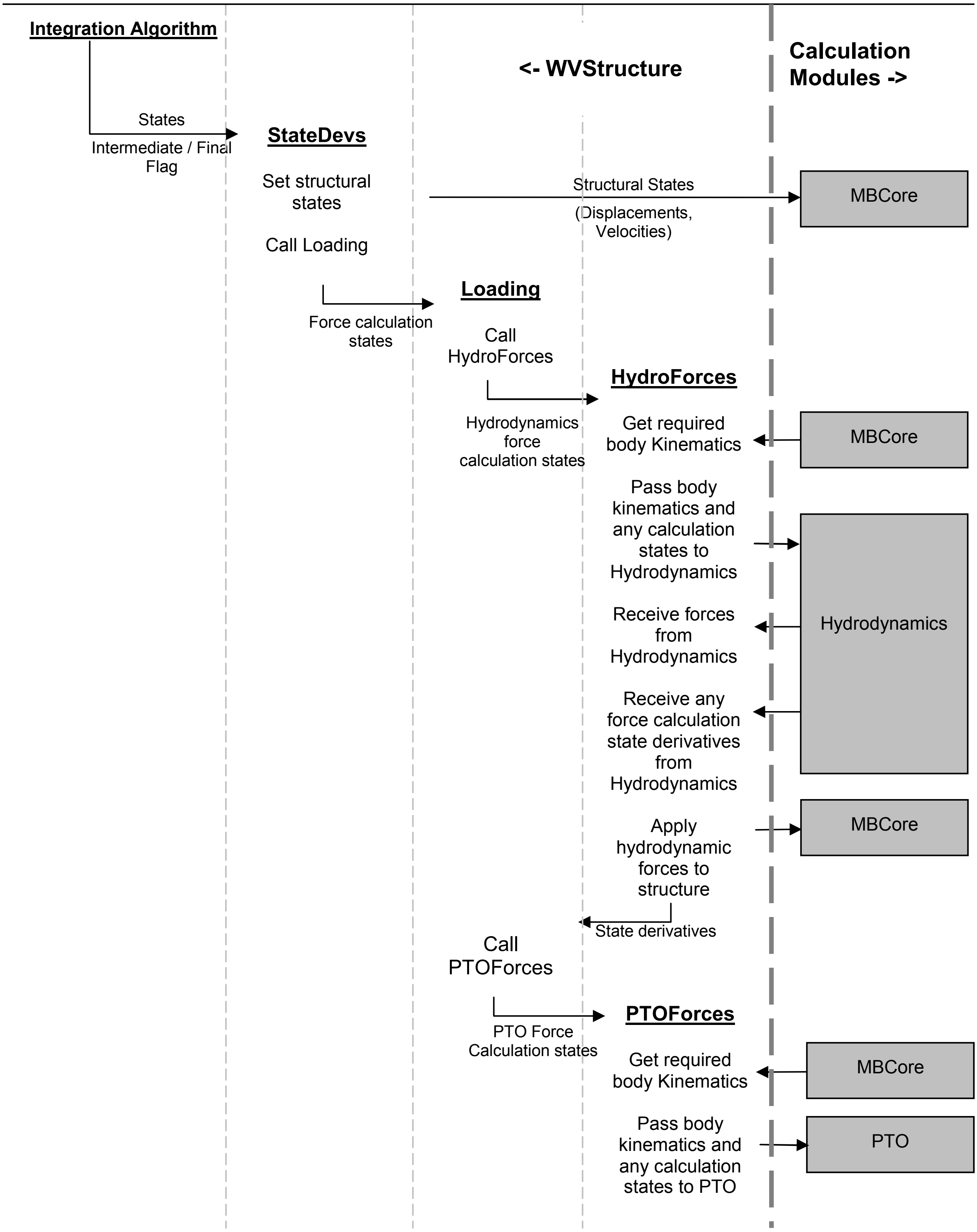
5 The Forces functions apply the forces to the structure as prescribed and return the applied force calculation state derivatives they receive to Loading

6 Loading compiles all of the applied force state derivatives and returns these to the StateDevs function.

7 StateDevs queries MBCore for the structural state derivatives (accelerations) and combines these with the applied force state derivatives, returning the complete set to the integrator.

The complete process is illustrated graphically in Figure 2.7. Note that the Control Module acts on the system through the PTO and so has not been included in the diagram. It is possible for internal control states to exist in the Control calculation, in which case these would be returned through the StateDevs function to the integrator as well.

The integration process requires an initialisation step to be completed at the beginning of the simulation, in which each of the Calculation Modules is required to return the number of states to be solved. The states are then assigned suitable initial values. At present, the initial values are simply zero for all states, however a more developed time-domain simulation may attempt to determine an initial equilibrium state to set in order to minimise the length of starting transients at the beginning of the simulation.


Integration Algorithm
States
Intermediate / Final Flag
<- WVStructure
Calculation Modules ->
StateDevs
Set structural states
Call Loading
Structural States
(Displacements, Velocities)
MBCore
Loading
Force calculation states
Call HydroForces
HydroForces
Hydrodynamics force calculation states
Get required body Kinematics
MBCore
Pass body kinematics and any calculation states to Hydrodynamics
Hydrodynamics
Receive forces from Hydrodynamics
Receive any force calculation state derivatives from Hydrodynamics
Apply hydrodynamic forces to structure
MBCore
State derivatives
Call PTOForces
PTOForces
PTO Force Calculation states
Get required body Kinematics
MBCore
Pass body kinematics and any calculation states to PTO
PTO

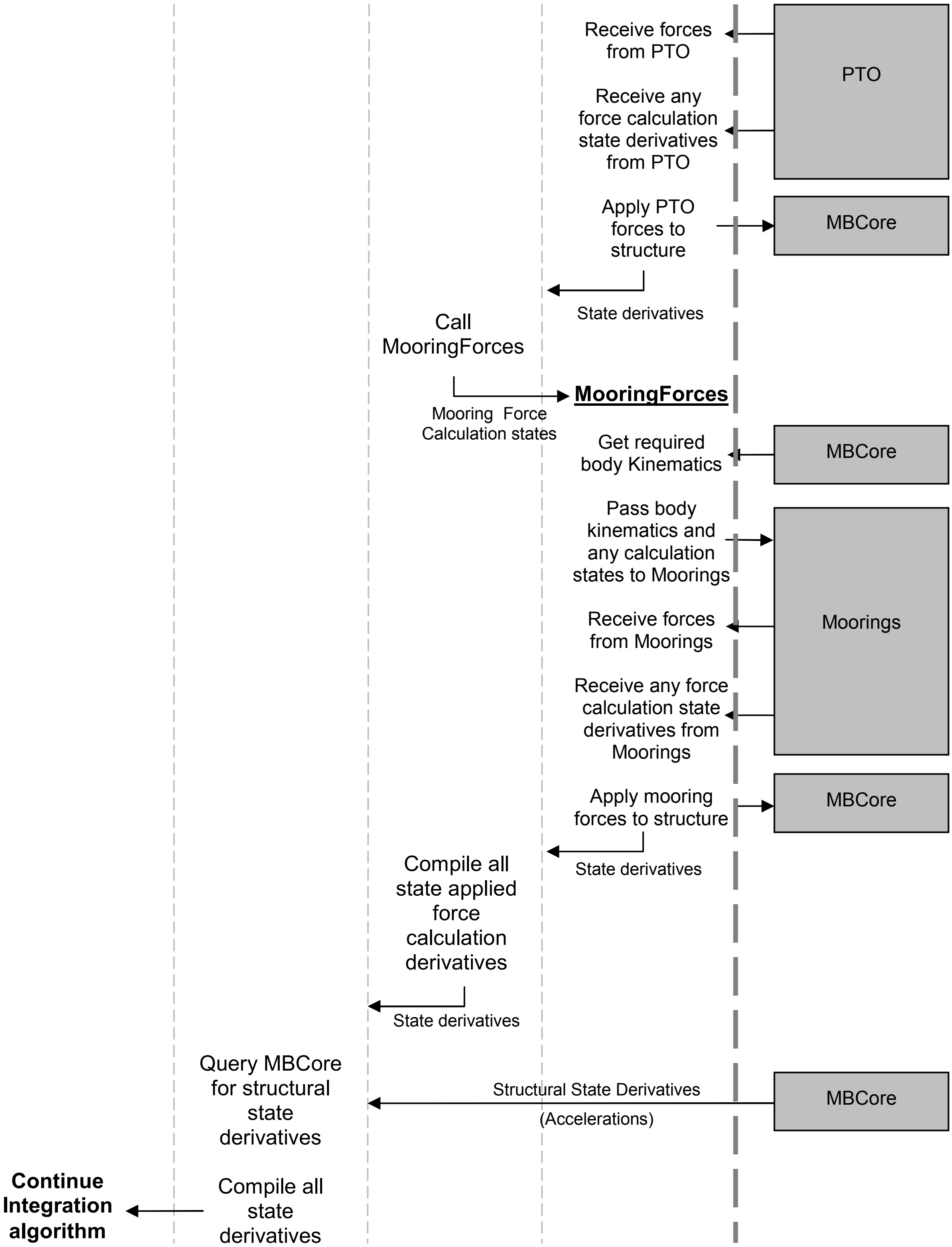

Receive forces from PTO
PTO
Receive any force calculation state derivatives from PTO
Apply PTO forces to structure
MBCore
State derivatives
Call MooringForces
MooringForces
Mooring Force Calculation states
Get required body Kinematics
MBCore
Pass body kinematics and any calculation states to Moorings
Moorings
Receive forces from Moorings
Receive any force calculation state derivatives from Moorings
Apply mooring forces to structure
MBCore
State derivatives
Compile all state applied force calculation derivatives
State derivatives
Query MBCore for structural state derivatives
Structural State Derivatives
(Accelerations)
MBCore
Continue Integration algorithm
Compile all state derivatives

Figure 2.7: The State Derivatives Calculation Process in WVStructure

2.10 The role of WaveDyn

The WaveDyn multi-body modelling approach described above covers the scope of the 'FD', 'TD', 'Basic Controller', 'External forces code' and 'Wave data loader' modules originally envisaged for the WaveFarmer code structure in the Methodology Report (WG1 WP1 D1b, Section 1), which is reproduced in Figure 2.8 below. The 'Basic Controller' and 'External forces code' blocks now operate as the series of parallel multi-body Calculation Modules shown in Figure 2.5, whilst the FD and TD blocks are represented in the higher level WVStructure co-ordinating code. The wave data loader is one of the I/O routines in the Hydrodynamics Calculation Module. It is envisaged that the Calculation Modules may be developed to give the user the option of replacing the core functionality of each with a DLL interface, however this interface will itself be part of the Calculation Module from an overall software architecture perspective.

The name 'WaveFarmer' as a software package now refers purely to the array design code (and includes the Optimiser and Mapping blocks shown in Figure 2.8). The WaveFarmer Optimiser has the ability to prescribe and analyse the results from WaveDyn simulations, for a WEC operating in any position in an array environment (subject to appropriate hydrodynamic data being provided), as described in Section 8.

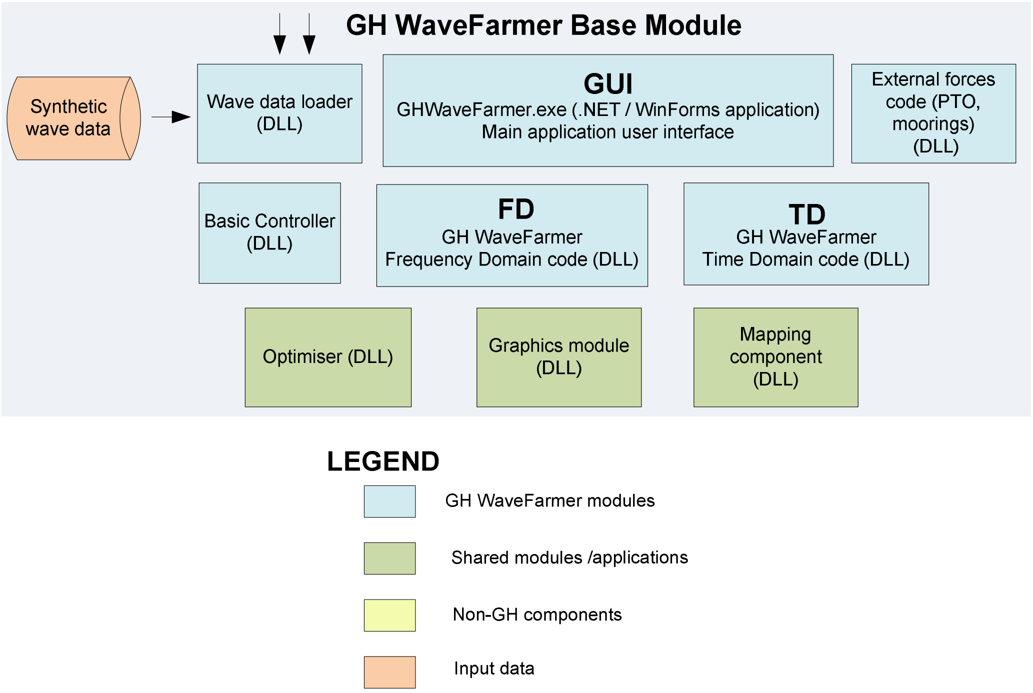


Figure 2.8 Summary of GH WaveFarmer's envisaged structure under the PerAWaT programme

3 HYDRODYNAMICS MODULE

3.1 Key theoretical aspects

In a pure frequency-domain approach, the Hydrodynamics Calculation Module would be required to read the hydrodynamic data for every wave activated body from the .IN file and return complex excitation force amplitude, radiation damping and hydrostatic stiffness matrices for the wave frequencies used to evaluate Equation 2.11. The task is restricted to correctly sorting through the user supplied hydrodynamic data and applying any necessary co-ordinate system transformations.

In the time-domain, the task is more complex: the hydrodynamics module is required to evaluate the excitation, radiation and hydrostatic forces acting on each wave activated body in the system and return a set of resultant forces to be applied at particular locations in the WEC structure. It is useful to refer to the time-domain equation of motion for a single body under wave loading in the form described in detail in the WG1 WP1 D1b Methodology report, reproduced below for convenience.

$$\left(m_m + m_r(\infty)\right)\ddot{x} + f_{hs}(x) + \int_{-\infty}^{t} k(t-\tau)\dot{x}(\tau)\ d\tau = f_e(t) + f_{ext}(x, \dot{x}, \ldots, t) \qquad [3.1]$$

The structural elements of the equation of motion are compiled by the MBCore Calculation Module, so that in reality the equation of motion is solved for the complete WEC in an expanded, more generalise form; however on an individual body basis, the hydrodynamic forces in terms of a constant, infinite frequency added-mass, a hydrostatic stiffness, a radiation force convolution and a wave excitation force are applied in the format indicated. The constant added-mass at infinity values for the wave activated bodies, $m_r(\infty)$, may be added directly to the MBCore structure at the beginning of the simulation and so the task in the Hydrodynamics Calculation module is to evaluate:

- the excitation force, $f_e(t)$;
- the radiation convolution term, $\int_{-\infty}^{t} k(t-\tau)\dot{x}(\tau)\ d\tau$, and
- the hydrostatic force, $f_{hs}(x)$ which is a linear stiffness term if a water-plane area is assumed constant (for the case for a linear formulation),

for each wave activated body in the system.

3.2 Data formats and co-ordinate systems

The user is responsible for supplying the hydrodynamic data for the wave activated bodies in the WaveDyn .IN file format. A complete description of the .IN file hydrodynamic inputs is given in Table 3.1 below. The data must appear between the "<hy >hy" sub-block identifiers for each body and must be provided in the order presented.

Data output from commercial flow solvers such as WAMIT may need to be processed to meet the WaveDyn requirements; this may be done manually or with a simple script written in MATLAB (or an alternative). It is foreseeable that GH may write a collection of MATLAB scripts to produce the

WaveDyn format from the output data provided by the most commonly used flow solvers. Currently the WAMIT format is followed for the WEC hydrodynamic parameters.

<hy >hy Sub-block Parameters			
Input Parameter (as presented in the .IN file)	**Data Format**	**Parser Dependencies**	**Description**
`-Morrison 0+BEM`	Integer Value. ≥ 0 : Boundary Element Method Body <0 : Morrison Element Body (not currently supported)	None Parameter is compulsory, even though it is currently redundant.	Simple Flag. Morrison element data is not currently supported; all bodies must be modelled using a Boundary Element Method (BEM) flow solver data.
Radiation Force Data			
`AddMassInf`	3x1 Vector of surge, sway and heave added-mass in the body-fixed reference plane.	None Compulsory	Added-mass at infinity
`AddInerInf`	3x3 Matrix – An added inertia tensor for Roll, Pitch, Yaw of the body about the centre of mass.	None, Compulsory	Added inertia at infinity
`ImpResAvailable`	Integer Value >0 – Impulse response function data provided ≤ 0 – Impulse response function not provided	None, Compulsory	If the impulse response function is not provided it may be calculated from the radiation damping data, however this is not currently supported.
`NumRDampFreq`	Integer	None, Compulsory	Number of frequencies at which radiation damping data will be provided.
`RDampFreq`	Vector of `NumRDampFreq` elements. Each element is a double precision, floating	Read if `NumRDampFreq` > 0 Otherwise parameter should be omitted.	Frequency ordinate of radiation damping data.

	point number.		
RDampMat	NumRDampFreq 6x6 Matrices. Each new matrix must start on a new line. New lines mid-matrix may still be used instead of a "," to mark a new row.	Read if NumRDampFreq > 0 Otherwise parameter should be omitted.	Radiation damping data. Matrix format is Surge, Sway, Heave, Roll, Pitch, Yaw.
IRFEqualTimeSpace	Double precision floating point number. Impulse response function time ordinate is equally spaced, by this amount if >0. <0 – Spacing is uneven.	Read if ImpResAvailable > 0 Otherwise parameter should be omitted.	Equal spacing of the time ordinate speeds up interpolation of the data during evaluation of the radiation forces convolution integral at run-time.
NumImpResTimes	Integer	Read if ImpResAvailable > 0 Otherwise parameter should be omitted.	Number of time values in the impulse response function time ordinate.
IRFTimes	Vector of NumImpResTimes elements. Each element is a double precision, floating point number.	Read if ImpResAvailable > 0 Otherwise parameter should be omitted.	Impulse response function time ordinate.
IRFData	NumImpResTimes 6x6 matrices. Each new matrix must start on a new line. New lines mid-matrix may still be used instead of a "," to mark a new row.	Read if ImpResAvailable > 0 Otherwise parameter should be omitted.	Impulse response function data. Matrix format is Surge, Sway, Heave, Roll, Pitch, Yaw.
IRFCutTime	Double precision floating point number.	Read if ImpResAvailable > 0 Otherwise parameter should be omitted.	Length of time-history used in the evaluation of the radiation force convolution for the body in question. Short time will speed up the simulation but will result in loss of accuracy.

`MinConvInterval`	Double precision floating point number.	Read if `ImpResAvailable > 0` Otherwise parameter should be omitted.	The minimum time spacing between values stored in the convolution time history buffers. A short time reduces simulation speed, but improves accuracy.
Excitation Force Data			
`NumExciteFreq`	Integer	None, Compulsory	Number of frequency values in the excitation force frequency ordinate.
`ExciteFreqEqSpace`	Double precision floating point number. Impulse response function frequency ordinate is equally spaced, by this amount if >0. <0 – Spacing is uneven.	None, Compulsory	Equal spacing of the frequency ordinate speeds up interpolation of the data during evaluation of the wave excitation forces at run-time.
`NumExciteDirs`	Integer	None, Compulsory	Number of direction values in the excitation force direction ordinate.
`ExciteDirEqSpace`	Double precision floating point number. Impulse response function direction ordinate is equally spaced, by this amount if >0. <0 – Spacing is uneven.	None, Compulsory	Equal spacing of the direction ordinate speeds up interpolation of the data during evaluation of the wave excitation forces at run-time.
`-Naut 0+Cart`	Integer ≥ 0 : Direction ordinate provided in global cartesian angles. <0 : Direction ordinate provided in nautical angles.	None, Compulsory	Incident wave direction ordinate for the excitation force data. In WaveDyn, the positive global x-axis points South

`NorthACFromBodyX`	Double precision floating point number.	Only read if `-Naut 0+Cart < 0` (Nautical) Otherwise parameter should be omitted.	Cartesian angle of excitation force direction ordinate North axis, measured anti-clockwise from global X direction.
`ExciteFreqs`	Vector of `NumExciteFreq` elements. Each element is a double precision, floating point number.	None, Compulsory	Excitation force data frequency ordinate.
`ExciteDirs`	Vector of `NumExciteDirs` elements. Each element is a double precision, floating point number.	None, Compulsory	Excitation force data directions ordinate.

`Amp`	6 x `NumExciteFreq` Matrix. Each element is a double precision, floating point number.	`Amp` and `Phase` pair provided `ExciteDirs` number of times.
		`Amp =`
		`Phase =`
		`Amp =`
		`Phase =`
		…
		At least one set is compulsory.
			Excitation force amplitude data in Surge, Sway, Heave, Roll, Pitch, Yaw for a single direction, over the complete frequency range.

`Phase`	6 x `NumExciteFreq` Matrix. Each element is a double precision, floating point number.		Excitation force phase data in Surge, Sway, Heave, Roll, Pitch, Yaw for a single direction, over the complete frequency range.
Hydrostatic Force Data			
`HydStaticStiff`	6x6 Matrix. Each element is a double precision, floating point number	None, Compulsory	Hydrostatic stiffness matrix for Surge, Sway, Heave, Roll, Pitch, Yaw.
`BodyFixedHyOrigin`	6 Element vector of double precision, floating point numbers	None, Compulsory	Global position of hydrodynamic co-ordinates reference location.

Table 3.1: Hydrodynamics Force Calculation Data .IN File Parameters

It is typical for flow solvers to provide hydrodynamic data for a body in a form that refers the motion about a fixed reference position in global space. This is usually the body's static equilibrium position and is provided as a user input to the flow solver. In WaveDyn, the global co-ordinate system origin is located at the Ground body (as long as the Ground body has its location parameter set to *X*,*Y*,*Z* = (0.0,0.0,0.0), otherwise the origin is at the specified offset from the Ground body), with the positive *Z* axis pointing upwards, and the positive *X* axis pointing South. The body hydrodynamic coefficients reference location supplied to the flow solver must be supplied to WaveDyn in global co-ordinates as the position of the body proximal node using the parameter `BodyFixedHyOrigin`, which accepts an (*X*,*Y*,*Z*,Roll,Pitch,Yaw) vector. Hydrodynamic force calculations for a body are based the instantaneous displacement of its proximal node from its `BodyFixedHyOrigin` position and its absolute proximal node velocity. The forces are converted back to the global co-ordinate system before being passed to WVStructure for application to the MBCore structure.

MBCore returns body kinematics for use in the applied force Calculation Modules in the global co-ordinate system only and so it is necessary to apply the following transformations to convert the data relative to the `BodyFixedHyOrigin` before the force calculation may proceed. The approach assumes that the reader is familiar with the theory previously provided in Section 2.5, however the cases are not identical.

The `BodyFixedHyOrigin` is an inertial reference frame positioned at a stationary point in global space, described by position vector d_{gHY}. The `BodyFixedHyOrigin` is shown in green in the diagrams below, whilst the global frame of reference is shown in black. At a given point in time, a wave activated body may be positioned at any point in global space. Its position and orientation is represented by the instantaneous body-fixed reference frame, which is marked in red.

Conversion of global body X,Y,Z displacements to surge, sway and heave values in the hydrodynamics co-ordinate system:

The task is to determine position vector $d_{HY\ BF}$ from the known global vectors d_{gHY} and d_{gBF}, as shown in Figure 3.1. $d_{HY\ BF}$ is a position vector relative to the `BodyFixedHyOrigin` reference plane and so it is necessary to convert the global position vectors to this plane so that the difference between the two will yield the required answer. The conversion requires pre-multiplication by the inverse (transpose) of the rotation matrix describing the orientation of the `BodyFixedHyOrigin` relative to the global origin, R_{HY}, so that the end result takes the form:

$$d_{HY\ BF} = R_{HY}{}^{T} d_{g\ BF} - R_{HY}{}^{T} d_{g\ HY} = R^{T}\left(d_{g\ BF} - d_{g\ HY}\right) \qquad [3.2]$$

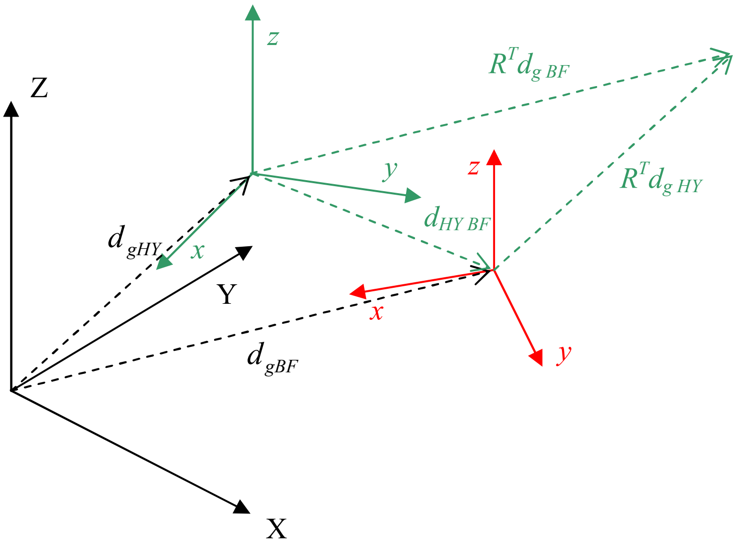


Figure 3.1: Determining the instantaneous body position relative to the hydrodynamic reference plane, `BodyFixedHyOrigin`

Conversion of global body Roll, Pitch and Yaw displacements to the hydrodynamics co-ordinate system:

The procedure described above provides a means of calculating the surge, sway and heave motions of the body only. The roll, pitch and yaw displacements are also required and may be determined as follows:

MBCore can be queried to return the Euler parameter quartonian describing the instantaneous global orientation of the body-fixed (red) reference frame. The Euler parameter can be converted to roll, pitch and yaw angles which may in turn be compiled into a rotation matrix R_{BF} that would rotate the global axes to the orientation of the body-fixed frame. The rotation matrix R_{HY} is also known; this rotates the global axes to the `BodyFixedHyOrigin` orientation.

The objective is to determine the rotation matrix that rotates the `BodyFixedHyOrigin` to the body-fixed reference plane; this is equivalent to the net rotation achieved by rotating the `BodyFixedHyOrigin` to the global orientation and then rotating the result to the body-fixed frame, a process achieved by the combined rotation matrix:

$$R_{HY}{}^{T} R_{BF} \qquad [3.3]$$

as shown in Figure 3.2. A single set of roll, pitch and yaw angles may be extracted from the combined matrix using equation 2.2. These angles represent the instantaneous orientation of the body relative to the `BodyFixedHyOrigin` as required.

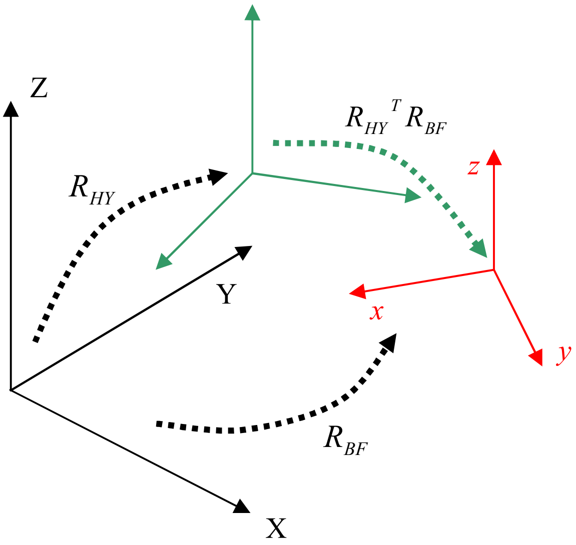


Figure 3.2: Rotation matrix combinations for determining the instantaneous body rotation relative to the Hydrodynamic reference frame, `BodyFixedHyOrigin`

Conversion of global body velocity and force values to the hydrodynamics co-ordinate system:

The velocity transformations are simpler than those for the displacements, as the three different reference frames are not offset from one another in velocity. As a result, the global (*X*,*Y*,*Z*) and (Roll,Pitch,Yaw) velocity vectors describing the body-fixed reference plane, $v_{g\ BF}$ and $\omega_{g\ BF}$ may be rotated directly into the hydrodynamics frame of reference, `BodyFixedHyOrigin` using the R_{HY} rotation matrix alone:

$$v_{HY\ BF} = R_{HY}{}^{T} v_{g\ BF} \qquad [3.4]$$

$$\omega_{HY\ BF} = R_{HY}{}^{T} \omega_{g\ BF} \qquad [3.5]$$

The force calculations completed once the body kinematic data from MBCore has been converted are described in Sections 3.4. The force vectors must be converted back into the global co-ordinate system to be applied to the structure. The conversion once again involves the hydrodynamics reference frame rotation matrix only:

$$F_{g\ BF}(X,Y,Z) = R_{HY} F_{HY\ BF}(Surge, Sway, Heave) \qquad [3.6]$$

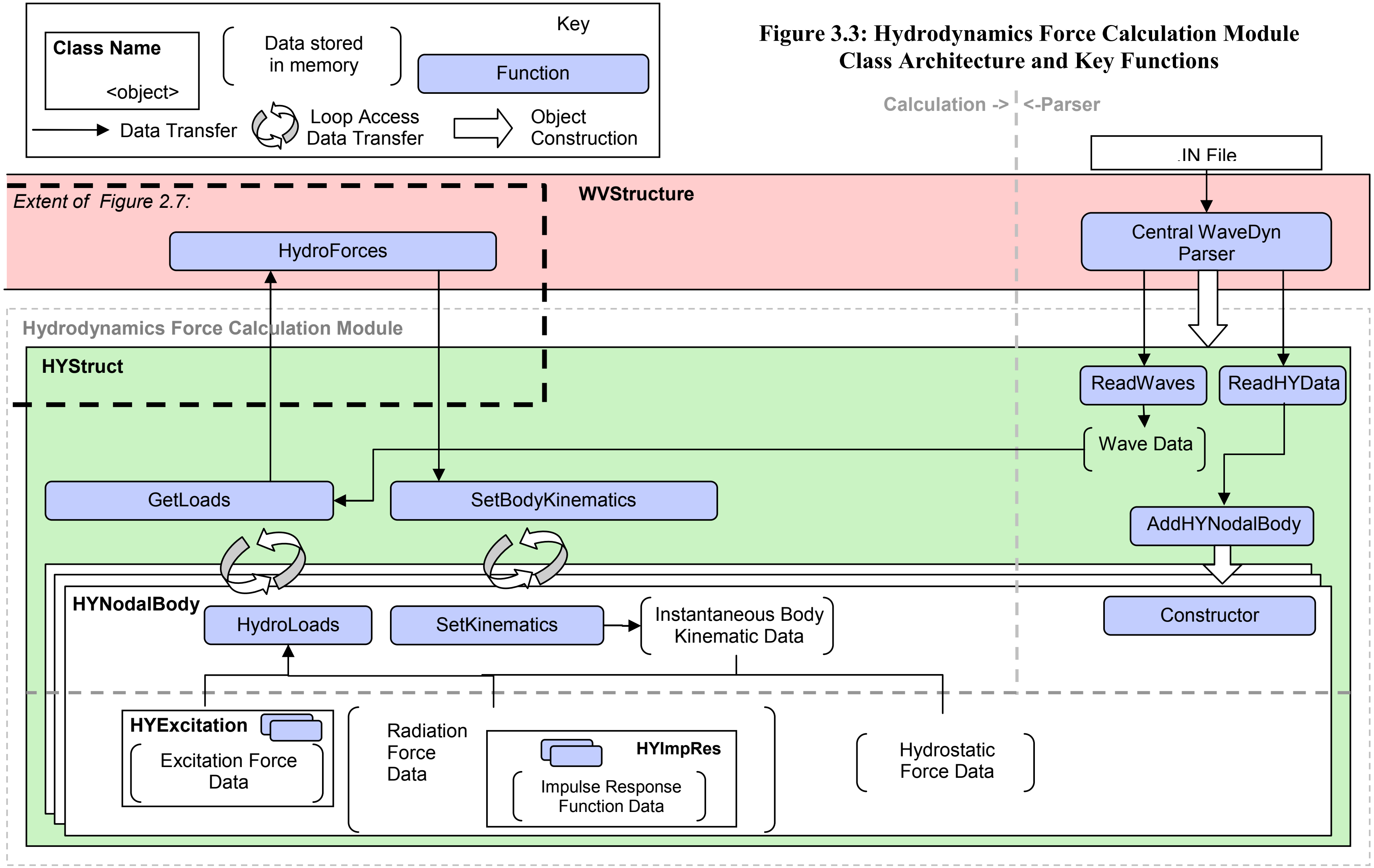
$$F_{g\ BF}(Roll, Pitch, Yaw) = R_{HY} F_{HY\ BF}(Roll, Pitch, Yaw) \qquad [3.7]$$

3.3 The multi-body hydrodynamics approach

The user may choose to define hydrodynamic properties for any body making up the WEC definition and so the hydrodynamic force calculations need to be completed on a body-by-body basis. The Hydrodynamics module is built on two levels to accommodate this. The lowest body level contains instances of a hydrodynamic body class, 'HYNodalBody'. The HYNodalBody objects contain variables storing all of the hydrodynamic data and, in the time-domain, the hydrodynamics co-ordinate system referenced instantaneous body kinematics for the current time-step. A member function in the class, SetKinematics sets the body kinematics variables by completing the transformations from global values as described in Section 3.2 above. Further member functions complete the excitation, radiation and hydrostatic forces calculations. These are co-ordinated by the HydroLoads function shown in Figure 3.3 and will be discussed in greater detail in Section 3.4. The proximal node name of the bodies is also stored in HYNodalBody so that when the forces are transformed back to global co-ordinates and returned to WVStructure, the code at that level knows which location in the MBCore structural definition to apply the forces.

The HYNodalBody objects are enclosed within a container object instance of a 'HYStruct' class, the higher code level in the Calculation Module. The HYStruct object operates as the interface for communication with the WVStructure coordinating code and incorporates a number of member functions designed to manipulate the HYNodalBodies list: ReadWaves and ReadHYData are the Hydrodynamics module parsing functions for reading the flow solver coefficients and wave data; AddHYNodalBody adds a new NodalBody to the HYStruct; and SetBodyKinematics and GetLoads are called during time-domain simulations and fulfil the role of looping through the HYNodalBodies, setting the current kinematic values from MBCore and collecting the resultant forces and application locations for the entire WEC structure. A specially designed return type class (not shown in Figure 3.3) has been defined to accommodate the force-location combinations of data that are produced by the GetLoads functions in all of the applied force Calculation Modules, including Hydrodynamics, allowing the information to be efficiently returned to the HydroForces function in WVStructure (see Figure 2.7).

Figure 3.3: Hydrodynamics Force Calculation Module Class Architecture and Key Functions



3.4 Force calculation algorithms

The hydrodynamic force calculations within each HYNodalBody in the multi-body HYStruct object are co-ordinated by the HydroLoads function. The calculation process for each of the three main hydrodynamic force components is described below.

3.4.1 Excitation force

The excitation force data read by the parser (see Table 3.1) is stored directly in an instance of a bespoke class built specifically to support the calculation process, HYExcitation. HYExcitation contains three-dimensional amplitude and phase information storage variables and the matching frequency and direction ordinates (the other dimensions distinguishes between surge, sway, heave, pitch, roll and yaw data). Two different member functions are called during the read process to ensure that the data stored in the object is consistent: the calling code must first supply the frequency ordinate, before sequentially pushing on the `Amp` and `Phase` matrices in combination with a direction value. Much more significantly, HYExcitation incorporates a member function that may be called at any time to return two 6x1 vectors, representing the excitation for amplitude coefficient and phase data for a supplied frequency and direction. This is useful in allowing the HydroLoads to loop through each of the wave frequency-direction components in the sea-state description, computing the excitation force contribution from each. Linear wave theory allows the component results to be summed to form an overall excitation force as follows:

$$f_e = \sum_n f_{e,n} = \sum_n \Re\left(F_{e,n} e^{j(\omega t+\phi_n)}\right) \qquad [3.8]$$

where the complex excitation force amplitude, $F_{e,n}$ for wave frequency component n has an amplitude that is equal to the product of the wave amplitude, A and the amplitude coefficient value determined by the flow solver (that stored in the `Amp` matrices), Γ. The complex amplitude phase, ϕ_e, is given directly by the phase data in the `Phase` matrices. ϕ_n in Equation 3.8 above is also a phase term, but is the wave component phase provided by the sea-state definition., so that for each wave component, the excitation force is calculated as:

$$f_{e,n} = \Gamma_n A_n \sin\left(\omega_n t + \phi_{e,n} + \phi_n\right) \qquad [3.9]$$

The calculation is completed for each of the surge, sway, heave, pitch, roll and yaw direction contributions.

The hydrodynamic force calculations in the time-domain are completed for each wave activated body, at least once every time-step (the amount depends on the numerical integrator algorithm) and the number of wave frequency components defining the sea-state may be high. As a result it is important for the HYExcitaton member function responsible for returning the amplitude and phase data to be as efficient as possible. It may be that the wave frequencies and directions in the sea-state description do not match the values in the frequency and direction ordinates for the stored coefficient data (the user may choose to define a sea-state for the WaveDyn simulation with different frequency components than those used in the initial flow solver calculations); in this case, the HYExcitation class will need to complete a two-dimensional interpolation of the data (as an interpolation in frequency and direction may be necessary). The interpolation process may be thought of as consisting of two key parts:

1 A search to find the data values either side of the interpolation point

2 The interpolation calculation itself.

If a linear interpolation strategy is used then the calculation (point 2) is quick and simple, however the search process (point 1) can take potentially take large amounts of time and it is important to employ a quick search algorithm. A search can be avoided altogether if the user supplies the data with equally spaced direction and frequency ordinates, as specified by the ExciteFreqEqSpace and ExciteDirEqSpace parameters described in Table 3.1 (in which case the indices of the data values either side of the interpolation point can be calculated directly). If this is not the case, then, rather than simply looking through the frequency and direction ordinates sequentially, the software has been designed to search using a mid-point bisection method, where the required value is reached by sequentially halving the search range using a single comparison with the middle data value, as shown in Figure 3.4. The code has been written so that data may be supplied by the user with the frequency and direction ordinates in either ascending or descending order.

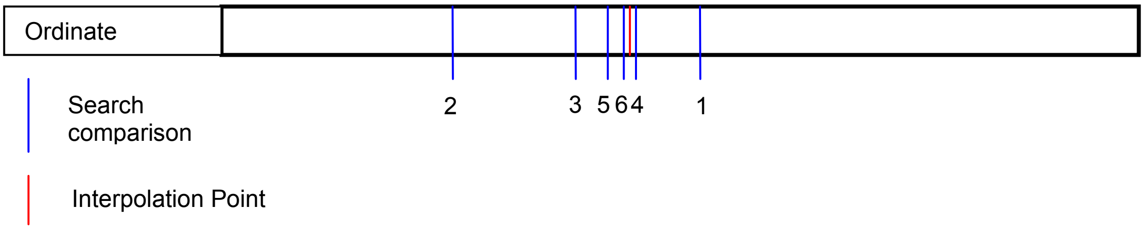


Figure 3.4: Mid-point bisection interpolation method. The first 6 search comparisons have been numbered.

3.4.2 Radiation force

The radiation force calculation process involves a numerical evaluation of the radiation forces convolution integral, as discussed in Section 3.1:

$$f_r' = \int_{-\infty}^{t} k(t-\tau)\dot{x}(\tau)\ d\tau \qquad [3.10]$$

It is envisaged that the user may choose to either enter 6x6 radiation damping matrices for a number of wave frequencies, in which case the Hydrodynamics Calculation Module will need to compute the impulse response function from these (not yet a feature of the code), or to provide impulse response function data directly, in which case the data is again provided in 6x6 matrices, but with a time ordinate as third dimension (see Table 3.1). The latter option alone has been implemented in the code at this stage.

Once again, a specifically designed data handling class has been created to assist with the calculation process. The HYImpRes class contains variables used to store the impulse response function data and member functions designed to read and write values to these variables during the parse process (ensuring data is stored in ascending time order) and interpolate the impulse response function for a given time. The interpolation algorithm is analogous to that described for the excitation force coefficients in Section 3.4.1 above, with the `IRFEqualTimeSpace` parameter being used to determine whether or not the data is equally spaced. The data is interpolated in one direction only (time), with the function returning a 6x6 surge, sway, heave, roll, pitch, yaw matrix of the approximate impulse response function values at the time requested.

Another member function in HYImpRes incorporates the convolution calculation itself. This final function requires a velocity time history (two vector containers, one storing past time-step velocity values, the other storing the times at which they were calculated) as its input and completes the convolution, in a matrix format (handling surge, sway, heave, roll, pitch and yaw values simultaneously), using a trapezium rule approximation. The current body velocity kinematic data is pushed on to the velocity time history at the start of the calculation so that the final time value stored in the time history vector is the current time, t. The convolution time parameter τ runs from the furthest point back in the time history (towards $-\infty$) to the current time, so a vector representing impulse response function kernel, $k(t-\tau)$ can be compiled by interpolating the impulse response function data at each $t-\tau$ combination provided by the discrete set of τ values. The $\dot{x}(\tau)$ vector is simply the velocity history, so a vector c containing the element-by-element products of the N values stored in each of the velocity and kernel vectors can now be compiled to contain the values to be used for the integration:

$$c[1] = k[1] * \dot{x}[1] \;\; ; \quad c[2] = k[2] * \dot{x}[2] \, ; \; \ldots \, ; \; c[N] = k[N] * \dot{x}[N] \qquad [3.11]$$

which is coded as the summation:

$$f_r' = \sum_{i=0}^{N-1} \left(0.5 * \left(T[i+1] - T[i] \right) * \left(c[i] + c[i+1] \right) \right) \qquad [3.12]$$

where T represents the vector of velocity history times and [ ] is used to denote access to a particular vector value.

The algorithm described above accommodates the use of variable time-step integrators which may mean that the velocity time history is not stored with equally spaced time intervals. Once the integration has been completed, a check is made to see if the calculation is part of an intermediate integrator call, or a complete time-step result evaluation. If the call is intermediate, then the current velocity value, which is added to the velocity time history at the beginning of the hydrodynamic forces calculation, is removed again. This ensures that only complete time-step values have an effect on the force calculation in future time-steps.

3.4.3 Hydrostatic force

The hydrostatic force calculation is completed as a simple pre-multiplication of the current body displacement vector by the constant hydrostatic stiffness value, provided by the user using the .IN file parameter `HydStaticStiff` (the stiffness matrix is determined by the flow solver but may also be readily calculated by hand):

$$f_{hs}(x) = K_{hs} x(t) \qquad [3.13]$$

3.5 Time-domain performance for frequency analysis

The applied force calculations completed each simulation time-step represent the most performance critical part of the WaveDyn code. This is particularly the case when large numbers of short time responses need to be analysed to build up a frequency response spectrum. The design of interpolation algorithms to account for this has already been described in Section 3.4.1; however it is the numerical convolution process that is the greatest computational burden in the hydrodynamic force calculations. It was noted in Methodology Report WG1 WP1 D1b, Section 3.3.4 that the need to complete a numerical convolution may be overcome by using a system identification technique to approximate the radiation effect using a finite-state-space model. The implementation of such an approach is reserved for the time-domain report, deliverable WG1 WP1 D3. In the interim it is possible however to substantially speed up the convolution algorithm by making two adjustments to the pure numerical process described in Section 3.4.2.

The first numerical adjustment is the inclusion of a parameter that allows the user to specify how far back in time the convolution should be evaluated. It is common for impulse response functions to be such that only the most recent 30s of time history have a significant effect on the force calculation for the current time-step. Convolving over a 30s, or even 1 minute, time history, rather than the complete 30 minutes that may otherwise build up towards the end of a long simulation, can save large amounts of computational time (recall that the time-step may be of the order of hundredths or thousandths of a second, making the element-by-element vector operations described in Section 3.4.2, which must be completed for every wave-activated body, very lengthy). The shape of the impulse response function changes with each body and each device and so the user will need to determine what a suitable cut-off time may be and specify an appropriate value in the `IRFCutTime` .IN file parameter. It is possible that in the future, WaveDyn may incorporate an impulse response function analysis function, which suggests what value this should be, by determining the time taken by the impulse response function to decay below a particular threshold. This option will be explored for the pure time-domain implementation (WG1 WP1 D3).

The second numerical adjustment incorporated in the current WaveDyn code allows the Hydrodynamics Calculation Module to complete the radiation convolution over coarser time-step that that used for the simulation by the numerical integrator. The user may specify a minimum time interval between the values saved in the velocity time history using the `MinConvInterval` parameter. The calculation process only allows new velocity time history values to be saved if at least this period of time as passed since the last was added. The convolution is still completed using accounting for the time difference between the current time and saved time history values (so that even if no new velocity time history values are added between successive time-steps, the computed convolution result will not be the same), however the effective step-size for the integration process is much larger (as the time history contains many fewer values). A switch to a higher order accuracy numerical integration algorithm (rather than the trapezium rule approach described in Equation 3.12, may help to offset some of the loss in accuracy, however the approximation is likely to be reasonable if the user keeps the `MinConvInterval` relatively small in comparison to the time-periods associated with the device response dynamics. A more detailed study on the effect of setting a MinConvInterval parameter for devices that respond both continuously and discontinuously will be investigated in detail in WG1 WP1 D3, however just one case is illustrated in Figure 3.5 which shows the change in the computed radiation force time history for the system shown in Figure 2.1 using a MinConvInterval value of zero (time history values saved for every integrator time-step) and 1s. The 1s simulation runs significantly (of the order of 30-40%) faster, but the radiation force time history is visibly rough, as shown in Figure 3.5. In this case the effect on the body displacement is minimal, because the radiation force amplitude is much smaller that those of the excitation and hydrostatic forces.

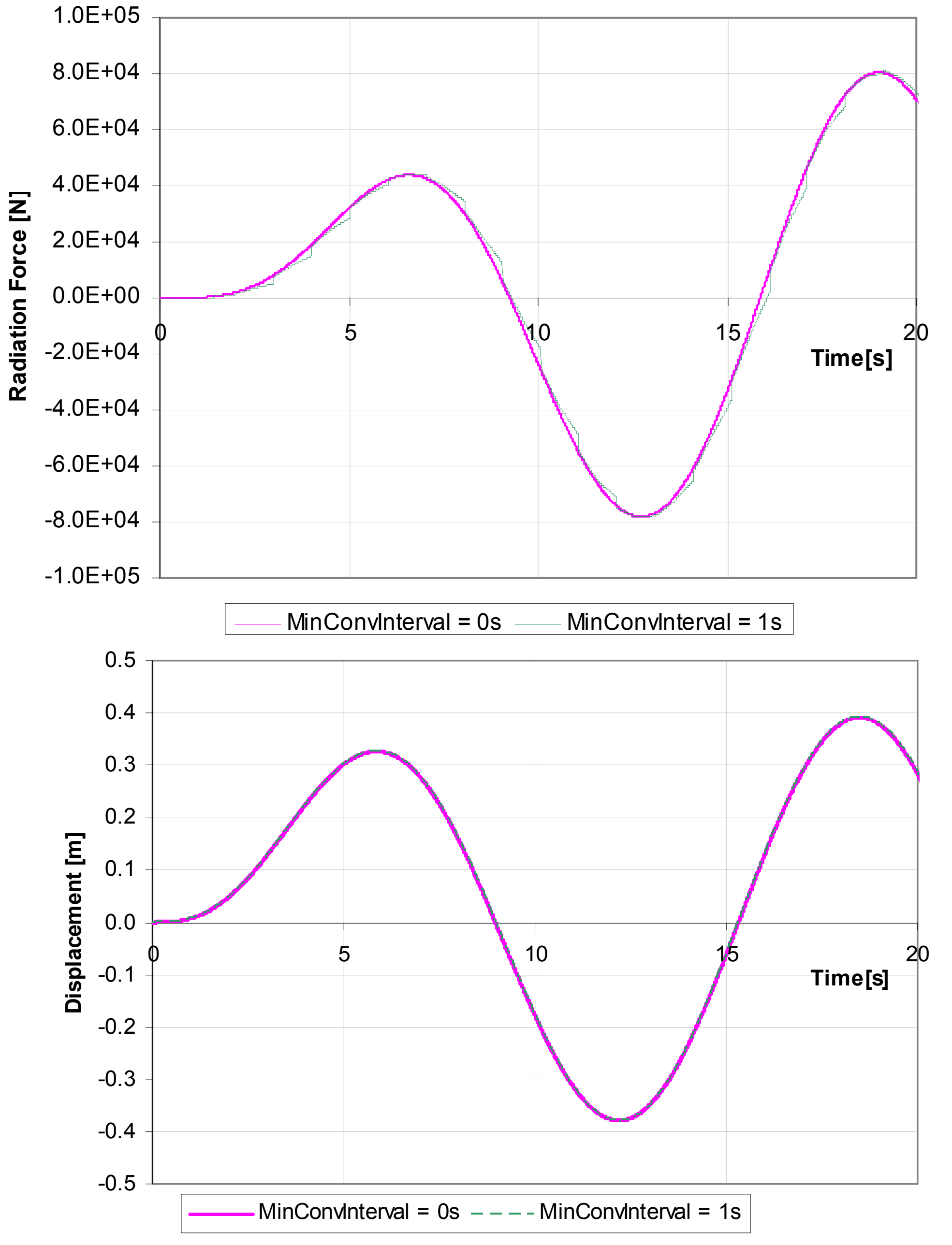


Figure 3.5: Radiation force and displacement time histories for a short section of simulation using MinConvInterval values of 0s and 1s. The simulation model is shown in Figure 2.1. The radiation force is not computed as precisely in the 1s simulation, but the effect on the device response is minimal.

4 WAVE ANALYSIS MODULE

4.1 Current inputs to the simulation

The target specifications for the wave base module to be developed in PerAWaT were defined in WG1 WP1 D1b. The base module must have the following core capabilities:

- Generate standard shape frequency spectra and directional spreading functions for a given sea state parameter vector.
- Load directional spectra from measured data.
- Create a probabilistic parametric description of a site-specific wave climate from measured spectra.
- Output unidirectional and directional spectra to frequency and time-domain simulations.

It was noted that the processing of raw measured data to wave spectra and parameters was outside scope of PerAWaT. However, the base module will have the ability to load measured spectra from files, as described in Section 4.1.2.

Since WG1 WP1 D1b was written, it has been decided that the output of the wave module to the frequency-domain (FD) and time-domain (TD) codes will be a list of wave amplitudes, frequencies, directions and phases rather than a spectrum. It was previously envisaged that these values would be calculated from the spectrum within the FD or TD code. However, for consistency this information will be calculated within the wave module and output in an ASCII format text file with the extension .SEA. This allows both regular and irregular wave information to be output in a single format.

This section describes the current functionalities of the wave module. Section 4.1.1 describes the standard spectral shapes which can be used as input to the simulation. Section 4.1.2 describes the format in which spectra are loaded to and saved from the wave module. Finally, the simulation of the free-surface elevation from a wave spectrum and the format of the files output to the FD simulation is discussed in Section 4.1.3.

A method to describe the wave climate in parametric form is still in development. The progress so far is described in Section 4.2. Further developments will be documented in WG1 WP1 D3.

4.1.1 Standard models for wave spectra

The directional spectrum $S(f,\theta)$ is usually decomposed into two functions, one representing the total energy at each frequency, and the other describing how the energy at each frequency is distributed with direction:

$$S(f,\theta) = E(f)D(f,\theta) \qquad [4.1]$$

$E(f)$ is called the omnidirectional spectrum (or sometimes the frequency spectrum) and is related to the directional spectrum by

$$E(f) = \int_0^{2\pi} S(f,\theta)d\theta \qquad [4.2]$$

$D(f,\theta)$ is the directional spreading function (or directional distribution) and satisfies two properties:

1. $$\int_0^{2\pi} D(f,\theta)d\theta = 1 \qquad [4.3]$$

2. $D(f,\theta) \geq 0$ over $[0\ 2\pi]$ [4.4]

Standard models for frequency spectra are described in Section 4.1.1.1 and standard models for the directional distribution are described in Section 4.1.1.3. Standard spectra from the wave module are generated as the product of a frequency spectrum and directional distribution as expressed in Equation 4.1.

4.1.1.1 Frequency spectra

There are several commonly used forms for unimodal spectra which result from a combination of theoretical considerations and empirical guidance. In deep water the shape of the spectrum is controlled by the balance between the wind input, dissipation from white-capping and nonlinear quadruplet wave-wave interactions. During active wave growth, when the waves are relatively steep, nonlinear quadruplet interactions play a central role in controlling the shape of the spectrum, forcing it towards 'standard' unimodal shapes and smoothing local deviations (Young and Van Vledder, 1993). If the wind input varies sufficiently slowly (as is normally the case in a storm) then nonlinear quadruplet interactions dominate and a JONSWAP type spectrum will evolve. However, if the wind drops or the waves leave their generation area then the steepness of the waves will decrease (due to frequency-dispersion and direction-dispersion) and the quadruplet interactions will decrease accordingly. In this case the spectral shape will depend upon the history of the individual wave components and a 'standard' shape cannot be expected. This can result in spectra with multiple peaks, from one or more swells possibly together with a local wind sea. In these cases parametric descriptions of the frequency spectrum can be formed as a sum of two or more standard unimodal spectra.

A useful review of models for spectral shapes is presented by Michel (1999). The author notes that the most commonly used forms of unimodal spectra belong to the family given by:

$$E(f) = \alpha f^{-r} \exp(-\beta f^{-s})\gamma^{\delta(f)}\,, \quad \text{for } \alpha,\beta,r,s > 0\,,\ \gamma > 1 \qquad [4.5]$$

where

$$\delta(f) = \exp\left(\frac{(f-f_p)^2}{2\sigma^2 f_p{}^2}\right) \qquad [4.6]$$

and it is usually assumed that

$$\sigma = \begin{cases} 0.07 \text{ for} f < f_p \\ 0.09 \text{ for} f \geq f_p \end{cases} \qquad [4.7]$$

The parameters *r* and *s* control the shape of the spectrum, *α* is the scale parameter, *β* is the location parameter (in terms of frequency) and *γ* is known as the peak enhancement factor. The peak frequency of the spectrum is given by

$$f_p = \left(\frac{s\beta}{r}\right)^{\frac{1}{s}} \qquad [4.8]$$

The high frequency tail of spectrum is proportional to f^{-r}. There is some debate on whether the spectral tail follows an f^{-4}-shape or an f^{-5}-shape. Most recent theoretical and empirical evidence suggests that an f^{-4}-shape is more appropriate (see Holthuijsen, 2007, for a review). However, the most commonly used spectra in ocean engineering have an f^{-5} tail. For practical purposes the difference appears to be negligible.

The family of spectra given by Equation 4.5 have five free parameters. To describe the sea state with fewer variables, some of these parameters can be fixed whilst the others are left free. The most commonly used families of spectra with one, two and three free parameters are summarised in Table 4.1. Note that β is a free parameter, but is always determined by the peak frequency together with r and s according to Equation 4.8. Note also that for the Gamma spectra it could have been written equivalently that s is free and r is fixed as s + 1.

Table 4.1. Free and fixed parameters for families of unimodal spectra given by Equation 4.5.

	α	β	r	s	γ
Pierson-Moskowitz	5.0×10^{-4}	free	5	4	1
Bretschneider	free	free	5	4	1
JONSWAP	free	free	5	4	free
Ochi	free	free	free	4	1
Gamma	free	free	free	r - 1	1

When $\gamma = 1$ the moments of the spectrum (and hence spectral parameters) can be expressed as explicit functions of α, β, r and s:

$$m_n = \frac{\alpha}{s}\beta^{\left(\frac{n-r+1}{s}\right)}\Gamma\left(\frac{r-n-1}{s}\right) \quad \text{for} \quad n < r-1 \qquad [4.9]$$

If $\gamma \neq 1$ then numerical integration must be used to compute the moments (see Appendix A of WG1 WP1 D1b).

Examples of the JONSWAP, Ochi and Gamma families are shown in Figure 4.1. The spectra have been normalised by H_s and f_p, and are shown for a range of the third free parameter. In each case the Bretschneider spectrum is a special case and is indicated with a bold line. For the JONSWAP family the Bretschneider spectrum is the limiting form, corresponding to the most broad-banded member. As the peak enhancement factor, γ, increases the spectra become more peaked, but a spread of energy remains between about $0.6f_p$ and $2f_p$. Gamma and Ochi spectra can take more broad-banded forms than JONSWAP spectra, albeit with the possibility of a physically unrealistic amount of energy in the tail for low values of r. For Ochi spectra there is little variation in the shape for frequencies less than f_p, whereas for Gamma spectra the proportion of energy below f_p increases as r decreases. For higher values of r, both Gamma and Ochi spectra can have an arbitrarily narrow concentration of energy about the peak frequency.

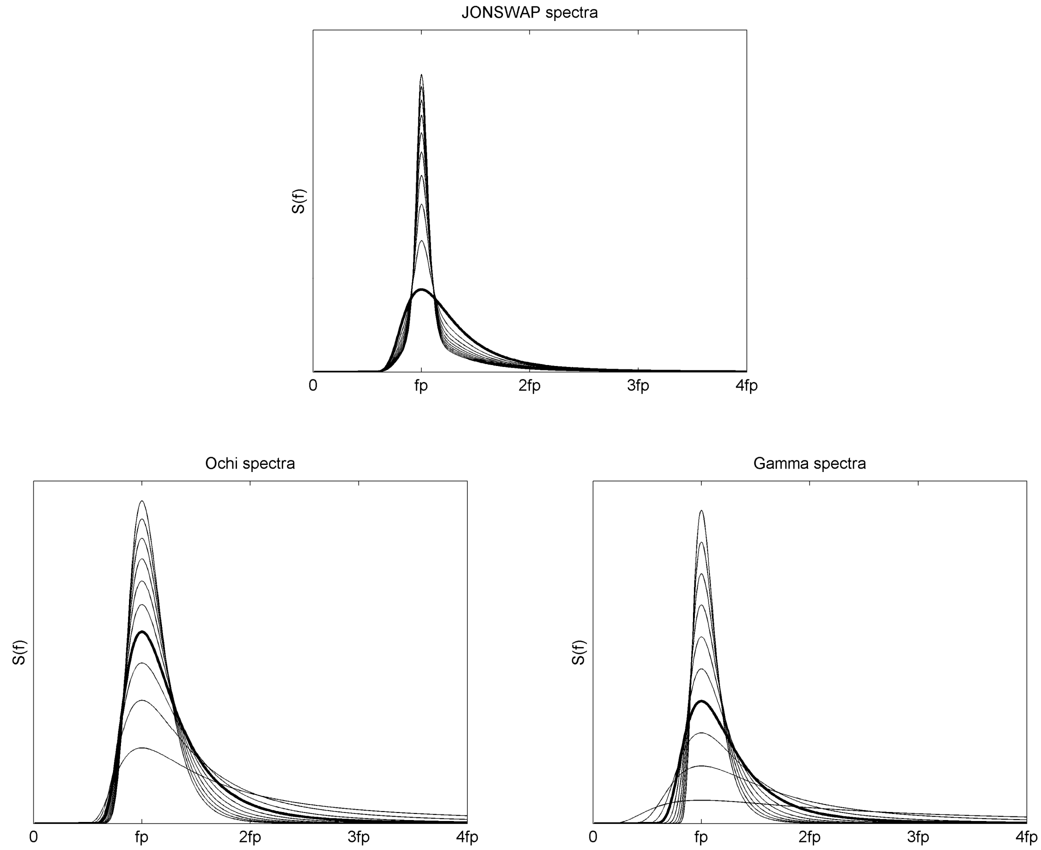


Figure 4.1. JONSWAP spectra for γ = 1-10 and Ochi and Gamma spectra for r = 2-11. Spectral densities have been normalised so that m_0 = 1. In each plot the Bretschneider spectrum is shown in bold.

Formulations for spectra in shallow water such as the TMA spectrum (Bouws et al, 1985) or the form proposed by Young and Babanin (2006) have not been implemented at present as only deep water WECs are considered in PerAWaT.

The most commonly used multimodal spectral forms are formulated as the summation of either JONSWAP, Gamma or Ochi spectra. Ochi and Hubble (1976) proposed a six-parameter spectrum formed as the sum of two Ochi spectra. However, each of the six free parameters was given as a function of H_s (together with 10 spectra representing a 95% confidence interval), so in essence this is a one parameter spectrum. Guedes Soares (1984) proposed a bimodal spectrum formed as the sum of two JONSWAP spectra, but γ fixed as 2 for both components, resulting in a four parameter spectrum. Torsethaugen (1993) and Torsethaugen and Haver (2004) have proposed a form consisting of two JONSWAP spectra. However, the values of the parameters of each spectrum are determined by the values of H_s and f_p, so the number of free parameters is reduced from six to two. Finally, Boukhanovsky and Guedes Soares (2009) modelled multi-modal spectra as the summation of Gamma spectra, imposing no restrictions on the parameter values, resulting in a true six parameter spectrum.

4.1.1.2 A note on directional spread

Before discussing directional distributions it is worth revisiting some the definitions of directional parameters presented in WG1 WP1 D1b. The RMS directional spread was defined in Appendix A of the WG1 WP1 D1b report (Equation A.20) as

$$\sigma_l(f) = \left[\int_{-\pi}^{\pi} D(f,\theta)(\theta-\theta_m)^2 d\theta \right]^{1/2} \qquad [4.10]$$

Here the subscript *l* has been added to indicate that this is a line moment definition. Kuik et al. (1988) advocated the use of RMS directional spread defined in terms of circular moments as

$$\sigma_c(f) = \left[\int_{-\pi}^{\pi} D(f,\theta)\left[2\sin\left((\theta-\theta_m)/2\right)\right]^2 d\theta \right]^{1/2} \qquad [4.11]$$

This definition approximates the line moment definition for narrow directional bandwidths since $2\sin\left(x/2\right) \approx x$ for small values of *x*. Moreover, Kuik et al. showed that the circular moment definition of spread can be expressed as a function of the first two Fourier coefficients of the directional distribution, a_1 and b_1:

$$\sigma_c(f) = \left\{ 2\left(1-\left[a_1^2+b_1^2\right]^{1/2}\right)\right\}^{1/2} \qquad [4.12]$$

This is useful for measured data, since the full directional distribution is not known and must be estimated from a small number of Fourier coefficients (see Appendix B of WG1 WP1 D1b).

Although, the difference between σ_l and σ_c is small for narrow bandwidths, σ_l increases faster than σ_c as the bandwidth increases. It is therefore important to distinguish between the two definitions as they appear in different contexts in the following discussion.

4.1.1.3 Directional distribution

There are two aspects to consider when modelling the directional distribution. The first is how the energy is distributed with direction at a particular frequency and the second is how this distribution changes with frequency.

The most commonly used forms of the directional distribution at a particular frequency are the ‘cosine2s’ distribution and the wrapped normal distribution. These have been implemented for the wave base module. The ‘cosine2s’ distribution is given by

$$D(\theta, f) = F(s)\cos^{2s}\tfrac{1}{2}\left(\theta-\theta_m(f)\right) \qquad [4.13]$$

where $F(s)$ is a factor necessary to satisfy Equation 4.3:

$$F(s) = \frac{1}{2\sqrt{\pi}}\frac{\Gamma(s+1)}{\Gamma(s+\frac{1}{2})} \qquad [4.14]$$

The RMS angular spread (circular moment definition) is given by

$$\sigma_c^{\ 2} = \frac{2}{1+s} \qquad [4.15]$$

The second commonly used formulation is the wrapped-normal distribution:

$$D(\theta, f) = \frac{1}{\sigma_l(f)\sqrt{2\pi}} \sum_{k=-\infty}^{\infty} \exp\left[-\frac{1}{2}\left(\frac{\theta-\theta_m(f)-2\pi k}{\sigma_l(f)}\right)^2\right] \qquad [4.16]$$

The summation over k in Equation 4.13 is to ensure that energy outside the interval [0 2π] is added back in. In practice the summation can be taken over range $k = -2,..,2$. This formulation directly includes the spread parameter σ_l (line moment definition). The circular RMS spread can be calculated from the Fourier coefficients which are given by:

$$\begin{aligned} a_1(f) &= \exp\left(\frac{-\sigma_l^2(f)}{2}\right)\cos\left(\theta_m(f)\right) \\ b_1(f) &= \exp\left(\frac{-\sigma_l^2(f)}{2}\right)\sin\left(\theta_m(f)\right) \\ a_2(f) &= \exp\left(-2\sigma_l^2(f)\right)\cos\left(2\theta_m(f)\right) \\ b_2(f) &= \exp\left(-2\sigma_l^2(f)\right)\sin\left(2\theta_m(f)\right) \end{aligned} \qquad [4.17]$$

Setting $\theta_m = 0$ and substituting into Equation 4.12 gives

$$\sigma_c(f) = \sqrt{2\left[1-\exp\left(\frac{-\sigma_l^2(f)}{2}\right)\right]} \qquad [4.18]$$

For narrow directional bandwidths the ‘cosine2s’ and wrapped-normal distribution have very similar shapes, as illustrated in Figure 4.2. An overview of other standard forms used for the directional distribution is given by Kahma et al (2005).

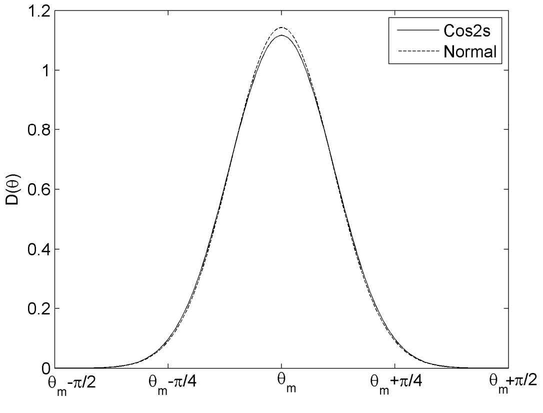


Figure 4.2. Comparison of Cos2s and wrapped normal directional distributions for σ_l = 20°.

For fetch-limited wind seas the directional distribution is bimodal at frequencies greater than about twice the peak frequency (see e.g. Young et al., 1995; Ewans, 1998; Hwang et al., 2000). Ewans (1998) has proposed the use of a double Gaussian distribution to model this bimodality. It can be written as

$$D(f,\theta)=\frac{1}{\phi(f)\sqrt{8\pi}}\sum_{k=-\infty}^{\infty}\left\{\exp\left[-\frac{1}{2}\left(\frac{\theta-\theta_1(f)-2\pi k}{\phi(f)}\right)^2\right]+\exp\left[-\frac{1}{2}\left(\frac{\theta-\theta_2(f)-2\pi k}{\phi(f)}\right)^2\right]\right\}$$

[4.19]

where

$$\theta_1(f)=\theta_m+\Delta\theta(f)/2$$

$$\theta_2(f)=\theta_m-\Delta\theta(f)/2$$

and $\Delta\theta$ is the separation between the peaks of the two modes. The values of $\Delta\theta$ and ϕ are given as functions of frequency:

$$\Delta\theta=14.93 \text{ for } f<f_p \qquad [4.20]$$

$$\Delta\theta=\exp\left[5.453-2.750\left(\frac{f}{f_p}\right)^{-1}\right] \text{ for } f\geq f_p \qquad [4.21]$$

$$\phi=11.38+5.357\left(\frac{f}{f_p}\right)^{-7.929} \text{ for } f<f_p \qquad [4.22]$$

$$\phi=32.13-15.39\left(\frac{f}{f_p}\right)^{-2} \text{ for } f\geq f_p \qquad [4.23]$$

The resulting distribution is unimodal for $f<2f_p$ and becomes bimodal at higher frequencies. This formulation results in a directional distribution which is qualitatively the same as earlier studiers (e.g. Mitsuyasu et al., 1975; Hasselmann et al., 1980; Donelan et al., 1985) in the way that the spread varies with frequency. However earlier studies made the *a priori* assumption that the distribution was unimodal. Mitsuyasu et al. (1975) and Hasselmann et al. (1980) also suggested that the distribution was dependent on the wave age (a function of the wind speed and phase speed of the waves), whereas no such dependence was noted in later studies.

For the directional distribution of swell Ewans (2001) found less evidence of bimodality than for wind seas. He therefore proposed the use of the wrapped normal distribution (Equation 4.16) with

$$\sigma_l(f) = 6 + 4\left(\frac{f}{f_p}\right)^{-5} \quad \text{for } f < f_p \qquad [4.24]$$

$$\text{for } f \geq f_p \qquad [4.25]$$

A comparison of the directional spread (circular moment definition) as a function of frequency for the wind sea and swell distributions proposed by Ewans (1998, 2001) is shown in Figure 4.3. The circular RMS spread has been obtained from the Fourier coefficients of the bimodal Gaussian distribution using formulas analogous to those for the single Gaussian distribution given in Equation 4.17.

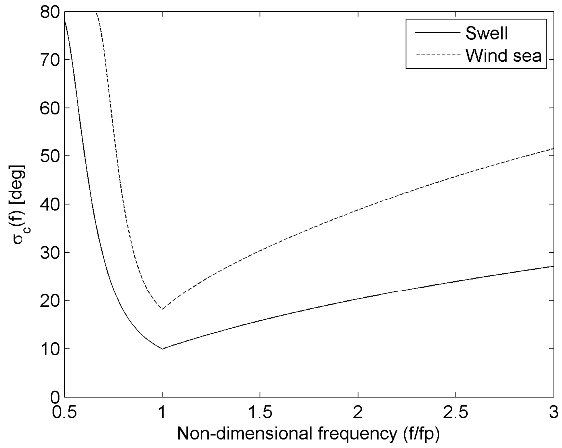


Figure 4.3. Comparison of directional spreading with frequency for the swell and wind sea directional distributions proposed by Ewans (1998, 2001).

In the wave base module the user has four options for specifying the directional distribution:

- Cosine2s spreading, user defined σ_c, constant with frequency.
- Wrapped normal spreading, user defined σ_c, constant with frequency.
- Wind sea directional distribution of Ewans (1998), σ_c varies with frequency.
- Swell directional distribution of Ewans (2001), σ_c varies with frequency.

Further options may be introduced at a later date where the user can specify a directional distribution where the mean direction and spreading varies as an arbitrary function of frequency.

4.1.2 Loading spectra to the wave module: the .SM file

Spectra can be loaded into the wave module in a from an ASCII format text file with the extension .SM (SM stands for spectral matrix). The SM file contains header lines describing the frequency and directional resolution, range and units, followed by a list of spectral densities. An example of the first few lines of an SM file is shown below:

```
startfreq = 0.01
freqstep = 0.01
endfreq = 0.5
funit = Hz
startdir = 0
dirstep = 15
enddir = 345
dunit = deg
specdensity =
s11 s12 s13 ...
s21 s22 s23 ...
...
```

From the header lines it can be seen that the file has spectral densities at 50 frequencies and 24 directions, but only the first few entries are shown here. The entry *Sij* corresponds to the spectral density at the frequency *startfreq* + (*i*-1)×*freqstep* and the direction *startdir* + (*j*-1)×*dirstep*.

Unidirectional spectra can also be saved in this format by setting *dirstep* to zero and *startdir* and *enddir* to the same value.

The field 'funit' can either be entered as 'Hz' or 'rad/s'. Similarly, the field 'dunit' can be entered as 'deg' or 'rad' depending on whether degrees or radians have been used to specify directions. It is assumed that directions are always specified as nautical bearings, with the angle measured clockwise from North and angles corresponding to the direction that the waves are coming from.

When a standard shape spectrum is generated by the wave base module or calculated from raw data by one of the add-on modules, it is saved in an SM file.

4.1.3 Simulation of the sea surface from spectra: the .SEA file

The output of the wave module to FD and TD code is a file listing a series of wave amplitudes, frequencies, directions and phases which make up the sea surface in an ASCII format text file with the extension .SEA. The contents and format of the .SEA files are described at the end of this section, but first it is worth making some notes on how the free-surface elevation is simulated from a spectrum.

When simulating waves from a given spectrum, whether in a numerical simulation or in a laboratory wave tank, there are several issues to consider. It is usually assumed that the phases are random with uniform distribution in [0 2π], but the wave amplitudes and directions can either be chosen randomly or deterministically. The pros and cons to each method are discussed below. Unidirectional aspects are considered first and directional aspects are considered at the end of the section. First, the basic equations for simulating the linear free-surface elevation are presented.

Suppose that it is desired to simulate unidirectional waves for a duration L at a time step of $dt = 1/f_s$, where f_s is the sampling frequency. The total number of samples generated will be $N = L \times f_s$. The free-surface elevation can then be simulated as

$$\eta(t) = \sum_{n=0}^{N-1} a_n \cos(2\pi f_n t + \phi_n) \qquad [4.25]$$

where a_n, f_n, and ϕ_n are the amplitude, frequency and phase of the n^{th} component. The summation is usually performed as in inverse FFT. Phases are selected randomly from a uniform distribution over $[0\ 2\pi]$. The frequency of the n^{th} component is given by $f_n = nf_s / N$. This ensures that the n^{th} wave component repeats exactly n times within the duration of the simulation and the repeat period of simulated waves is L.

Amplitudes can either be set deterministically as:

$$a_n = \sqrt{2E(f_n)df}\ , \qquad [4.26]$$

where $df = 1/L$, or selected as random variables from a Rayleigh distribution with an RMS value of $\sqrt{2}\sigma_n = \sqrt{2E(f_n)df}$:

$$a_n \sim Rayleigh(\sigma_n)\ . \qquad [4.27]$$

where $\sigma_n = \sqrt{E(f_n)df}$. Using deterministic amplitudes ensures that the spectrum of the simulated waves is exactly equal to the input spectrum, and consequently all spectral parameters of the simulated waves exactly match their target values. However, this neglects the fact that waves in the ocean are subject to sampling variability, i.e. for a theoretically stationary sea state, the wave spectrum measured in two locations over a finite time will differ, with the difference decreasing to zero as the sampling period increases to infinity. Using random amplitudes given by Equation 4.27 correctly reproduces this effect (see e.g. Tucker et al, 1984).

For short record lengths, the distributions of zero-crossing wave heights and periods produced by the deterministic method are restricted compared to the random amplitude method. However, as the record lengths increases the wave height and period distributions from the two methods converge. Elgar et al (1985) conducted simulations with random and deterministic amplitudes and compared various group statistics such as mean run length, variance of run length, distribution of the number of waves per group. They showed that the differences are negligible when more than 1000 frequencies are used but that there are some small differences for shorter runs. They also noted that the wave statistics from the two methods converge faster for broad-banded spectra.

Increasing the sampling frequency of the output signal, f_s, increases the maximum frequency used in the simulation and therefore the total number of frequencies used. However, in the method described above the frequency step is fixed at $df = 1/L$, thus increasing f_s only adds higher frequency components to the sum, which have little affect on the results since they typically contain little energy. So for a given spectrum, the convergence between the two methods is effectively determined by the simulation length L alone.

The differences between random and deterministic methods have been discussed in depth by Funke and Mansard (1987) and Huntington (1987). The main argument for the use of random amplitudes is related to extreme responses. Advocates of the deterministic amplitude method argue that the response of a device to an extreme condition can be tested by using record containing a large wave group. Whereas advocates of the random amplitude method point out that the extreme response will not necessarily be caused by the largest wave group. However, Huntington (1987) notes that it is more appropriate to frame the argument in terms of simulation length rather than random versus deterministic amplitudes, since for shorter run lengths there is a high chance that extreme events could be missed using either random or deterministic amplitudes and for longer run lengths the wave statistics from the two methods converge.

For performance assessment, the choice between the two methods therefore seems to be entirely dependent on whether the user is interested in sampling variability or not. It should be noted here that there are potentially two sources of sampling variability in the WEC response, one resulting from the amplitudes and the other resulting from the phases. If the WEC is completely linear then the response will not be dependent on the phases used, i.e. for any realisation with deterministic amplitudes the simulated response will be the same. However, if there are nonlinearities in the WEC response, either due to nonlinear hydrodynamics on nonlinear applied forces (PTO and moorings), then the different realisations with deterministic amplitudes and random phases may result in different average power. In the FD code all device responses will be linear, and nonlinearities will only be considered in the TD approach.

Simulation of the free-surface elevation under directionally spread waves involves several subtleties not involved in the unidirectional case. The method presented in Appendix A of WG1 WP1 D1b is known as the double sum method and is the logical extension to Equation 4.25. However, the free-surface elevation obtained from this method is not ergodic and not spatially homogeneous (i.e. the temporal distribution of the surface elevation varies with location). This is a result of phase-locking, where the directional components at each frequency are in constant phase-shift to each other and therefore create a spatial pattern of nodes and anti-nodes (see Jefferys (1987) for a detailed discussion of this effect). The non-homogeneous wave field simulated using the double summation method could potentially be confused with array-interaction effects, so the double sum method has not been implemented.

Instead, directionally spread waves are simulated using a single summation method, where the direction varies at each frequency. The surface elevation at location (x, y) and time t is given by

$$\eta(x,y,t) = \sum_{n=1}^{N} a_n \cos\left(2\pi f_n t + \phi_n + k_n (x\cos\theta_n + y\sin\theta_n)\right) \qquad [4.28]$$

Amplitudes, frequencies and phases are chosen as before (with the option for either random or deterministic amplitudes). Here k_n is the wavenumber at frequency f_n given by the linear dispersion relation $\omega_n{}^2 = gk_n \tanh(k_n h)$, where $\omega_n = 2\pi f_n$ and h is the water depth. The direction at frequency f_n is denoted θ_n and is a random variable from the distribution $D(f_n,\theta)$. The number of directions M in a frequency interval Δf is given by $M = \Delta f \,/\, df = L\Delta f$ and is therefore entirely determined by the simulation length.

The sea surface elevation calculated for a given spectrum is saved as an ASCII format text file with the extension .SEA. This file contains header lines describing the spectrum file from which the surface elevation was simulated and whether random or deterministic amplitudes were used. The FD code calculates the wave excitation force as a sum of the force from the individual components, so these are listed rather than the sea surface elevation with time. The first few lines of the file are shown below:

```
source = 'filename'
duration = 1800
amplitudes = rnd / det
waves =
f1 a1 d1 p1
f2 a2 d2 p2
```

```
f3 a3 d3 p3
...
```

The units of the wave components listed is

- Frequency in Hz
- Amplitudes in meters
- Directions in degrees (clockwise from North)
- Phases in degrees

For the FD code all simulations are based on regular waves, so the phase information is ignored. Results for irregular waves are calculated as the summation from regular wave contributions. However, the phase information is included as it will be used in the TD simulations. The duration of the simulated waves is also listed (with value in seconds) so that the repeat period is not exceeded in the TD simulations.

4.2 Current status of wave climate parameterisation

In WG1 WP1 D1b the trade-off between the accuracy in describing a wave climate and the number of parameters used was discussed. Using a larger number of parameters to describe a wave spectrum generally results in a more accurate description, but is also likely to result in a more complex description of the wave climate, needing a larger number of cases to cover the range of sea states described.

The approach taken in PerAWaT has been to investigate the use of standard spectral models, described in Section 4.1.1, and their accuracy for describing the range of spectra at a site. This approach involves partitioning the spectra into component wave systems, fitting standard shape spectra and quantifying the goodness-of-fit. The initial results are described below.

4.2.1 Partitioning wave spectra

As discussed briefly in Section 4.1.1, nonlinear quadruplet interactions within the spectrum cause wind sea spectra to tend towards ‘preferred’ shapes. This means that they can often be described reasonably accurately by standard unimodal shapes. But as soon as the wind drops or turns or the waves move out of their generation area then the influence of nonlinear interactions decreases and the spectral shape depends on the history of the individual components. This normally results in spectra with multiple peaks, composed of wind sea components together with one or more swell systems, which are not well modelled by unimodal shapes.

Identifying the cases which are composed of multiple components requires the use of a partitioning algorithm. This is essentially a set of rules to define what constitutes a separate system and the peaks which are just noise within the spectrum. This is always a subjective process. Some methods incorporate knowledge of the local wind speed and direction to separate wind sea components from swell, but this information is not always available.

Portilla et al. (2009) presented a review of partitioning methods which use information from the wave spectrum alone. The authors group the methods into two categories, depending on whether directional aspects are considered or not. The non-directional methods use only the shape of the frequency spectrum to identify separate systems, whereas directional methods also take into account the separation between the directions of adjacent peaks. For both types of method rules are implemented to decide whether peaks which have been identified are significant or belong to the same system. The most common checks are:

- The total energy of the partition in relation to the total H_s and/or the peak frequency of the partition.
- The minimum spectral density between adjacent peaks in relation to the spectral density of each peak.
- Separation of peaks in frequency and/or direction in relation to the bandwidth of the partition.

Both directional and non-directional algorithms have been trialled. Several directional algorithms include a check in the frequency domain to group peaks where the minimum spectral density between adjacent peaks is greater than a percentage of the spectral density at the lower peak (e.g. Hanson and Phillips, 2001). These methods therefore do not identify any partitions which would not be identified by a non-directional model. Moreover, non-directional methods tend to be much less computationally demanding to run. Therefore a non-directional partitioning algorithm has been implemented at present. Further directional checks may be implemented at a later date for spectra for which only one system is identified and which are not-well modelled by unimodal shapes. The criteria used to define separate partitions have been established based on trial and error and appear to give reasonably robust results. The criteria used are:

- The minimum spectral density between two peaks must be less than 65% of the spectral density of both adjacent peaks.
- The total energy of each partition must be greater than 5% of the total energy and have an H_s greater than 0.25m.
- All peaks with frequency greater than 0.3Hz are grouped together (these are all assumed to be wind sea).
- Partitions with peak periods less than 1.5s apart are grouped together.

Where wave systems overlap in frequency or direction it can be difficult to distinguish between distinct systems using automated procedures. However, the purpose of the partitioning algorithm used here is to identify when spectra cannot adequately be described by a unimodal spectrum, rather than to identify separate systems. In cases where two systems are close in frequency and/or direction, then representing the resulting spectrum with a unimodal shape may not be too serious a simplification. However, there are likely to be cases of overlapping systems where which are neither clearly defined nor well modelled by unimodal spectra.

The aim of the partitioning and fitting is to investigate the distribution of the parameters of the fitted spectra in order to describe the wave climate. If the fitted spectra have a poor correspondence to the measured spectra the describing the distribution of the fitted parameters may be a poor description of the wave climate. Therefore a key metric to quantify is the accuracy of the fitted spectra, this is discussed in Section 4.2.3. First the fitting procedure is described.

4.2.2 Fitting standard models to the frequency spectrum

Fitting routines have been implemented for the three families of frequency spectra described in Section 4.1.1: Ochi spectra, Gamma spectra and JONSWAP spectra (all of which include the Pierson-Moskowitz and Bretschneider spectra as special cases). The algorithm begins by using a simplex search technique to find the parameters which minimise the squared distance between the measured and fitted spectra:

$$\varepsilon = \int_{f_0}^{f_1} \left(E_m(f) - E_f(f) \right)^2 \mathrm{d}f\,, \qquad [4.29]$$

where E_m is the measured spectrum, E_f is the fitted spectrum and f_0 and f_1 are the lower and upper limits of the partition. If more than one partition has been identified these parameters are then used as a first guess for the total spectrum, which is formed as a sum of the spectra for each partition.

4.2.3 Goodness-of-fit statistics

The goodness-of-fit of the resulting spectrum is defined in as the absolute difference between the measured and fitted spectrum, normalised by m_0:

$$\Delta = \frac{1}{m_0} \int_0^\infty \left| E_m(f) - E_f(f) \right| \mathrm{d}f \qquad [4.30]$$

Due to the uncertainties related to the method used to estimate the directional spectrum (see Appendix B of WG1 WP1 D1b), only the accuracy of model-free parameters has been considered. The goodness-of-fit parameters for the mean direction and spread are defined as:

$$\delta(\theta) = \frac{1}{m_0} \int_0^\infty E(f) \left| \theta_m(f) - \theta_f(f) \right| \mathrm{d}f \qquad [4.31]$$

$$\delta(\sigma) = \frac{1}{m_0} \int_0^\infty E(f) \left| \sigma_m(f) - \sigma_f(f) \right| \mathrm{d}f \qquad [4.32]$$

4.2.4 Initial results

Test cases have been run using data recorded by buoys operated by the US National Data Buoy Centre (NDBC), which is available to download from ftp://ftp.nodc.noaa.gov/pub/f291/. To illustrate the results, wave measurements from a buoy off the coast of Oregon (USA), are considered. The buoy (NDBC number 46029) is located approximately 35km from the coast in a depth of 135m. The data covers the period 03/1984 - 07/2010 and consists 145,895 hourly records. All the spectra have been used in the analysis in order to gain as full a picture of the range of sea states as possible. The length of data required to estimate the long-term climate will be considered after the method to describe the climate has been defined. At present only non-directional aspects have been considered.

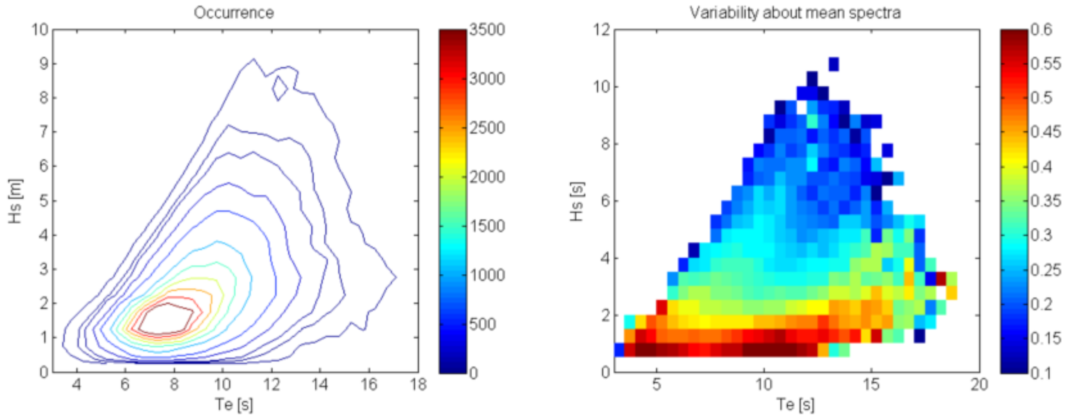


Figure 4.4. Left: occurrence of sea states in bins of width 0.5m by 0.5s. Right: Variability of spectral shapes about mean spectral shape, binned by H_s and T_e.

Figure 4.4 shows the occurrence of sea states binned by H_s and T_e, contours are shown for 10, 25, 50, 100, 250, 500 and then at intervals of 500 records. The right hand plot shows the variability about the mean spectrum, defined as the mean value of Δ in each bin, with the mean spectrum of each bin used in place of a fitted spectrum in Equation 4.30. It is clear that the highest variability in spectral shape occurs for the lowest sea states, with the spectra tending towards more uniform shapes with increasing H_s.

Each spectrum has been analysed as follows:

- The partitioning algorithm is run to identify up to three partitions.
- A Bretschneider spectrum is fitted (2 free parameters).
- A JONSWAP spectrum is fitted (3 free parameters).
- If two or more partitions are identified a spectrum formed as the sum of two JONSWAP spectra is fitted (6 free parameters).
- If three or more partitions are identified a spectrum formed as the sum of three JONSWAP spectra is fitted (9 free parameters).
- Goodness-of-fit statistics are calculated for each of the fitted spectra.

To quantify the differences in accuracy between using various parameterisations of the spectrum, the mean value of Δ can be compared. Figure 4.5 shows the ratio of the mean value of Δ, binned by H_s and T_e, for various parameterisations. The top left plot shows the difference in accuracy when using the Bretschneider spectrum (2 free parameters) compared to the mean spectrum (also a 2 parameter spectrum, as this is the mean shape for a given H_s and T_e). The ratio of the mean value of Δ is always greater than one, indicating that the mean spectrum in a bin is a better description on average than the Bretschneider spectrum fitted to each individual spectrum. Interestingly, the difference between the mean spectrum and Bretschneider spectrum is lowest for the steeper sea states. The top right plot shows the relative improvement from the Bretschneider spectrum by introducing the peak enhancement factor as a third further free parameter. The ratio of the mean value of Δ is everywhere less than one, with improvements of 25% in some bins. The improvements are smallest for the wind-sea dominated sea states. The lower left plot shows the ratio of the mean value of Δ between using the standard JONSWAP spectrum (3 free parameters) and spectra formed as the sum of two JONSWAP spectra (6 free parameters). This is the mean value over all spectra, not only those with two or more partitions were identified. In cases where only one partition was identified the spectrum is fit with just one JONSWAP spectrum so there is no difference between the two methods and the ratio of Δ in these instances is one. Including the cases where there was only one partition identified gives an indication of the relative importance of modelling cases with more than one peak. The pattern of relative improvements roughly follows the pattern of variability of spectral shapes, shown in Figure 4.4. Finally, the lower right plot shows the relative improvement in accuracy by introducing a third JONSWAP partition, to give 9 free parameters. Again, the mean is taken over all spectra, so the ratio of Δ in these instances where less than 3 partitions were identified is one. The improvement for sea states with H_s < 2m is typically less than 1% and negligible above this, indicating that the relative occurrence of spectra with three or more peaks is very low and on average they are not important.

The next steps in the investigation of methods to parameterise the wave climate are:

- Compare the accuracy of fitting spectra with Ochi and Gamma spectra with JONSWAP spectra.
- Assess accuracy of models for mean direction and spread.
- Examine distribution of fitted parameters to find efficient description of the wave climate.
- Determine length of measured data required to estimate distribution of spectral shapes.
- Quantify effect of various descriptions of the wave climate on accuracy of WEC yield prediction.

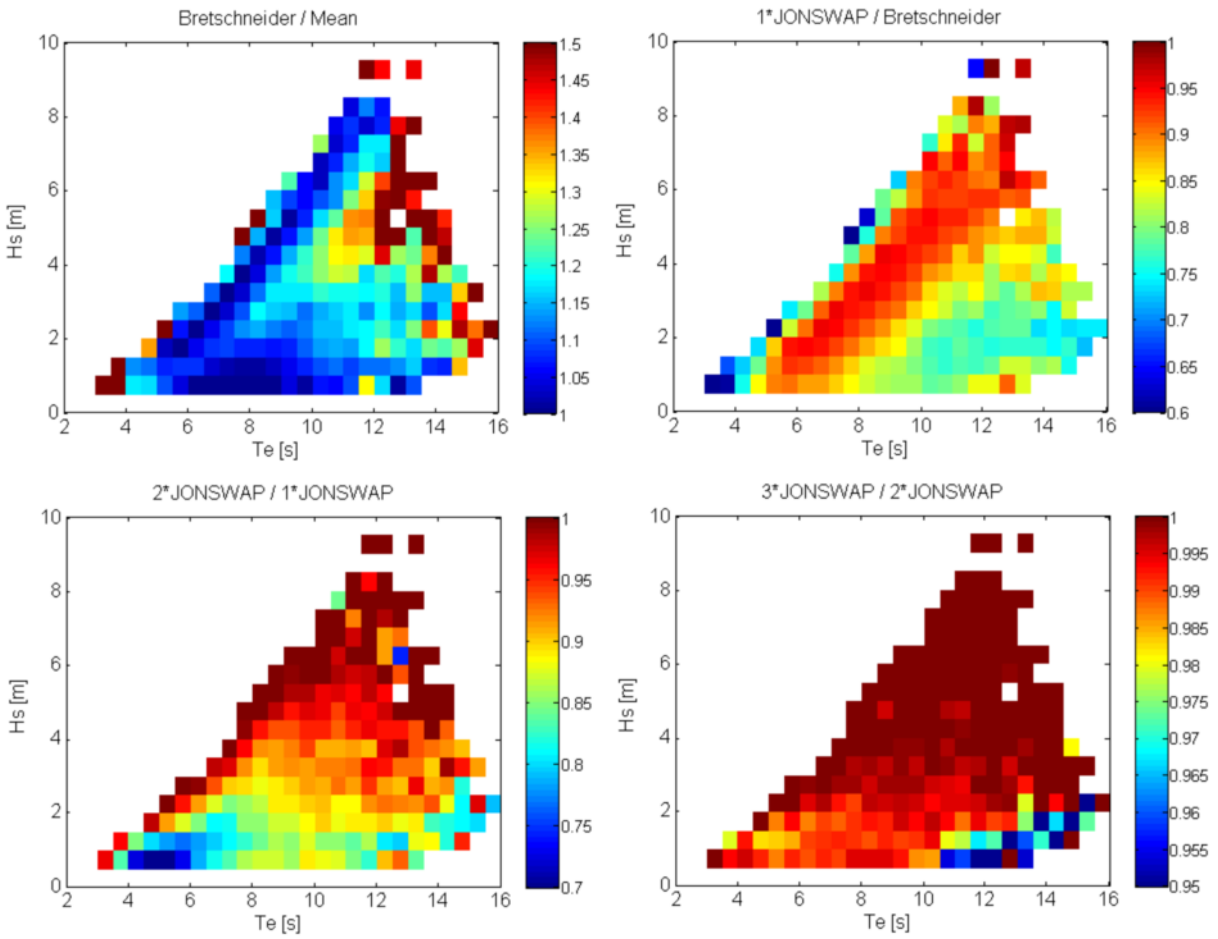


Figure 4.5. Ratios of the mean deviance, Δ, for various spectral parameterisations.

5 MOORINGS MODULE

5.1 Overview of the approach

To better understand the methodologies and commercial options to model mooring configurations a survey of the available software tools which might be suitable for modelling the behaviour of the moorings of a floating WEC was conducted. A list of candidates was drawn up and the most suitable options assessed, taking into account the associated computational effort, flexibility and ease of adoption for PerAWaT's purposes. As well as comparing tools, the several types of modelling approaches were also compared. The aim of this exercise is to obtain a clear view of the modelling approaches that are applied in commercial packages and their accuracy, in order to further develop them and meet the project requirements by developing a moorings module to be coupled with the main WEC simulation.

Table 5.1shows a list of various candidate tools / packages which currently exist, along with a brief account of their properties. The tools shown in bold are those which have been investigated in detail by GH. These four were chosen for detailed investigation mostly because:

- ROMEO was developed within the GL group, so GH has access to the source code.
- MDD is open-source so can be altered and used freely.
- AQWA and OrcaFlex are both widely used within the offshore engineering community.

This approach has allowed GH to derive some key findings regarding the development of a moorings module under PerAWaT. These key findings can be summarised as follows:

- Having investigated MDD, ROMEO, AQWA, and OrcaFlex, it is concluded that of these four only AQWA and OrcaFlex are likely to have suitable functionalities in their current state, given that:
  - MDD has no wave-loading or dynamic load capability.
  - ROMEO is suitable only for frequency-domain modelling. A time-domain model will also be required for PerAWaT, so this makes ROMEO only a partial solution. A single tool that covers both frequency- and time-domain modelling might be preferable.
- The modelling approach to be adopted initially involves using look-up tables to represent the damping/inertia/stiffness properties of the mooring configuration. This is an established approach which is also used in ROMEO and AQWA.
- A third-party tool such as OrcaFlex or AQWA may be useful in compiling the information to populate the look-up tables before starting a simulation in the GH model. This should be seen as a preliminary step which can also be completed with a detailed bibliographic search, to allow the user to have default choices (in addition to the introduction of custom parameters).
- If necessary, GH will explore the option of a runtime link to (e.g.) AQWA or OrcaFlex so that a commercial mooring analysis package can be fully integrated into a GH time-domain simulation run, at the expense of additional computational effort. This option will be investigated under WG1 WP1 D3.

Tool / package / code	Main characteristics
ANSYS AQWA	Consists of several sub-components. AQWA-NAUT, AQWA-DRIFT, AQWA-FER, …, each with specific properties (e.g. time-domain analysis, frequency-domain analysis)
Martec WaveLoad	Intended for ships. Frequency- or time-domain. Mooring line analysis available in FD only.
Flexcom	Strong focus on riser analysis; limited range of mooring options
Ariane-3Dynamic	Specialised mooring tool. Restricted water depth.
Optimoor	Ship-shaped bodies only; limited types of analysis possible. (Focussed on extreme-load analysis for mooring lines).
MDD	Open-source, static solutions only, no wave loads. See Section 5.5.2.
ROMEO	GL Noble Denton mooring and riser package. Frequency-domain only. See Section 5.5.3.
GMoor32	Catenary mooring and riser analysis
BMT SPM	Quasi-static analysis (plus transient analysis for line breaks only).
SEAMOOR 2000	Operational tool for station-keeping / line-break analysis. Limited modelling options.
TERMSIM	Tanker mooring simulation. Limited modelling options.
MOSES	A modelling language for marine dynamic simulations and stress analysis.
OrcaFlex	Marine dynamics tool, widely used by the offshore engineering community.

Table 5.1: Overview of selected moorings modelling packages

5.2 Mooring design and Dynamics (MDD)

Mooring Design and Dynamics (MDD) is an open-source MATLAB package produced by Richard Dewey at the University of Victoria, British Columbia. It was written with oceanographic research applications in mind. Specifically, it helps with the design of anchor and mooring systems for stationary oceanographic instruments, and can also perform calculations for moving (i.e. towed) instruments. MDD takes current and wind profiles into account, but does not deal with wave forces. Immediately this indicates that a considerable development effort would be necessary to adapt MDD; the main attraction to such a tool is therefore the fact that it is open-source.

5.2.1 Using MDD

Basics

MDD consists of a package of MATLAB files and is run from the MATLAB command-line. It allows the user to create / modify mooring or tow-line configurations. User interaction is done via a simple GUI (see Figure 5.1). Designs can be saved and loaded as .MAT files, and displayed as MATLAB 3-D or 2-D graphics. The package also allows the creation of movie sequences so that the dynamic behaviour of a mooring can be visualised. Mooring configurations can be plotted, as shown in Figure 5.2, and the components are also listed in detail in the MATLAB console window.

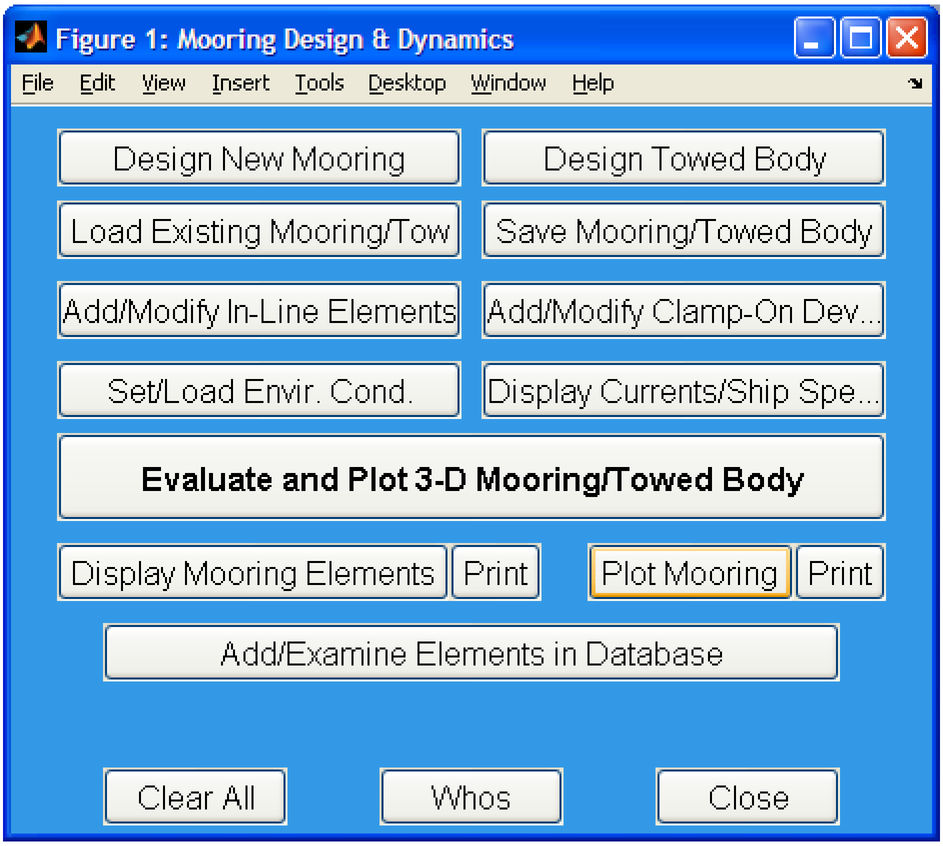


Figure 1: Mooring Design & Dynamics
File Edit View Insert Tools Desktop Window Help
Design New Mooring
Design Towed Body
Load Existing Mooring/Tow
Save Mooring/Towed Body
Add/Modify In-Line Elements
Add/Modify Clamp-On Dev...
Set/Load Envir. Cond.
Display Currents/Ship Spe...
Evaluate and Plot 3-D Mooring/Towed Body
Display Mooring Elements
Print
Plot Mooring
Print
Add/Examine Elements in Database
Clear All
Whos
Close

Figure 5.1: MDD's main GUI form.

The user can also enter current-depth profiles (i.e. a set of depth/speed/direction values), and wind profiles (speed and direction). The software uses this information to calculate the drag forces on the various elements of the mooring, in order to derive a steady-state solution for the mooring shape.

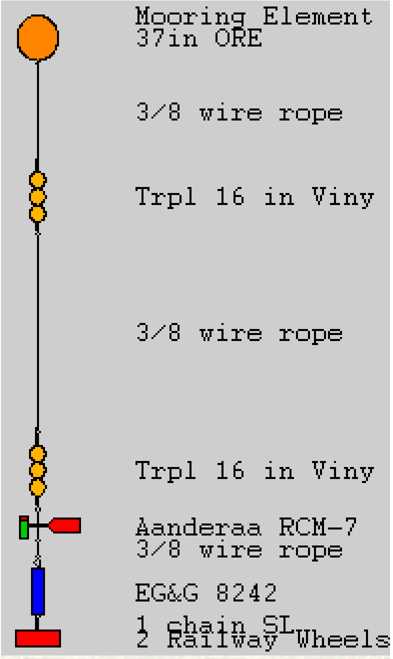


Figure 5.2: Typical mooring configuration (design view) in MDD.

MDD takes an iterative approach; it starts from a vertical configuration for the mooring and iteratively "moves" the mooring horizontally until a steady-state solution is reached. It can deal with time-varying inputs, i.e. a time-series of current-depth profiles can be provided as an input. When handling these time-series, it is assumed that the mooring has time to reach an equilibrium state between time-steps. In other words, the variation in the current is slow compared to the response time of the mooring. There are currently no detailed plans to extend MDD to include the action of wave forces.

Outputs

MDD can produce any of the following outputs:

- Graphical plots of the steady-state shape of the mooring;
- Movie sequences showing the changing shape of the mooring in response to a time-varying current;
- Tension in each segment of the mooring;
- Final steady-state position of each component (horizontal and vertical);
- Final angle of each component to the vertical;
- Anchor mass required;
- Adequacy of buoyancy elements in maintaining vertical position of the mooring.

5.2.2 Key findings - MDD

MDD is designed as a stand-alone, self-contained mooring design package. It is not designed to interact dynamically with other software. In principle, as the MDD m-files are freely available, it would be possible to modify it so that it could be used as a mooring-force calculation module for the GH model, but this would not be straightforward. Also it only produces what are effectively steady-state solutions, which are suitable for slowly-varying forces like those from wind or currents, but will not be appropriate for wave-induced forces.

MDD as it currently exists is not suitable as a mooring analysis module for interfacing with the GH model. Its most serious drawback is its inability to carry out a true dynamic analysis of moorings subjected to rapidly-varying (wave-induced) loads, which would require a considerable extension. The principles of the approach are similar and some components (e.g. drag force calculation algorithms) and some of the post-processing capabilities may be useful when developing the GH moorings module (in both the frequency and time-domain).

5.3 ROMEO

ROMEO is a mooring and riser analysis package produced by GL Noble Denton. It can perform both static and dynamic (frequency-domain) analysis in response to wind, wave and current loadings. Systems of up to 16 mooring lines are supported. Riser calculations are ignored here.

5.3.1 Functionality of ROMEO

ROMEO carries out three types of analysis on a mooring system: static analysis, dynamic analysis (both wave-frequency and low-frequency) and then a quasi-static analysis which is a combination of the first two.

The inputs for all analyses include the details of the moored device or ship, “no-load” mooring geometry, and environmental conditions.

Static analysis

The environmental inputs for static analysis are steady wind and current forces. It is possible to enter current as a depth-dependent velocity profile. The algorithm uses Morison’s equation to calculate the drag on the mooring lines. This equation gives the inline force (i.e. force in the direction of the flow) on the mooring line element as

$$F = \rho V\dot{u} + \rho C_a V(\dot{u} - \dot{v}) + \frac{1}{2}\rho C_d (u - v)|u - v| \qquad [5\text{-}1]$$

(i.e. the sum of Froude-Krylov force, hydrodynamic mass force, and drag force)

where

ρ	= density of water;
C_a	= added-mass coefficient;
A	= cross-flow area of the mooring line element;
V	= volume of the mooring line element;
C_d	= drag coefficient;

u = flow velocity;

v = line element's velocity.

It also calculates wind and current forces on the ship (or the WEC) using a simple drag equation. The wind and current calculations require a cross-flow area, height coefficients and shape coefficients to be supplied for the body in question:

$$\begin{aligned} C_{curr} &= \sum_{i=1}^{N} C_{si} A_i \\ C_{wind} &= \sum_{i=1}^{N} C_{hi} C_{si} A_i \end{aligned} \qquad [5\text{-}2]$$

where C_{curr} and $\mathbf{C_{wind}}$ are force coefficients, C_s are shape coefficients, C_h height coefficients and A_i are cross-flow areas. The force coefficients are used to calculate the drag forces as follows:

$$\begin{aligned} F_{curr} &= C_{curr} V_{curr}^2 \\ F_{wind} &= C_{wind} V_{wind}^2 \end{aligned} \qquad [5\text{-}3]$$

where V_{wind} and V_{curr} are wind and current flow speeds. The force coefficients are loaded in as part of the vessel definition file, so they have to be obtained externally. They are typically obtained either by manual estimation or, if higher accuracy is needed and there is no published data available for similar-shaped objects, they would be scaled-up from model tests.

The static analysis outputs consist of:

- steady-state shapes of mooring catenaries.
- steady-state (mean) position of the moored device.

Dynamic and quasi-static analysis

Dynamic analysis is carried out in the frequency-domain only. The following parameters are used to characterise the wave environment:

- significant wave height.
- zero up-crossing period.
- spectral shape factor (peak enhancement factor, γ).
- incident wave direction.

The user manual draws a distinction between wave-frequency and low-frequency analysis. The low-frequency effects are driven by both time-varying wind and longer-period wave activity

The outputs from dynamic analysis are:

- amplitude of the wave-induced motion of the moored vessel/device.
- maximum dynamically-induced tensions in the mooring lines.

Dynamic analysis is run after static analysis – the static phase gives the steady-state positions of all components, and this position in then used as a starting-point for the dynamic analysis. Combining the static and dynamic results produces what is effectively a "quasi-static" analysis, i.e. a worst-case scenario in which the maximum dynamic displacement is added to the static displacement. This is primarily intended as a safety calculation to determine the minimum required breaking strain for mooring lines.

5.3.2 Key findings - ROMEO

The most serious limitation of ROMEO is the lack of any capability to conduct a time-domain dynamic analysis. This means that only frequency-domain simulations could be conducted. It may, however, be proven while developing the software that a full time-domain solver is not necessary if the moorings package is to be used merely to establish the mooring configuration properties, as discussed in Section 5.6. As a frequency-domain tool ROMEO offers all the suitable functionalities and thus is a good template when developing the GH capability in this domain while keeping in mind the potential to share the new formulation with the time-domain module.

5.4 AQWA

The ANSYS AQWA brand name covers an entire suite of tools:

- The AQWA Graphical Supervisor – this is the GUI which is used to control operations and allow the user to interact with other tools in the suite. It includes graphical renderings of the system being modelled, plotting facilities for output data, export to spreadsheets, etc.
- AQWA-LIBRIUM – for steady-state equilibrium load calculations.
- AQWA-LINE – for wave loading calculations (regular waves only).
- AQWA-FER – wave loading in irregular waves (frequency-domain only).
- AQWA-NAUT – time-domain analysis of floating body (or several bodies) in either regular or irregular waves. Includes several different types of basic mooring-line model and a cable dynamics module as an optional add-on.
- AQWA-DRIFT – time-domain modelling of long-period loads under irregular wave conditions. Complementary to AQWA-NAUT which deals with the wave-frequency loads.
- AQWA-WAVE – A link between AQWA-LINE and external FEA packages.

5.4.1 AQWA-LINE

AQWA-LINE can be used as a stand-alone tool for some types of wave-loading calculations, or as a pre-processor which calculates hydrodynamic coefficients for use by other products in the AQWA suite. The main analysis technique used in this code is Radiation/Diffraction theory. It can calculate both wave-frequency forces and second-order drift forces in regular waves. The Quadratic Transfer Function (QTF) matrix for second-order drift forces can also be passed on to AQWA-DRIFT for a full time-domain analysis in irregular waves.

AQWA-LINE only handles regular waves, but can take account of current and wind forces as well. It calculates the wave response of the structure in the frequency-domain and stores them as a set of RAOs for a range of different frequencies. These can be subsequently used by other tools such as AQWA-NAUT.

5.4.2 AQWA-NAUT

Overview

AQWA-NAUT is a time-domain analysis program for modelling the motion of large floating bodies in ocean waves. It deals with both regular and irregular waves, and can also include the effects of current and wind in the model. AQWA-NAUT does not calculate the effects of wave drift forces – AQWA-DRIFT is required for that. It can be used as a stand-alone tool or with AQWA-LINE as a pre-processor. It solves the second-order differential equations of motion and uses a "two-stage predictor-corrector" integration scheme to derive a time-history from the solutions. Being originally developed for the oil & gas industry, there are no specific PTO or control modules.

It uses linear hydrodynamic coefficients for the various degrees of freedom at different frequencies, supplied by AQWA-LINE or an equivalent code, along with other hydrostatic and hydrodynamic information. In regular waves, RAOs are used to calculate the time-domain response of the device in waves of a given frequency, using the following relation for each degree of freedom:

$$x(t) = aXe^{-i\omega t + ikx_p} \qquad [5\text{-}4]$$

where:

X = complex RAO at wave frequency ω for this degree of freedom;

ω = angular frequency of the regular wave component

k = wave number (=2π/wavelength)

x_p = distance from the origin of the wave system perpendicular to the wave direction

a = amplitude of the regular wave component

$x(t)$ = instantaneous displacement at time t

The hydrostatic and Froude-Krylov forces are normally calculated by AQWA-NAUT itself, and used in conjunction with the linear coefficients to build up a complete envelope of the loadings and motions of the body. As part of the process, AQWA-NAUT requires a mesh of the body's surface to be built. Four different wave models are available for calculating the Froude-Krylov forces:

1. Linear wave theory – deep-water
2. Linear wave theory – finite depth
3. Second-order wave theory – deep-water
4. Second-order wave theory – finite depth

The Froude-Krylov force on a submerged body (e.g. an element of mooring line) is due to the unsteady pressure field produced by undisturbed waves, and is given by

$$\overline{F}_{inc} = -\int_{S(t)} p\overline{N}ds \qquad [5\text{-}5]$$

where:

$\overline{F}_{inc}$ = Froude-Krylov force;

S(t) = wetted surface of the body at an instant in time *t;*

p = instantaneous pressure at a point on the surface;

$\overline{N}$ = the outward normal vector of the body surface.

The unsteady pressure field value *p(x,y,z,t)* is calculated in different ways according to the type of wave model used.

For linear wave theory in deep-water, the relation is:

$$p(x,y,z,t) = \rho gae^{kz} \cos(k\chi - \omega t) \qquad [5\text{-}6]$$

For linear wave theory where the depth is finite, it is:

$$p(x,y,z,t) = \rho gae^{kz} \frac{1+e^{-2kz}e^{-2kd}}{1+e^{-2kd}} \cos(k\chi - \omega t) \qquad [5\text{-}7]$$

For second-order wave theory in deep-water:

$$p(x,y,z,t) = \rho ga[e^{kz} \cos(k\chi - \omega t) - \frac{1}{2}kae^{2kz}] \qquad [5\text{-}8]$$

For second-order wave theory with finite depth:

$$p(x,y,z,t) = \rho gae^{kz} \frac{1+e^{-2kz}e^{-2kd}}{1+e^{-2kd}} \cos(k\chi - \omega t) + \rho g \frac{ka^2e^{-2kd}}{1-e^{-4kd}} . \frac{6e^{2kz}(1+e^{-4kz}e^{-4kd})-1}{(1-e^{-2kd})^2} \cos 2(k\chi - \omega t)$$
$$- \rho g \frac{ka^2e^{2kz}}{2} . \frac{1+e^{-4kz}e^{-4kd}}{1-e^{-4kd}} \qquad [5\text{-}9]$$

where:

χ $= x \cos \theta + y \sin \theta$

θ = wave direction

k = wave number (=2π/wavelength)

ω = wave frequency

a = wave amplitude

d = water depth

ρ = density of water

g = acceleration due to gravity

t = time

AQWA-NAUT can model the interactions between floating bodies in an array (including radiation-coupling and shielding effects), for up to 20 bodies. This is a total number of bodies, thus if a WEC has e.g. 3 bodies, arrays of up to six WECs can be modelled. Furthermore the structural restraints between bodies are not standard elements, alongside the PTO and control aspects that in essence define a WEC. These elements can be approximated by modifications to the characteristics of standard elements available in AQWA, on a case-by-case basis. This direct inability to model a variety of specific WECs from a database of elements specifically developed for wave energy conversion partly justified the creation of WaveDyn (see Section 2), ensuring that a multi-purpose platform for the modelling of loads and performance variables is available.

Moorings

In the core code of AQWA-NAUT, moorings are modelled as either linear or nonlinear cables. Five linear cable models are available:

- Linear elastic cables.
- Winch cables.
- Constant force cables.
- Pulleys.
- Drum winch cables.

Of these five, the one which may be of most interest for modelling WECs is the linear elastic cable, assuming that winches, pulleys etc will not be used as frequently. The linear elastic cable is basically modelled as a linear spring. Constant-force cables may also be useful for highly simplified mooring models where cable details are not available. The following nonlinear cable models are also available:

- Nonlinear steel wire: allows modelling of the nonlinear properties of steel wire.
- Nonlinear elastic hawsers: These are treated as nonlinear springs, whose force-extension curve is represented by a polynomial (up to 5th order).
- Composite catenary lines: the lines themselves can be elastic, and the loading effects due to the catenary shape are also modelled.
- Clump weights and buoys: these can be added to the line model.

Wave and current forces on mooring lines are ignored in all the above types of mooring model, unless the “Coupled Cable Dynamics” module is used. This module has a dedicated user-interface window (see Figure 5.3), and allows the code to model mooring cables more realistically by calculating the drag and inertia (including added-mass) forces on the cable.

In the absence of the Cable Dynamics module, it is still possible to model the effects of wave and current drag on mooring cables in AQWA-NAUT, using the “Slender Tube” (STUB) Morison element. For both of these latter approaches, the cable is divided into a number of rod-type elements and the force on each element worked out separately. The hydrodynamic force on an element of cable, in the direction perpendicular to the element’s axis, is given by

$$dF = \frac{1}{2}\rho D C_d (u_f - u_s)\left|u_f - u_s\right| + \rho A C_m \dot{u}_f - \rho A (C_m - 1)\dot{u}_s \qquad [5\text{-}10]$$

(drag force) (wave force) (inertia force)

(The “inertia force” is effectively an added-mass term.)

where

C_d = drag coefficient

D = effective cable diameter for drag purposes

u_f = fluid velocity perpendicular to element axis

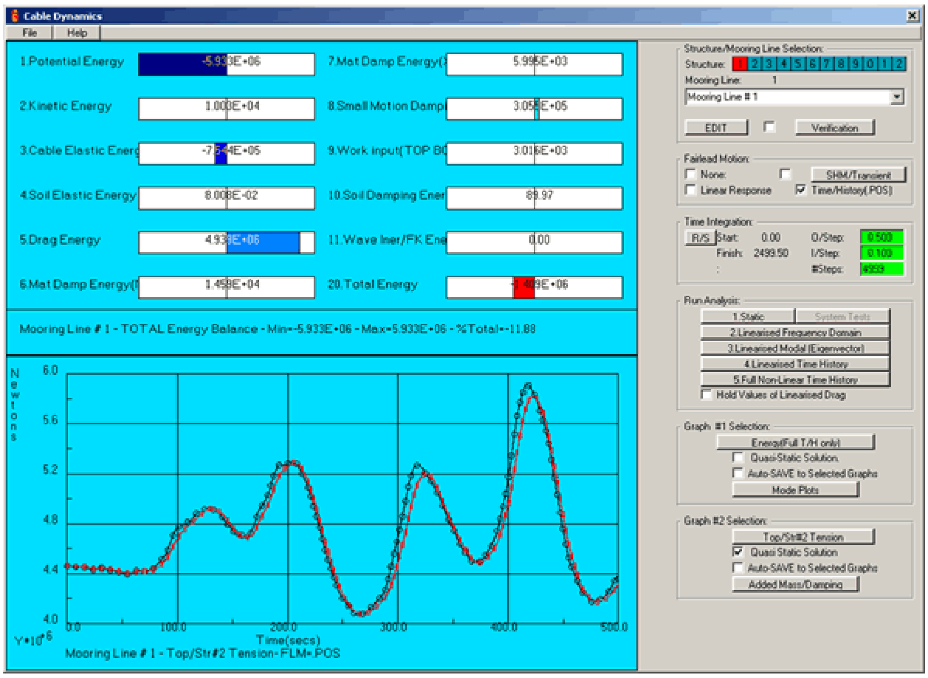
u_s = cable element’s own velocity perpendicular to element axis

C_m = inertia coefficient

A = effective cross-section area (for a chain which does not have a uniform cross-section, this is actually the volume of a unit length of cable)

ρ = water density

These hydrodynamic forces are combined with the other loads on the cable (inertial / gravitational / tension loading) and then forces on all the elements are used to model the overall dynamic behaviour of the cable.



Cable Dynamics

1.Potential Energy -5.933E+06
2.Kinetic Energy 1.000E+04
3.Cable Elastic Energ -7.544E+05
4.Soil Elastic Energy 8.008E-02
5.Drag Energy 4.939E+06
6.Mat Damp Energy(I 1.459E+04
7.Mat Damp Energy(1 5.995E+03
8.Small Motion Dampi 3.058E+05
9.Work input(TOP BC 3.016E+03
10.Soil Damping Ener 89.97
11.Wave Iner/FK Ene 0.00
20.Total Energy -1.409E+06

Mooring Line # 1 - TOTAL Energy Balance - Min=-5.933E+06 - Max=5.933E+06 - %Total=-11.88

Structure/Mooring Line Selection:
Structure: 1 2 3 4 5 6 7 8 9 0 1 2
Mooring Line: 1
Mooring Line # 1

Fairlead Motion:
□ None:
□ Linear Response
☑ Time/History(.POS)

Time Integration:
Start: 0.00 O/Step: 0.500
Finish: 2499.50 I/Step: 0.100
#Steps: 4999

Run Analysis:
□ Hold Values of Linearised Drag

Graph #1 Selection:
□ Quasi-Static Solution
□ Auto-SAVE to Selected Graphs

Graph #2 Selection:
☑ Quasi Static Solution
□ Auto-SAVE to Selected Graphs

Figure 5.3 The user interface window for the optional Coupled Cable Dynamics module in AQWA-NAUT.

Key findings - AWQA

AQWA has the capability to perform detailed simulations of mooring systems in both time and frequency domains. It is likely to be suitable for preparing lookup tables off-line for use in a WaveDyn simulation. As it is a stand-alone software suite, rather than a library, it is not suitable in its current form for full integration into the WaveDyn time-domain simulation. Such integration may not be necessary in any case. Should it prove necessary however, a commercial agreement with ANSYS for developing a DLL interface to AQWA would be required.

5.5 OrcaFlex

5.5.1 Overview

OrcaFlex is a dedicated "marine dynamics" analysis package produced by Orcina Software. It covers moorings and risers, as well as towing systems. OrcaFlex can carry out full modal, static and dynamic analysis in the time-domain. It supports a wide range of wave models as well as drift forces, wind drag, currents etc. The software has a Windows UI which includes animated or static 3-D graphical displays, including some quite realistic rendering. OrcaFlex can also be run in non-interactive (batch) mode.

There is an interactive click-and-drag type UI which allows the user to build up a model in a fairly intuitive way. Objects such as vessels, buoys, mooring lines etc. can be added from a library of predefined objects or custom-defined by the user. Environmental conditions (wind, waves, currents) can be similarly defined and added to the model. When a model has been completed a simulation is started. The behaviour of the model during the simulation can be visualised on the animated 3-D view window:

Output quantities (load, motions, etc) can be sampled at user-defined intervals during the simulation and output to a file. Outputs can be easily imported to Excel, etc. for further processing. OrcaFlex also has its own graphical display function.

5.5.2 Theoretical aspects

The code supports a wide range of different wave models, including regular and irregular waves (four standard wave spectra are supported), and both linear and nonlinear waves. Wave drift forces are also available (in contrast to AQWA).

Hydrodynamic forces on the mooring lines and the moored device are derived using an extended form of Morison's equation. Mooring lines are modelled on a finite-element basis; each line is divided into a number of straight-line elements each with a clump weight, buoyancy and drag. Figure 5.4 shows the coordinate frame of a line element. Each element is treated as a stiff massless rod with a telescopic sliding joint in the centre, so that it can change in length (and can also twist around its own axis). The user is given the option of specifying both stiffness and damping coefficients for the following degrees of freedom:

- Axial deflections (changes in the length of the rod element).
- Torsional deflections about the rod element's central axis. This means one end of the element twisting relative to the other end.
- Bending deflections (changes in the relative angle between the rod element and its immediate neighbouring elements).

N.B. The damping discussed above is only for the cable itself; external (hydrodynamic) damping is treated separately and is discussed below.

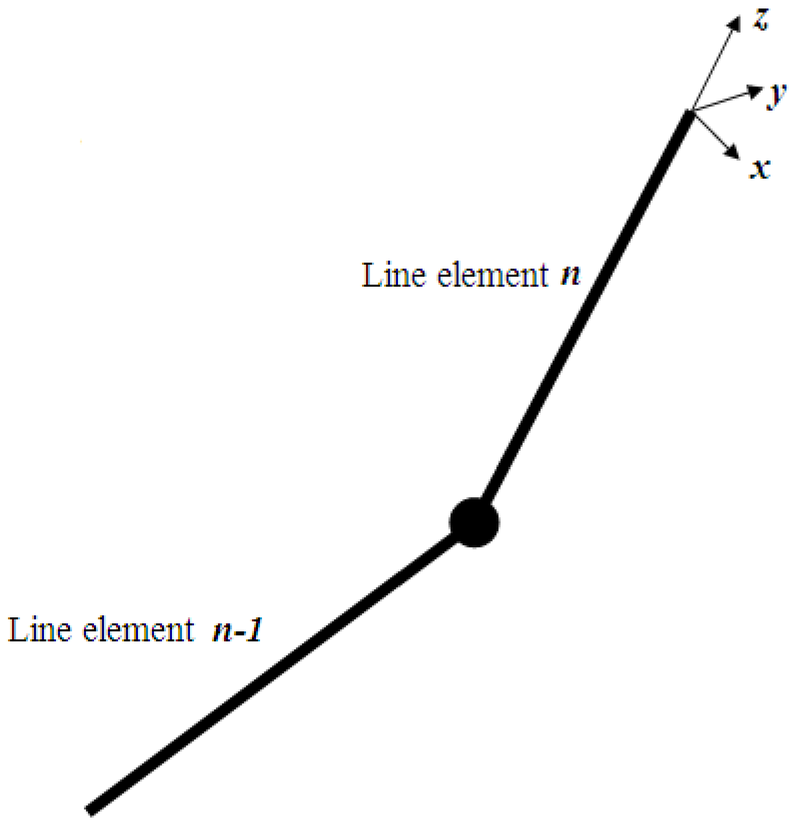


Figure 5.4 Line element representation in Orcaflex, showing the element's frame of reference.

The tension in each element of the line is given by the relation

$$T_e = EA.\varepsilon + EA.e\frac{(dL/dt)}{L_0} \qquad [5\text{-}11]$$

where

T_e = effective tension.

EA = axial stiffness of line (i.e. Young's modulus x effective cross-section area).

ε = mean axial strain = $(L-\lambda L_0)/(\lambda L_0)$.

L = instantaneous length of segment.

λ = expansion factor of segment.

L_0 = unstretched length of segment.

e = numerical damping of the line, in seconds.

Hydrodynamic drag forces on the line are found using a choice of three different relations, all based on Morison's equation. Drag forces are applied both in water and optionally also in air (wind forces). The same relations are used for both water and air, with appropriate values for fluid density of course. The three available relations are as follows:

- Standard formulation – the most commonly used; suited to general flow conditions.
- Pode formulation – preferred by some modellers for situations where flow is nearly tangential to the line.
- Eames formulation – sometimes preferred for bare (unsheathed) mooring cables.

The standard formulation is given by

$$\begin{aligned} F_x &= \frac{1}{2} P\rho D_n L C_{d_x} V_x |V_n| \\ F_y &= \frac{1}{2} P\rho D_n L C_{d_y} V_y |V_n| \\ F_z &= \frac{1}{2} P\rho D_n L C_{d_z} V_z |V_n| \end{aligned} \qquad [5\text{-}12]$$

where (all *x,y,z* subscripts are in the line element's frame of reference; see Figure 5.4 for coordinate system):

P = proportion of this element that is in water (for the hydrodynamic case) or air (for the wind-loading case).

F_x, F_y, F_z = drag forces in the *x,y,z* directions respectively

C_{dx}, C_{dy}, C_{dz} = drag coefficients of the cable element in the *x,y,z* directions.

D_n = effective diameter of the line, in the normal direction, for drag purposes.

V_n = component of fluid velocity in direction normal to the line element.

V_x, V_y, V_z = *x,y,z* components of fluid velocity.

The code formulation is the same as the standard one except for the Fz component:

$$F_z = +/- \frac{1}{2} P\rho\pi D_a L C_{d_z} |V|^2 \qquad [5\text{-}13]$$

where

D_a = effective diameter of the line in the axial direction for drag purposes.

V = overall fluid velocity vector.

The Eames formulation is as follows:

$$F_x = \frac{1}{2}P\left(\rho D_n L C_{dx} V_x |V_n| + \frac{1}{2}\rho \pi D_a L C_{dz} V_x (|V| - |V_n|)\right)$$

$$F_y = \frac{1}{2}P\left(\rho D_n L C_{dy} V_y |V_n| + \frac{1}{2}\rho \pi D_a L C_{dz} V_y (|V| - |V_n|)\right) \qquad [5\text{-}14]$$

$$F_z = \frac{1}{2}P\rho \pi D_a L C_{dz} V_z |V|$$

As far as the wave-induced forces on the floating bodies are concerned, OrcaFlex relies on an external code (AQWA and WAMIT are the ones it explicitly supports) to provide the hydrodynamic coefficients.

The following data can be imported from an external hydrodynamic solver:

- Displacement RAOs (amplitude and phase).
- Load RAOs.
- Quadratic transfer functions (QTFs) for wave drift calculations.
- Added-mass and damping coefficients.
- Hydrostatic stiffness.

5.5.3 Key findings - Orcaflex

Although it is clear that commercial packages such as OrcaFlex provide dedicated solutions for mooring problems, a full integration with the GH model is not feasible as a first step. Furthermore, it is not clear whether such integration is necessary. The coupling of the software packages could be achieved via a DLL, but this would involve an explicit commercial agreement with Orcina.

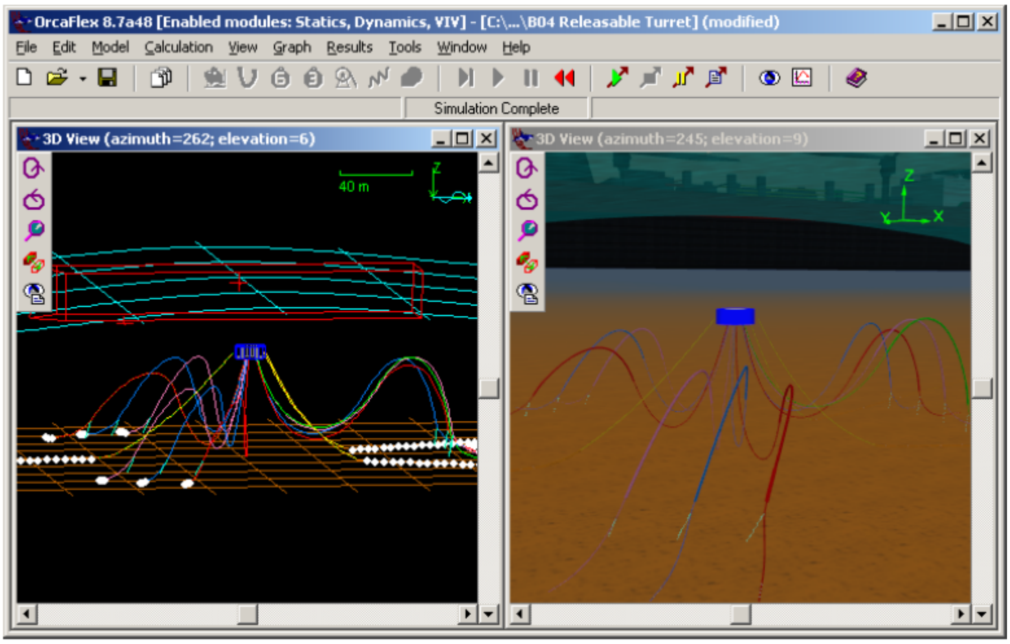

OrcaFlex 8.7a48 [Enabled modules: Statics, Dynamics, VIV] - [C:\...\B04 Releasable Turret] (modified)
File
Edit
Model
Calculation
View
Graph
Results
Tools
Window
Help
Simulation Complete
3D View (azimuth=262; elevation=6)
40 m
Z
3D View (azimuth=245; elevation=9)
Z
Y
X

Figure 5.5: OrcaFlex screenshot showing both wire-frame and shaded 3-D views.

5.6 Development strategy: moorings module

As a first step (and possibly the only step needed), mooring configurations will be modelled using a look-up-table approach. These tables allow nonlinear behaviour to be modelled, by specifying the inertial, stiffness, and damping properties of the mooring configurations. These quantities can vary with displacement, velocity, acceleration, and also with the frequency of the incident wave. As part of the design process for this software, a decision must be made as to which property is “plotted” against which kinematic quantity for the purposes of the look-up tables.

For example, the damping force can vary with velocity in a nonlinear manner, but it also varies with the tautness of the cable, and thus with the displacement of the top end of the cable. It also depends on excitation (wave) frequency.

The look-up tables could be selected by the user at runtime from a database of predefined mooring elements or strategies. For some applications, however, this approach may be inadequate and in this case task-specific “custom” tables would be calculated on demand, using alternative moorings software or industry standard codes. On the basis of the investigations described in this section, it is concluded that if the use of commercial tools is needed, OrcaFlex and AQWA are the most suitable codes for this purpose. Also unknown is whether there will in future be a need for a WaveDyn simulation to include a more complex mooring analysis than can be achieved using the look-up-table approach, such as the definition of Morison elements for all (or most) key aspects of the mooring configuration. If this does become necessary, this need will be satisfied either by a more advanced moorings module developed in-house by GH, or by means of a DLL interface to one of these two, or an equivalent package.

Following Fitzgerald and Bergdahl (2008) and Johanning et al. (2006), mooring lines display a substantial level of frequency-dependent impedance (both resistive and reactive). It may accordingly be necessary to incorporate this into the time-domain model, where an impedance of this type would be represented by a convolution in time. Convolution would have the effect of representing a large number of states describing the line dynamics. A system identification technique could be used to reduce the system to a finite-state model, as has been suggested for the radiation force convolution in the Methodology Report (WG1 WP1 D1b, Section 3.3.4).

5.7 Implementation details

5.7.1 Algorithm

The coefficients which determine the mooring forces are summarised in three matrices. These are the inertia or mass matrix M_{moor}, the damping matrix B_{moor} and the stiffness matrix K_{moor}.

The inertia (mass) matrix contains the following elements:

$$M_{moor} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} \\ m_{31} & m_{32} & m_{33} & m_{34} & m_{35} & m_{36} \\ m_{41} & m_{42} & m_{43} & m_{44} & m_{45} & m_{46} \\ m_{51} & m_{52} & m_{55} & m_{54} & m_{55} & m_{56} \\ m_{61} & m_{62} & m_{63} & m_{64} & m_{65} & m_{66} \end{pmatrix}$$

where the subscripts 1,2,…6 correspond to the x, y, z, ϕ ,θ, ψ directions of the mooring line's own coordinate system and where the symbols ϕ ,θ, ψ refer to roll, pitch and yaw rotations respectively.

The damping matrix is as follows:

$$B_{moor} = \begin{pmatrix} b_{xx} & b_{xy} & b_{xz} & b_{x\phi} & b_{x\theta} & b_{x\psi} \\ b_{yx} & b_{yy} & b_{yz} & b_{y\phi} & b_{y\theta} & b_{y\psi} \\ b_{zx} & b_{zy} & b_{zz} & b_{z\phi} & b_{z\theta} & b_{z\psi} \\ b_{\phi x} & b_{\phi y} & b_{\phi z} & b_{\phi\phi} & b_{\phi\theta} & b_{\phi\psi} \\ b_{\theta x} & b_{\theta y} & b_{\theta z} & b_{\theta\phi} & b_{\theta\theta} & b_{\theta\psi} \\ b_{\psi x} & b_{\psi y} & b_{\psi z} & b_{\psi\phi} & b_{\psi\theta} & b_{\psi\psi} \end{pmatrix}$$

Finally, the stiffness matrix is as follows:

$$K_{moor} = \begin{pmatrix} k_{xx} & k_{xy} & k_{xz} & k_{x\phi} & k_{x\theta} & k_{x\psi} \\ k_{yx} & k_{yy} & k_{yz} & k_{y\phi} & k_{y\theta} & k_{y\psi} \\ k_{zx} & k_{zy} & k_{zz} & k_{z\phi} & k_{z\theta} & k_{z\psi} \\ k_{\phi x} & k_{\theta y} & k_{\psi z} & k_{\phi\phi} & k_{\phi\theta} & k_{\phi\psi} \\ k_{\phi x} & k_{\theta y} & k_{\psi z} & k_{\theta\phi} & k_{\theta\theta} & k_{\theta\psi} \\ k_{\phi x} & k_{\theta y} & k_{\psi z} & k_{\psi\phi} & k_{\psi\theta} & k_{\psi\psi} \end{pmatrix}$$

If there are N moorings attached to the body and the matrices for the ith mooring are given by K_i,B_i,M_i, then the overall force on the body due to the moorings is

$$F_{moor} = \sum_{i=1}^{N} (K_i x + C_i \dot{x} + M_i \ddot{x}) \qquad [5\text{-}15]$$

where x is the 6x1 displacement vector (three linear and three angular displacements). This is similar to the approach taken in ROMEO. The implementation has predicted the potential of expanding the functionality into the time-domain, by allowing nonlinear representation of the above expression to be developed from the frequency-domain basic functionality.

The algorithm is implemented in code as follows. Each non-zero element of each of the matrices K_{moor}, B_{moor}, and M_{moor} is not represented as a single value, but as a look-up table. As an example, consider the top-left element in the damping matrix, b_{xx}. This element represents the amount of mooring damping force experienced by the body in the *x*-direction as a result of the *x*-component of the velocity. The software will store several different values of b_{xx} in a look-up table, corresponding to different values of displacement of the attachment point of the mooring on the body. An example of such a look-up table is shown in Table 5.2 and a graphical representation of the table is given in Figure 5.6. The numbers used are for illustrative purposes only. The displacement in this example is measured relative to an equilibrium position.

In some approaches (the one used in AQWA, for example), a "mooring force vs, extension" curve is used instead. For a taut mooring line, this value can be translated to a "force vs, displacement" curve quite simply: the extension, or stretch, of the line is just the component of the displacement vector in the direction of the line. For a catenary line, the effective stiffness force is due primarily to the catenary shape rather than the stretching of the line.

Displacement, *x* (m)	Damping coefficient, b_{xx} (kNm^{-1}s)
-4.0	3.8
-2.0	4.3
0	5.7
2.0	7.8
4.0	10.6

Table 5.2: Example of a look-up table as used to represent an element of the damping matrix. matrix.

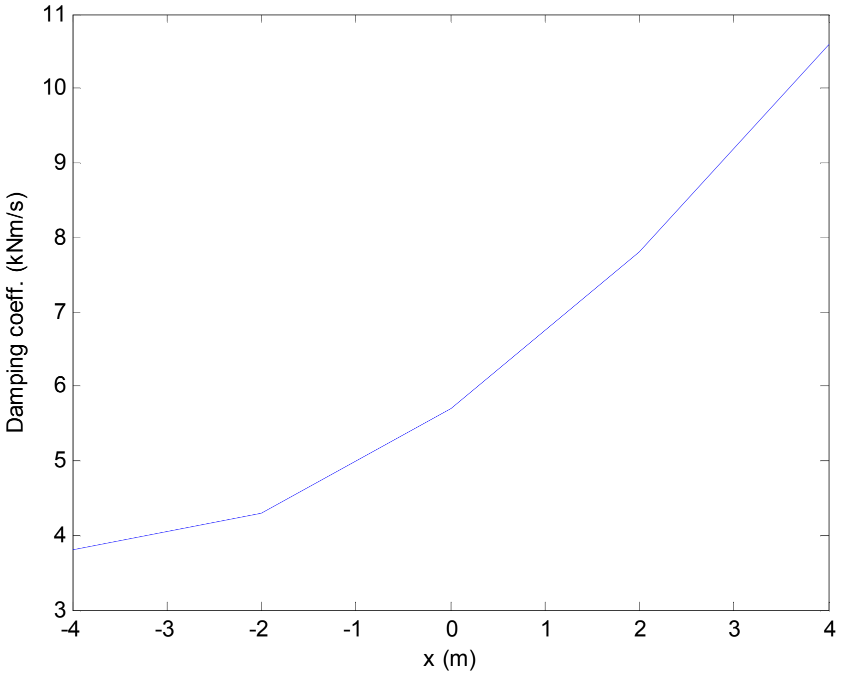


Figure 5.6: Graphical representation of the look-up table (Table 5.2)

5.7.2 Coordinate system

The kinematics and loads on which the lookup tables are based are all expressed in the mooring line's own coordinate system. The definition of this system is illustrated in Figure 5.7. As the figure shows, the anchor point is the origin of the mooring's coordinate system. The *z*-axis of this coordinate system is vertical (as is the global z-axis), and the ***xz*** – plane contains both the anchor point and the attachment point (i.e. where the mooring line is fixed to the floating body). Thus the transformation from the global frame to the mooring-based frame consists of two elements: a translation given by vector $\underline{\boldsymbol{a}}$ (the position of the anchor in global coordinates), and a rotation by an angle α about the z-axis:

$$\underline{x}_m = R\underline{x}_g - \underline{a} \qquad [5\text{-}16]$$

where:

$\underline{x}_m$ is a position vector in mooring-based coordinates;

$\underline{x}_g$ is the same position in space but expressed in global coordinates;

R is the rotation matrix corresponding to a rotation by angle α about the z-axis, where the rotation direction is from the mooring x-axis towards the global x-axis (anticlockwise in Figure 5.7).

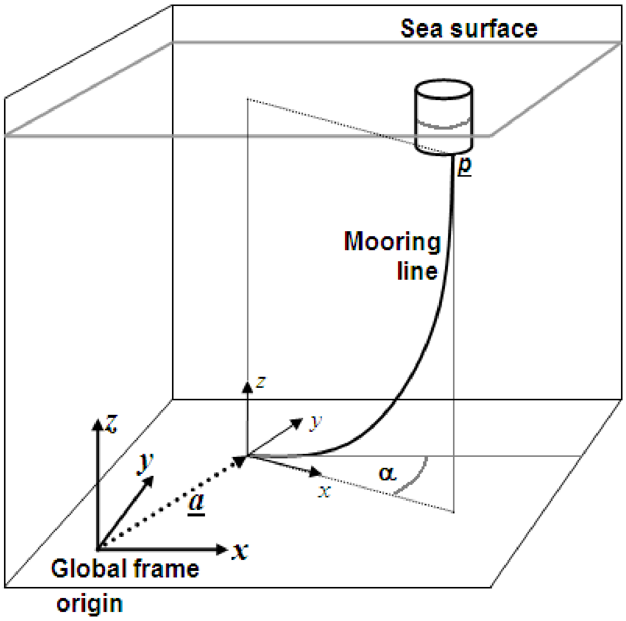


Figure 5.7: The mooring line layout, illustrating the basis of the mooring line's coordinate system.

For each mooring line, the displacement and velocity vectors which are used in the look-up tables refer to the line's attachment point $\underline{\boldsymbol{p}}$, and are expressed in that mooring line's coordinate system. The displacement vectors are measured relative to $\underline{\boldsymbol{p}}$'s equilibrium position $\underline{\boldsymbol{p}}_{eq}$, which is its steady-state position in the absence of any waves. Thus the displacement vector to use in the look-up, $\underline{\boldsymbol{p}}_m$, is given by:

$$\underline{p}_m = R(\underline{p} - \underline{p}_{eq}) \qquad [5\text{-}17]$$

where both $\underline{\boldsymbol{p}}$ and $\underline{\boldsymbol{p}}_{eq}$ are expressed in the **global** frame.

The look-up procedure's output values of force and torque are also expressed in the mooring-line coordinate system. They are converted back into global coordinates before they are applied to the floating body.

5.7.3 Integration with existing software architecture

This section should be read in the context of Section 2 which gives an overall description of the WaveDyn architecture. Figure 2.5 shows an overview of the code structure and how the moorings module fits into it. The moorings module is one of three "applied-force" calculation modules, the other two being the PTO module and the hydrodynamics.

WaveDyn uses a multi-body dynamics approach (see Section 2.1). Any part of the WEC that moves relative to other parts will be modelled as a single body in this system. Mooring lines are not modelled as bodies in the system, but are treated separately. This is done for two main reasons:

- To fully model a mooring line as a flexible body is computationally intensive. The look-up table approach offers a more efficient alternative to this, and thus can be seen as a starting point. Commercial packages still offer similar alternatives to date. To quantify the implications and the potential loss of accuracy of such approach (under performance related wave conditions) is part of the ongoing effort.
- The multi-body approach places some restrictions on the configuration of the multi-body system. Specifically it requires the system to have a tree-like topology (as explained in Section 2.2), with exactly one possible path to ground from anywhere in the structure. Multiple mooring-line bodies in the structure would violate this rule, while multiple mooring line force profiles do not. If mooring line bodies prove to be necessary, changes to the MB core code will be necessary (again this justifies the investigation of the first step outlined above).

As shown in Figure 5.7, the moorings module has its own input/output (I/O) functionality; this allows it to parse the relevant section of the common data file (the .IN file described in Section 2.7.1) which holds all information about the WEC and the simulation. For each mooring line, the relevant information is within a special block (see Table 2.2) demarcated by the symbols

{MOORING

...

}MOORING,

and consists of:

- The mooring line's name;
- The body which it is attached to;
- The position of the attachment point (expressed in the body-fixed coordinate system of the attached body);
- The position of the anchor (expressed in the global coordinate system);
- The stiffness, damping and inertia matrices (stored as 6-by-6 arrays of look-up tables, all quantities being expressed in the frame of reference of the mooring line).

Once this information has been read in, the coordinating (WaveDyn core) code starts the simulation. Firstly the core WaveDyn code passes on relevant details to the mooring module about the current state of the system. For each mooring line, the required information is the current position and velocity (both linear and rotational) of the body to which the line is attached. The mooring module can then calculate, in the mooring line coordinate system, both the velocity of attachment point $\underline{\boldsymbol{p}}$ and its relative displacement from its equilibrium position. Once these values are known it can perform the look-up operation to obtain the loads, which it then transforms back into the global coordinate system. Before the loads can be applied to the bodies, a further step is needed: in the multi-body code the loads are applied at the proximal node of the body (see Section 2.2 for definition of this term). In general, the proximal node of the body is not at the same point in space as the attachment point of the mooring. This means the loads need to be transformed into a new set of loads acting at the proximal node. The transformed mooring and other loads (PTO and hydrodynamic) are then applied to the body or bodies by the multi-body module (MBCore) code.

6 POWER TAKE-OFF (PTO) AND CONTROL MODULES

Although the PTO and control modules are in fact two separate modules (as it is clear in Figure 2.5), the frequency-domain implementation considerably limits the domain of applicability of both. In particular the PTO force (or torque) profile must be linear (or linearised) with regard to the system's internal variables such as (relative) velocity between elements of the WEC. In addition, the control approaches are often limited to simple, conservative methodologies (e.g. passive damping, constant throughout the wave frequency range) or optimal strategies (e.g. phase control).

Given the above mentioned limitations (which have been further detailed in WG1 WP1 D1b) the development of the more complex functionalities for both the PTO and control modules will occur during the time-domain model implementation stage, which will be presented in WG1 WP1 D3. The target functionalities are overviewed in the 'future work section of this report.

The following sections describe the implementation of the PTO and control modules to date, the potential of the functionalities and the key limitations associated with the approach.

6.1 The PTO modelling approach

WaveDyn users have the option of applying PTO forces at any joint body in a WEC structure. The PTO resists the relative motion in the joint as it extracts useful energy and so a realistic description of the PTO design and the actuator forces it applies to the structure is necessary to solve accurately for the motion of the device and to assess the amount of power generated. The multi-body approach introduced for the Hydrodynamics and Moorings Calculation Modules has again been adopted for the PTO implementation: PTO definitions for the joints are represented computationally as a system of PTO bodies stored within a higher level containment and interfacing object. In the initial PTO implementation, the PTO bodies will represent a simple explicit force calculation based on a parameterised linear mathematical equation; this approach is described in Section 6.2. The PTO bodies will support increasingly more detailed descriptions of the PTO design as WaveDyn develops and it is envisaged that a number of 'PTO templates' will be available to the user. The templates will each be implemented as a PTO body type, containing mathematical models designed to create actuator forces at the structural joints with similar *characteristics* to the real system, which may consist of many different dynamic components which will not all be individually modelled. A mathematical example of a hydrodynamic or electrical rectification and smoothing PTO system template was provided in the Methodology Report WG1 WP1 D1b, Section 5.5.4. The PTO templates will generate nonlinear and potentially discontinuous forces and will be available in the time-domain only. The template calculations may also incorporate internal states, so that the Calculation Module returns state derivative values to the integrator and receives updated state values at the beginning of each time-step as described in Section 2.9.

WaveDyn users wishing to model PTO systems in greater detail, potentially to assist with the design of the PTO system itself, will have the option of producing an external DLL PTO model that receives structural joint kinematic information and returns PTO forces. The interface may also incorporate the transfer of PTO state values and state derivatives if the PTO model is solved for using the WaveDyn integrator; alternatively, the DLL may incorporate an integrator of its own. If a DLL is used to define the PTO then this must also incorporate any control strategies to be implemented, as the WaveDyn code will not have access to the PTO itself. Again this is an option that will be available in the time-domain approach, as a highly discontinuous PTO force profile can be expected for real applications.

The depth of PTO modelling available in WaveDyn will be an important aspect of the software capability, which, as a specialised WEC loading and performance tool, differs from more generic flow solver and hydrodynamics modelling software packages originally developed for more traditional maritime or offshore oil and gas industries.

6.2 Initial formulation of the PTO module: applying forces to structural elements

The initial PTO Calculation Module formulation, designed to support frequency-domain simulations, allows the user to specify parameters for a linear PTO force expression to be applied to the freedoms in the SlidingJoint and Hinge structural bodies. The PTO force expression takes the simple, parameterised spring-damper form:

$$f_{PTO} = B_{Control} B_{PTO} \dot{x} + K_{Control} K_{PTO} x \qquad [6.1]$$

where x is the motion in the joint freedom relative to its position at the start of the simulation, B_{PTO} is a PTO damping value and K_{PTO} is a spring stiffness. The terms with a "$_{\text{Control}}$" subscript will be discussed in Section 6.3. The PTO model input parameters are provided by the user in the .IN file "<pto >pto" sub-block for the joint bodies. The format and order of the parameters is shown in Table 6.1 below.

<pto >pto Sub-block Parameters			
Input Parameter (as presented in the .IN file)	**Data Format**	**Parser Dependencies**	**Description**
`Damping`	Double precision floating point number	None Parameter is compulsory, but may be set to zero.	PTO damping. Energy dissipated is assumed to be converted by the PTO mechanism, producing some useful electrical power output.
`Stiffness`	Double precision floating point number	None Parameter is compulsory, but may be set to zero.	PTO stiffness

Table 6.1: PTO sub-block .IN file parameters for the initial, linear PTO model

6.3 Initial formulation of the control module: restriction to the applied PTO forces

Under the frequency-domain limitations the control module can be interpreted as a means to restrict and / or adapt the applied PTO force, under a series of assumptions and approximations.

As presented in Equation [6.1], the terms with a "$_{\mathrm{Control}}$" subscript indicate those associated with the control matrix. Immediately several control methodologies can be envisaged; these can be grouped into two main categories:

- Baseline control strategies;
- Optimal control strategies.

Under the former the control terms ($B_{Control}$ and $K_{Control}$) can only assume constant, frequency independent values. Such approach allows the estimation of the WEC(s) behaviour under conservative conditions, if the PTO terms (B_{PTO} and K_{PTO}) are also kept to frequency independent values. The PTO terms may also be given as intervals (rather than single values), and the control module will assess all the passive responses to select (via exhaustive search) which coefficients are associated with best performance for such input conditions. As an example, if $K_{Control} = 0$ then the PTO force will depend exclusively on a linear dependence between the velocity in the joint freedom ($\dot{x}$) and the $B_{Control}B_{PTO}$ product. This can be further simplified if B_{PTO} is kept constant (at a mean value) and $B_{Control}$ is defined as a range of values that allows iterations around the mean value (e.g. $B_{Control} = [0.5;0.75;1;1.25;1.5]$, which would lead to a range of solutions that allow the evaluation of the WEC(s) performance under the influence of a B_{PTO} coefficient spanning between 50% and 150% of the original (mean) value). Such simplified scenarios can lead to the identification of the under-damped and over-damped regimes in a quick and effective way.

For the optimal control strategies, the product of the control and the PTO terms may be associated with frequency dependent quantities which in turn can be related to the hydrodynamic characteristics of the WECs (or array of WECs). This approach, albeit less conservative than the baseline control strategies, has the potential to lead to better performance. Careful monitoring of the output variables in necessary to ensure that the derived variables do not violate the fundamental principles of a frequency-domain approach, nor do they violate engineering feasibility limits associated with specific components; this may lead to the implementation of structural constraints (e.g. motion constraints) in the core definition of the WEC (see Section 2).

For array performance the control module is particularly linked to the optimiser module – see Section 8.2 for a preliminary exercise. In the time-domain approach it is envisaged that both the PTO and the control modules will further reduce the wide range of options that the optimiser module can use as starting points for the array optimisation process.

In the Beta versions the user will have the option to define the control strategy (and the associated coefficients) from a list of options which may be associated with the above categories. Alternatively, the user may load a .DLL file (compiled in a constrained format) that defines either f_{PTO} directly (in which case a combined DLL operates in place of both the control and PTO modules, as described in Section 6.1), or just the control coefficients (only an option if the PTO model is not supplied as a DLL, so that the nature of the coefficients is known). This .DLL link will allow technologies developers to supply specific routines which the software tools can use without disclosing intellectual property, or sensitive information, even in baseline control strategies.

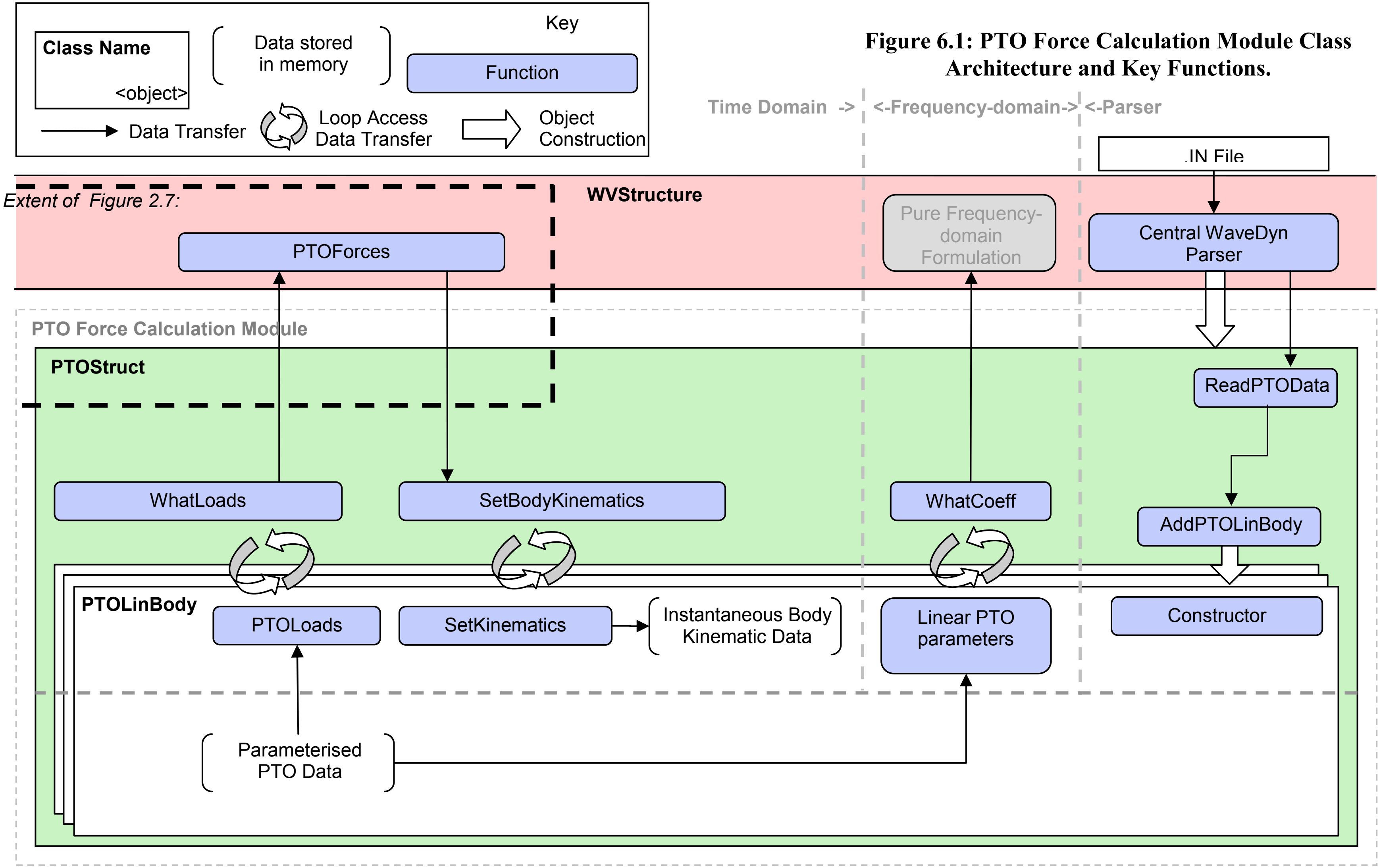
The frequency-domain formulation does not allow internal states to be defined, thus all f_{PTO} are treated as explicit functions of the motion and velocity in the joint freedom relative to its position at the start of the simulation. The ability to describe f_{PTO} as a highly discontinuous function will be handled in the time-domain module (defined in WG1 WP1 D3).

6.4 Calculation Module Software Architecture

The PTO module multi-body software architecture has a very similar form to that described in Section 3.4 for the Hydrodynamics module, but is much less developed at this stage as it must only conduct comparatively simple calculations. The class object and member functions arrangement is shown schematically in Figure 6.1. At present a single PTO body type, PTOLinBody exists, which evaluates the linear expression given in Equation [6.1] in the time-domain and would return transformed damping and stiffness matrices in a pure frequency-domain formulation.

Future development will allow the higher level PTOStruct class to contain sets of bodies of a number of different types; each body type representing a different PTO template model, a simpler nonlinear mathematical form, or a DLL interface.

Figure 6.1: PTO Force Calculation Module Class Architecture and Key Functions.



7 OPTIMISER MODULE

The role of the optimiser module is to assist in the design a wave farm under realistic engineering constraints. To do so, it utilises the functionality of the modules described in the previous sections of this report. In this task, the aim is to accentuate certain desirable characteristics and minimise the impact of undesirable features, whilst ensuring that specific requirements are met. As described in Section 1.1, the key importance of the optimiser module and its dependence on WEC specific load and performance characteristics (calculated via all the other modules) have led to the conclusion that the optimiser module will in fact form a separate tool (WaveFarmer) to the other software components. This will depend on outputs (or outputs in a compatible format with) a tool that aggregates all other modules (WaveDyn).

The specification of the intended solution and the analysis of it are not part of the functionality of the optimiser, so the module must interact with several others as detailed in Section 7.1. The form of the associated input and outputs that are currently included in the optimiser module are also given in that section, along with possible future extensions to them. A range of current and potential methodologies for the optimisation are presented in Section 7.2, including an update to one of the planned procedures. Finally in Section 7.3, a description is given of the optimisation algorithms employed thus far, along with an outline of the case studies used to demonstrate them.

7.1 Functionality

7.1.1 Structure

There are several categories of information that interact within the optimiser. Such information may come from other modules or be passed to them in the course of the operation. The following labels will be used to refer to these categories:

- Objective function
- Fixed inputs
- Constraints
- Penalties
- Design variables

The ‘objective’ is the quality of a design that is required to be enhanced or reduced. This should have the ability to be expressed numerically, in the form of a function of the design (the ‘objective function’). The fixed inputs are variables that the optimiser receives from elsewhere, over which it has no control. Note that these are not necessarily fixed with respect the software package as a whole, since the user may be able to change them. The constraints are the features of the solution that must be satisfied by the output from the optimisation. These may come from the user directly or be stipulated by the software to ensure valid designs are produced. Penalties are more flexible way of specifying undesirable characteristics than constraints because they lower the estimation of how suitable a particular solution is without excluding it. In other words they condition the objective function so that designs that have unwanted but not unfeasible features are less likely to be selected as the final design output. In practice, however, a constraint may sometimes be expressed as a strong penalty and vice versa. Finally, the design variables form the specification for any particular solution. These are the values that are altered by the optimisation algorithm in order to change the features of the design. In particular, they will determine what value the objective function and penalty function(s) take and whether or not the constraints are satisfied.

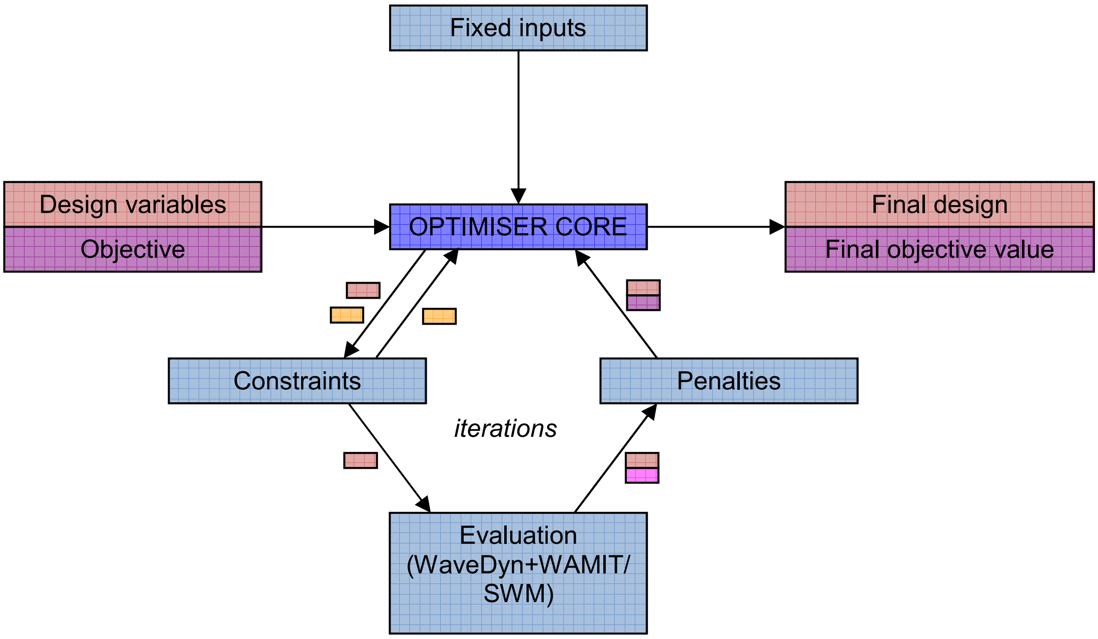


Figure 7.1: Structure of optimisation algorithm.

The general structure of an optimisation algorithm, including the above mentioned categories of information, is shown in Figure 7.1. The fixed inputs are fed into the core of the optimisation routine along with the specification of what form the design variables and objective function take. One or more solutions are then generated in the form of values taken by the design variables. These are either produced in such a way as the constraints are satisfied or they are checked after production to make sure that they are permissible, with alternative solutions being generated if not. The set of designs are then evaluated using other modules of the software (such as WaveDyn or potentially the spectral wave model) and thus paired with a value of the objective function. This is often the most time-consuming part of the whole process since it generally involves the assessment of many different physical scenarios. The penalty function is able to modify the objective value before it is passed back to the central optimiser code. Depending on the algorithm used, further solutions may be generated based on the information regarding the performance of the previous set. This may repeat as one or more iterations before the final solution is selected from the last set to be considered, if necessary. It should be noted that although this is a fairly general outline of an optimisation procedure, some algorithms may involve slight variations from the structure given here.

7.1.2 Inputs and outputs

The variables in each of the categories specified in the previous section are described below. Only the inputs and outputs that have been implemented in the optimiser to date will be described, inline with the methodologies that have been implemented (Section 7.2) and the original methodology plan presented in WG1 WP1 D1b. Possibilities for extensions are reviewed in Section 7.2. Note that the classification into the categories given here is not absolute and is dependent on the exact expression of the optimisation problem.

Objective function

The following is the only objective currently considered in the optimiser module:

- Maximisation of the power absorbed in a certain wave climate

This objective function is chosen since it constitutes the primary performance indicator for a wave farm. As the calculations in the frequency-domain model are by definition linear, the principle of linear superposition may be applied for each incident wave period and direction. Therefore an average annual sea-state can equivalently represent the wave climate over the year. The calculation of the absorbed power for each set of design variables is currently preformed by WaveDyn, with calls to WAMIT where necessary.

Fixed inputs

The fixed inputs to the optimisation module are:

- Sea-state
- Water depth
- Number of devices
- Geometry of the device
- Modes in which the device can move
- Modes of motion in which power is converted

The sea-state may be regular or irregular and in the latter case can represent the annual wave climate, as previously mentioned. The water is assumed to have a constant finite depth specified by another of the inputs. It is also assumed in the current implementation that the user knows how many devices are being deployed and their fundamental properties. The last three inputs in the list are defined in WaveDyn, as outlined in Sections 3, 5 and 6.

Constraints / Penalties

As previously stated, constraints may be implemented as penalties and thus the two categories are grouped together in this description. The list therefore comprises:

- Minimum spacing between devices
- Minimum and maximum PTO settings
- Allowable control strategies to be tested

The minimum spacing requirement serves a number of purposes. Perhaps the most important is to ensure that WECs that permit some level of horizontal movement do not collide with each other or allow their moorings to do so. In addition, access routes for maintenance as well as other planned (and unplanned) activities will demand a minimum spacing requirement. In the current implementation, this constraint is expressed by setting the objective function to be zero for arrays that violate the condition. The final two items on the list are inputs that arrive in the optimiser via the PTO module. The range of permitted control strategies determine the form of the PTO variables used in the optimiser. In the present frequency-domain implementation, two representations of the PTO control are considered; a spring and spring-damper combination.

Design variables

The two main design variables relating to an array are as follows:

- Layout of devices within the array
- PTO settings for each WEC

Either of these sets of variables may alternatively be entered by the user as fixed inputs with the optimisation performed on the remaining set. They both may have a significant influence on the performance of the array as a whole due to their effect on the hydrodynamic interactions between devices. In the following paragraphs, the way that each is represented as design variables will be described.

Firstly, the variables associated with the array layout are defined. In the general case, a configuration of axi-symmetric devices within the array can be specified by two coordinates (e.g. x and y) for every element following the first. The coordinates of the first converter do not need to be specified since these determine the position of the whole array leaving the objective function, constraints and penalties applied unaffected. For devices that are not radially symmetric, a further variable is needed for each to determine its heading angle.

However, the initial array definition used in the implementation is a special case of the aforementioned scheme. It consists of a 'regular grid,' where devices are placed at the intersections of two sets of equally spaced parallel lines. Although this is a more restrictive definition than that above, there are a number of reasons why it might be more attractive. The first is that it significantly reduces the number of variables needed to describe arrays of many bodies. This reduction in the size of the search space has the potential to lead to an improvement in optimiser efficiency. Secondly, free optimisation of the layout (that is, without formation patterns imposed) has been seen to produce regular arrangements in a number of scenarios (Child and Venugopal, 2010). Hence it is anticipated that a regular grid regime would assist the optimisation in finding similar optimal configurations more effectively. It is worth noting that in complex problems such as the present one, the 'true' optimum is often never reached. Instead an algorithm may be judged on how quickly it finds a solution and how well that solution performs.

There are also some practical advantages of regular array formations. Firstly, high packing densities can be achieved with this type of array for a given minimum inter-body spacing. Also, the given formation allows the natural extension of arrays involving a small number of devices to larger ones. Finally, the possibility exists of sharing some components of the mooring arrangement if the devices are in some sort of regular formation, which could represent significant cost savings. This is the focus of some recent academic research (Vicente and Falcao, 2009), although further associated issues still need to be resolved.

The regular grid formation of an array of N devices, illustrated in Figure 7.2, is defined as follows. Firstly two directions are defined, making angles ψ and σ with the positive x-direction (mean wave direction). Next, a grid is constructed such that lines are placed at regular intervals with spacing p in the direction defined by ψ and s in the σ-direction. An integer number of adjacent gridlines M are chosen in the σ-direction, and a sufficient number, ceil(N/M), in the ψ-direction such that their product of the two values is at least as big as the number of devices. The intersections of the chosen gridlines then define the potential device locations. These positions are filled by devices sequentially in a down-wave direction from the most up-wave point of the array until all N converters have been allocated a position. In the case that more than one candidate for the final device location share the same x-coordinate, preference is given to the one with the least y-coordinate, without loss of generality. For WECs that are not radially symmetric, a further parameter χ is defined to represent the orientation of all devices with respect to the x-axis.

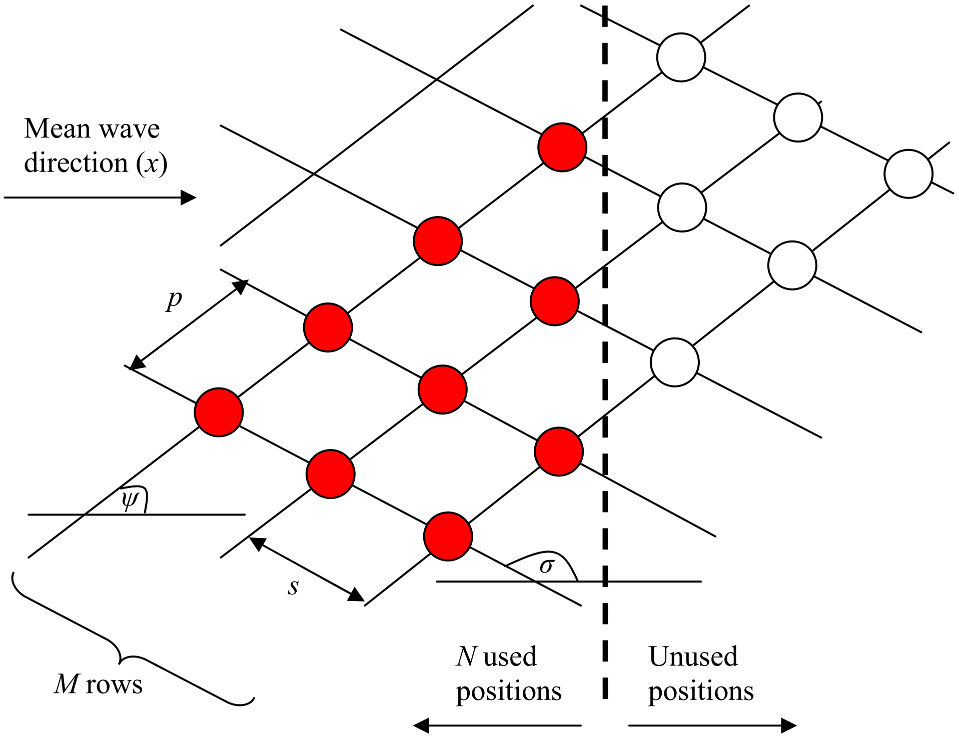


Figure 7.2: Regular grid definition sketch (axially symmetric devices).

There are therefore five parameters that govern any particular instance of arrays of axially symmetric WECs using this formation:

- ψ: Direction of gridline (Set 1)
- σ : Direction of gridline (Set 2)
- p: Spacing between gridline of Set 2 along gridlines from Set 1
- s: Spacing between gridline of Set 1 along gridlines from Set 2
- M: Number of gridlines from Set 1 selected to define potential device locations

If the devices are not axially symmetric a sixth parameter is also involved:

- χ: Device heading

A parameter vector [M, p, s, ψ, σ] or [M, p, s, ψ, σ, χ] is then sufficient to completely determine the geometry of the array. Therefore, the ‘array layout’ design variable consists in practice of five or six values to be optimised over.

The constraints must also be expressed in terms of these parameters. Due to equivalence of the formation when any of the angles is incremented by whole multiples of 2π, the following restrictions may be placed on the associated parameters to reduce the size of the search space:

$$0 < \psi \leq 2\pi$$

$$0 < \sigma \leq 2\pi$$

$$0 < \chi \leq 2\pi$$

A minimum device spacing of distance L may be expressed for adjacent devices on the same gridline as:

$$L \leq p$$

$$L \leq s$$

To prevent devices that are not on the same gridline from becoming too close, another pair of constraints is needed. From the cosine rule, the diagonal distances across grid cells must be greater than or equal to the minimum separation. Moreover, the distance between devices diagonally across multiple cells must also satisfy these conditions. Hence the following conditions need to be obeyed:

$$\sqrt{(np)^2 + (ms)^2 - 2(nr)(ms)\cos(\pi + \psi - \sigma)} \geq L$$

$$\sqrt{(np)^2 + (ms)^2 - 2(nr)(ms)\cos(\sigma - \psi)} \geq L$$

where $1 \leq n \leq N$ and $1 \leq m \leq N$ are integers.

The other main set of design variables involves the characteristics of the PTO for each WEC, bearing in mind that each WEC may have multiple PTO modes. Depending on the control strategies that are permitted by the constraints, these parameters will take a different form. If only a dissipative component is to be considered then the PTO settings may be expressed as a collection of damping coefficients; one for each mode in which power is converted, for each WEC in the array. If a reactive component is also permitted, then there are twice this number of parameters; a spring and a damping coefficient for each mode as previously mentioned.

7.1.3 Extensions

Below are listed the possible future extensions to each of the categories of input and output described in the preceding sections. These represent targets that may be investigated but not necessarily implemented if initial results show that their influence is minimal.

Objective function

- Minimisation of electrical cable used in the wave farm.
- Minimisation of the complexity of the mooring system.
- Minimisation of the total cost of energy by incorporating all major considerations weighted by economic impact. This is the undisputable ultimate goal of the optimisation, although the constituent costs would be difficult to quantify accurately within the project timeframe. The software should be developed to at least predict the possibility of a plug-in being developed in this regard once more detailed cost data is available in the public-domain.

Fixed inputs

- The mooring arrangement could be specified by the user (in WaveDyn), the nature of which will influence the behaviour of the system and therefore its performance.
- Maximum capacity of power transmission to land.

- Bathymetry and type of sea bed. These factors may affect the hydrodynamic solution (in the flow solver) or provide constraints on layout configuration.

Constraints

- Maximum area of site. The array must fit within the size and shape of the potential site. This is especially important for arrays involving a large number of devices, but is not included in the current implementation.
- Maximum amplitude of motion displacements such that the stroke length is not exceeded and the moorings are not over-extended. This will involve detailed information exchange with WaveDyn or a comprehensive WEC parameterisation procedure.
- Maximum mechanical loads that the PTO and the moorings can withstand.
- Maximum structural (wave induced) loads that do not damage the WEC.
- Maximum peak instantaneous power able to be transmitted to land, due to substation and cable limitations
- Active or passive control strategies permitted. That is to say, whether or not the PTO settings may be different for different incident wave frequencies.
- Different PTO templates permitted. In this approach, the PTO is described in terms of a number of physical components that have variables associated with them, which may then be optimised. This could lead to the conclusion that certain PTO types are more suited to certain sites and array configurations.

Penalties

- Large long- and short-term variability in total power output. This is undesirable since the grid needs smooth and reliable power sources.
- Large peak to average power ratio for each device. This is linked to higher rated, more expensive equipment.
- Large loads which do not contribute to power. These lead to shorter component lifetime and therefore higher maintenance costs.
- Excessive removal of energy from certain frequency or directional components of the incident wave energy spectrum. This can lead to an undesirable modification to coastal processes.

Design variables

- The regular grid structure could be parameterised in a different way in order to encourage the algorithm to efficiently find optimal solutions.
- Other formation strategies may be eventually be used. It may be found that the restrictions of regular grids are too great and that a free optimisation of device positions is more desirable.
- The layout of electrical cable and moorings between devices could be determined in future.

7.2 Implementation

7.2.1 Current approaches

The development of the optimiser module has been decomposed into several 'scenarios' as detailed in the Methodology Report (WG1 WP1 D1b). To facilitate the reading of the present report, a summary of these approaches is given here, along with an account of the modifications that have been made to them. Scenarios 1A and 1B relate to the Beta 1 version of the software, whereas the more advanced Scenarios 2A and 2B will only be implemented in the Beta 2 version.

Scenario 1A

In this approach, the optimisation problem is handled using an exhaustive method. A fixed set of cases and associated design variables are determined before evaluation of the solutions is made. The final design is then determined by comparing the objective function values, penalties and whether or not the constraints are satisfied before one is selected. This corresponds to one long iteration of the loop shown in the lower half of Figure 7.1.

The evaluation of solutions is performed using WaveDyn and WAMIT. It should be noted that here and in the following scenarios, the optimisation of layout requires one WAMIT simulation for every solution evaluation. This is in contrast to the optimisation of PTO settings for a fixed layout where the hydrodynamic coefficients may be determined with WAMIT in advance of the optimisation procedure.

Since the search space in the optimisation problems under consideration is typically very large and no pre-selection is made regarding the properties of the designs, correspondingly large number must be tested in order for this algorithm to be effective. The difficulty is especially acute for layout optimisation where time-consuming runs of WAMIT (or an equivalent flow solver) are involved. Therefore this approach is less efficient than dedicated optimisation algorithms. However, it is simple to implement and is able to provide a baseline for more advanced methodologies. Details of the implementation of Scenario 1A are contained in Section 7.3.1.

Original Scenario 1B

Scenario 1B as described in the Methodology Report (WG1 WP1 D1b) would involve a pre-selection of solutions from a given set prior to evaluation. This reduces the number of candidates that need to be tested. General selection rules would be formed on the basis of, for example, features of the isolated device behaviour or the radiation pattern of the device within an array. The remaining calculations regarding the array performance would then be executed as in Scenario 1A before the final solution chosen.

Further investigation has revealed that the number of remaining solutions after pre-selection may still be considerable, so the execution time may not be substantially reduced when migrating from Scenario 1A to 1B. Furthermore, general rules for different classes of FDC would need to be formed, potentially invalidating the methodology for a wider (non-FDC) range of concepts. A revised methodology has therefore been developed, as described below (Updated Scenario 1B).

Updated Scenario 1B

The new methodology that has been implemented for Scenario 1B involves the evaluation of solutions at every stage of an iterative procedure. Thus, the algorithm generates solutions automatically which are evaluated by WaveDyn (and WAMIT if necessary) before adaptively generating further solutions based on the success or failure of these initial solutions. This process corresponds to several executions of the loop pictured in the lower half of the optimisation scheme of Figure 7.1 instead of only one as planned in the original version of Scenario 1B. The whole process can be handled by an existing optimisation procedure, such as the Genetic Algorithm used by Child and Venugopal (2010). This type of algorithm was described in Section 4.2 of the Methodology Report (WG1 WP1 D1b) which was referred to under Scenario 1B in Section 5.5.7.

Note that, as in Scenario 1A, the layout optimisation is the most time-consuming task since it involves calls to WAMIT. Care must be taken to ensure that this process can be executed efficiently so that the optimisation algorithm can run a sufficient number of function evaluations in order to be effective. It is envisaged that this task will be made easier in Scenario 2A (described in Section 7.2.2), where a method for evaluating hydrodynamic interaction effects will be developed as part of the optimiser.

7.2.2 Extensions

The planned extensions to the optimiser module follow the template set out in the Methodology Report (WG1 WP1 D1b). The implementation of Scenarios 1A and 1B will be developed and improved for the Beta 1 version of the software.

Following the delivery of Beta 1, Scenarios 2A and 2B will be adopted for the Beta 2 version of the software. The first of these involves the inclusion of the capability to estimate the hydrodynamic interactions between devices within the software. Hence WAMIT will be run once prior to the execution of the optimiser module and all other calculations will be done within the software package under development. As previously mentioned, it is anticipated that this could lead to an improvement in the efficiency of the optimiser.

Scenario 2B involves the use of the spectral wave methodology developed by QUB as the design evaluation tool. Such a method would significantly improve the efficiency of the optimisation such that much larger arrays could be considered. It has not yet been determined the applicability of this type of model to the present situation nor the exact formulation of the optimisation problem to accompany the associated large arrays. These issues will be (at least partly) approached in WG1 WP2 D4, where a verification exercise comparing all modelling approached will be presented.

7.3 Detailed algorithm description and case studies

Following the approach described in Section 7.2, an initial implementation of Scenarios 1A and 1B of the optimiser module has been successfully carried out in the frequency-domain. In this section, details are given regarding the associated optimisation algorithms along with examples of their application. The associated results will be given in Section 8.3 of the case studies chapter.

All of the algorithms share the same objective function which is, as specified previously, to maximise the total power absorbed by the array. In practice, since the optimisation algorithms generally operate as minimisation routines, the negation of this quantity will be used as the objective function. The other inputs and outputs of the optimisation from the categories described in Section 7.1 (namely the fixed inputs, constraints, penalties and design variables) will be described in each case that follows.

The values associated with the variables in the input and output categories may clearly be altered from those given here. In particular, the software has the capacity for the following to be changed, as described in the Methodology Report (WG1 WP1 D1b):

- Wave climate inputs. This includes regular sea-states (of different frequency) and uni-directional irregular sea-states (Pierson Moskowitz, Bretschneider and JONSWAP with spectral parameters H_{m0} and T_p or T_{-10}) with different wave directions or equivalently orientations of array.

- Control methodology. The software is currently capable of considering both damper and spring-damper methodologies although only the former is used for the purposes of demonstrating the functionality of the optimiser here.
- Constraints. Different values for the minimum separation may be imposed as well as for the bound on the PTO settings.

The cases of a fixed array layout and the variation of performance with geometrical input data are considered below.

7.3.1 Scenario 1A

Under Scenario 1A, all of the design variables may vary simultaneously in a pre-defined database of arrays. However, to clearly demonstrate the preliminary optimisation exercises here, results relating to Scenario 1A are split into two examples where each of the main design variables is analysed separately. Firstly only the layout of the array is changed whilst PTO coefficients are kept constant. Subsequently, the PTO settings of each WEC in the array are iterated upon whilst the layout is held fixed.

Layout optimisation

In Scenario 1A, the task of layout optimisation reduces to the trial of different array configurations and the choice of the one that optimises the objective. A user of the software may load a number of pre-defined array layouts for the optimiser to test which then returns results to the user for consideration. The process is demonstrated here by computing results for set-ups I and II described below.

Set-up I

The layout for this case is shown in Figure 7.3 with the remaining properties given in summary form.

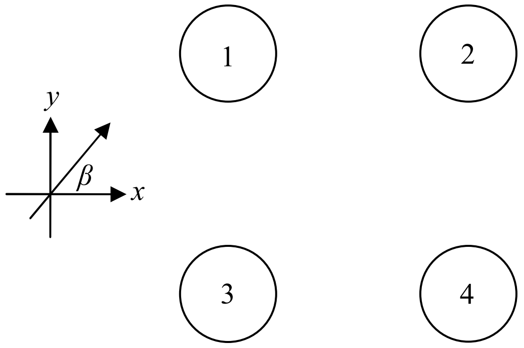


Figure 7.3: Array layout for Set-up I.

- Device properties
  - Geometry: Vertical truncated circular cylinder
  - Radius: a=10m
  - Draft: d-h=2a=20m (d is water depth, h is clearance beneath devices)
  - Mass: Neutrally buoyant at rest

  PTO: Power converted in the vertical (heave) direction only. Linear damper to represent PTO. All associated coefficients B_{pto}^{j} identical and equal to a value that maximises power outputput from a single device in the Bretschneider sea-state described below. This quantity was determined by testing values of the damping coefficient from 0 to 1000kNs/m in increments of 10kNs/m. A value of B_{pto}=640kNs/m resulted from these computations and has therefore been used here.
- Moorings: None
- Array properties
- Number of devices: N=4
- Array geometry: Square. Cylinder centres have coordinates 1: (0m,0m), 2: (40m,0m), 3: (0m,-40m), 4: (40m,-40m).
- Separation: L=4a=40m. Centre-to-centre distance separating devices on adjacent sides of the square.
- Wave direction: β=π. That is to say waves approach the array in the negative x-direction.

- Wave climate
  - Regular waves with periods: 6s ≤ T ≤ 20s
  - Bretschneider spectrum with H_{m0}=4m, T_p=10.2s
- Bathymetry
  - Flat
  - Water depth: d=4a=40m

Results are shown for this set-up in Section 8.3.1.

Set-up II

This case consists of exactly the same parameters as in set-up I but with the array layout modified as shown in Figure 7.4 and described below

- Array geometry: as for a square array except the row with the greatest x-coordinate is offset in the y-direction from the other row such that its least y-coordinate is the mean of those relating to the two devices in the other row. Cylinder centres have coordinates 1: (0m,0m), 2: (40m,20m), 3: (0m,-40m), 4: (40m,-20m).

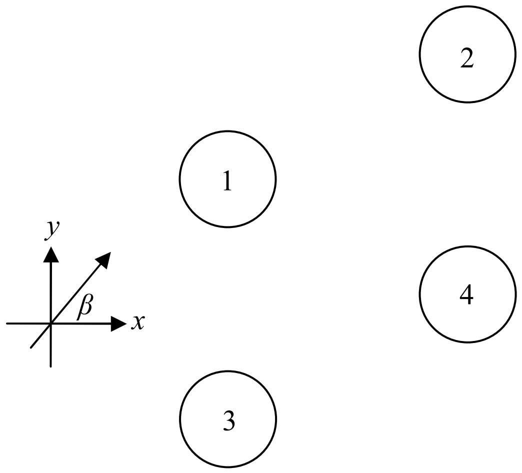


Figure 7.4: Array layout for Set-up II.

Results are shown for this set-up in Section 8.3.1.

PTO optimisation

The exhaustive optimisation of the PTO settings involves the damping coefficient in each mode associated with energy absorption being altered individually. These therefore constitute the design variables whilst the other inputs including the layout are held fixed. It should be stressed that the same methodology could be applied to spring-damper PTO regimes if permitted by the user input.

Minimum and maximum PTO settings are loaded and the interval split into equally spaced values within that range. This is the set of values that each PTO mode is permitted to take. Clearly, the computation time and accuracy will both increase if a finer step-size is chosen, so a compromise must be reached. This must be guided by the number of PTO modes since that figure must raised to the power of the number of permitted values in order to obtain the total number of cases to be tested. A link to the WaveDyn engine is then used to evaluate each case consisting of a unique combination of PTO settings, with the best one then selected as the output.

The set-up used to demonstrate this process is exactly as described in set-up I with the exception of the following details:

- Device PTO: Damping coefficients for each device independent and from the set B_{pto}=100,600,1100kNm/s for the course step size and 550 $\leq B_{pto} \leq$900 kNm/s in steps of 50kNm/s as the fine step size.
- Wave direction: β=5π/4.
- Wave climate: Bretschneider with H_{m0}=4m, T_p=10.2s.

Results are shown for the PTO optimisation in Section 8.3.1.

7.3.2 Scenario 1B

PTO optimisation

Power take-off optimisation in arrays has been approached by a number of researchers in recent years. Cruz et al. (2009) used an exhaustive approach using the methodology described in Scenario 1A. However, local optimisation algorithms have also been used successfully for this purpose. Bellew et al. (2009) used a Gauss-Newton method to solve the problem and De Backer et al. (2009) used a sequential quadratic programming (SQP) algorithm. These routines have the advantage that they are much quicker than exhaustive search of the solution space, particularly when the number of variables gets larger than a small number. They also typically lead to a much more accurate solution than an exhaustive search which is limited by the step-size.

One disadvantage of local optimisation algorithms is that there is a possibility that they converge to a local optimum, rather than a global optimum. This occurs when the solution is more favourable than those in the immediately surrounding parameter space but is less so compared to some other solution contained in the entire parameter space. It is the nature of the problem that will determine if such solutions are possible with they likelihood of finding one rising for more multi-peaked solution spaces. De Backer et al. mention the possibility of this occurring in the current problem, recommending a judicious choice of initial point to make convergence on the global solution more likely. However, preliminary investigations performed here with a number of starting points have indicated that the solution has low sensitivity to this choice. Hence a local optimisation algorithm is applied in the present circumstance without undue concern as to whether or not a global optimum has been reached. Nevertheless, out of an abundance of caution, the initial solution is chosen to represent a reasonable estimate of the final solution (all PTO coefficients set to 640kNs/m, the optimum for an isolated device in the sea-state).

More specifically, the method is provided by the function ‘fmincon’ from the Optimization Toolbox associated with the MATLAB programming environment. This is designed to find the optimum in constrained nonlinear multivariable problems such as the one under consideration. The medium-scale line search algorithm used by the function employs an SQP method. This involves the solution of a quadratic programming (QP) sub-problem at every iteration, using the Kuhn-Tucker (KT) equations (The Mathworks, Inc., 2010).

The set-up for the demonstration of this algorithm is identical to that described under the same direction in Scenario 1A. Since the values of the design variables are generated automatically, it is also necessary to impose:

- Constraints: Each PTO damping setting must satisfy $0 \leq B_{pto} \leq 1000$ kNs/m

Results are shown for this optimisation in Section 8.3.2.

The development of the code may include further investigations as to how appropriate local optimisation algorithms are in the current problem. Multiple starting points may be used to increase the likelihood of finding the true global optimum. Alternatively a global optimisation technique such as the one described for the optimisation of array layout may be employed.

Layout optimisation

The solution space for the layout optimisation is multi-peaked, due to the spatial periodicity of the wave field emanating from each of the devices. Furthermore, there are a great number of such local optima. Therefore a local optimisation technique is unlikely to converge to the global optimum and so is not appropriate in this circumstance. One class of global search procedure is the Genetic Algorithm (GA) which takes its inspiration from the theory of evolution. An example of this type of method has been successfully applied by Child and Venugopal (2010) for the optimisation of array layout in regular wave climates. A GA has therefore been selected as the first optimisation method to be implemented for the current problem.

In a genetic algorithm, a group of solutions ‘evolve’ in a guided semi-random process towards an optimum. Solutions (also known as ‘individuals’) are represented by a number of defining variables called ‘genes’. A collection of solutions termed a ‘population’ is initiated, after which there follows sequentially several other populations at stages called ‘generations’. After each is created, the value of the objective function is calculated for each individual in the population with the best ones more likely to be selected to proceed. These then form the basis of individuals belonging to the next generation by three reproduction mechanisms: ‘crossover’, ‘mutation’ and ‘elitism’ as illustrated in Figure 7.5. Crossover involves creating new individuals combining the features of two ‘parent’ individuals; Mutation involves a random perturbation to the genetic make-up of a single individual. The highest-rated ‘elite’ individuals pass into the next generation unchanged. After certain stopping criteria have been met, the final solution is the most highly-rated solution in the final generation. For more details, see Mitchell (1998).

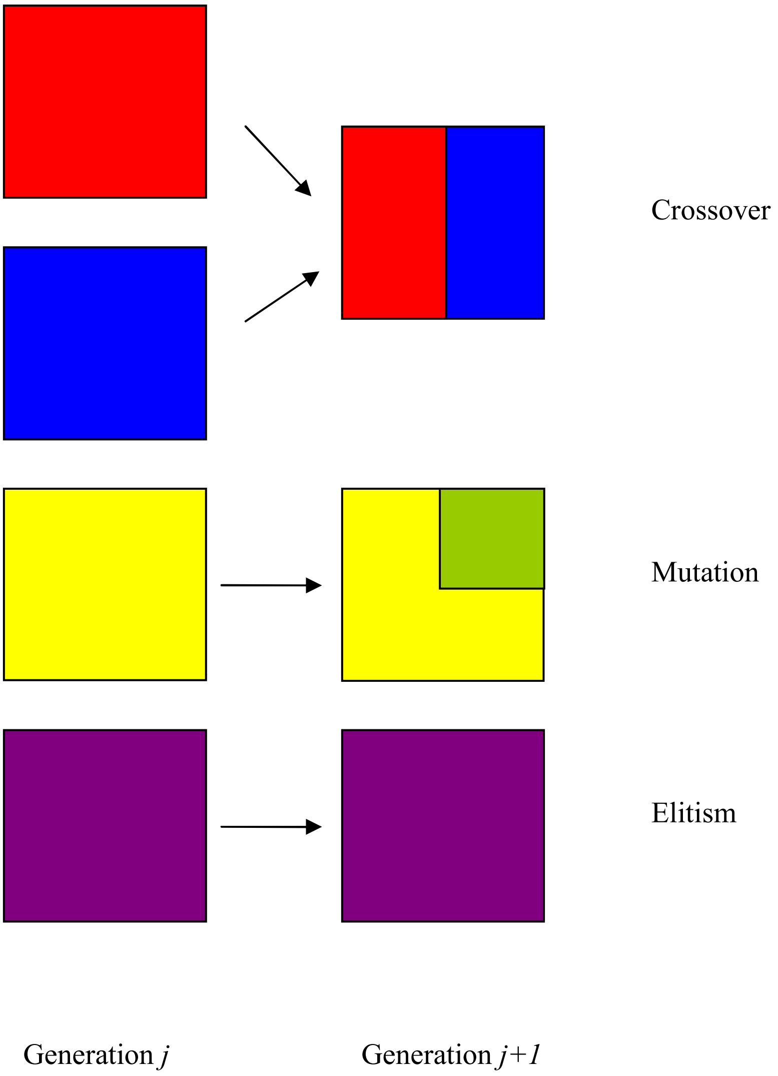


Figure 7.5: Genetic Algorithm reproduction operators

The algorithm has been implemented using the Genetic Algorithm and Direct Search Toolbox of MATLAB. There are several parameters relating to the operation of the algorithm that can have a significant impact on the performance of the optimisation. Detailed analysis has not been undertaken here in order to determine these (this may be addressed more fully in WG1 WP1 D3); where there is no clear choice for their values, the settings used by Child and Venugopal (2010) have been adopted has a starting point. The following therefore summarises the settings adopted:

- PopulationType: DoubleVector. The vector [M, p, s, ψ, σ] of real numbers was used to represent each solution, as described in Section 7.1.2. The map M→ceil(M) was used to ensure that the number of rows is an integer.
- PopulationSize: 50

- Generations: 20; StallTimeLimit: Inf
- PopInitRange: [eps L_0 L_0 eps eps; N 20*a 20*a 2*pi 2*pi] where L_0=4*a* is the minimum separation between devices; CreationFcn: @gacreationuniform
- FitnessScalingFcn: @fitscalingprop; SelectionFcn: {@selectiontournament, 4}
- EliteCount: 2; CrossoverFraction: 0.8
- MutationFcn: @mutationadaptfeasible
- CrossoverFcn:{@crossoverheuristic,1.2}

The evaluation of individual layouts requires WAMIT to be called every time. This is a time consuming process since the objective function is assessed a great number of times. Hence the evaluation needs to run as quickly as possible. One way of reducing the computation time is to increase the panel size used in the discretisation of the body surface in WAMIT. Although this makes the solution less accurate, what is more important for the optimiser is the *relative* performance of different layouts. If any array that performs better than another with the course grid resolution shows the same property using the fine resolution, then the optimiser may distinguish which are the best arrangements adequately with the larger panel size. A similar procedure is applied in another GH tool (WindFarmer), where a fine resolution is only implemented in final layout selection (for an accurate estimation of performance related variables), but not in the layout selection process. Initial tests have indicated that this is indeed the case and thus the value PANEL_SIZE=20 has been used for these calculations. A significant improvement in optimiser performance has consequently been observed. Further investigations into this may be carried out in the course of the development of the software. However, in Scenario 2A, the need for WAMIT is removed since the interaction calculations will be performed within the software. This could lead to further improvements in the performance of the optimiser.

The set-up for the optimisation of the layout is the same as was detailed in Scenario 1A. The exception is that the layout may now vary in line with the regular grid definition of Section 7.1.2 as a design variable. Therefore the following must also be applied:

- Constraints (as described in Section 7.1.2): Lower bounds on solution vector [eps L_0 L_0 eps eps] where L_0=4*a* is the minimum device separation. Note that eps is used instead of 0 to prevent *M*=0 and to prevent multiple angles defining the same array. Upper bounds [N Inf Inf 2*pi 2*pi]. The nonlinear constraints on the distance between devices diagonally across grid cells are enforced by setting the objective function to zero for arrangements that violate them.

Results are shown for this optimisation set-up in Section 8.3.2.

8 CASE STUDIES

In this section preliminary results associated with the frequency-domain functionality of both the loads calculations and the wave farm design tools are presented. An effort was made to ensure that these preliminary studies are aligned with the verification scenarios listed in WG0 D1, in particular the array of four point-absorbers. It is expected that this will ensure that comparisons related to initial results from different project partners (GH, UoOx and WUB) can be more easily made. As listed in WG1 WP1 D1B, these cases should not be confused with a set of representative scenarios for which results will be presented in WG1 WP1 D4 and D15 (versions A and B). Finally, the implementation case studies allowed the test of the functionality of each module, and the definition of the immediate next steps for extension of the core functionality and the time-domain implementation (Section 9.2).

8.1 Definition of the case studies

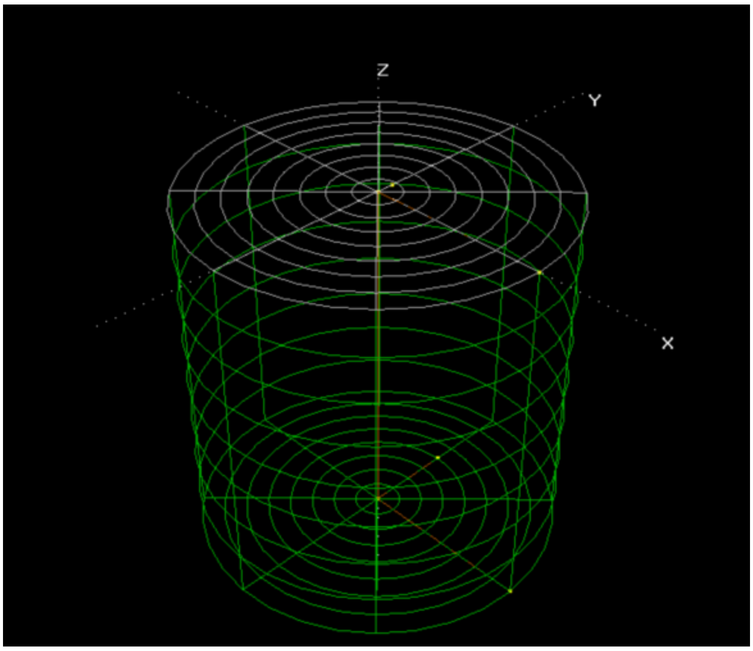
The case studies presented in the following subsections have been defined with the primary objective of showing the baseline functionality of the software tools under the limitations of a frequency-domain approach. The detail in the description and in the analysis of the results relative to each case study is such that the implementation can be replicated by a third-party. The following key aspects are covered in the case studies:

- The ability of the tool(s) to define different FDCs with different characteristics;
- The influence of different wave input conditions in the response of a WEC;
- The influence of the array layout in the power absorption characteristics;
- The influence of the control of the PTO of each array element in the power absorption characteristics (by treating the array as a power plant).

To cover the first point listed above, two different concept geometries were defined: one which is characteristic of a point-absorber, and another one which is characteristic of an attenuator. Both FDC types were therefore modelled for the loads calculations and performance characterisation exercise (Section 8.2). The geometry of the first FDC (point-absorber) was then used in the first array layout exercise (Section 8.3).

The numerical pre-processing methodology involved the use of a CAD package to estimate the mass matrix of the WEC, the use of a second CAD package for the creation of a detailed mesh and the geometry definition files for the flow. Internally GH uses SolidWorks and MultiSurf for the above described tasks (two commercially available packages). The remaining input files for the flow solver, from which the baseline hydrodynamic characteristics of the geometry are derived, are then created and the flow solver can in turn be used (internally GH uses WAMIT for first-order hydrodynamic calculations). However the developed tools are able to load solutions from other packages if minor extensions to the pre-processing routines are implemented.

Figure 8.1 and Figure 8.2 define the geometries under considerations. Outputs from the different CAD packages utilised are presented; these include the numerical mesh created for the point-absorber (Figure 8.1) and a 3-D rendering of the attenuator design (Figure 8.2). The mesh of one of the array layouts simulated in Section 8.3 is also presented in the bottom of Figure 8.2. Finally, the basic physical properties (geometrical and mass information) for each FDC are listed in Table 8.1 and Table 8.2.



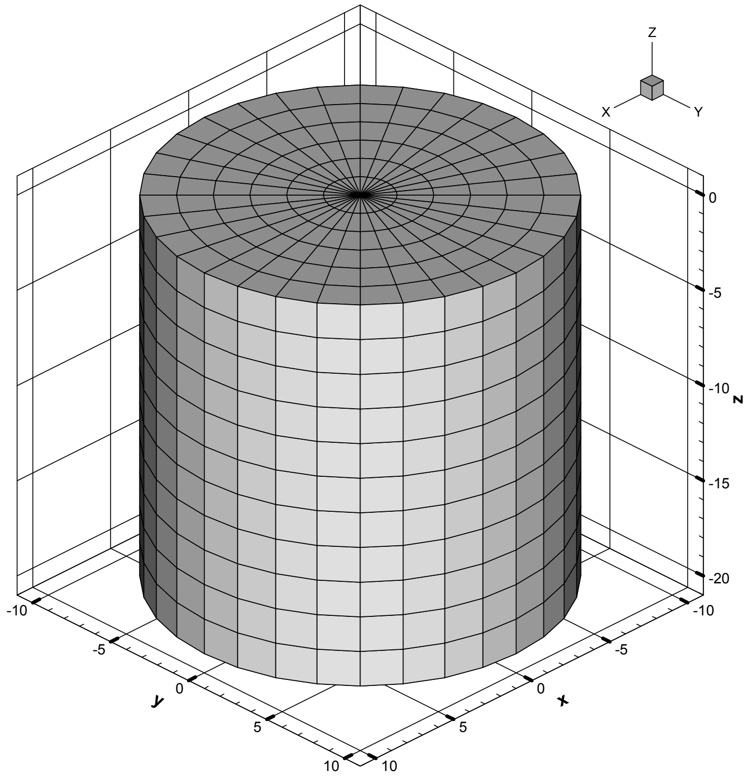


Figure 8.1: Graphical representation of the mesh (top) and WAMIT .GDF file (bottom); case study #1 (point-absorber)

Table 8.1: Definition of the physical properties of case study #1 (point-absorber)

	Truncated Cylinder
Diameter (m)	20
Draft (m)	20
Mass (Ton)	1455.6
Ballast (Ton)	4987.7

Table 8.2: Definition of the physical properties of case study #2 (attenuator)

	Long cylinder	**Short cylinder**
Diameter (m)	3.50	3.50
Length (m)	30.00	5.00
Total Volume (m^3)	288.63	48.11
Displaced mass of water (kg)	147925	24654
Outer Shell (Hull):		
Outer shell thickness (m)	0.025	0.025
Outer shell volume	8.188	1.365
End cap volume	0.241	0.241
Outer shell mass (kg)	67617	14396
Inner Ballast:		
Mass (kg)	80308	10258
Outer diameter (m)	3.449	3.449
Inner diameter (m)	3.227	3.280

Notes:

Densities (kg/m3)	
Carbon steel	7800
Sea water	1025
Concrete	2300

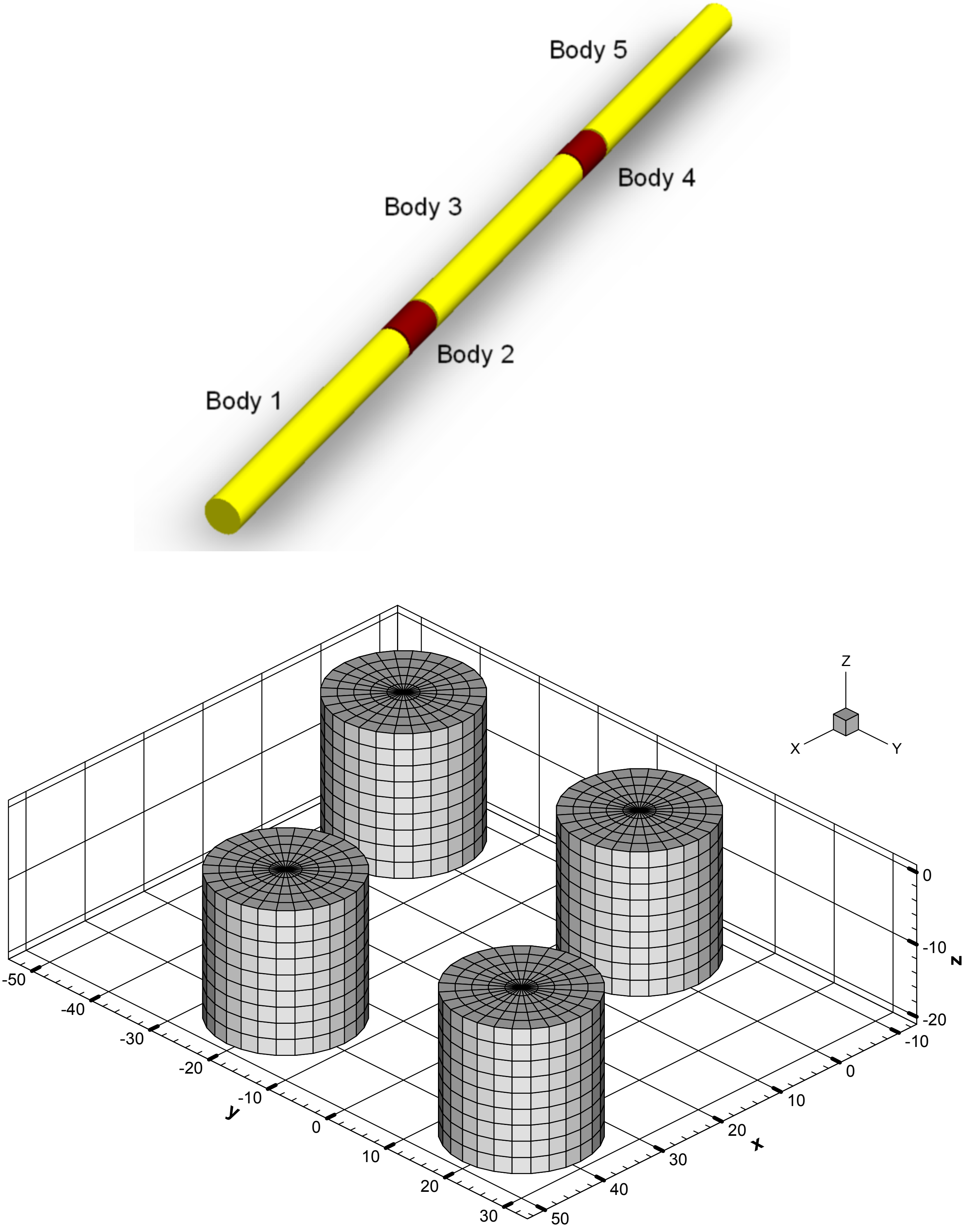


Figure 8.2: 3-D rendering of the case study #2 geometry (attenuator - top) and 3-D mesh of an array of point-absorbers (bottom)

8.2 Preliminary results – single FDCs

Having defined the physical properties, the geometry and the relevant pre-processing files hydrodynamic data were obtained for the concepts defined in Section 8.1. In particular, added-mass, radiation damping and the wave exciting force were obtained for each body that constitutes each WEC.

A simulation model was created for each FDC using the core elements presented in Section 2.2. Figure 8.3 presents a multi-body representation of the point-absorber and the attenuator defined in Section 8.1. The use of the different elements available (such as bodies, rigid links, slides and joints) is clear from the graphical representation of the model. Note that all the hydrodynamic data is loaded per body element. Both simulations shared the same wave input (in a 40m deep, flat seabed environment): regular wave with periods spanning from 5 to 17s were tested along with a range of Bretschneider spectra. Head-on sea states were considered (with no spreading).

The PTO modes were limited to heave relative to the seabed for the point-absorber case, and relative pitch between each body for the attenuator case (in a total of four independent PTO modes for such case study). The point-absorber is effectively a one DOF model (only allowed to have), while the attenuator is a ten DOF model.

Figures 8.4 to 8.6 show the preliminary results for the single point-absorber case defined in Section 8.1. Firstly variables related to regular wave calculations are presented, namely:

- The heave response amplitude operator (RAO) – Figure 8.4;
- The PTO force RAO – Figure 8.5;
- The relative capture width (RCW) – Figure 8.6

Note that all the above variables were defined in WG1 WP1 D1b and can be considered the key loading and performance indicators under regular waves.

The results are presented for a total of three PTO damping conditions, is a purely linear (passive, frequency independent – see Section 6.3) control methodology. The influence of the PTO damping coefficient is clear in all the variables listed above.

Overall the results are well aligned with the expectations; in particular, and under all the simplify assumptions and the conservative control approach, it was expected that a heaving point-absorber would respond in a relatively narrow bandwidth and with RCW barely exceeding 1. The results have been verified against the MATLAB simulations previously conducted by GH for the EWTEC 2009 conference and presented in Cruz et al. (2009), showing excellent agreement. Finally, irregular wave results are presented in Table 8.3 for the point-absorber case. These show the ability of the tool to derive power absorption estimates which are dependent on the input sea state. Note that under the frequency-domain approach (and its associated simplifications) further irregular wave results can be obtained by recognising that for any given wave period the results are directly proportional to H_s^2.

Such result provides increased confidence in the new C++ formulation, which is more robust and allows the definition of more complex WECs. A five-body, attenuator FDC was defined (see also Section 8.1). Preliminary results are presented in Figures 8.7 and 8.8, and were obtained following a similar input methodology (albeit different PTO settings were used, owing to the different PTO mode when compared to the point-absorber case). Irregular wave power absorption estimates are also presented in Table 8.3, for the same three sea states tested with the point-absorber. These preliminary results allow an assessment of the code's ability to model a range of structures under varied input conditions and technical constraints.

Table 8.3: Irregular wave results for case study#1 (point-absorber) and case study #2 (attenuator)

		Point-absorber	**Attenuator**
H_s (m)	T_e (s)	P_{abs} (kW)	P_{abs} (kW)
0.5	5	3.4	2.6
2.5	7	140.8	70.3
5	10	475.5	131.1

Notes:

- Spectral shape: Bretschneider (head-on direction, no spreading)
- PTO damping coefficient for the point-absorber: 0.5MNs/m
- PTO damping coefficient for the attenuator: 50MNms (per PTO mode)

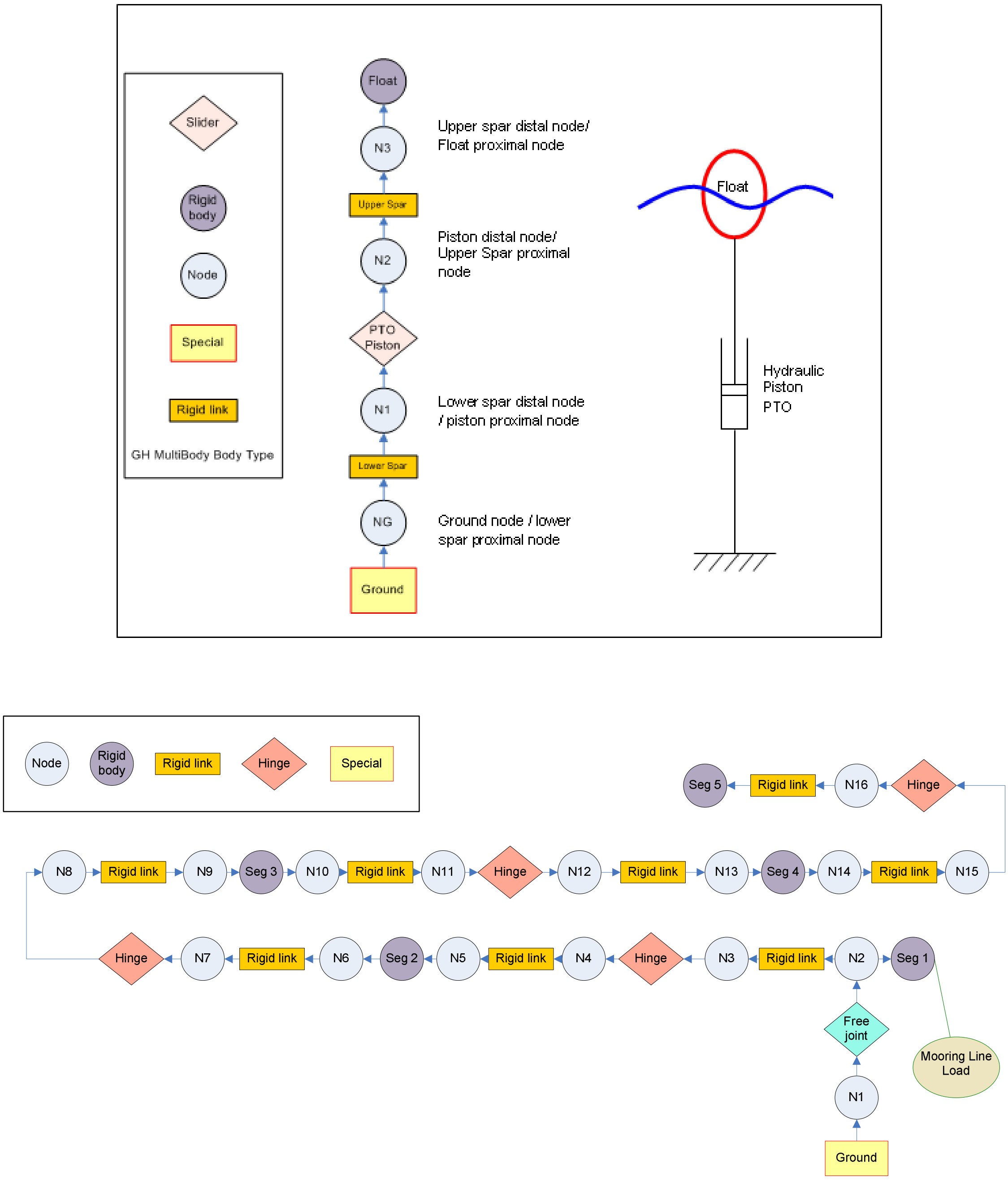


Figure 8.3: A multi-body representation of a case study #1 (point-absorber – top) and case study #2 (attenuator – bottom).

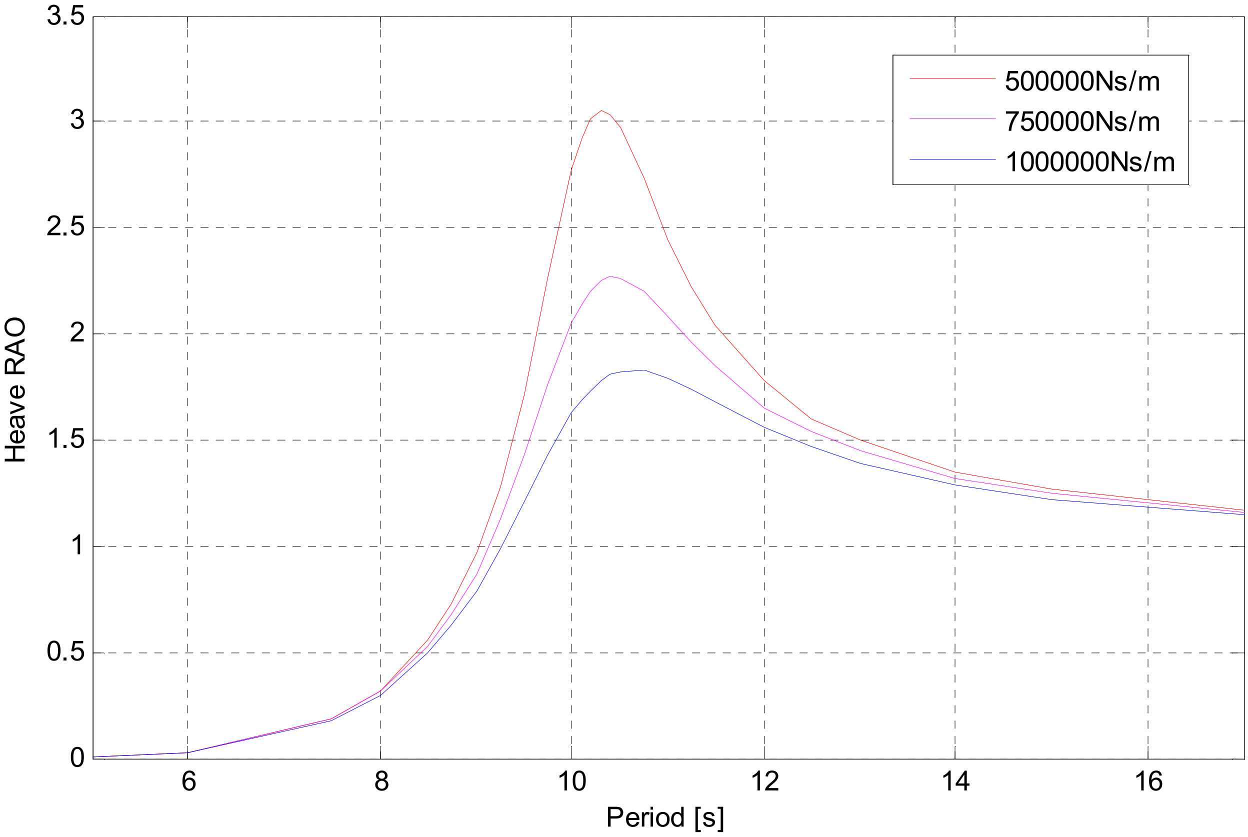


Figure 8.4: Heave RAO – case study #1 (point-absorber)

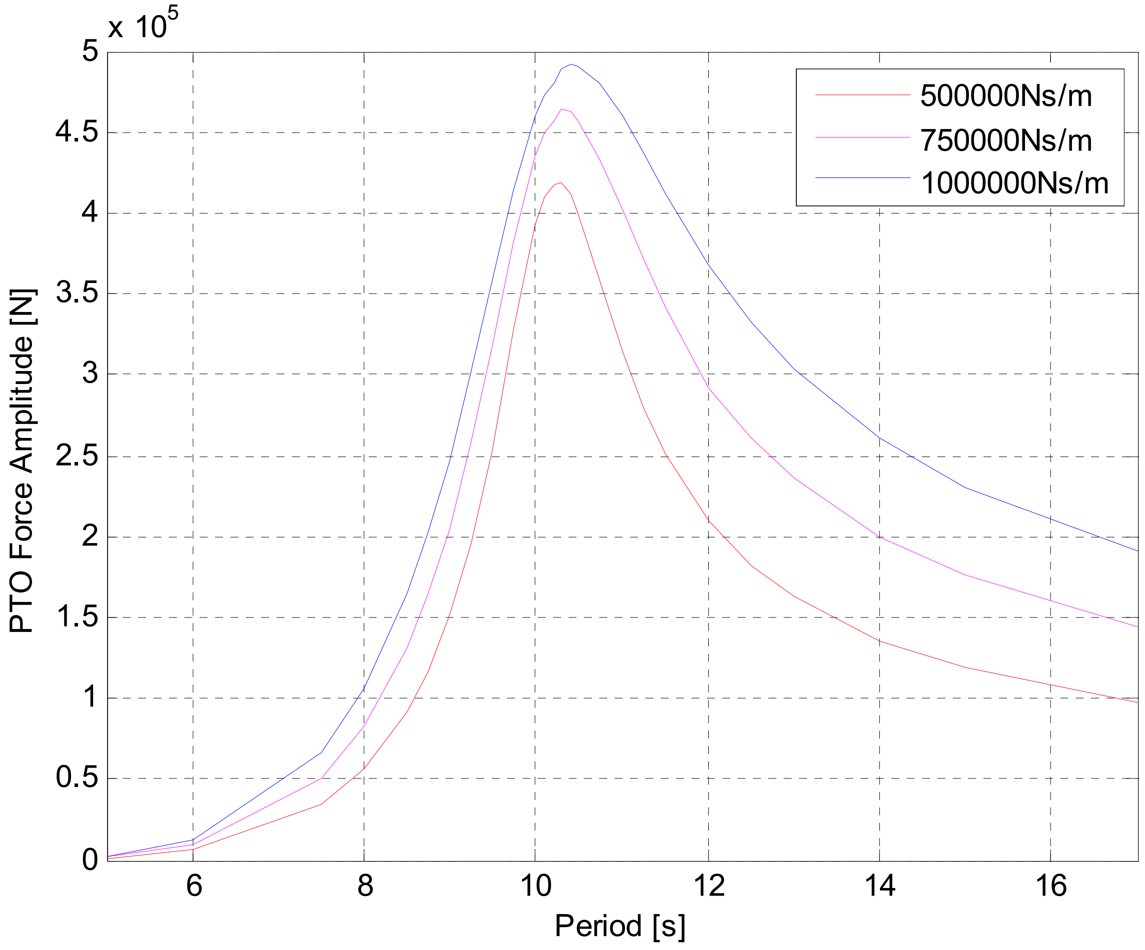


Figure 8.5: PTO force RAO – case study #1 (point-absorber)

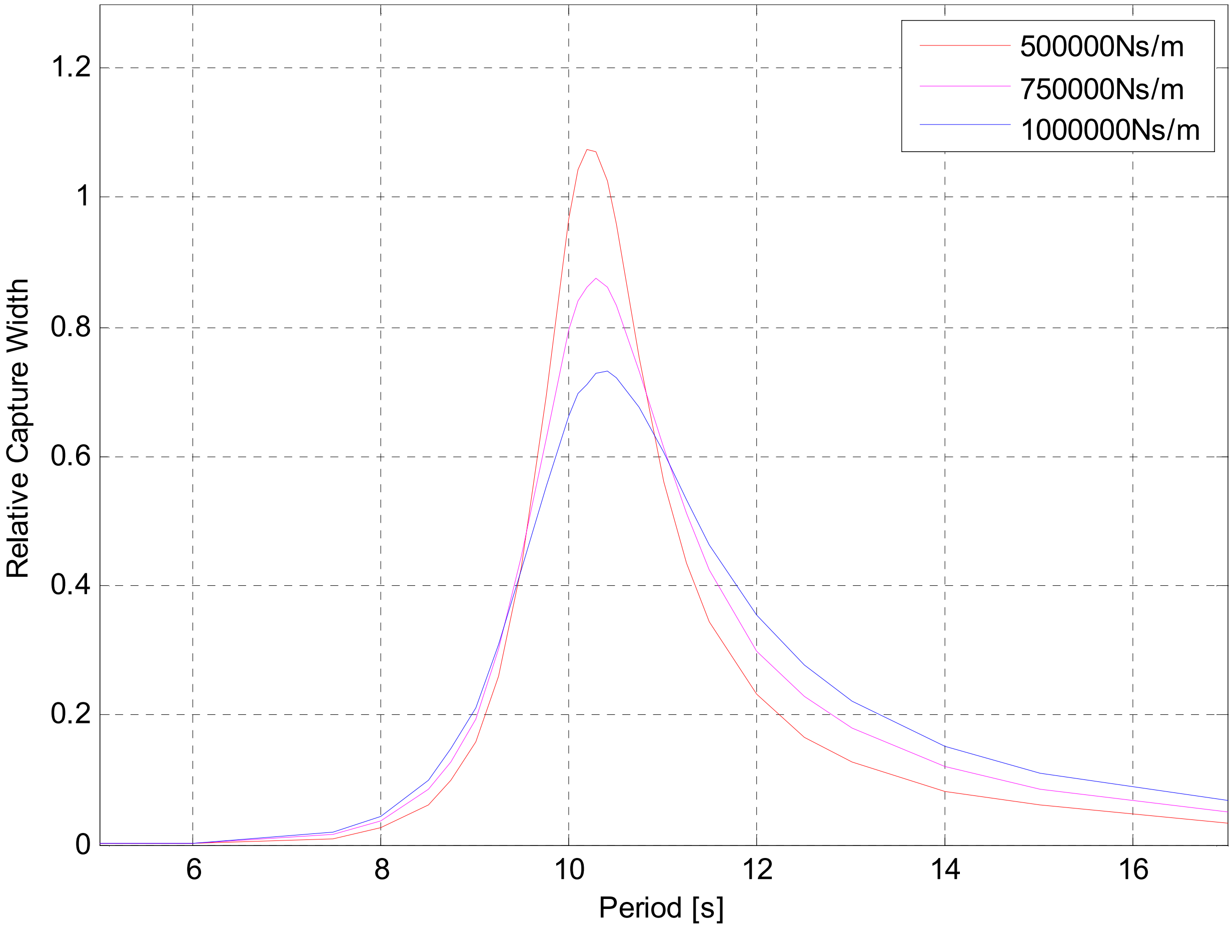


Figure 8.6: RCW – case study #1 (point-absorber)

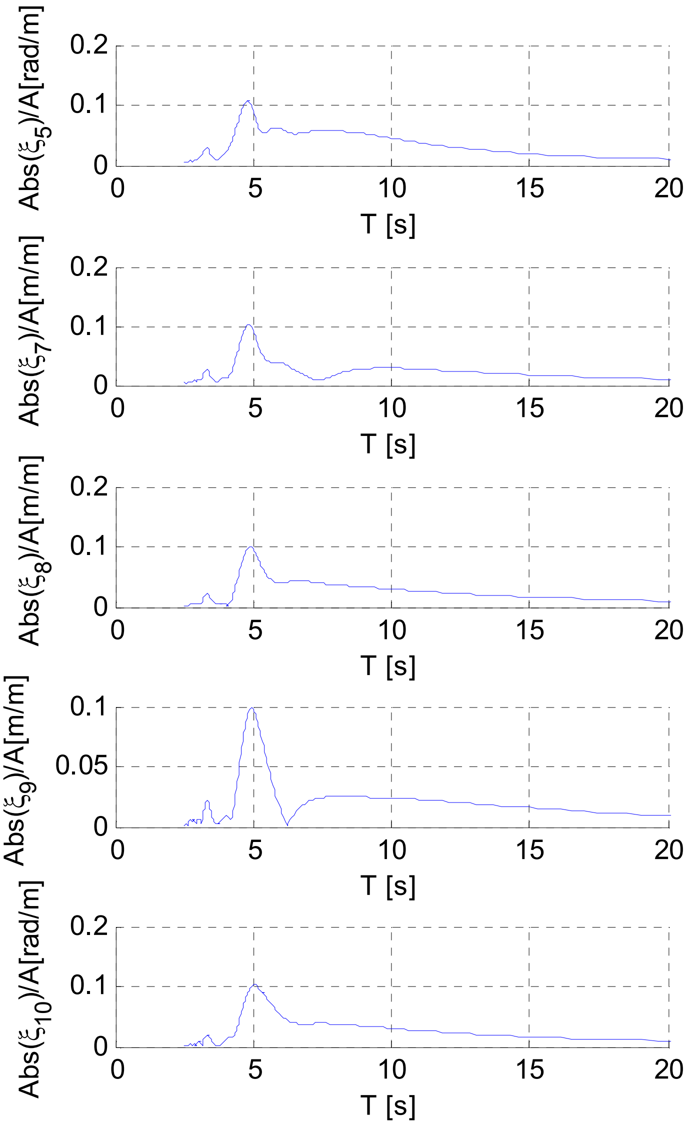


Figure 8.7: Pitch RAO – case study #2 (attenuator); PTO damping: 50MNms (per PTO mode)

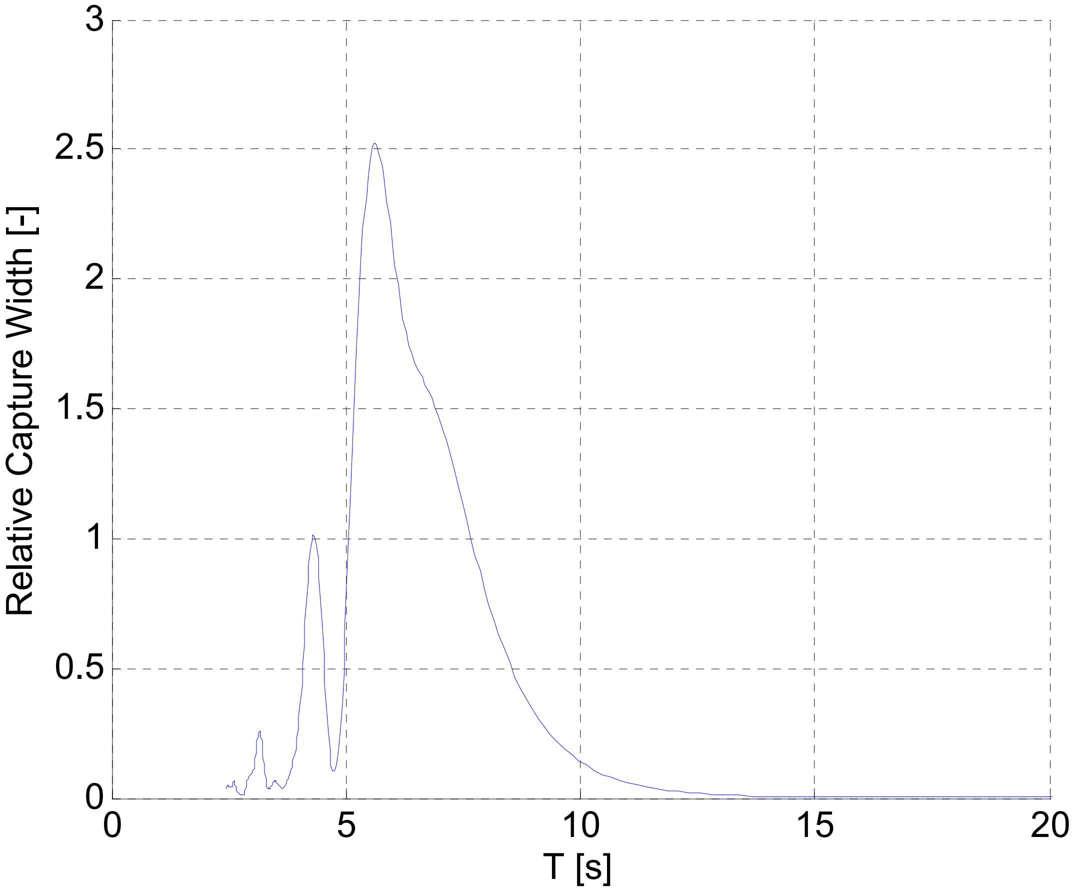


Figure 8.8: RCW – case study #2 (attenuator) ; PTO damping: 50MNms (per PTO mode)

8.3 Preliminary results – arrays of FDCs

With regard to arrays of FDCs, the set-ups for the case studies were described in Section 7.3 and are summarised here in Table 8.4. In the following subsections, the results of each case study will be presented and discussed.

Property	Scenario 1A			Scenario 1B	
	Layout		PTO	PTO	Layout
	Set-up I	Set-up II			
PTO	All 640kNs/m	All 640kNs/m	*Design variables* (discrete values for each)	*Design variables* (continuous variables for each)	All 640kNs/m
Layout	Square, separation: $L=4a=40$m	Staggered, separation: $L=4a=40$m	Square, separation: $L=4a=40$m	Square, separation: $L=4a=40$m	*Design variables*
Wave direction	$\beta=\pi$	$\beta=\pi$	$\beta=5\pi/4$	$\beta=5\pi/4$	$\beta=0$
Wave climate	Regular waves ($6s \leq T \leq 20s$), Bretschneider (H_{m0}=4m, T_p=10.2s)	Regular waves ($6s \leq T \leq 20s$), Bretschneider (H_{m0}=4m, T_p=10.2s)	Bretschneider (H_{m0}=4m, T_p=10.2s)	Bretschneider (H_{m0}=4m, T_p=10.2s)	Bretschneider (H_{m0}=4m, T_p=10.2s)
N	4				
WEC geometry	Truncated vertical cylinder				
Radius	a=10m				
Draft	d-h=2a=20m				
Mass	Neutrally buoyant at rest				
Bathymetry	Flat, water depth: d=4a=40m				
Moorings	None				

Table 8.4: Set-ups for Array case studies

8.3.1 Scenario 1A

Layout optimisation

Set-up I

The power absorbed by each individual WEC (normalised by the square of the wave amplitude) is shown in Figure 8.9 as a function of wave period. Only two curves are shown because of the symmetry of the array at this wave direction. The mode index in the legend refers to the heave motion of each device in the 6N degrees of freedom of the array (that is to say the mode of body n is 6(n-1)+3). The graph clearly shows the down-wave devices (Modes 3 and 15) performing less well than the up-wave devices (Modes 9 and 21) due to the sheltering effect of the latter converters. Figure 8.10 shows the total power from all WECs in regular waves. The net power produced by the array in the Bretschneider sea-state is 1.509MW.

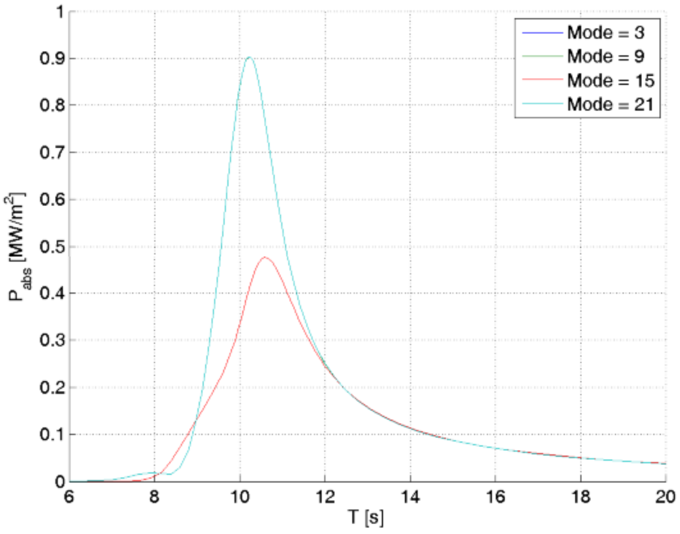


Figure 8.9: Power absorbed by each device in Set-up I.

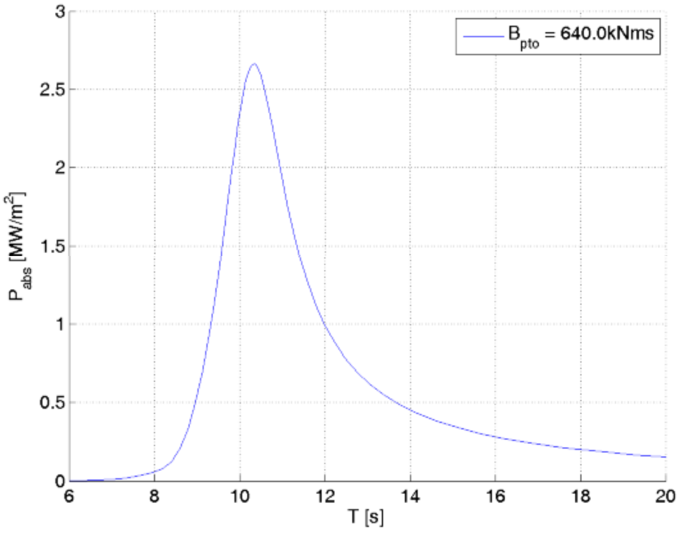


Figure 8.10: Total power absorbed in Set-up I.

Set-up II

Figure 8.11 shows the individual power values from the devices in the array. It can be seen that all four devices perform differently because the symmetry of set-up I has been broken. Also the up-wave devices (2 and 4) perform better than their down-wave counterparts. Figure 8.12 shows the total power from this array, which is not significantly different from set-up I because the power from the up- and down-wave rows average out to approximately those of the square array. The net power from the array in the Bretschneider sea-state is 1.506MW.

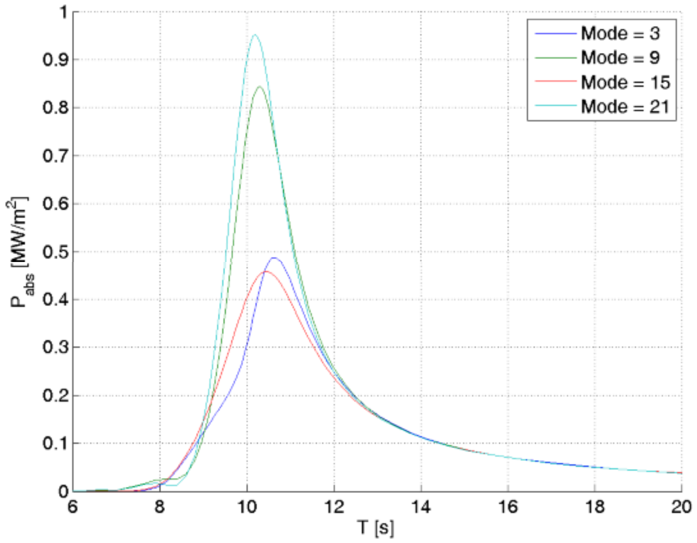


Figure 8.11: Power absorbed by each device in Set-up II.

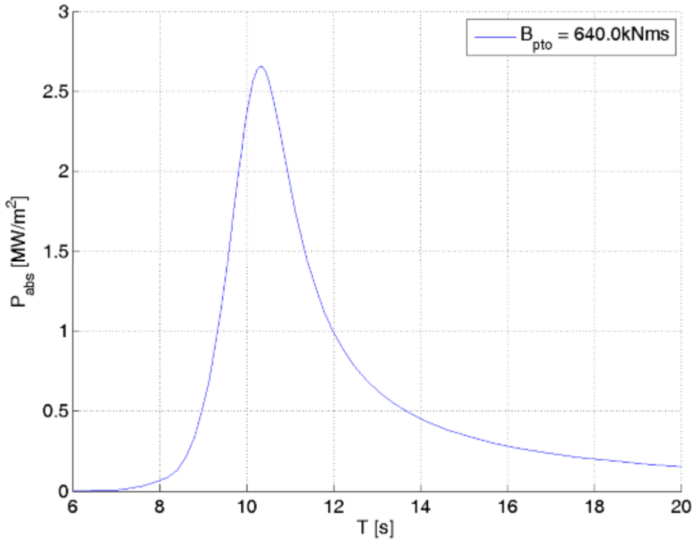


Figure 8.12: Total power absorbed Set-up II.

PTO optimisation

Figure 8.13 shows the total power output from the array in the given sea-state using the combination of PTO settings for devices 1-4 given on the *y*-axis as horizontal bars. These results with the course step-size show that altering the PTO coefficients can significantly affect the power output from the array.

Figure 8.14 is a plot of the same quantities except that the step-size has been reduced as previously specified. Text output from the software in this case is also given below

```
>> Maximum Array Power in Bretschneider = 1570235.19W

At Maximum Array Power, Power for PTOMode 3 in Bretschneider = 395602.4106W

This occurs for a damping of 750000Nms

At Maximum Array Power, Power for PTOMode 9 in Bretschneider = 506279.0917W

This occurs for a damping of 650000Nms

At Maximum Array Power, Power for PTOMode 15 in Bretschneider = 272617.1651W

This occurs for a damping of 850000Nms

At Maximum Array Power, Power for PTOMode 21 in Bretschneider = 395736.5226W

This occurs for a damping of 750000Nms
```

It can thus be seen that the best PTO settings may be different for each element of the array (750, 650, 850, 750kNs/m for devices 1-4). In fact, the difference between the highest and the lowest values is over 30%. Moreover, the optimal PTO values for devices in the array in this sea-state are all different to the optimal PTO value for an isolated device in the same sea-state (640kNs/m).

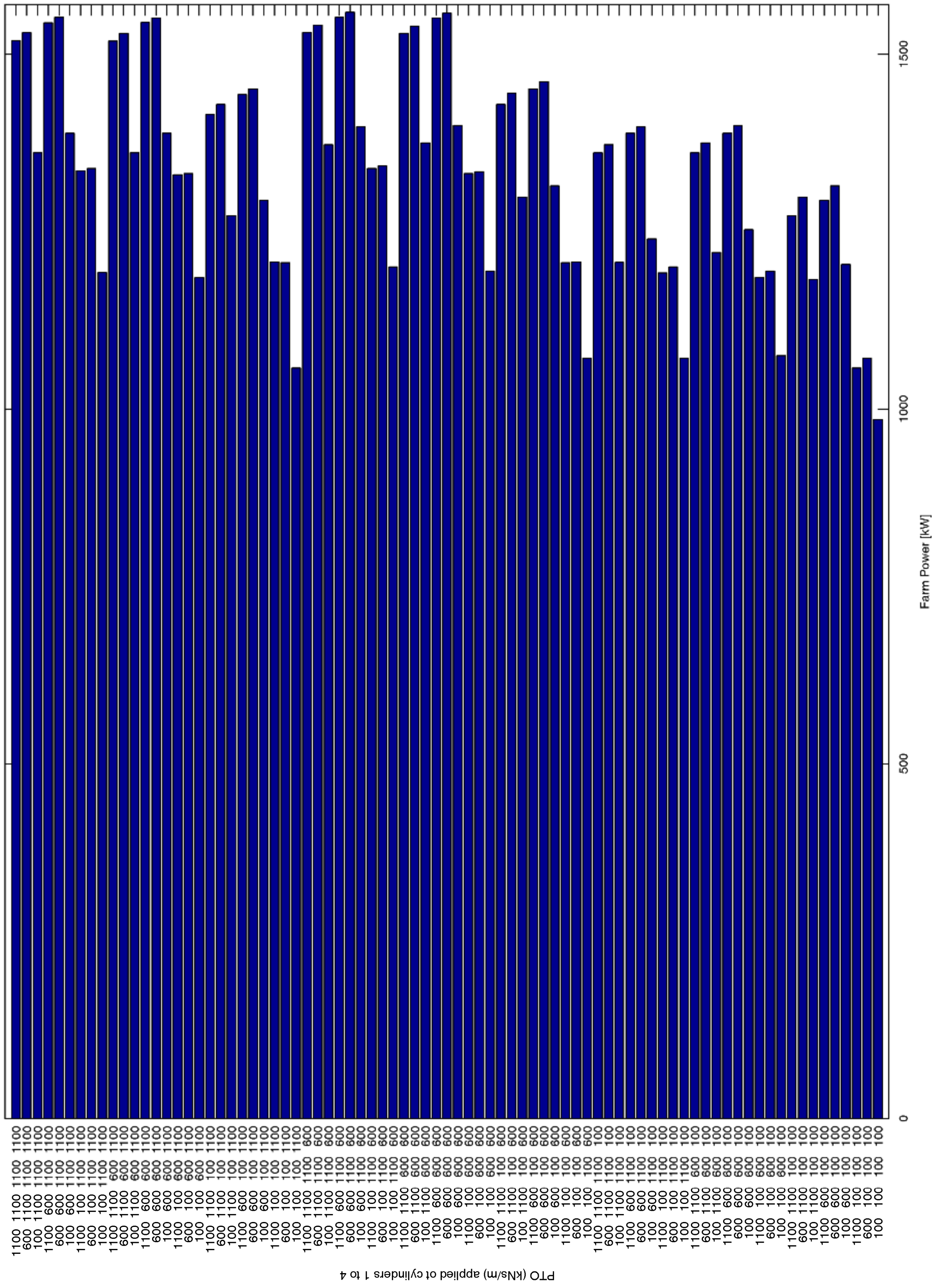


Figure 8.13: Power generated from an array of 4 devices with different PTO damping settings for each device (Course step-size).

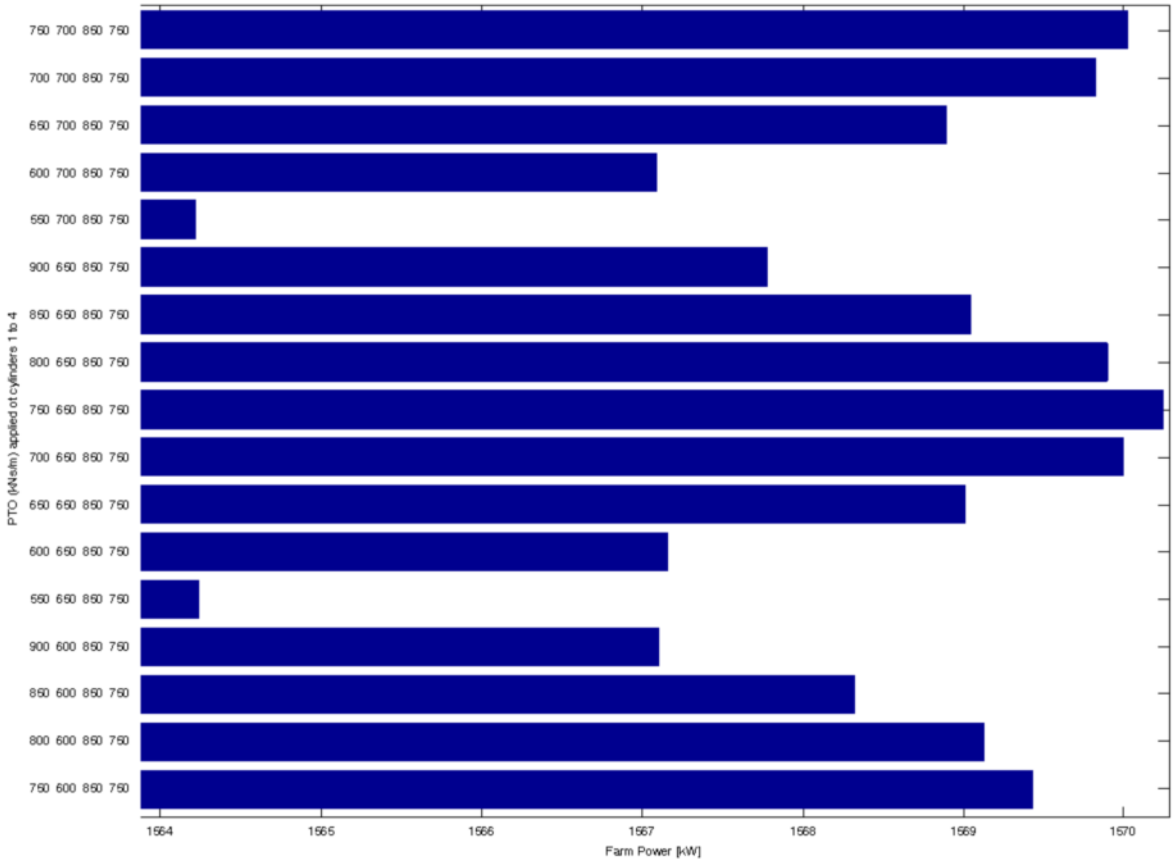


Figure 8.14: Power generated from an array of 4 devices with different PTO damping settings for each device (Fine step-size, not all values shown).

8.3.2 Scenario 1B

PTO Optimisation

The text output from the optimisation is shown below:

```
x0 =


      640000       640000       640000       640000


                                           max     Line search  Directional   First-order
 Iter F-count         f(x)     constraint   steplength   derivative   optimality Procedure
    0       5 -1.56375e+006    -3.6e+005
    1      10 -1.56375e+006    -3.6e+005            1     -0.00367       0.0382
    2      15 -1.56375e+006    -3.6e+005            1       -0.184       0.0382  Hessian modified
twice
    3      20 -1.56933e+006  -2.674e+005            1    -2.54e+003       0.0171  Hessian modified
    4      25 -1.56991e+006  -2.304e+005            1         -171       0.0102
    5     30 -1.56993e+006   -2.229e+005            1         -2.1       0.0107
    6     35 -1.56993e+006   -2.224e+005            1     -0.00189       0.0107
    7     40 -1.56993e+006   -2.224e+005            1    -0.000207       0.0107
    8     45 -1.56993e+006   -2.224e+005            1     -0.00204       0.0108  Hessian modified
    9     50 -1.56993e+006   -2.223e+005            1     -0.00411       0.0108
   10     55 -1.56993e+006   -2.222e+005            1     -0.00997       0.0108
   11     60 -1.56993e+006    -2.22e+005            1       -0.029       0.0108
   12     65 -1.56993e+006   -2.217e+005            1      -0.0715       0.0108
   13     70 -1.56993e+006   -2.213e+005            1       -0.193       0.0109
   14     75 -1.56993e+006   -2.205e+005            1       -0.502        0.011
   15     80 -1.56993e+006   -2.191e+005            1         -1.3       0.0111
   16     85 -1.56994e+006   -2.167e+005            1        -3.38       0.0113
   17     90 -1.56995e+006   -2.126e+005            1        -8.54       0.0114
   18     95 -1.56998e+006   -2.053e+005            1        -20.8       0.0112
   19    100 -1.57004e+006   -1.934e+005            1        -44.4       0.0099
   20    105 -1.57014e+006   -1.777e+005            1        -70.9      0.00643
   21    110 -1.57026e+006   -1.674e+005            1        -60.1      0.00497
   22    115 -1.57031e+006   -1.704e+005            1        -22.4      0.00301
   23    120 -1.57032e+006   -1.772e+005            1        -2.88      0.00306
   24    125 -1.57032e+006   -1.793e+005            1      -0.0657      0.00275
   25    130 -1.57032e+006   -1.794e+005            1    -0.000374      0.00269
   26    135 -1.57032e+006   -1.794e+005            1   -2.05e-005      0.00269
   27    140 -1.57032e+006   -1.794e+005            1   -2.99e-005      0.00269  Hessian modified
   28    145 -1.57032e+006   -1.794e+005            1   -7.91e-005      0.00268  Hessian modified
   29    150 -1.57032e+006   -1.794e+005            1    -0.000259      0.00268  Hessian modified
   30    155 -1.57032e+006   -1.793e+005            1     -0.00067      0.00267  Hessian modified
   31    160 -1.57032e+006   -1.793e+005            1     -0.00149      0.00266  Hessian modified
   32    165 -1.57032e+006   -1.793e+005            1     -0.00574      0.00263  Hessian modified
   33    170 -1.57032e+006   -1.793e+005            1      -0.0139      0.00259
   34    175 -1.57032e+006   -1.792e+005            1      -0.0371      0.00252
   35    180 -1.57032e+006    -1.79e+005            1      -0.0945       0.0024
   36    185 -1.57032e+006   -1.787e+005            1       -0.251      0.00221
   37    190 -1.57032e+006   -1.781e+005            1       -0.634      0.00221
   38    195 -1.57032e+006   -1.769e+005            1        -1.58      0.00235
   39    200 -1.57033e+006   -1.746e+005            1        -3.57      0.00243
   40    205 -1.57034e+006   -1.706e+005            1        -6.58      0.00223
   41    210 -1.57035e+006   -1.657e+005            1        -7.59      0.00149
   42    215 -1.57036e+006   -1.632e+005            1        -3.84      0.00104
   43    220 -1.57036e+006   -1.637e+005            1       -0.698     0.000326
   44    225 -1.57036e+006   -1.645e+005            1      -0.0357    2.83e-005
   45    230 -1.57036e+006   -1.646e+005            1    -0.000431    3.43e-006
   46    235 -1.57036e+006   -1.646e+005            1    -8.8e-008    1.45e-006
Optimization terminated: magnitude of directional derivative in search
 direction less than 2*options.TolFun and maximum constraint violation
```

```
  is less than options.TolCon.
No active inequalities.


x =


  1.0e+005 *


    7.4357    6.6356    8.3538    7.4347



fval =


 -1.5704e+006
```

Thus the optimal PTO settings for devices 1-4 are 744, 664, 835, 743kNs/m, giving rise to a total power output of 1.570MW. Recall that the exhaustive search returned optimal settings of 750, 650, 850, 750kNs/m and an output of 1.570MW. Hence two sets of results coincide to within a single multiple of the step-size used in the latter (to the nearest 50kNs/m). It is also interesting to note the number of function evaluations (that is to say, computational burden) needed to find the solution; in Scenario 1A, 8^4=4096 iterations were needed compared to just 235 here. It is therefore apparent that the local optimisation technique adopted here is preferable to an exhaustive approach.

Layout optimisation

The text output from the optimisation is shown below:

```
                               Best           Mean      Stall
Generation      f-count        f(x)           f(x)    Generations
    1             50     -1.839e+006      -1.52e+006        0
    2            100     -1.849e+006     -1.524e+006        0
    3            150     -1.854e+006     -1.498e+006        0
    4            200     -1.858e+006     -1.725e+006        0
    5            250     -1.858e+006     -1.856e+006        0
    6            300     -1.858e+006     -1.845e+006        0
    7            350     -1.858e+006     -1.845e+006        0
    8            400     -1.858e+006     -1.829e+006        0
    9            450     -1.858e+006     -1.837e+006        0
   10            500     -1.858e+006     -1.826e+006        0
   11            550     -1.858e+006     -1.842e+006        0
   12            600     -1.858e+006     -1.845e+006        1
   13            650     -1.858e+006     -1.838e+006        2
   14            700     -1.858e+006     -1.844e+006        3
   15            750     -1.858e+006     -1.858e+006        4
   16            800     -1.858e+006     -1.837e+006        5
   17            850     -1.858e+006     -1.815e+006        6
   18          900     -1.858e+006     -1.858e+006       7
   19          950     -1.858e+006     -1.814e+006       8
   20         1000     -1.858e+006     -1.837e+006       9
Optimization terminated: maximum number of generations exceeded.


x =


    3.6361  122.8310   80.1209    5.5073    4.7000


fval =
 -1.8581e+006
```

The optimisation has converged to a total array power of 1.858MW. The corresponding array layout is shown in Figure 8.15 which in this case is a linear arrangement, approximately perpendicular to the oncoming wave direction (travelling from the left to the right hand side of the figure). The simplicity of the formation is most likely a result of the relatively simple optimisation exercise. However, the results may change once tighter site boundaries are imposed on the solution and more constraints are included in the optimisation process.

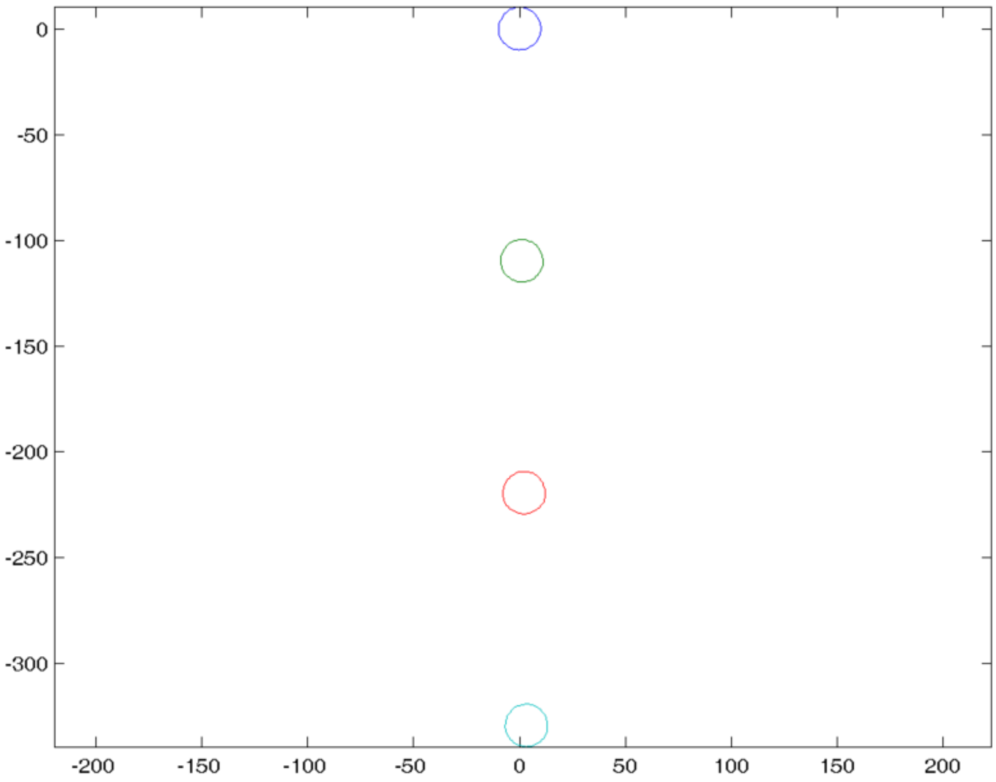


Figure 8.15: Array layout resulting from optimisation. Axes in meters and waves approaching from the left-hand side.

9 FINAL REMARKS AND NEXT STEPS

The final section of this report compiles a list of the findings which can be derived from the implementation to-date and compared with the original WG1 WP1 D2 targets, and also lists a series of next steps which can be seen as the immediate actions for the time-domains extension to be implemented under WG1 WP1 D3. As listed in WG0 D1, once the time-domain implementation is complete the core functionalities of the first beta version will be in place.

9.1 Final remarks: frequency-domain implementation report

The present report describes the current implementation status of the WEC numerical modelling software being developed by GH under the PerAWaT project. The report builds on previous deliverables, namely the Methodology Report (WG1 WP1 D1B), and details the core aspects and key functionalities of the developed code, with emphasis to performance variables related to frequency-domain simulations.

The report begins by presenting background information regarding the GH multi-body code (Bladed) used as the kernel unit to describe the WECs. Under PerAWaT GH has developed a series of software routines that plug into the Bladed code, allowing the definition of a wide range of WECs. This new package (code named WaveDyn) can be considered a loads calculation tool which allows the reduction of the uncertainty in the estimation of the relevant (single WEC) performance variables and the definition of the parameterisation of the WEC behaviour in order to use this in the optimiser module (code named WaveFarmer).

The WaveDyn multi-body modelling approach covers the scope of the ‘FD’, ‘TD’, ‘Basic Controller’, ‘External forces code’ and ‘Wave data loader’ modules originally envisaged for the WaveFarmer code structure in the Methodology Report (WG1 WP1 D1B, Section 1). The ‘Basic Controller’ and ‘External forces code’ blocks now operate as the series of parallel multi-body Calculation Modules shown in the figure on the next page, whilst the FD and TD blocks are represented in the higher level WVStructure co-ordinating code. The wave data loader is one of the I/O routines in the Hydrodynamics Calculation Module. It is envisaged that the Calculation Modules may be developed to give the user the option of replacing the core functionality of each with a DLL interface, however this interface will itself be part of the Calculation Module from an overall software architecture perspective. WaveFarmer as a software package now refers purely to the array design code (and includes the Optimiser and Mapping blocks shown in WG1 WP1 D1B). The WaveFarmer Optimiser has the ability to prescribe and analyse the results from WaveDyn simulations as described in Section 7.

Following WG1 WP1 D1B, a multi-body dynamic solver has been created, providing a means of mathematically describing the structural forces within the WEC structure. All of the non-structural forces applied to the WEC, resulting from the hydrodynamics, PTO or moorings must be calculated separately in code modules parallel to the core code (MBCore). If the flexible nature of the multi-body structural approach, which allow a range of WECs to be defined and simulated, is to be maintained then a multi-body theme must be propagated throughout these applied force calculations, with the hydrodynamic and mooring forces being calculated on a body-by-body basis and PTO calculations being associated with a particular joint in the multi-body structure.

In this way the complete software formulation for the WEC modelling code may be envisaged as a collection of parallel multi-body Calculation Modules, including MBCore:

- The geometrical/structural definition formulated using the MBCore code.
- A hydrodynamic definition containing the hydrodynamic information and force calculation functions for the wave activated bodies in the system (any wave activated structural body having a parallel hydrodynamic body containing its particular hydrodynamic properties).
- A moorings system built up from multiple mooring line 'bodies', each containing information on the line properties and attachment location on the geometrical structure.
- A PTO system containing information on the PTO properties and force calculation functions for any joint contributing to the energy capture of the device. Each PTO 'body' describes the PTO mechanism operating on a parallel structural joint.
- A control system containing the control algorithm used to control / operate the PTO.

Each of these modules (apart from the MB core) was specifically developed (and will continue to be developed) under PerAWaT. The module functionalities discussed in greater detail in the corresponding sections of this report.

The initial results presented in this report show the baseline functionality of the code under the limitations of a frequency-domain approach. In Section 8 the following aspects are covered in the case studies:

- The ability to define different FDCs with different characteristics;
- The influence of different wave input conditions in the response of the WEC;
- The influence of the array layout in the power absorption characteristics;
- The influence of the control of the PTO of each array element in the power absorption characteristics (by treating the array as a power plant).

Even thought the results in Section 8 are preliminary, an effort was made so that these are aligned with the verification scenarios listed in WG0 D1, in particular the array of four point-absorbers. It is expected that this will ensure that comparisons related to initial results from different project partners (GH, UoOx and QUB) can be more easily made. As listed in WG1 WP1 D1B, these cases should not be confused with a set of representative scenarios for which results will be presented in WG1 WP1 D4 and D15 (versions A and B).

9.2 Next steps

WaveDyn Code Architecture

Much of the initial implementation work has focussed on the WaveDyn code architecture, which has been developed to accommodate time-domain simulation and the frequency-domain results processing. No significant structural changes to the code architecture are envisaged for the D3 deliverable (indeed regular wave simulations in the time-domain already support the frequency-domain analysis presented), allowing future development work to focus on the applied force Calculation Modules. The numerical integrator functionality will be developed however, to give the user control over the integrator type used and any related performance parameters. The output data format is also yet to be finalised, although the underlying C++ data processing classes have been written.

Hydrodynamics module

Hydrodynamics Calculation Module development work for the time-domain will focus on performance improvements to the baseline implementation described in Section 3. These include:

- Automated analysis and selection of an appropriate impulse response function cut-off time (preventing the user from having to decide how far back in time the radiation force convolution should be completed);
- Development of a system-identification technique to approximate the radiation force impulse response function convolution integral with a finite state-space model (see Methodology Report WG1 WP1 D1b, Section 3.3.4), thereby removing the numerical burden of the convolution process (note the above point on the cut-off time will still be necessary in determining the length of the impulse response function required for the identification process);
- A study on the effect of completing the radiation force convolution integral over increasingly coarse time-steps. A coarse convolution time-step may compare favourably in computational time with the finite state-space model described above.

In addition to this, functionality is required to convert a set of user supplied radiation damping data to impulse response function data, in the event that the latter is not supplied by the user. The conversion process requires the completion of a numerical integration at the start of the simulation, as described in the Methodology Report WG1 WP1 D1b, Equation 3.30.

Further development to incorporate nonlinear hydrodynamic force modelling will be the subject of later deliverables and will be closely linked to the first experimental and numerical verification datasets.

Wave module

The wave module will not require any major changes for the time-domain implementation other than the parameterisation of wave climate. The next steps in the investigation of methods to parameterise the wave climate are:

- Compare the accuracy of fitting spectra with Ochi and Gamma spectra with JONSWAP spectra.
- Assess accuracy of models for mean direction and spread.
- Examine distribution of fitted parameters to find efficient description of the wave climate.
- Determine length of measured data required to estimate distribution of spectral shapes.
- Quantify effect of various descriptions of the wave climate on accuracy of WEC yield prediction.

Moorings module

The moorings module currently uses a look-up table approach, as discussed. This may need to be complemented (for some simulation cases at least) by a more explicit mooring-line model based on a Morison-element approach. Post-processing and visualisation functionality will also be expanded. Next steps for the moorings module are therefore:

- Compare look-up-table approach to Morison-element approach for different simulation cases, to determine the degree of similarity in results;
- Based on conclusions from the previous step, make a decision as to whether to the currently functionality is sufficient, or there is a need to add the Morison-element approach to the existing code, or in alternative to create a run-time link to a third-party tool, or to continue with just the look-up-table approach.
- Improve post-processing capabilities, including a graphical tool to allow the user to visualise the mooring line shapes and configurations.
- Investigate the possibility of creating a finite-state model of the mooring system with the aim of carrying out convolutions in the time-domain to represent frequency-dependent impedance.
- Adding static-analysis capability to the software so that a realistic equilibrium state to be computed. This equilibrium state is used as the starting-point for time-domain simulations.

PTO and Control modules

It is envisaged that the development of the PTO and of the control modules will be, as highlighted in Section 6, closely linked.

The PTO module at present supports only a simple, explicit PTO force calculation description. The first PTO templates, designed to support much more complex PTO models, with internal system states will be developed in the next few months. An initial template description, for a hydraulic / electrical rectification and smoothing system was provided in Methodology Report WG1 WP1 D1b, Section 5.5.4. Work is also expected to begin on an external PTO definition DLL interface, providing device manufacturers with the option to create their own PTO models, subject to a set of input and output constraints. The PTO module development work is expected to occupy a significant proportion of the WaveDyn implementation time prior to the WG1 WP1 D3 deliverable. Such external link will also create the facility for developers to provide the control parameters in the .DLL format. Further extensions to template specific control strategies are also planned (in the hydraulics example above, template specific strategies can involve constant pressure (fluctuation in power) and constant power output (fluctuations in pressure) methodologies).

Optimiser module

Finally, the basic next steps for the optimiser development can be summarised in the following points:

- Improve the optimisation of array layout by using different settings with the Genetic Algorithm or possibly employing an alternative procedure.
- Investigate the trade-off between efficiency and accuracy in the computation of the hydrodynamic solution for the layout optimisation objective function.
- Investigate the suitability of local optimisation routines to the problem of PTO coefficient optimisation and possibly use an alternative approach.
- Integrate the optimiser module with the time-domain WaveDyn tool for the objective function.
- Investigate procedures for estimating the interaction effects between WECs and incorporate into the optimiser module.

REFERENCES

General

Alves, M (2002). “Incident Wave Identification”, MARETEC / AWS Internal Report 2/2002, Lisbon, Portugal

American Petroleum Institute (1997); API RP 2SK, Recommended Practice for Design and Analysis of Stationkeeping Systems for Floating Structures, USA.

Babarit A, Duclos G, Clément A.H, 2004. Comparison of latching control strategies for a heaving wave energy device in random sea. Applied Ocean Research 26, 227-238.

Babarit A, Clément A.H, 2006. “Optimal latching control of a wave energy device in regular and irregular waves”; Published in Applied Ocean Research 28, 77-91.

Brito-Melo, A, Sarmento, A, Clément, A and Delhommeau, G (1998). “Hydrodynamic Analysis of Geometrical Design Parameters of Oscillating Water Columns”, Proc. 3rd European Wave Energy Conference, Patras, Greece, Vol. 1, pp.23-30.

Brito-Melo, A, Hofmann, T, Sarmento, A, Clément, A and Delhommeau, G (2000a). “Numerical Modelling of OWC-shoreline Devices Including the Effect of the Surronding Coastline and Non-Flat Bottom”, Proc. 10th International Offshore and Polar Eng Conference, Seatle, USA, Vol.1, pp.743-748.

Brito-Melo, A and Sarmento, A (2000b). “Numerical Study of the Performance of a OWC Wave Power Plant in a Semi-Infinite Breakwater”, Proc. 4th European Wave Energy Conference, Aalborg, Denmark, pp. 283-289.

Brito Melo, A (2002). “AWS Physics Characterisation”, IST Internal Report, Instituto Superior Técnico, Lisbon, Portugal.

Brooke, J (2003). *Wave Energy Conversion*, Elsevier Science.

Budal, K and Falnes, J (1975). “A resonant point absorber of ocean-wave power”, *Nature*, Vol. 256, pp. 478-479.

Budal, K (1977). “Theory of absorption of wave power by a system of interacting bodies”, *Journal of Ship Research*, Vol. 21, pp. 248-253.
Budal, K and Falnes, J in Count B (Editor), 1980. Power from Sea Waves. Academic Press (London). ISBN: 0-12-193550-7.

Bryden, I (1983). “Long floating cylinders in three-dimensional seas,” PhD Thesis, The University of Edinburgh.

Callaghan, J and Bould, R (2006). *Future Marine Energy*, The Carbon Trust.

Centre for Marine and Petroleum Technology (CMPT), 1998. Floating Structures: A guide for design and analysis. Volume 2. Oilfield Publications Limited. ISBN 1 870553 357.

Clare, R, Evans, DV and Shaw, TL (1982). "Harnessing sea wave energy by a submerged cylinder device", *Journal of The Institution of Civil Engineers*, Vol. 73, pp. 356-385.

Clément, A, Babarit, A, Gilloteaux, J, Josset, C and Duclos, G (2005). "The SEAREV Wave Energy Converter, Proc. 6th European Wave & Tidal Energy Conference, Glasgow, United Kingdom.

Count B M, Jefferys E R, Wave power, the primary interface. Proceedings of the 13th Symposium on Naval Hydrodynamics, Paper 8. p1-10. The Shipbuilding Research Association of Japan, Tokyo.

Cruz, J and Salter, S (2006). "Numerical Modelling of a Modified Version of the Edinburgh Duck Wave Energy Converter". *Proc. IMechE Part M: Journal of Engineering for the Maritime Environment*, Vol. 220, pp. 129-147.

Cruz, J and Sarmento, A (2007). "Sea state characterisation of the test site of an offshore wave energy plant", *Ocean Engineering*, Vol. 34 (5-6), pp. 763-775.

Cruz, J, (2008). *Ocean Wave Energy* (Editor), Springer-Verlag.

Cummins W.E, 1962. The impulse response function and ship motions. Schiffstechnik 9, p101–109.

Jefferys in Count B (Editor), 1980. Power from Sea Waves. Academic Press (London). ISBN: 0-12-193550-7.

Dean, R, and Darymple, R (1991). *Water Wave Mechanics for Engineers and Scientists*, World Scientific.

Delauré, Y and Lewis, A (2003). "3D hydrodynamic modelling of fixed oscillating water column by a boundary element methods", *Ocean Engineering*, Vol. 30 (3), pp. 309-330.

Det Norske Veritas (DNV) Software. 'Mimosa' Flyer. Available online, February 2010.

Det Norske Veritas (DNV) Software. 'Riflex – Riser System Analysis Program' Flyer. Available online, February 2010.

Det Norske Veritas (DNV), 2004. DNV-OS-E301, Position Mooring, Norway.

Duclos G, Clément A H, Chatry G, 2001. Absorption of Outgoing Waves in a Numerical Wave Tank using a Self-Adaptive Boundary Condition. Int. Journal of Offshore and Polar Engineering, 11(3), 168–175.

Edinburgh University Wave Power Project, 1987. Solo Duck Linear Analysis. A report to the wave energy steering committee covering work supported by the United Kingdom department of Energy.

Eatock Taylor, R and Jeffreys, E (1985). "Variability of Hydrodynamic Load Predictions for a Tension Leg Platform", *Ocean Engineering*, Vol. 13 (5), pp. 449-490.

Evans, DV (1976). "A theory for wave-power absorption by oscillating bodies", *Journal of Fluid Mechanics*, Vol. 77 (1), pp. 1-25.

Evans, DV (1979). "Some theoretical aspects of three-dimensional wave-energy absorbers", Proc. 1st Symp. Wave Energy Utilization, Gothenburg, Sweden.

Evans , DV, Jeffrey, DC, Salter, SH and Taylor JR (1979). "Submerged cylinder wave energy device: theory and experiment", *Applied Ocean Research*, Vol. 1 (1), pp. 3-12.

Evans, DV (1981a). "Power from Water Waves", *Annual Review of Fluid Mechanics*, Vol. 13, pp. 157-187.

Evans, DV (1981b). "Maximum wave-power absorption under motion constraints", *Applied Ocean Research*, Vol. 3, pp. 200-203.

Evans, DV and Falcão, AF (1986). *Hydrodynamics of Ocean Wave-Energy Utilization* (Editors), Proceedings IUTAM Symposium, Lisbon, Portugal, 8-11 July 1985, Springer-Verlag.

Evans, D., Linton, CM (1993). "Hydrodynamics of wave-energy devices", Annex Report B1: Device Fundamentals/Hydrodynamics, Contract JOU2-0003-DK, Commission of the European Communities.

Falcão, A (2003). "First-generation wave power plants: current status and R&D requirements", Proc. 22nd International Conference on Offshore Mechanics and Arctic Engineering, Cancun, Mexico, Paper No. OMAE2003-37366.

Falcão A F de O, 2007. Modelling and control of oscillating-body wave energy converters with hydraulic power take-off and gas accumulator. Ocean Engineering, vol. 34, 2021–2032.

Falcão A F de O, 2008. Phase control through load control of oscillating-body wave energy converters with hydraulic PTO system. 'Ocean Engineering' 35, p358–366.

Falnes, J (1980). "Radiation impedance matrix and optimum power absorption for interacting oscillators in surface waves", *Applied Ocean Research*, Vol. 2, pp. 75-80.

Falnes J, 2002. Ocean Waves and Oscillating Systems. Linear Interactions Including Wave Energy Extraction. Cambridge University Press. ISBN 0-521-01749-1.

Faltisen, O (1990). *Sea loads on ships and offshore structures*, Cambridge University Press.

Fitzgerald J, Bergdahl L, 2007. Considering Mooring Cables for Offshore Wave Energy Converters. Proceedings of the 7th European Wave and Tidal Energy Conference, Porto, Portugal

Fitzgerald J, Bergdahl L, 2008. Including moorings in the assessment of a generic offshore wave energy converter: A frequency-domain approach. Marine Structures Volume 21, Issue 1. 23-46

Iserles, A, 1996. A First Course in the Numerical Analysis of Differential Equations. Cambridge University Press. ISBN 0521556554.

Goodman T R, Breslin J P, 1976. Statics and dynamics of anchoring cables in water. Journal of Hydronautics vol. 10, no. 4, 113-120.

Greenhow, M and Ahn, SI (1988). "Added-mass and Damping of Horizontal Circular Cylinder Sections", *Ocean Engineering*, Vol. 15 (5), pp. 495-504.

Haskind, M (1957). "The exciting forces and wetting of ships in waves", *Izv. Akad. Nauk. SSSR, Otd. Tekh. Nauk*, Vol. 7, pp. 65-79.

Havelock, T (1942). “The damping of the heaving and pitching motion of a ship”, *Philosophical Magazine*, Vol. 33 (7), pp. 666-673.

Havelock, T (1955). “Waves due to a floating sphere making periodic heaving oscillations”, *Proc. Royal Society*, Vol. 231, pp. 1.

Henderson, R, 2006. Design, simulation, and testing of a novel hydraulic power take-off system for the Pelamis wave energy converter. Published in ‘Renewable Energy’ Vol. 31, 271–283.

Henderson, R (2008). “Case Study: Pelamis”, In: Chapter 5, *Ocean Wave Energy* (Ed.: J Cruz), Springer-Verlag.

Hess, J and Smith, A (1964). “Calculation of nonlifting potential flow about arbitrary three-dimensional bodies”, *Journal of Ship Research*, Vol. 8, pp. 22-44.

Hess, J (1990). “Panel methods in computational fluid dynamics”, *Annual Review of Fluid Mechanics*, Vol. 22, pp. 255-274.

Ho B, Kalman R, 1966. Effective Reconstruction of Linear State-Variable Models from Input/Output Functions. Regelungstechnik 14 (12), 417-441

Jeffrey, DC, Keller, GJ, Mollison, D, Richmond DJ, Salter, SH, Taylor, JM, Young, IA (1978). “Study of mechanisms for extracting power from sea waves”, Fourth Year Report of the Edinburgh Wave Power Project, The University of Edinburgh.

John, F (1950). “On the Motion of Floating Bodies”, *Communications on Pure and Applied Mathematics*, Vol. 3, pp. 45-101.

Justino, P and Clément A (2003). “Hydrodynamic performance for small arrays of sub-merged spheres”, Proc. 5th European Wave Energy Conference, Cork, Ireland, pp. 266-273.

Johanning L, Smith G H, Wolfram J. Towards Design Standards for WEC Moorings. Proceedings of the 6th European Wave and Tidal Conference, Glasgow UK, 223-230

Josset C, Babarit A, Clément A H, 2007. A wave-to-wire model of the SEAREV wave energy converter. Proceedings of the IMechE vol. 221 Part M: J. Engineering for the Maritime Environment.

Katory, M (1976). “On the motion analysis of large asymmetric bodies among sea waves: an application to a wave power generator”, *Naval Architecture*, pp. 158-159.

Kinsman, B (1965). *Wind Waves*, Prentice-Hall, pp. 460-471.

Koterayama W, 1978. Motions of a moored floating body and dynamic tension of mooring lines in regular waves. Report of the Research Institute for Applied Mechanics, Kyushu University, Fukuoka, vol. 6 no. 82.

Kristiansen E, Egeland O, 2003. Frequency Dependent Added-mass in Models for Controller Design for Wave Motion Damping. 6th IFAC Conference on Maneuvering and Control of Marine Craft, 17th – 19th September 2003, University of Girona, Spain.

Kung S Y, 1978. A New Identification and Model Reduction Algorithm via Singular Value Decompositions. Proc.Twelth Asilomar Conf. on Circuits, Systems and Computers, November 6-8, 1978, 705-714.

Lamb, H (1932). *Hydrodynamics*, 6th Ed., Cambridge University Press.

Le Méhauté, B (1976). *An Introduction to Hydrodynamics & Water Waves*, Springer-Verlag.

Lee, C-H, Maniar, H, Newman, JN and Zhu, X (1996a). "Computations of Wave Loads Using a B-Spline Panel Method", Proc. 21st Symposium on Naval Hydrodynamics, Trondheim, Norway, pp. 75-92.

Lee, C-H, Newman, JN and Nielsen, F (1996b). "Wave Interactions with Oscillating Water Column", Proc. 6th International Offshore and Polar Engineering Conference, Los Angeles, USA, Vol.1, pp. 82-90.

Lee, C-H, Farina, L and Newman, J (1998). "A Geometry-Independent Higher-Order Panel Method and its Application to Wave-Body Interactions", Proc. 3rd Engineering Mathematics and Applications Conference, Adelaide, Australia, pp. 303-306.

Lin, C-P (1999). "Experimental studies of the hydrodynamic characteristics of a sloped wave energy device", PhD Thesis, The University of Edinburgh.

Linton, CM (1991). "Radiation and diffraction of water waves by a submerged sphere in finite depth, *Ocean Engineering*, Vol. 18 (1/2), pp. 61-74.

Livingstone M J, 2009. Modelling and Control of a Wave Energy Converter Power Take-Off. Final Year Project Supported by Garrad Hassan and Partners Ltd. Machine Systems Group. University of Bath.

Livingstone M J, Plummer A R, 2010. The design, simulation and control of a wave energy converter power take-off. (submitted for publication in Proceedings of the 7th International Fluid Power Conference, Aachen, Germany).

Longuet-Higgins M S, 1977. The mean forces exerted by waves on floating or submerged bodies with applications to sand bars and wave power machines. Proceedings of the Royal Society of London A. 352, 463-480.

Maniar, H (1995). "A three-dimensional higher order panel method based on B-splines", PhD Thesis, Massachusetts Institute of Technology.

Martins, E, Ramos, FS, Carrilho, L, Justino, P, Gato, L, Trigo, L and Neumann, F (2005). "CEODOURO: Overall Design of an OWC in the new Oporto Breakwater", Proc. 6th European Wave Energy Conference, Glasgow, UK, pp. 273-280.

Mavrakos, S and McIver, P (1997). "Comparison of methods for computing hydrodynamic characteristics of arrays of wave power devices", *Applied Ocean Research*, Vol. 19, pp. 283-291.

McCabe A P, Aggidisa G A, Stallard T J, 2006. A time-varying parameter model of a body oscillating in pitch. Applied Ocean Research 28, 359-370.

McCormick, M (1981). *Ocean Wave Energy Conversion*, John Wiley & Sons.

McIver, P (1993). “The hydrodynamics of arrays of wave-energy devices”, Annex Report B1: Device Fundamentals/Hydrodynamics, Contract JOU2-0003-DK, Commission of the European Communities.

McIver, P and McIver, M (1995). “Wave-power absorption by a line of submerged horizontal cylinders”, *Applied Ocean Research*, Vol. 17, pp. 117-126.

Mei, CC (1976). “Power Extraction from Water Waves”, *Journal of Ship Research*, Vol. 20, pp. 63-66.

Mei, CC (1989). *The Applied Dynamics of Ocean Surface Waves*, Advanced Series on Ocean Engineering, Vol. 1, World Scientific (revised edition in 2005).

Montgomery, DC (2001). *Design and Analysis of Experiments*, John Wiley & Sons, 5th Edition.

Mynett, AE, Serman, DD and Mei, CC (1979). “Characteristics of Salter’s cam for extracting energy from ocean waves”, *Applied Ocean Research*, Vol. 1 (1), pp. 13-20.

Nebel, P (1992). “Optimal Control of a Duck”, Report of the Edinburgh Wave Power Project, Edinburgh, UK.

Newman, JN (1976). “The interaction of stationary vessels with regular waves”, Proc. of the 11th Symposium on Naval Hydrodynamics, London, UK, pp. 491-501.

Newman, JN (1977). *Marine Hydrodynamics*, MIT Press.

Newman, JN (1985). “Algorithms for the free-surface Green’s function”, *Journal of Engineering Mathematics*, Vol. 19, pp. 57-67.

Newman, JN (1992). “Panel methods in marine hydrodynamics”, Proc. 11th Australasian Fluid Mechanics Conference, Hobart, Australia, Keynote Paper K-2.

Newman, JN and Lee, CH (1992). “Sensitivity of Wave Loads to the Discretization of Bodies”, Proc. of the 6th Behaviour of Offshore Structures (BOSS) International Conference, London, UK, Vol. 1, pp. 50-63.

Newman, JN and Lee, CH (2002). “Boundary-Element Methods in Offshore Structure Analysis”, *Journal of Offshore Mechanics and Artic Engineering*, Vol. 124, pp. 81-89.

Ogilvie, TF (1963). “First and second-order forces on a submerged cylinder submerged under a free-surface”, *Journal of Fluid Mechanics*, Vol. 16, pp. 451-472.

O’Leary, M (1985). “Radiation and scattering of surface waves by a group of submerged, horizontal, circular cylinders”, *Applied Ocean Research*, Vol. 7 (1), pp. 51-57.

Orcina Ltd. OrcaFlex Manual Version 9.2a.

Payne, G (2002). “Preliminary numerical simulations of the Sloped IPS Buoy”, Proc. of the MAREC Conference, Univ. Newcastle upon Tyne, UK.

Payne, G (2006). “Numerical modelling of a sloped wave energy device”, PhD Thesis, The University of Edinburgh.

Pinkster, JA (1997). “Computations for Archimedes Wave Swing”, Delft University Technology, Delft, The Netherlands, Report No. 1122-O.

Pizer, D (1992). “Numerical Predictions of the Performance of a Solo Duck”, Report of the Edinburgh Wave Power Project, Edinburgh, UK.

Pizer, D (1993). “The Numerical Prediction of the Performance of a Solo Duck”, Proc. European Wave Energy Symposium, Edinburgh, UK, pp. 129-137.

Pizer, D (1994). “Numerical Models”, Report of the Edinburgh Wave Power Project, Edinburgh, UK.

Pizer, D, Retzler, C, Henderson, R, Cowieson, F, Shaw, F, Dickens, B and Hart, R (2005). “PELAMIS WEC - Recent Advances in the Numerical and Experimental Modelling Programme”, Proc. 6th European Wave Energy Conference, Glasgow, UK, pp. 373-378.

Prado, M., and Gardner, F (2005). “Theoretical analysis of the AWS dynamics during the submergence operation”, Proc. 6th European Wave Energy Conference, Glasgow, UK, pp. 395-400.

Prado, M, Neumann, F, Damen, M and Gardner, F (2005). “AWS Results of Pilot Plant Testing 2004”, Proc. 6th European Wave Energy Conference, Glasgow, UK, pp. 401-408.

Prado, M (2008). “Archimedes Wave Swing”, In: *Ocean Wave Energy* (Ed.: J Cruz), Springer-Verlag.

Press, W, Teukolsky, S, Vetterling, W and Flannery, B (1992). *Numerical Recipes in FORTRAN*. 2nd Ed., Cambridge University Press.

Rademakers, L, Van Schie, R, Schitema, R, Vriesema, B and Gardner, F (1998). “Physical Model Testing for Characterising the AWS”, Proc. 3rd European Wave Energy Conference, Patras, Greece, Vol. 1, pp. 192-199.

Retzler C, Pizer D, Henderson R, Ahlqvist J, Cowieson F and Shaw M (2003). “PELAMIS: Advances in the Numerical and Experimental Modelling Programme”, Proc. 5th European Wave Energy Conference, Cork, Ireland, pp. 59-66.

Roache, PJ (1997). “Quantification of Uncertainty in Computational Fluid Dynamics”, *Annual Review of Fluid Mechanics*, Vol. 29, pp 123-160.

Roache, PJ (1998). *Verification and Validation of Computational Science and Engineering*, Hermosa Publishers.

Roache, PJ (2003). “Error Bars for CFD”, 41th Aerospace Sciences Meeting, Reno, USA.

Romate, JE (1988). “Local Error analysis in 3-D Panel Methods”, *Journal of Engineering Mathematics*, Vol. 22, pp. 123-142.

Romate, JE (1989). “The Numerical Simulation of Nonlinear Gravity Waves in Three Dimensions using a Higher Order Panel Method”, PhD Thesis, Universiteit Twente.

Ross, D (1995). *Power from the Waves*, Oxford University Press.

Salter, SH (1974). "Wave Power", *Nature*, Vol. 249, pp. 720-724.

Sarmento, A (1992). "Wave flume experiments on two-dimensional oscillating water column wave energy devices", *Experiments in Fluids*, Vol.12, pp. 286-292.

Sarmento, A, Luís, A, Lopes, D (1998). "Frequency-Domain Analysis of the AWS", Proc. 3rd European Wave Energy Conference, Patras, Greece, Vol. 1, pp. 15-22.

Sarpkaya, T and Isaacson, I (1981). *Mechanics of Wave Forces on Offshore Structures*, Von Nostrand Reinhold Company.

Shaw, R (1982). *Wave Energy - A Design Challenge*, Ellis Harwood Ltd, John Wiley & Sons.

Skyner, D (1987). "Solo Duck Linear Analysis", Report of the Edinburgh Wave Power Project, Edinburgh, UK.

Standing, MG (1980). "Use of Potential Flow theory in Evaluating Wave Forces on Offshore Structures", *Power from Sea Waves*, Proc. Conf. Institute of Mathematics and its Applications (Ed. B. count), pp. 175-212, Academic Press, London.

Sumer, BM and Fredsøe, J (1997). *Hydrodynamics around Cylindrical Structures*, Advanced Series on Ocean Engineering, Vol. 12, World Scientific.

Sykes, R, Lewis, A and Thomas, G (2007). "A Physical and Numerical Study of a Fixed Cylindrical OWC of Finite Wall Thickness", Proc. 7th European Wave and Tidal Energy Conference, Porto, Portugal, September 2007.

Taghipour R, Perez T, Moan T, 2008. Hybrid frequency-time-domain models for dynamic response analysis of marine structures. Ocean Engineering 35, 685-705.

Thomas, GP and Evans DV (1981). "Arrays of three-dimensional wave-energy absorbers", *Journal of Fluid Mechanics*, Vol. 108, pp. 67-88.

Thomas in Cruz J (Editor), 2008. Ocean Wave Energy, Current Status and Future Perspectives. Springer-Verlag. ISBN978-3-540-74894-6

Triantafyllou M S, Bliekl A, Shin H, 1986. Static and fatigue analysis of multi-leg mooring system. MIT Sea Grant Program Report, August 1986.

Tucker, MJ and Pitt, EG (2001). *Waves in Ocean* Engineering. Elsevier Science Ltd.

Ursell F, 1964. The decay of the free motion of a floating body. Journal of Fluid Mechanics 19, 305-319

van Daalen, E (1993). "Numerical and Theoretical Studies of Water Waves and Floating Bodies", PhD Thesis, University of Twente, The Netherlands.

van der Pluijm, D and Voors, E (2002). "Power from ocean waves – A study on power conversion with the Archimedes Wave Swing", MSc Thesis, Delft University of Technology.

Vugts, JH (1968). “The Hydrodynamic Coefficients for Swaying, Heaving and Rolling Cylinders in a Free-surface”, Netherlands Ship Research Centre TNO, Report No. 112 S.
Walker N S. GMOOR32 Manual v9.2. Global Martime Consultancy Limited.

WAMIT Inc, 2006. WAMIT User Manual Versions 6.3, 6.3PC, 6.3S, 6.3S-PC

Wehausen, J and Laiton, E (1960). “Surface Waves”, *Encyclopaedia of Physics*, Vol. 9, pp. 446-778.

Wehausen, J.V., 1971. The motion of floating bodies. Annual Review of Fluid Mechanics 3, 237–268

Yeung, RW (1982). “Numerical methods in free-surface flows”, *Annual Review of Fluid Mechanics*, Vol. 14, pp. 395-442.

Yemm, R, Pizer, D and Retzler, C (1998). “The WPT-375 – a near-shore wave energy converter submitted to Scottish Renewables Obligation 3”, Proc. 3rd European Wave Energy Conference, Patras, Greece, Vol. 2, pp.243-249.

Yu Z, Falnes J, 1995. State-space modelling of a vertical cylinder in heave. Applied Ocean Research 17, 265-275.

Zentech Inc. ZenMoor by Zentech Flyer. Available online, February 2010.

Section 2

LAPACK – Linear Algebra PACKage, (2010). Developed since 2004 with the support of the National Science Foundation (USA) under Grant No. NSF-0444486. http://www.netlib.org/lapack/

Section 4

Boukhanovsky AV and Guedes Soares C, (2009). Modelling of multipeaked directional wave spectra. Applied Ocean Research, 31, 132-141.

Bouws E, Gunther H, Rosenthal W, Vincent CL, (1985). Similarity of the wind wave spectrum in finite depth. Part 1: Spectral form. J. Geophysical Res. 90 (C1), 975-986.

Donelan MA, Hamilton J, Hui WH, (1985). Directional spectra of wind-generated waves. Philos. Trans. Roy. Soc. London., A315, 509-562.

Elgar S, Guza RT, Seymour RJ, (1985). Wave group statistics from numerical simulations of a random sea. Applied Ocean Research, 7(2), 93-96

Ewans KC , (1998). Observations of the Directional Spectrum of Fetch-Limited Waves. J. Phys. Oce., 28, 495-512.

Ewans KC , (2001). Directional Spreading in Ocean Swell. Proc. Ocean Wave Measurement and Analysis (2001), pp517-529.

Funke E and Mansard E, (1987). A rationale for the use of the deterministic approach to laboratory wave generation. Proc. IAHR Seminar on Wave Analysis and Generation in Laboratory Basins, Lausanne, Switzerland, 1987, pp153-195.

Guedes Soares C, (1984). Representation of Double-Peaked Sea Wave Spectra. Ocean Eng., 11, 185–207.

Hanson JL and Phillips OM, (2001). Automated Analysis of Ocean Surface Directional Wave Spectra. J. Atmos. Ocean. Technol., 18, 277–293.

Hasselmann DE, Dunckel M, Ewing JA, (1980). Directional wave spectra observed during JONSWAP 1973 . J. Phys. Oceanogr., 10, 1264-1280.

Holthuijsen LH, (2007). Waves in Oceanic and Coastal Waters. Cambridge University Press.

Huntington S, (1987). A rationale for the use of the probabilistic approach to laboratory wave generation. Proc. IAHR Seminar on Wave Analysis and Generation in Laboratory Basins, Lausanne, Switzerland, 1987, pp197-208.

Hwang PA, Wang W, Walsh EJ, Krabill WB, Swift RN, (2000). Airborne Measurements of the Wavenumber Spectra of Ocean Surface Waves. Part II: Directional Distribution. J. Phys. Oceanogr., 30, 2768-2787.

Jefferys ER, (1987). Directional seas should be ergodic. Applied Ocean Research, 1987, 9(4), 186-191.

Kahma K, Hauser D, Krogstad HE, Lehner S, Monbaliu JAJ, Wyatt LR, (2005). Measuring and Analysing the Directional Spectra of Ocean Waves. EU COST Action 714, EUR 21367, ISBN 92-898-0003-8.

Kuik AJ, Van Vledder GP, Holthuijsen LH, (1988). A Method for the Routine Analysis of Pitch-Roll-Heave Buoy Wave Data. J. Phys. Oceanography, Vol 18, 1020-1034.

Medina JR and Diez JJ, (1984). Discussion of “Comparisons of Numerical Random Sea Simulations” by Hang Tuah and Robert T. Hudspeth (November, 1982). J. Wtrwy., Port, Coast., and Oc. Engrg., 110(1), 114-116 .

Michel WH, (1999). Sea spectra revisited. Marine Technology, 36(4), 211-227.

Miles MD and Funke ER, (1989). Comparison of methods for synthesis of directional seas. J. Offshore Mech. Arct. Eng., 111(1), 43-48.

Mitsuyasu H, and Coauthors, (1975). Observations of the directional spectrum of ocean waves using a cloverleaf buoy . J. Phys. Oceanogr., 5, 750-760.

Ochi MK and Hubble EN, (1976). Six-Parameter Wave Spectra. Proc. 15th Coastal Engineering Conference, pp. 301–328.

Portilla J, Ocampo-Torres FJ, Monbaliu J, (2009). Spectral Partitioning and Identification of Wind Sea and Swell. J. Atmospheric & Oceanic Technology, 26, 107-122.

Torsethaugen K, (1993). A Two Peak Wave Spectrum Model. Proc. 12th Int. Conf. Offshore Mech. Arctic Eng., Vol. 2, pp. 175–180.

Torsethaugen K and Haver S, (2004). Simplified double peak spectral model for ocean waves. Proc. 14th Int. Offshore and Polar Eng. Conf., Toulon, France.

Tuah H and Hudspeth RT , (1982). Comparisons of Numerical Random Sea Simulations. J. Waterway Port Coastal and Ocean Division, 108(4), 569-584.

Tucker MJ, Challenor PG, Carter DJT, (1984). Numerical simulation of a random sea: a common error and its effect upon wave group statistics. Applied Ocean Research, 6(2), 118-122.

Young IR and Babanin AV, (2006). The form of the asymptotic depth-limited wind wave frequency spectrum. J. Geophys. Res., 111, C06031, doi:10.1029/2005JC003398, 2006.

Young IR and Van Vledder GPh, (1993). A Review of the Central Role of Nonlinear Interactions in Wind-Wave Evolution. Philosophical Transactions: Physical Sciences and Engineering, Vol. 342, No. 1666. (Mar. 15, 1993), pp. 505-524.

Young IR, Verhagen LA, Banner ML, (1995). A note on the bimodal directional spreading of fetch-limted waves. J. Geophys. Res., 100 (C1), 773-778.

Section 5

ANSYS Inc (2009). AQWA-NAUT / AQWA-LINE User Manuals, Release 12.0, April 2009. www.ansys.com.

Dewey, R (2003). Mooring Design and Dynamics manual. www.canuck.seos.uvic.ca/rkd/mooring

Faltisen, O (1990). *Sea loads on ships and offshore structures*, Cambridge University Press.

Fitzgerald J, Bergdahl L (2007). Considering Mooring Cables for Offshore Wave Energy Converters. Proceedings of the 7th European Wave and Tidal Energy Conference, Porto, Portugal.

Fitzgerald J, Bergdahl L (2008). Including moorings in the assessment of a generic offshore wave energy converter: A frequency-domain approach. Marine Structures Volume 21, Issue 1. 23-46.

GL Noble Denton Ltd (2008). ROMEO User Manual, Release 1.6, July 2008.
www.gl-nobledenton.com.

Johanning L, Smith G H, Wolfram J (2005). Towards Design Standards for WEC Moorings. Proceedings of the 6th European Wave and Tidal Conference, Glasgow UK, 223-230.

Johanning L, Smith G H, Wolfram J (2006). Interaction between mooring line damping and response frequency as a result of stiffness alterations in surge. Proceedings of 25th International Conference on Offshore Mechanics and Arctic Engineering (OMAE 2006), Hamburg, paper no. 2006-92373.

McCormick M E (2010). *Ocean Engineering Mechanics*. Cambridge University Press.

Orcina Ltd (2008). OrcaFlex Manual, Version 9.2a. www.orcina.com.

Vicente P, Falcao A, Gato M, and Justino P (2009). Hydrodynamics of triangular-grid arrays of floating-point-absorber wave energy converters with inter-body and botton slack-mooring connection. Proc. 8th EWTEC, Uppsala, Sweden.

***Section** 7*

Bellew S, Stallard T and Stansby PK, (2009). Optimisation of a Heterogeneous Array of Heaving Bodies. Proceedings of the 8th European Wave and Tidal Energy Conference, 519-527.

Child BFM and Venugopal V, (2010). Optimal configurations of wave energy device arrays. Ocean Engineering, 36(16), 1402-1417.

Cruz J, Sykes R, Siddorn P and Eatock Taylor R, (2009). Wave Farm Design: Preliminary Studies on the Influences of Wave Climate, Array Layout and Farm Control. Proceedings of the 8th European Wave and Tidal Energy Conference, 736-745.

De Backer G, (2009). Optimisation of a Heterogeneous Array of Heaving Bodies. Proceedings of the 8th European Wave and Tidal Energy Conference, 519-527.

De Backer G, Vantorre M, Beels M, De Rouck J and Frigaard P, (2009). Performance of closely spaced point absorbers with constrained floater motion. Proceedings of the 8th European Wave and Tidal Energy Conference.

The Mathworks, Inc., (2010). Constrained Nonlinear Optimization. http://www.mathworks.com/help/toolbox/optim/ug/brnoxzl.html

Mitchell M, (1998). An introduction to genetic algorithms. MIT Press.

Vicente PC, de O Falcão AF, Gato LMC and Justino PAP, (2009). Dynamics of arrays of floating point-absorber wave energy converters with inter-body and bottom slack-mooring connections. Applied Ocean Research, 31(4), 267-281.

NOMENCLATURE

Standard Index (S I) Units are used unless stated otherwise

An overdot, $\dot{x}$ indicates differentiation of the quantity x with respect to time.

A X^* indicates the complex conjugate of the complex quantity X

The complex operator, $j = \sqrt{-1}$

A standard glossary of terms can be found in the 2007 Ocean Energy Glossary (IEA-OES), which was followed throughout this document. This document is available at

http://www.wavec.org/client/files/Ocean_Energy_Glossary_Dec_2007.pdf.

Section 4	
a	Wave Amplitude
$D(f, \theta)$	Directional Spreading Function (or Directional Distribution)
$E(f)$	Omnidirectional spectrum or Frequency Spectrum
f	Frequency
f_p	Peak Frequency
f_s	Sampling Frequency
g	Acceleration due to gravity
H_s	Significant Wave Height
$k = 2\pi / \lambda$	Wave Number
L	Duration of simulation
m_n	nth moment of the omnidirectional spectrum
$MDIR$	Mean Direction
r	Generalised JONSWAP spectrum high-frequency tail index
s	Generalised JONSWAP spectrum shape parameter
$S(f, \theta)$	Directional Variance Spectrum

SDIR	Mean Spread
T_p	Peak Period
T_e	Energy Period
T_m	Mean Period
T_z	Zero-Crossing Period
α	Generalised JONSWAP spectrum scale parameter
β	Generalised JONSWAP spectrum location parameter
γ	Generalised JONSWAP spectrum peak enhancement factor
Δ	Deviance between measured and fitted spectra
$\delta(\theta)$	Deviance between measured and fitted mean direction
$\delta(\sigma)$	Deviance between measured and fitted spreading function
θ	Direction of wave propagation
θ_m	Mean wave direction (frequency dependent)
λ	Wavelength
η	Sea surface elevation
$\omega = 2\pi f$	Angular frequency
σ_c	Directional Spread (circular moment definition)
σ_l	Directional Spread (line moment definition)
ϕ	Phase
Section 5	
B_{moor}	Damping matrix for the mooring line
K_{moor}	Stiffness matrix for the mooring line
M_{moor}	Inertia matrix for the mooring line

C_a	Added-mass coefficient
C_{curr}	Current force coefficient in ROMEO
C_{wind}	Wind force coefficient in ROMEO
C_s	Shape coefficient for wind/current force calculations in ROMEO
C_h	Height coefficient for wind/current force calculations in ROMEO
X	Complex RAO at a given wave frequency
x_p	Distance from origin of wave system perpendicular to wave direction
χ	$= x \cos \theta + y \sin \theta$ where θ = wave direction
u_f	Fluid velocity component normal to cable element (Orcaflex)
u_s	Velocity of cable element
C_m	Inertia coefficient
ε	Mean axial strain
λ	Expansion factor of cable element
L	Instantaneous length of cable element
L_0	Original (unstretched) length of cable element
E	Young's modulus
D_n	Effective diameter of cable for drag purposes (for flow in normal direction)
D_a	Effective diameter of cable for drag purposes (for flow in axial direction)
$\underline{x}_m$	General position vector in mooring line coordinates
$\underline{p}$	Position vector of mooring attachment point in global coordinates
$\underline{p}_{eq}$	Position vector of mooring attachment when system is at equilibrium
Section 7	
ψ	Direction of gridline (Set 1)
σ	Direction of gridline (Set 2)

p	Spacing between gridline of Set 2 along gridlines from Set 1
s	Spacing between gridline of Set 1 along gridlines from Set 2
M	Number of gridlines from Set 1 selected to define potential device locations
χ	Device heading
a	Device radius
h	Clearance beneath the device
d	Water depth
B_{pto}	Damping coefficient of PTO
L	Device spacing
N	Number of devices in the array
β	Wave direction
T	Period
T_{-10}	Energy period
T_p	Peak period
H_{m0}	Significant wave height
P_{abs}	Absorbed power

KEY TO ACRONYMS

ADCP	Acoustic Doppler Current Profiler
AIC	Akaike's Information Criterion
BCS	Body-fixed Coordinate System
BDM	Bayesian Directional Method
BEM	Boundary Element Method
DDD	Double Direction Decomposition
DFTM	Direct Fourier Transform Method
DLL	Dynamic-Link Library
DOF	Degree-Of-Freedom
EMEC	European Marine Energy Centre
EMLM	Extended Maximum Likelihood Method
EMEP	Extended Maximum Entropy Principle
EWTEC	European Wave and Tidal Energy Conference
FD	Frequency-Domain
FDC	Fundamental Device Concept
FFT	Fast Fourier Transform
FSP	Full-Scale Prototype
GH	Garrad Hassan and Partners Ltd
GCS	Global Coordinate System
GUI	Graphical User Interface
IMLM	Iterated Maximum Likelihood Method
JONSWAP	Joint North Sea Wave Project
NURBS	Non-Uniform Rational B-Splines
OWC	Oscillating Water Column
PM	Pierson-Moskowitz

PR	Power Ratio (ratio between measured and expected power)
PTO	Power Take-Off
QUB	Queen’s University Belfast
RAO	Response Amplitude Operator
RCW	Relative Capture Width
RMS	Root-Mean-Square
SDD	Single Direction Decomposition
TD	Time-Domain
TFSM	Truncated Fourier Series Decomposition Method
UoOx	University of Oxford
WEC	Wave Energy Converter
WR	Waverider