

Instituto Tecnológico de Costa Rica  
Mechatronics Engineering Academic Area

Rijstuniversiteit Groningen  
Faculty of Science and Engineering  
Ocean Grazer

# **Model Predictive Control Implementation for the Ocean Grazer Wave Energy Converter with a port-Hamiltonian Model**

A thesis submitted in fulfilment for the  
Licentiate degree in Mechatronics Engineering

by

**Joel Alpízar Castillo**

July, 2018

## Declaration of Authorship

I declare that this graduation project is entirely made by myself, with the help of scientists from the University of Groningen and the Costa Rica Institute of Technology, who gave me the opportunity to collaborate with the Ocean Grazer project located in the Netherlands. In the cases I have used literature, I mentioned the sources by references. Therefore, I assume total responsibility for the work done and for the content of the present final report.

Signed:  \_\_\_\_\_

Date: September 4, 2018

Instituto Tecnológico de Costa Rica  
Mechatronics Engineering Academic Area  
Rijstuniversiteit Groningen  
Faculty of Science and Engineering  
Ocean Grazer  
Graduation Project  
Assessment Tribunal

Graduation Project presented to the Assessment Tribunal as requisite to obtain the title of Mechatronics Engineer with the academic grade of Licentiate, from the Instituto Tecnológico de Costa Rica.

Assessment Tribunal

---

Dr. Juan Luis Crespo Mariño  
Assessment Professor

---

MSc. Juan Luis Guerrero Fernández  
Assessment Professor

---

Dr. Gabriela Ortiz León  
Supervisor Professor

The members of this Tribunal attest that the present graduation project has been approved and fulfil the standards established by the Mechatronics Engineering Academic Area.

Cartago, 10th July 2018.

# Resumen

El océano es una fuente perpetua de energía que se ha estudiado durante las últimas décadas para convertir la energía de las olas y las mareas en electricidad, a través de un proceso limpio. Hasta ahora, se han diseñado varios dispositivos para ese fin, incluido el Ocean Grazer, un novedoso dispositivo de recolección de energía que permitirá extraer hasta 260 GWh por año y almacenar hasta 800 MWh de energía de las olas, a través de un sistema de extracción por puntos, asegurando un suministro continuo a la red. El sistema utiliza un novedoso concepto de pistón múltiple y bombeo múltiple (MP<sup>2</sup>) para maximizar la cantidad de energía extraída, sin embargo, se requiere una estrategia de control. En este proyecto se presenta una estrategia de control predictivo por modelo (MPC) con un modelo port-hamiltoniano que utiliza el lenguaje de código abierto Python, con la ventaja sobre otras estrategias de control en la literatura al no requerir una predicción del oleaje. Su validación en lazo abierto mostró una precisión aceptable cuando se compara con una contraparte de MATLAB, pero con un tiempo de computación considerablemente menor ( $\sim 28$  veces menos). La estrategia de control se probó usando un arreglo de  $2 \times 1$  flotadores, lo que permitió obtener una configuración de pistón para el MP<sup>2</sup> en pocos segundos, y garantizando una absorción de energía con menos del 5% de error en comparación con el valor máximo teórico.

**Palabras clave:** control predictivo por modelo, energía undimotriz, modelado port-Hamiltoniano, Python.

## Abstract

The ocean is a perpetual source of energy that has been studied for the last decades in order to convert the energy from the waves and tides into electricity, through a clean process. So far, several devices have been designed for that purpose, including the Ocean Grazer, a novel energy harvesting device that will allow to extract up to 260 GWh per year and store up to 800 MWh from the waves, through a point absorber take-off system, ensuring a continuous supply to the grid. The system uses a novel multiple-piston multiple-pump (MP<sup>2</sup>) concept to maximize the extracted energy, however, a control strategy is required. In this project is presented a model predictive control (MPC) strategy with a port-Hamiltonian (pH) model using the open source language Python, with the advantage over other control strategies in the literature that doesn't require a wave prediction. Its open loop validation showed an acceptable accuracy when compared against a MATLAB counterpart, but taking considerable less computing time ( $\sim 28$  times less). The control strategy was tested using a  $2 \times 1$  floater array, resulting possible to obtain a piston configuration for the MP<sup>2</sup> in few seconds, and guaranteeing an energy absorption with less than 5 % of error when compared with the theoretical maximum value.

**Keywords:** model predictive control, port-Hamiltonian modelling, Python, wave energy.

# Dedicatory

This document only reaffirms the end of an academic stage. During this process, there were people who contributed positively, either in the academic perspective or personally, and it seems fair to me to dedicate to all these people, not this document, but the title it implies. On the other hand, there were people who hampered and even opposed many of the achievements that allowed me to reach this point, and precisely for that reason, I would like to dedicate this achievement to them as well, as a demonstration of the will and character that my parents, friends and professors helped to forge.

# Acknowledgements

There were a considerable amount of people involved in the academic process concluded by this document; my parents, professors, familiars and friends. I am deeply grateful to all of you.

I would like to thank to my supervisor Gabriela Ortiz León, and my reading committee members, Juan Luis Crespo and Juan Luis Guerrero Fernández, for their observations. Also, to professors Gilberto Vargas, Carlos Piedra and Gabriela Amador for their support throughout my studies.

A special mention to the Physics Department of the Costa Rica Institute of Technology, highlighting Dionisio Gutiérrez, Ernesto Montero and Laura Rojas, with whom I worked with for many years, and Álvaro Amador, for their invaluable support and advices. To you, I am heartily grateful.

Regarding to this project, I would like to thank to Mauricio Muñoz, who helped me to find the project itself and also during its execution. To the Ocean Grazer team, specially Bayu Jayawardhana, Antonis Vakis Zaki Almuzakki, Yanji Wei and Marijn van Rooij, with whom I worked and gave me crucial feedback and advises. Also to Arlette van Berkel and Conny Dokter, who helped me with all the paperwork behind the internship and the immigration process. To all of you, thank you very much.

Joel Alpízar Castillo  
Cartago, September 2018.

# Contents

List of Figures . . . . .	iv
List of Tables . . . . .	vi
Abbreviations . . . . .	vii
Constants . . . . .	viii
Variables . . . . .	ix
<b>1 Introduction</b>	<b>1</b>
1.1 State-of-the-art . . . . .	2
1.2 Objectives . . . . .	2
1.2.1 General objective . . . . .	2
1.2.2 Specific objectives . . . . .	3
1.3 Main Contribution . . . . .	3
1.4 Structure of the Proposal . . . . .	4
<b>2 Generalities</b>	<b>5</b>
2.1 Waves . . . . .	5
2.2 Wave Energy . . . . .	7
2.3 Wave Energy Extraction Methods . . . . .	8
2.3.1 Knick Absorber Systems . . . . .	8
2.3.2 Point Absorber Systems . . . . .	9
2.3.3 Oscillating Water Column Systems . . . . .	9
2.3.4 Over-topping Terminator Systems . . . . .	9
2.3.5 Oscillating Wave Surge Converter Systems . . . . .	10
2.4 Port-Hamiltonian Systems . . . . .	11
2.4.1 pH Model of a Mass-Spring-Damper System . . . . .	13
2.5 Model Predictive Control . . . . .	16
2.5.1 Dynamic Matrix Control . . . . .	18
2.5.2 Model Algorithmic Control . . . . .	18
2.5.3 Predictive Functional Control . . . . .	18
2.5.4 MPC in Renewable Energies . . . . .	18
2.6 Ocean Grazer . . . . .	19
2.6.1 WEC Operation Principle . . . . .	21
2.6.2 WEC Previous Control Proposals . . . . .	21
2.7 Concluding Remarks . . . . .	22

<b>3</b>	<b>Solution Proposal</b>	<b>23</b>
3.1	Problem Determination . . . . .	23
3.2	Problem Synthesis . . . . .	24
3.3	User Requirements . . . . .	25
3.4	System Requirements . . . . .	25
3.5	Solution Approach . . . . .	25
3.6	Control Strategy Selection Criteria . . . . .	26
3.7	Considered Control Strategies . . . . .	26
3.7.1	Control Strategies Evaluation . . . . .	27
3.8	Programming Language Selection Criteria . . . . .	28
3.9	Considered Languages . . . . .	30
3.9.1	Considered Languages Evaluation . . . . .	30
3.10	Possible Solutions . . . . .	31
3.10.1	A Script with All the Functions Needed . . . . .	31
3.10.2	A Set of Functions as a Toolbox . . . . .	32
3.11	Chosen Solution . . . . .	32
3.12	Viability Criteria . . . . .	32
3.13	Concluding Remarks . . . . .	33
<b>4</b>	<b>Model Description and Control Strategy</b>	<b>35</b>
4.1	Previous port-Hamiltonian Model Analysis . . . . .	35
4.2	Proposed Model . . . . .	41
4.2.1	Assumptions and Simplifications . . . . .	43
4.3	Control Strategy . . . . .	44
4.4	Control Algorithm Design . . . . .	47
4.5	Concluding Remarks . . . . .	49
<b>5</b>	<b>Results Validation</b>	<b>52</b>
5.1	Design of the Python Algorithm . . . . .	52
5.1.1	Open Loop Testing of Python's Equivalent of the MATLAB Model . . . . .	54
5.1.2	Open Loop Testing in Python Using the Proposed Model . . . . .	56
5.2	Implementation of the MPC . . . . .	59
5.3	Discussion . . . . .	62
<b>6</b>	<b>Conclusions, Limitations and Recommendations</b>	<b>67</b>
6.1	Conclusions . . . . .	67
6.2	Limitations . . . . .	68
6.3	Recommendations for Future Research . . . . .	69
	<b>Bibliography</b>	<b>71</b>

# List of Figures

2.1	Character of the waves according with its frequency spectrum width. . . . .	6
2.2	Oscillating Water Column systems examples. . . . .	8
2.3	Point Absorber systems examples. . . . .	9
2.4	Oscillating Water Column systems examples. . . . .	10
2.5	Over-topping Terminator systems examples. . . . .	10
2.6	Oscillating Wave Surge Converter systems examples. . . . .	11
2.7	Port-Hamiltonian elements relation. . . . .	12
2.8	Single mass-spring-damper system . . . . .	14
2.9	MPC main process, were $r(k)$ is the reference trajectory, $u(k)$ is the process input, $y(k)$ the process output, $\hat{y}(k)$ the predicted output, $\hat{y}(k + 1)$ the future predicted output, $\epsilon(k)$ is the error between the predicted output and the process output. . . . .	17
2.10	MPC basic principle using the error between the output of the system and a set-point trajectory as cost function, were $k$ is the current instant, $y(t)$ the output of the system, $s(t)$ the set-point trajectory, $r(t k)$ the reference trajectory with respect to the instant $k$ , $\hat{y}(t k)$ the predicted output with respect to the instant $k$ , $\hat{y}_f(t k)$ is predicted free response to the system with respect to the instant $k$ , and $H_p$ the prediction horizon, [39]. . . . .	17
2.11	Multi-piston multi-pump (PTO 1) concept, [47]. . . . .	20
2.12	OG PTO 3 and 4 representation, [47]. . . . .	20
2.13	Representation of the Ocean Grazer MP <sup>2</sup> PTO. . . . .	21
3.1	Ishikawa diagram with the causes of the computational cost of controlling the current time-domain model. . . . .	24
3.2	Pareto chart summarizing the criteria for the solution validation. . . . .	33
4.1	Diagram of the multi-floater system used in [18]. . . . .	36
4.2	Flowchart of the Matlab code of the pH model of the WEC in open loop. . .	39
4.3	Response of the pH model with the Matlab code using a regular wave with 4 m of height and a period of 5 s, and the parameters shown in Table 4.1. . . .	40
4.4	Diagram of the proposed model, considering the hydrodynamics of the piston. . .	41
4.5	Representation of the prediction horizon time window with irregular waves, were $h_n$ is the height of each one of the up-strokes, $H_n$ is the prediction horizon and $K$ is the peak instant. . . . .	48

4.6	Representation of the prediction horizon time window, filtering the low energy peaks in irregular waves, where $h_n$ is the height of each one of the up-strokes, $H_n$ is the prediction horizon and $K$ is the sampling instant. . . . .	49
4.7	Flowchart of the control algorithm. . . . .	51
5.1	Error between the MATLAB model and the WEC-Sim results in the pH calculations with respect to the radiation approximation order, when using a regular wave with 4 m of height and a period of 5 s. . . . .	53
5.2	Flowchart of the Python code of the pH model of the WEC in open loop. . .	54
5.3	Comparison of results of the previous pH model in Python against the Matlab model. . . . .	55
5.4	Response of the proposed model with the Python code using a regular wave with 4 m of height and a period of 5 s. . . . .	57
5.5	Response of the proposed model with the Python code, using a regular wave with 4 m of height and a period of 5 s, with different piston area configurations. . . . .	58
5.6	Computation time of the proposed model when simulating a 100 s interval with regular waves, with respect to the time interval used in the Euler method. . . . .	59
5.7	Irregular wave used as testing conditions. . . . .	60
5.8	Level of the working fluid in the upper reservoir against the piston combination after the irregular wave shown in Figure 5.7. The maximum value is 0.1626 m, with the combination $\{0.48, 0.30\}$ . . . . .	61
5.9	Results of the optimization of the proposed model with the Python code, using the irregular wave shown in Figure 5.7, and a filtering factor to neglect up-strokes smaller than 0,5 m for the buoy 1 and 0,2 m for buy 2. . . . .	62
5.10	Response of the proposed model with the Python code against the irregular wave shown in Figure 5.7, using a filtering factor to neglect up-strokes smaller than 0,5 m for the buoy 1 and 0,2 m for buy 2. . . . .	63

# List of Tables

3.1	Criteria to compare the possible control strategies to chose the optimal proposal.	27
3.2	Evaluation of the considered control strategies. . . . .	28
3.3	Criteria to compare the possible open source languages to chose the optimal proposal. . . . .	29
3.4	Evaluation of the considered open source languages. . . . .	30
4.1	Parameters and constants used in the Matlab simulation, [18]. . . . .	40
4.2	Set of equivalent areas, [13], and damping coefficients for the possible piston configurations. . . . .	47
5.1	Parameters and constants used in the Python simulation with the modified model. . . . .	56
5.2	Resulting piston combination from the optimization strategy of the proposed model when the wave presented in Figure 5.7 is used. . . . .	60

# List of Abbreviations

An alphabetical ordered list of the abbreviations used in this thesis is given next.

<b>ANN</b>	<b>Artificial Neural Networks</b>
<b>BEM</b>	<b>Boundary Element Method</b>
<b>DMC</b>	<b>Dinamic Matrix Control</b>
<b>DoF</b>	<b>Degree of Freedom</b>
<b>EC</b>	<b>Evolutionary Computing</b>
<b>i/o</b>	<b>inputs and outputs</b>
<b>IRF</b>	<b>Impulse Response Function</b>
<b>MAC</b>	<b>Model Algorithmic Control</b>
<b>MP<sup>2</sup></b>	<b>Multiple-Piston Multiple-Pump</b>
<b>MPP</b>	<b>Multiple-Piston Pump</b>
<b>MPC</b>	<b>Model Predictive Control</b>
<b>ODE</b>	<b>Ordinary Differential Equation</b>
<b>OG</b>	<b>Ocean Grazer</b>
<b>pH</b>	<b>Port-Hamiltonian</b>
<b>PSO</b>	<b>Particle Swarm Optimization</b>
<b>PFC</b>	<b>Predictive Functional Control</b>
<b>PID</b>	<b>Proportional-Integral-Differential</b>
<b>PTO</b>	<b>Power Take-Off</b>
<b>SLSQP</b>	<b>Sequential Least Squares Programming</b>
<b>WEC</b>	<b>Wave Energy Converter</b>

# List of Constants

An alphabetical ordered list of the constants used in this thesis is given next.

Symbol	Constant	Value
$A_b$	Basal area of the buoy	49 m <sup>2</sup>
$A_p$	Area of the piston	Table 4.2
$A_{UR}$	Area of the upper reservoir	33,33 m <sup>2</sup>
$b_{pto}$	PTO damping coefficient	Table 4.2
$C_{UR}$	Capacitance of the upper reservoir	$3,40 \times 10^{-3} \text{ m}^4\text{s}^2/\text{kg}$
$g$	Gravitational acceleration	9,81 m/s <sup>2</sup>
$H_b$	Height of the buoy	2 m
$H_p$	Prediction horizon	3
$h_p$	Height of the piston	0,1 m
$k_i$	Stiffness of the n-th buoy	$4,9271 \times 10^5 \text{ N/m}$
$k_{pto}$	PTO stiffness	0 N/m
$L_{UR}$	Distance between the piston and the upper reservoir	115 m
$m_b$	Mass associated to the buoy	1 650 kg
$m_f$	Mass of the floater	1 500 kg
$m_p$	Mass of the piston	150 kg
$m_{\infty i,i}$	Added mass of the buoy due its own movement	$1,0545 \times 10^5 \text{ kg}$
$m_{\infty i,j}$	Added mass of the buoy due the movement of the other buoy	$1,1327 \times 10^4 \text{ kg}$
$P_{atm}$	Atmospheric pressure	101 325 Pa
$P_{UR}$	Pressure in the upper reservoir	101 325 Pa
$S_p$	Separation between the piston and the cylinder	400 $\mu\text{m}$
$\mu$	Viscosity of the working fluid	0,0734 Pa-s
$\rho$	Density of the working fluid	1000 kg/m <sup>3</sup>
$\rho_s$	Density of the sea water	1025 kg/m <sup>3</sup>

# List of Variables

An alphabetical ordered list of the variables used in this thesis is given next.

Symbol	Variable	Units	Definition
$A_{pH}$	State-space equivalent A matrix from the pH representation	-	(4.31)
$B_{pH}$	State-space equivalent B matrix from the pH representation	-	(4.32)
$b$	Damping coefficient	kg/s	(4.36)
$C_{pH}$	State-space equivalent C matrix from the pH representation	-	(4.33)
$C_{UR}$	Capacitance of the upper reservoir	$m^4s^2/kg$	(4.16)
$c$	Filtering factor	-	-
$E_t$	Energy per unit length	$kJ/m$	(4.37)
$e$	Effort	-	-
$e$	Error	-	-
$f$	Flow	-	-
$f_b$	Buoyancy force	N	(4.3)
$f_{ex}$	External excitation force	N	-
$f_{pto}$	Forces exerted by the PTO	N	-
$f_r$	Radiation force	N	(4.4)
$f_w$	Force of the column of water	N	(4.18)
$G$	pH input matrix	-	-
$H$	Hamiltonian function	J	(4.15)
$H_M$	Mean wave height	m	(2.1)
$H_p$	Prediction horizon	-	-
$H_s$	Significant wave height	m	(2.2)
$I$	Identity matrix	-	-
$h_b$	Height of the buoy's movement	m	-
$J$	pH interconnection matrix	-	-
$J(x, A_p)$	Cost Function	J	(4.35)
$K$	Matrix of spring coefficients	$N/m$	-
$K$	Instant of the peak in the buoy movement	-	-
$k$	Stiffness coefficient	$N/m$	(4.10)
$M_\infty$	Added mass matrix in infinite frequency	kg	(4.11)
$m$	Mass of the system	kg	(4.12)
$N$	Auxiliary Hamilton derivative matrix	-	(4.29)
$n$	Number of buoys	-	-
$O$	Off-diagonal block	-	(4.5)
$o$	Order of the radiation approximation	-	-
$P_w$	Power per unit length	$kW/m$	(2.5)
$P$	Pressure	Pa	(4.14)
$p$	Momentum	$kg\cdot m/s$	-
$\dot{p}$	Force	N	-
$q$	Displacement	m	-
$\dot{q}$	Velocity	$m/s$	-
$\ddot{q}$	Acceleration	$m/s^2$	-

$R$	pH dissipation matrix	-	-
$R_p$	Piston radius	m	-
$r$	Radiation component	-	-
$T_M$	Mean wave period	s	(2.3)
$T_s$	Significant wave period	s	(2.4)
$u$	Input of the system	-	-
$x$	State of the system	-	-
$y$	Output of the system	-	-
$y_{UR}$	Level of the water in the upper reservoir	m	(4.22)
$z$	Radiation component	-	-
$\eta$	Surface elevation	m	-
$\varphi$	Convolution kernel	-	-

# Chapter 1

## Introduction

The world is changing, and one of the main reasons is the inappropriate use of natural resources for the industry, transport, power and even agriculture, causing high environmental issues [1]. In fact, according to [2], in 2000 the 24 % of the greenhouse-gas emissions were produced by power generation, number that has been increasing because of the high dependence on fossil fuels. Moreover, according to the U.S. Energy Information Administration (EIA) International Energy Outlook 2016, in 2012 nearly 67,2 % of global electricity generation was supplied from fossil fuels, 21,9 % from renewable sources and 10,9 % from nuclear energy, [3].

Given the above, due the urgent need to support our energy generating capacity through the development of low carbon technologies [4], the scientific community has been working to develop new renewable energy extraction devices and reduce the greenhouse gasses emission. One alternative is wave energy, which extraction is based on the mechanical movement of waves. Since the waves are in a perpetual movement produced by wind, they are a virtually endless energy source; for that reason, the Ocean Grazer team is developing a device to extract up to 208 GWh each year from the waves. There is still, however, work to do in order to improve the current wave energy technologies used by the Ocean Grazer (OG) project, being the motivation of this project.

The problem to solve with the present project relies on the fact that the Ocean Grazer wave energy converter (WEC) uses a novel multiple-piston multiple-pump (MP<sup>2</sup>) that needs a controller to maximize the extracted energy, and its implementation is not straightforward. Therefore, a low computationally demanding time domain control strategy is required. The proposed solution is an open source model predictive control toolbox, that using a port-Hamiltonian model, can determine the optimal input parameters to the system.

During the design process, several assumptions and simplifications have to be considered, in order to start with a simple problem and, once a solution for that problem is found, more complex conditions can be added, approaching as most as possible to the real conditions.

## 1.1 State-of-the-art

The Ocean Grazer is a novel wave energy extraction device which uses a series of pistons connected to floaters that take advantage of the movement of the waves to pump water from a lower to an upper reservoir, using the multi-piston pump concept described in [5], which is a system of multiple pistons that can achieve different areas to extract as much energy as possible from the waves. The water can be stored in the upper reservoir or can be sent back to the lower reservoir through a turbine that converts the potential energy, that was stored in the upper reservoir through the pumping of the piston, into electricity with a model described in [6].

The movement of water is due to a piston with a specially designed ball valve, that has been described by [7] and experimentally analysed by [8] and [9]. The multi-piston pump has been modeled as a single piston that can change its area, denominated single-piston pump, and whose mechanics are described by [10]. Its movement is because it's attached to a buoy, and the group of buoys is called floater blanket, that has been modelled in the frequency domain by [11]. Likewise, the power absorption by the floater blanket was calculated by [12].

The whole mechanism used to extract the energy is called wave energy converter, and several models have been proposed to create the most efficient control strategy. A first non-linear control design was proposed by [13] used the single-piston pump approximation and a model predictive control strategy, but with the issue of a high computation time. A lumped dynamical model on the storage reservoir using a model predictive control strategy was proposed by [14], resulting in good performance in terms of generated electricity, however, it was still not suitable.

Then, the adaptability of the multi-piston pump was investigated by [15], proposing a high fidelity time domain model, but still with a high computational cost in terms of required hardware and computation time. A successful model predictive control implementation was presented by [16], allowing to optimize the captured energy by a  $5 \times 1$  array wave energy converter, however a full blanket is still not analysed, neither the interaction between buoys due to the radiation components generated by its movement in the water because of the computational cost of the current models.

A port-Hamiltonian modelling approach was first introduced by [17], considering the radiation effects between buoys, calculating them through a boundary-element method, and then continued by [18]. It is, then, introduced in the present investigation, a model predictive control implementation, based on the port-Hamiltonian model described by [18].

## 1.2 Objectives

### 1.2.1 General objective

Develop a numerically tractable algorithm to control the Ocean Grazer wave energy converter in time domain less computationally demanding than the current time domain model, which

allows to implement a Model Predictive Control strategy.

### 1.2.2 Specific objectives

- Determine a fast implementation method which allow to develop a time domain control algorithm for the Ocean Grazer wave energy converter.

**Indicator:** A rubric for each studied method that shows the viability of each for the implementation, including advantages, disadvantages, compatibility with MPC, documentation and computational requirements.

- Develop an equivalent time domain control algorithm for the Ocean Grazer wave energy converter pumping system.

**Indicator:** Results of different simulations of the developed model, which shows its behaviour under regular and irregular waves.

- Validate the developed control algorithm computational costs are less demanding than the current time domain model.

**Indicator:** A comparison between the computational cost of both models, including: elapsed time and computational resources used (RAM and processor).

## 1.3 Main Contribution

Technologies to exploit the power of the oceans and seas are still at an early stage of development, [4, 19]. The wave energy absorption is a hydrodynamic process of considerable theoretical difficulty, in which relatively complex diffraction and radiation wave phenomena take place. During the second half of the 1970s, large part of the work on wave energy published was on theoretical hydrodynamics, [20]. The time-domain model procedures are the appropriate tools for active-control studies of converter in irregular waves. However, it requires much more computing time as compared with the frequency-domain analysis, [20].

The results in [10] indicate that the control is essential in future designs, and [15] emphasizes the value of a computationally affordable hydrodynamic model which can be used as bases for the model-based control design. Thus, the development of a fast implementation time domain control for the pumping system is crucial for the Ocean Grazer. For that reason, a new port-Hamiltonian model, based in the model used in [18], is described, including the hydrodynamics of the piston that weren't considered, giving a more realistic approach.

Besides, a new control strategy, obtained through open source is proposed to decrease the computational cost of the model and facilitate the optimization strategy implementation. The control strategy will be delivered to the OG team as a preliminary model predictive control (MPC) optimization toolbox in an open source environment. The use of open source not only allows more free modification to the codes, but also can reduce the implementation cost in the device, since no license is required. At the same time, the set of functions can be extrapolated to other models, allowing to directly compare its behaviour.

## 1.4 Structure of the Proposal

Since the problem to solve is the current computational cost of the current time domain model, a control strategy over a port-Hamiltonian model is proposed. First of all, a bibliographic review will be done in order to analyse already existing similar applications and choose an appropriate paradigm, showing the main concepts related to the project in Chapter 2. Then, a solution analysis is done in Chapter 3, where the problem is analysed in detail in Section 3.1, indicating the main causes of the high computational cost, formally establishing the problem and indicating the requirements asked by the OG team.

The methodology to chose an appropriate solution is presented in Section 3.5, explaining the selection criteria and then, a rubric is made to determine the best solution from a list of several proposals. On the other hand, the base model presented by [18] and its modifications are detailed in Chapter 4, including the algorithm used in the MATLAB code used by [18] and the assumptions to simplify the newg model.

The description of the control strategy is presented in Section 4.3, describing the control variable, the cost function and the algorithm for the control implementation, which results are shown in Chapter 5, including open loop results of the previous and the proposed model, as well as the result when the MPC is implemented in a short interval of irregular waves. Finally, Chapter 6 is dedicated to summarize the main conclusions of this research, establishing the limitations of the developed strategy and proposing a series of recommendations for future research.

# Chapter 2

## Generalities

This Chapter aims to briefly present all the basic concepts related to the project's theoretical background that the reader would need. First, general information about waves is presented in Section 2.1, including the main parameters definition. Wave energy context will be summarized in Section 2.2 and some of the more frequent extraction methods are mentioned in Section 2.3.

After, the port-Hamiltonian modelling method is postulated in Section 2.4, and explained using a simple mass-spring-damper example, which is the base of the port-Hamiltonian model used in this project. Section 2.5 is dedicated to Model Predictive Control, explaining the strategy, describing some of the principal methods and concluding with a short list of application of MPC in renewable energies. Finally, Section 2.6 is dedicated to the Ocean Grazer project, presenting the device, its power take-off systems, the wave energy converter principle and some of the previous control strategies that have been proposed.

### 2.1 Waves

According with [23], a common mistake is to assume that the concept of surface elevation is the same as wave. The first, usually denoted as  $\eta(t)$ , is the instantaneous elevation of the surface of the water, relative to a reference level. On the other hand, a wave is the profile of the surface elevation, between two successive downward zero-crossings of the elevation.

The surface elevation movement is separated in two individual movements: the up-stroke, that is when the surface elevation increases, from a local minimum value to the next local maximum value; and the down-stroke, that is when the surface elevation decreases from that local maximum value to the next local minimum.

Likewise, the appearance of the waves can be inferred according to the frequency spectrum. If a spectral analysis of the wave is done, the tighter the spectrum is, the more regular the resulting wave will be, thus, the presence of a wider band will result in a chaotic wave field, also called irregular waves, because the components in the time record get out of phase

with one another quickly, [23], as can be seen in Figure 2.1.

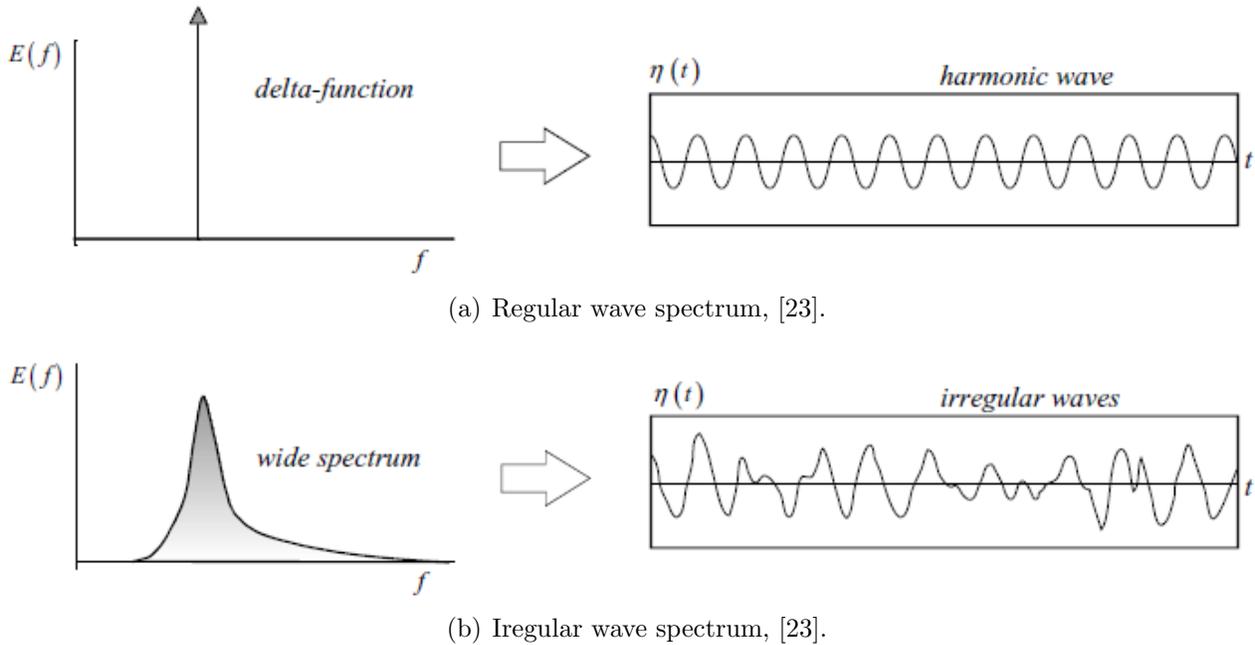


Figure 2.1: Character of the waves according with its frequency spectrum width.

Due the irregularity of the oceanic waves, several statistic methods have been developed to characterize the waves using time records. First, the sampling time must be short enough to be considered as stationary, but also long enough to obtain reliable data, commonly used periods are between 15 and 30 min, [23]. Once the data is registered, representative quantities are used to characterize the height and period of the waves.

The wave height is defined as the vertical distance between the highest surface elevation (also named peak) and the lowest (named valley) in a wave, [23]. Although each wave will have only one height, in a record with  $N$  waves, the mean wave height is defined as

$$H_M = \frac{1}{N} \sum_{i=1}^N H_i. \quad (2.1)$$

Another representation for the wave height in a record is the quadratically weighted average value, resulting in a root-mean-square wave height. Those measures are relevant for energy-related projects, since the wave energy is proportional to the wave height squared. However, they are not commonly used due its poor resemblance to the visually estimated wave height. In [23] is mentioned that, instead, the significant wave height  $H_s$  is used, which is defined as the mean of the highest one third of the waves of the record, as follows

$$H_s = \frac{3}{N} \sum_{j=1}^{N/3} H_j, \quad (2.2)$$

in this case,  $j$  doesn't refer to the sequence number of the wave in the record, but to its rank number based on their individual height, [23]. Analogously, the wave period (time between two consecutive zero-down crossing) can be defined as the mean period during the record

$$T_M = \frac{1}{N} \sum_{i=1}^N T_{0,i}. \quad (2.3)$$

However, as with the the mean wave height, if a relation with the visual estimation is done, following the same criteria than the significant wave height, the significant wave period is calculated as shown

$$T_s = \frac{3}{N} \sum_{j=1}^{N/3} T_{0,j}. \quad (2.4)$$

## 2.2 Wave Energy

Wave energy has being investigated since the 1970's; however, the European Commission included it in their R&D program on renewable energies until 1991, [20]. Thereby, and particularly in the last two decades, most of the R&D has being done in Europe, because the positive attitude adopted by some European national governments. As well, in the last few years, the interest in wave energy has been growing rapidly also in other countries, [20].

In 2012, according to [21], 23,4 % of the electricity produced came from renewable resources, and only 0,1 % where from ocean energy. This kind of energy is described by [13], and includes different kinds of energy that uses any phenomena that occur in the ocean. By 2014, the global ocean energy power capacity was nearly 530 MW, and is predicted that will reach 640 MW by 2021, [3].

The wave energy is normally expressed as power per unit of crest length, or wave energy transport flux; whose typical annual average range for good offshore locations is between 20 kW/m and 70 kW/m, occurring mostly in moderate to high latitudes, [20]. The wave power can be calculated from the spectrum. In deep water, however, the power wave per unit length (in kW/m) can be calculated as

$$P_w = \frac{\rho g^2 H_s^2 T_M}{64\pi}, \quad (2.5)$$

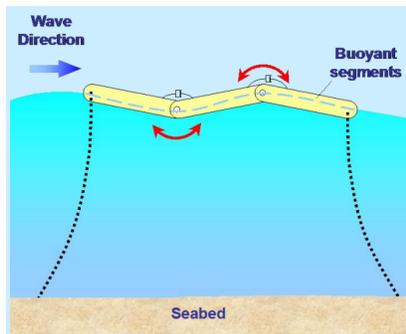
which depends on the wave mean period  $T_M$  and the significant wave height  $H_s$ , [24].

## 2.3 Wave Energy Extraction Methods

Ocean power can be exploited using five different technologies: Tidal rise and fall, ocean currents, waves, temperature gradients and salinity gradients. Of those, the main focus has given to the first two. [3]. Additionally, the work in [21] indicates that wave energy conversion has the highest theoretical potential and the present work will be focused on it. The main extraction methods are described in [5] and will be detailed next.

### 2.3.1 Knick Absorber Systems

Consist of several cylindrical sections that flex and bend because of the waving as shown in Figure 2.2(a). The movement pumps pressurized oil through hydraulic motors, driving electrical generators and producing electricity, [5]. This method is currently used in Portugal, in the 2,25 MW Agu adoura Wave plant, which is the world’s first grid connected wave farm, [4]. There, three 750 kW Pelamis devices, as shown in Figure 2.2(b), developed by Pelamis Wave Power and ScottishPower Renewables for the European Marine Energy Centre (EMEC), [5, 25]. In the future, the station objective is to expand to 25 Pelamis machines, increasing the capacity to 21 MW, [26].



(a) Knick absorber operating principle, [27]

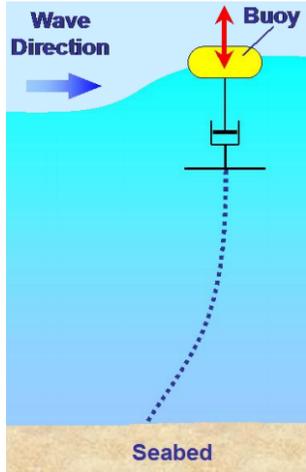


(b) Knick absorber Pelamis P2-001 in Portugal, [25]

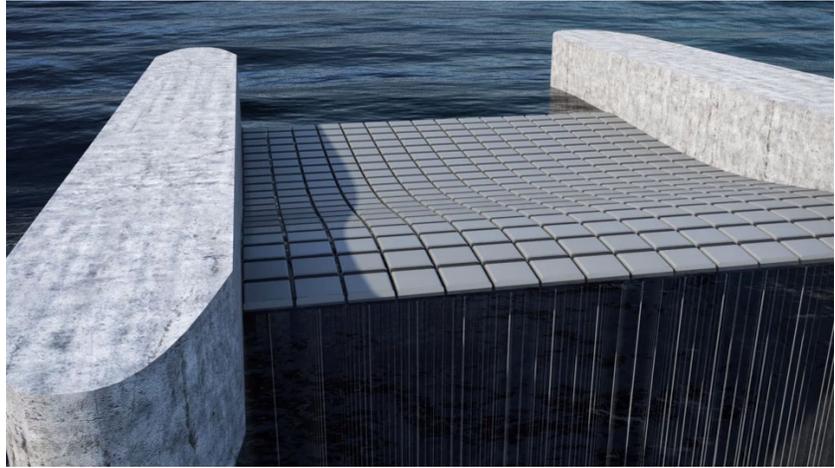
Figure 2.2: Oscillating Water Column systems examples.

### 2.3.2 Point Absorber Systems

Consist on a floating device which uses the vertical movement of the waves to pressurize gas or pumps a liquid, actuating a turbine, [5], as shown in Figure 2.3(a). The Ocean Grazer WEC, shown in Figure 2.3(b), is based on this method and will be detailed in Section 2.6.



(a) Point Absorber systems operating principle, [27].



(b) Ocean Grazer 1.0 concept, [47]

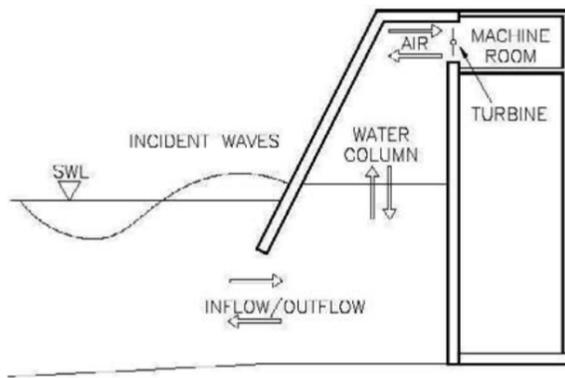
Figure 2.3: Point Absorber systems examples.

### 2.3.3 Oscillating Water Column Systems

Use a wave capture chamber and an air chamber. The waves enter in the wave capture chamber during the upstroke, forcing the air through a turbine which acts as outlet. Then, the down stroke creates an under pressure, sucking the air into the chamber through the turbine again, [5], as shown in Figure 2.4(a). The review in [28] mentions that the first experimental plants were constructed in early 80's, generating about 12 kW, but it was until 2001 when Wavegen (now VH Wavagen), with collaboration with Queens University Belfast, connected to the grid the plant named LIMPET, shown in Figure 2.4(b), which generated up to 500 kW, [29]. More recently, RWE npower renewables has proposed a 4 MW scheme at Siadar, on the outskirts of Scottish Isle Lewis, [28].

### 2.3.4 Over-topping Terminator Systems

Consist on a large floating reservoir with an entry side featuring as a ramp for the water to get into the reservoir. The overtopping wave has to leave the reservoir through a turbine to flow back into the ocean, [5], as shown in Figure 2.5(a). This method is used by the Wave Dragon, a Danish project, shown in Figure 2.5(b), which was the world's first offshore wave energy converter grid connected and producing power. Its main objective is to develop a



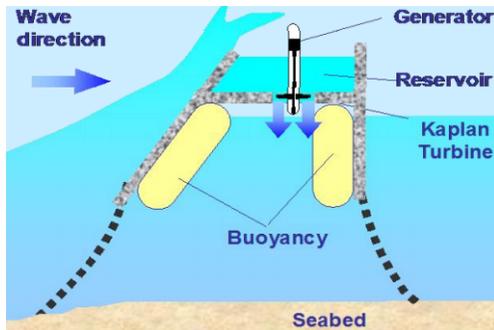
(a) Oscillating Water Column operating principle, [28].



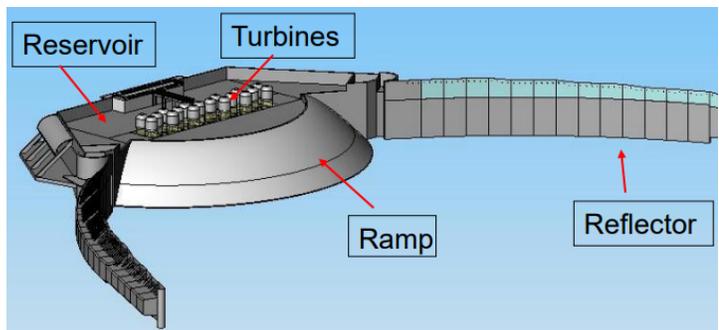
(b) LIMPET plant in Scotlan, [29].

Figure 2.4: Oscillating Water Column systems examples.

power plant unit to produce between 4 MW to 11 MW with a competitive production price per kWh, [30]. One of the main advantages mentioned by [27] is that is a non-resonating structure, thus, there is a lower risk of damage as the structure does not move, but the largest waves pass over the device.



(a) Overtopping Terminator operating principle, [27]



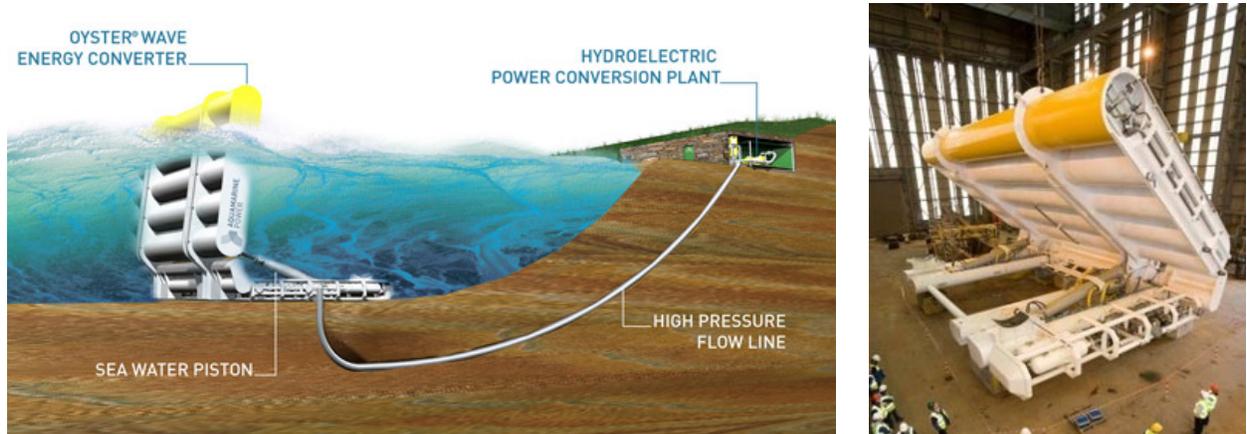
(b) Wave Dragon representation, [27]

Figure 2.5: Over-topping Terminator systems examples.

### 2.3.5 Oscillating Wave Surge Converter Systems

Consist on a large paddle that spins on an axis perpendicular to the direction of the waves. Its movement translate a horizontal piston, which compresses a water column through a turbine, [5], as shown in Figure 2.6(a). The Oyster wave energy device, launched in 2009 in Scotland is based on this method, and is designed to capture the energy found in near-shore waves in water depths between 10 m and 16 m, with all the electrical components onshore [31], as shown in Figure 2.6(b). According with [32] and [33], each Oyster device will produce

about 315 kW. Due to this full scale prove succes, the Oyster 2 project is under development, which consists on a three oscillator 2,4 MW system, [32].



(a) Oscillating Wave Surge Converter operating principle, [33].

(b) Oyster device in Scotland, [33].

Figure 2.6: Oscillating Wave Surge Converter systems examples.

## 2.4 Port-Hamiltonian Systems

The port-Hamiltonian modelling theory converges different traditional modelling approaches. First, the port-based modelling, which provides a unified framework of different physical domains establishing the energy as the connection between them, identifying the ideal system components and its physical characteristics. The second branch is the geometric mechanics, which basic paradigm remains in the representation of the dynamics in a coordinate-free manner using a state-space, together with a Hamiltonian function representing the energy of the system. Finally, systems and control theory emphasizes the dynamical systems as being open to interaction with the environment through inputs and outputs, which are susceptible to control interaction. Also, energy-dissipating elements are included, [34].

In port-based modelling, the physical system is regarded as the interconnection of three ideal components: energy-storing elements, energy-dissipating elements and energy-routing elements. The relation between the elements in a pH representation is given in ports-pairs of flow ( $f$ ) and efforts ( $e$ ) as presented in Figure 2.7.

**Energy-storing elements:** denoted by  $S$ , represent all the elements in the system that stores energy, *e.g.* inductors, capacitors and springs, [34].

**Energy-dissipating elements:** also known as resistive elements, are denoted by  $R$  and represent all the elements in the system that dissipates energy, *e.g.* resistors and dampers, [34].

**Energy-routing elements:** denoted by  $D$ , since are Dirac structures, represent the connection between the other elements, with the basic property that they conserve the power, *e.g.* transformers, gyrators and ideal constraints, [34].

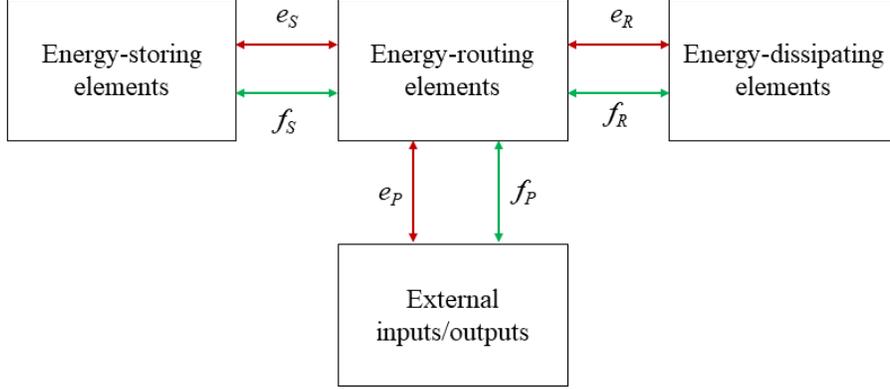


Figure 2.7: Port-Hamiltonian elements relation.

Mathematically, the port-Hamiltonian representation uses a state-space representation of the state  $\mathbf{x}$ , where its derivative is a function of the partial derivative of the Hamiltonian function  $\mathbf{H}(x)$ , that describes the energy of the system in any instant, associated to the states. According with [34] and [35], any system

$$\Sigma = \begin{cases} \dot{\mathbf{x}} = [\mathbf{J}(x) - \mathbf{R}(x)] \frac{\partial \mathbf{H}(x)}{\partial \mathbf{x}} + \mathbf{G}(x) \mathbf{u} \\ \mathbf{y} = \mathbf{G}^T(x) \frac{\partial \mathbf{H}(x)}{\partial \mathbf{x}} \end{cases}, \quad (2.6)$$

where  $\mathbf{x}$ , with  $\mathbf{x} \in \mathbb{R}^n$ , represents the states of the system and  $n$  the number of states. The matrix  $\mathbf{J}(x)$  is a skew-symmetric matrix called interconnection matrix and  $\mathbf{R}(x)$  is a positive semi-definited matrix called dissipation matrix, with  $\mathbf{J}(x), \mathbf{R}(x) \in \mathbb{R}^{n \times n}$ . The Hamilton function  $\mathbf{H}(x)$ , with  $\mathbf{H}(x) \in \mathbb{R}^n$  represents the total stored energy in the system in any state, and  $\mathbf{G}(x)$ , with  $\mathbf{G}(x) \in \mathbb{R}^{n \times m}$  is the input weighting matrix, been  $\mathbf{u}$ , with  $\mathbf{u} \in \mathbb{R}^m$ , the input of the system, and  $m$  the number of inputs, [36].

To verify the resulting model, the  $\mathbf{J}$  and  $\mathbf{R}$  matrices must fulfil several conditions. The interconnection matrix, since is a skew-symmetric, must be equal to the opposite of its transpose, that means,

$$\mathbf{J} = -\mathbf{J}^T, \quad (2.7)$$

but it also must fulfil the relation

$$\mathbf{x}^T \mathbf{J} \mathbf{x} = 0, \quad (2.8)$$

where  $\mathbf{x}$  corresponds to a non-zero column vector. The dissipation matrix, since its a positive semi-definite matrix, it must satisfy the inequity:

$$\mathbf{x}^T \mathbf{R} \mathbf{x} \geq 0, \quad (2.9)$$

where  $\mathbf{x}$  corresponds to a non-zero column vector. At the same time, it must be equal to its transpose, i.e.

$$\mathbf{R} = \mathbf{R}^T. \quad (2.10)$$

### 2.4.1 pH Model of a Mass-Spring-Damper System

As will be described in Chapter 4.3, the model to be used simplifies the WEC to a set of individual mass-spring-damper systems attached to the sea bed, with no direct connection between masses. To introduce the port-Hamiltonian method and simplify the understanding of the model, a simple mass-spring-damper system will be modelled using a pH representation.

First consider the mass-spring-damper system presented in Figure 2.8. In this case, the state variables are going to be the position  $q$  and the momentum  $p$  of the mass. If a force analysis is done, considering the velocity of the mass as  $\dot{q}$  and the acceleration as  $\ddot{q}$ , the mass, the sum of the forces, i.e. the external force  $u$ , minus the force of the spring  $f_k$ , minus the force of the damper  $f_b$  will be equal to the mass times the acceleration, as shown

$$\sum_{i=1}^n f_i = f_{k,i} + f_{b,i} + u_i = m_i a_i, \quad (2.11)$$

where  $n$  is the number of buoys.

Likewise, if (2.11) is rewritten in terms of the position, velocity, acceleration, stiffness and damping coefficient, it is obtained

$$m\ddot{q} = -kq - b\dot{q} + u. \quad (2.12)$$

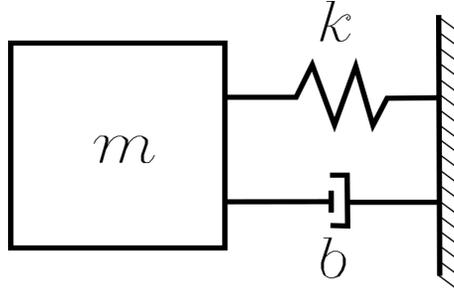


Figure 2.8: Single mass-spring-damper system

Furthermore, if the momentum of the mass is defined as  $p = m\dot{q}$ , and its derivative as  $\dot{p} = m\ddot{q}$ , (2.12) is rewritten as

$$\dot{p} = -kq - b\dot{q} + u. \quad (2.13)$$

On the other hand, the Hamiltonian of the system can be defined as the sum of the elastic energy stored in the spring and the kinetic energy of the mass while moving, that means

$$H(q, p) = \frac{k\Delta x^2}{2} + \frac{mv^2}{2} = \frac{kq^2}{2} + \frac{p^2}{2m}, \quad (2.14)$$

that when is derived with respect to position and momentum results respectively in

$$\frac{\partial H(q, p)}{\partial q} = kq \quad (2.15)$$

and

$$\frac{\partial H(q, p)}{\partial p} = \frac{p}{m}. \quad (2.16)$$

Given the definition of the derivative of the position and the momenta, together with (2.13), (2.15) and (2.16), it is possible to build the system presented in (2.6), resulting in

$$\Sigma = \begin{cases} \begin{bmatrix} \dot{\mathbf{q}}_{n \times 1} \\ \dot{\mathbf{p}}_{n \times 1} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{n \times n} & \mathbf{I}_{n \times n} \\ -\mathbf{I}_{n \times n} & -\mathbf{b}_{n \times n} \end{bmatrix} \begin{bmatrix} \frac{k\mathbf{q}_{n \times 1}}{\mathbf{p}} \\ \mathbf{m}_{n \times 1} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{n \times m} \\ \mathbf{I}_{n \times m} \end{bmatrix} \mathbf{u}_{m \times 1} \\ \mathbf{y}_{m \times 1} = \begin{bmatrix} \mathbf{0}_{m \times n} & \mathbf{I}_{m \times n} \end{bmatrix} \begin{bmatrix} \frac{k\mathbf{q}_{n \times 1}}{\mathbf{p}} \\ \mathbf{m}_{n \times 1} \end{bmatrix} \end{cases}, \quad (2.17)$$

where  $n$  is the number of elements and  $m$  the number of inputs. Thus, if the single mass-spring-damper system is considered, (2.17) turns into

$$\Sigma = \begin{cases} \begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -b \end{bmatrix} \begin{bmatrix} \frac{kq}{p} \\ m \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{kq}{p} \\ m \end{bmatrix} \end{cases}. \quad (2.18)$$

From (2.18) it's possible to obtain the interconnection and the dissipation matrices, resulting, respectively, in

$$\mathbf{J} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad (2.19)$$

and

$$\mathbf{R} = \begin{bmatrix} 0 & 0 \\ 0 & b \end{bmatrix}. \quad (2.20)$$

To ensure that the previous results are correct, the interconnection matrix, presented in (2.19) must fulfil the conditions presented in (2.7) and (2.8). Likewise, the dissipation matrix presented in (2.20) must fulfil the conditions presented in (2.9) and (2.10). The proof of (2.7) and (2.10) is considered trivial and won't be developed. Starting with the interconnection matrix, the condition presented in (2.8) can be easily checked when defining the vector  $\mathbf{x}$  as:

$$\mathbf{x} = \begin{bmatrix} a \\ b' \end{bmatrix}. \quad (2.21)$$

Given the vector  $\mathbf{x}$ , (2.8) can be rewritten as follows

$$\begin{bmatrix} a & b' \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} a \\ b' \end{bmatrix} = ab' - ab' = 0, \quad (2.22)$$

fulfilling the condition. Likewise, a similar result is expected when (2.20) and (2.21) are substituted in (2.9), resulting in

$$\begin{bmatrix} a & b' \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} a \\ b' \end{bmatrix} = bb'^2 \geq 0, \quad (2.23)$$

where the result is independent of the sign of  $b'$ , and the damping of the system obeys the relation  $b \in \mathbb{R}^+$ .

## 2.5 Model Predictive Control

Model Predictive Control (MPC) is a widely used control technique for multivariable control problems [37]. It consists on an ample range of control methods, instead an specific control strategy [38]. However, this methods are based on the same principle: first, the algorithm uses a model to predict the outputs of the system in a determined set of future instants, called horizon. Second, it calculates the control sequence by optimizing (maxmizing or minimizing) an objective function, called cost function, which is defined by the designer; and finally recedes the sequence each sampling instant, displacing the horizon to the future, as shown in Figure 2.9, [37, 38, 39]. MPC includes several control methods whose mathematical demonstration can be found in [38].

To explain the MPC process, for example, consider as cost function the error between the state of the system and the state it must have. In that case the variables are: the current output  $y(t)$ , the set-point  $s(t)$  which is the output the plant should follow, the reference trajectory  $\hat{r}(k|k)$  which is the ideal trajectory along which the plant should return to the set-point, the predicted free response of the plant  $\hat{y}_f(t|k)$  which corresponds to the response that would be obtained at the coincidence point if the future input remained at the latest value the response of the model after the applied input  $\hat{y}(t|k)$  and  $H_p$  is the prediction horizon, *i.e.* the number of steps ahead that the calculations are made, [39]. The previous variables are shown in Figure 2.10.

It is important to mention the notation when using MPC to indicate the step of the variables:  $(k+n|k)$ , in the right side of the vertical bar is indicated the real step in which the variable is analysed, and the left side indicate the  $n$ th predicted step over the predicted horizon, considering as reference the step  $k$ , *e.g.*  $y(k+2|k)$  is the predicted output of the second step after the  $k$ -th step.

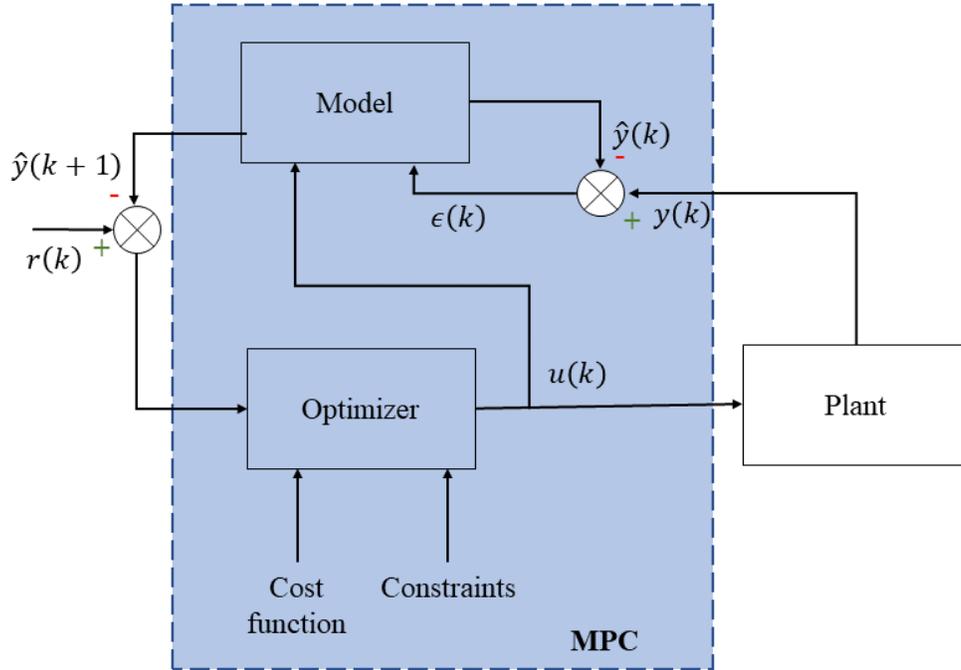


Figure 2.9: MPC main process, where  $r(k)$  is the reference trajectory,  $u(k)$  is the process input,  $y(k)$  the process output,  $\hat{y}(k)$  the predicted output,  $\hat{y}(k+1)$  the future predicted output,  $\epsilon(k)$  is the error between the predicted output and the process output.

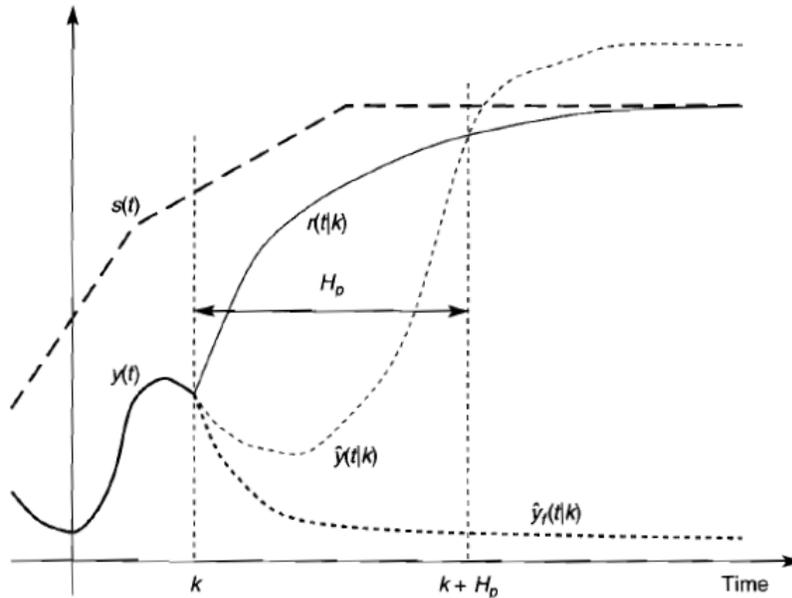


Figure 2.10: MPC basic principle using the error between the output of the system and a set-point trajectory as cost function, where  $k$  is the current instant,  $y(t)$  the output of the system,  $s(t)$  the set-point trajectory,  $r(t|k)$  the reference trajectory with respect to the instant  $k$ ,  $\hat{y}(t|k)$  the predicted output with respect to the instant  $k$ ,  $\hat{y}_f(t|k)$  is predicted free response to the system with respect to the instant  $k$ , and  $H_p$  the prediction horizon, [39].

### 2.5.1 Dynamic Matrix Control

Because its ability to deal with multivariable processes, Dynamic Matrix Control, or DMC, has been accepted in the industrial world. The process model employed in DMC is the step response of the plant, considering the disturbance as a constant along the horizon. The measure disturbances are taken as system inputs and, in general, disturbances are considered to be the sum of the following effects: the response input, the measurable and nonmeasurable disturbances and the actual process states. Then, DMC drives the output as close to the setpoint as possible, using the least-squares method, however, it is necessary to keep a safe zone around the operating point, since the perturbations can make the process violate the established constraints, [38].

### 2.5.2 Model Algorithmic Control

Similar to DMC, Model Algorithmic Control, also known as MAC, uses the truncated step response of the process to provide a simple explicit solution in absence of constraints. This model defines the output as a linear combination of past input values on a stable and causal system. Also, disturbances are assumed as constants in the future as the current value. The trajectory is usually a smooth approximation from the current state towards a known reference, where the shape of the trajectory determines the desired speed of approach to the setpoint, providing robustness to the control algorithm, which is proportional to the time constant, [38].

### 2.5.3 Predictive Functional Control

Predictive Functional Control, or PFC, has two main differences with respect to DMC and MAC: first, the control signal is a linear combination of basis functions (normally polynomial; such as step, ramp or parabolas), reducing the number of unknown parameters and resulting in an advantage when controlling nonlinear systems. Second, coincidence points are used to evaluate the cost function along the horizon, *i.e.* the predicted error is not considered all along the horizon, but only in certain instants called coincidence points, that can be used as tuning parameters, considering its influence on the stability and robustness of the control system. It's important to mention that the number of coincidence points must be, at least, the same number of the selected number of basis functions. Besides, are chosen to be optimal at each instant, therefore, are different at each step. Finally, PFC uses a state-space model of the process (including nonlinear systems) and can only be used for stable models, [38].

### 2.5.4 MPC in Renewable Energies

Due to the flexibility and ability to be used in multivariable applications, the popularity of MPC in renewable energies has been increasing for the last decades as response of the growth in renewable energies development (including research and installation). Therefore, control

and optimization strategies must be implemented to increase the efficiency of the systems and maximize the extracted energy, for example, in [40] is mentioned that MPC in solar photovoltaic and wind energy systems has been successfully implemented, using the output voltage and current as parameters to be optimized.

In wind energy, the work presented in [41] demonstrates that load frequency control (LFC), using MPC in closed-loop systems, is more robust against perturbations than classical integral control designs. However, MPC can be also implemented in a combination with conventional proportional-integral (PI); the simulation results in [42] using double fed induction generator (DFIG) wind turbines show faster response, more robustness against uncertainties and load changes than when those strategies are used individually.

In wave energy, MPC implementations on WECs start to appear in the early 2010's, previously, passive controllers based on impedance matching were mainly used to match the resonance frequency of the WEC with the dominant frequency of the incoming waves, [43]. Since then, many studies have been done to analyse different approaches of the MPC, specially those independent of wave prediction, but using only direct measurements, as did in [43], [44] and [45].

A complete description of the implementation of an MPC in a buoy type WEC is presented in [44]. The buoy is modelled as a single degree of freedom (DoF) oscillator, which will have its maximum range of motion when is in resonance with the excitation forces of the wave. Several configurations were tested, and even though long prediction horizons showed high profit, a prediction horizon of 10 s with a sampling time of 0,1 s (*i.e.* 100 samples) is close enough to the optimal controller since longer prediction horizons (more samples) does not make significant changes in the power extraction.

## 2.6 Ocean Grazer

As mentioned before, the ocean is a huge energy source, and the Ocean Grazer is an integral platform which is expected to produce a combined average output of 260 GWh from different sources. The OG energy harvesting farm exploits five different sources of energy using different power take-off systems (PTOs), which are mentioned by [5]. Moreover, due to the massive structure, it has a reservoir which can store 800 MWh of loss-free potential energy that can be extracted by hydroelectric turbines, [5, 13, 46].

**PTO 1 (wave energy):** consist on a multi-piston multi-pump (MP<sup>2</sup>PTO) based on the point absorber method, but with the difference that the OG uses a consecutive line of floaters (a blanket), organized in rows, as shown in Figure 2.11, producing between 160 GW and 200 GWh according with [5] and [13].

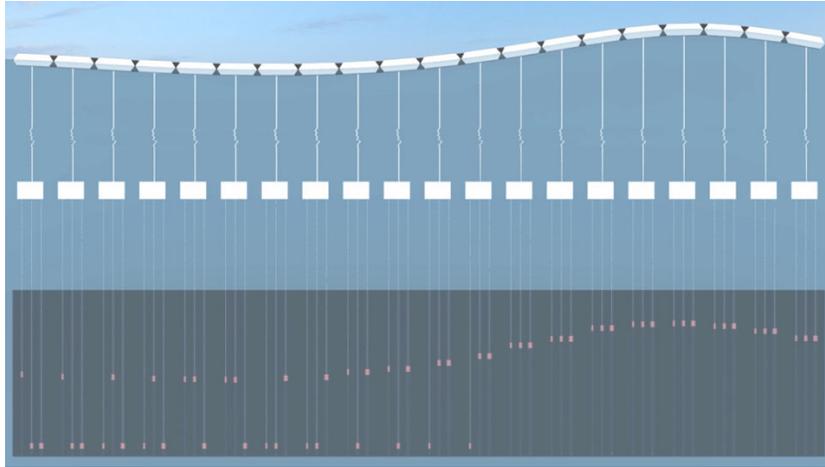


Figure 2.11: Multi-piston multi-pump (PTO 1) concept, [47].

**PTO 2 (wave energy):** taking advantage of the rise and fall of the ocean, an oscillating water column system generate electricity through the displacement of air in a pressure chamber, [5].

**PTO 3 (wind energy):** even though an exhaustive analysis hasn't been done yet, it is estimated that three 100 m height wind turbines can be placed on the surface of the OG, as shown in Figure 2.12, producing about 5 MWh, [5].

**PTO 4 (solar energy):** due the top surface of the OG is about 50 000 m<sup>2</sup>, as shown in Figure 2.12, solar panels can be used to obtain up to 10 MW, [5].



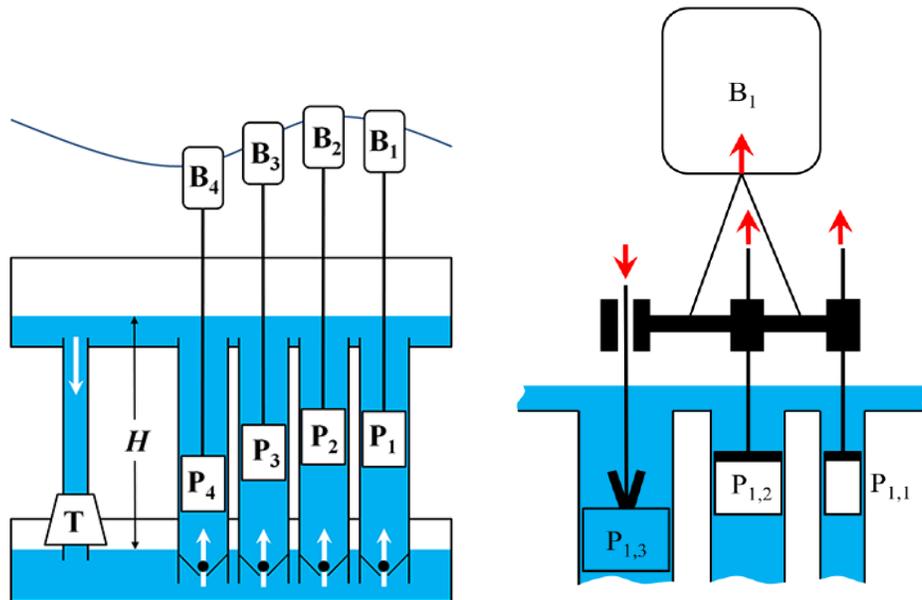
Figure 2.12: OG PTO 3 and 4 representation, [47].

**PTO 5 (additional energy):** additional energy sources, like horizontal wave movement, underwater currents or salinity differences can be exploited, however, its viability is still being investigated, [5].

## 2.6.1 WEC Operation Principle

The WEC is part of the Ocean Grazer energy harvesting platform, project that is being developed in the University of Groningen. This point absorber system consists on a series of interconnected floaters or buoys, which are connected to individual hydraulic multi-piston pumps. Since the waves height and period change with respect to time, producing a non-linear pumping force [10], pistons need a control strategy in order to optimize the buoy load during the upstroke movement and increase the system efficiency, [13].

The proposed WEC consists on a multi-pump, multi-piston power take-off (MP<sup>2</sup>PTO) system. A mechanical design and model is described in [10], indicating that the operating principle of the MP<sup>2</sup>PTO WEC is to create a pressure difference in the working fluid circulating between two reservoirs, in order to transform the potential energy stored by the device into electricity by releasing the aforementioned internal fluid through a turbine.



(a) Representation of the Ocean Grazer multi-pump concept, [46] (b) Representation of the Ocean Grazer multi-piston concept, [13]

Figure 2.13: Representation of the Ocean Grazer MP<sup>2</sup> PTO.

## 2.6.2 WEC Previous Control Proposals

The work in [10] describes a mechanical design and modeling of a single-piston pump for the Ocean Grazer WEC, [16] proposes a equivalent single piston pump of a multi-piston pump (MPP) model for the OG WEC and [13] develop a non-linear control design for the OG WEC, and they agree in the complexity of the OG WEC. Nonetheless, the computational cost of those studies is not viable for a large number of elements.

A first approach for a model predictive control (MPC) is described in [16], resulting in an effective solution. The MPC strategy consist in making predictions based on the system dynamics and use those solutions to obtain an optimal sequence of controls or decisions, and then apply them on the next step of the problem [37]. However, if the model is already computationally demanding, it turns in an even more complex in closed loop.

The adaptability of the MP<sup>2</sup>PTO system is carefully detailed in [15], where a MATLAB/SIMULINK environment is used to build the model and using the open-source code NEMOH to calculate the hydrodynamic coefficients required for the calculations by the boundary element method. The results accuracy was satisfactory, but the model is time consuming, making it difficult the implementation of a control strategy.

## 2.7 Concluding Remarks

A general perspective of wave energy was presented, giving the mathematical expression for the principal parameters used to describe the waves. Also, some of the main extraction methods were briefly described, giving specific examples of their implementation in past, current and future applications. Then, the port-Hamiltonian modelling was proposed as a method which allows to understand the energetic behaviour of the system modelled, that was demonstrated using as example a simple mass-spring-damper system, that is also used as base by [18] to model the OG WEC.

Furthermore, the Model Predictive Control strategy was proposed, including some of its approaches and specific cases where the MPC has being successfully applied in renewable energies, including wave energy. Finally, the Ocean Grazer project is presented, announcing the power take-off systems and focusing on the wave energy converter, giving a recap of previous models and their control proposals.

# Chapter 3

## Solution Proposal

This Chapter considers the context presented previously and the information given by the OG team to summarize the reason why the control strategy currently demands a high computational cost. Then, based on the user requirements established by the OG team and presented in Section 3.3, the system requirements to quantify the research goals are listed in Section 3.4.

In order to develop a solution, it is important to consider several possible solutions and evaluate them to choose the optimal. Likewise, the developed solution must be validated. Thus, a selection criteria is specified in the Section 3.6 to choose between the control strategies proposed in Section 3.7, and in Section 3.8 to choose between the programming languages considered in Section 3.9. Then, two possible solutions are proposed in Section 3.10 and analysed in Section 3.11. Finally, Section 3.12 establishes the criteria to validate the solution.

### 3.1 Problem Determination

Wave energy is inherently stochastic, as a consequence of wind energy. The conversion of wave energy into usable energy is extremely complex due to the hydrodynamic processes presented in the diffraction and radiation of waves as they propagate to shore, [4]. The main disadvantage of wave power, as with the wind from which it is originated, is its random variability in several time-scales: from wave to wave, with sea state, and from month to month, [20].

The work in [13] indicates the importance of a proper control strategy for wave energy systems efficiency, and specifically with the OG WEC, using the piston area as control variable. Therefore a high fidelity model is required, but because of the high amount of variables used by [10] and [16] and the Boundary Element Method nature, described by [48] and [49], used to calculate the hydrodynamic coefficients in the time domain model, the simulations computational cost is considerable in both, hardware and time.

A model predictive control strategy, which is described by [37] and [50], can be used in order to increase the system efficiency, but turns very computationally demanding with com-

plex models, like the current one. The current time domain model, using ten floater elements, presented by [15], takes about one day to run on a computer with an Intel Xeon Processor at 3.5 GHz and 64 GB of RAM. It uses the multi-body dynamics solver Multibody<sup>TM</sup>, which is based on the open-source tool WEC-Sim, and the open-source code NEMOH to calculate the hydrodynamic coefficients, and the its results are validated against experimental data and other analytical models.

To summarize the problems which leads to the need for a fast implementation control strategy, an Ishikawa diagram is presented in Figure 3.1.

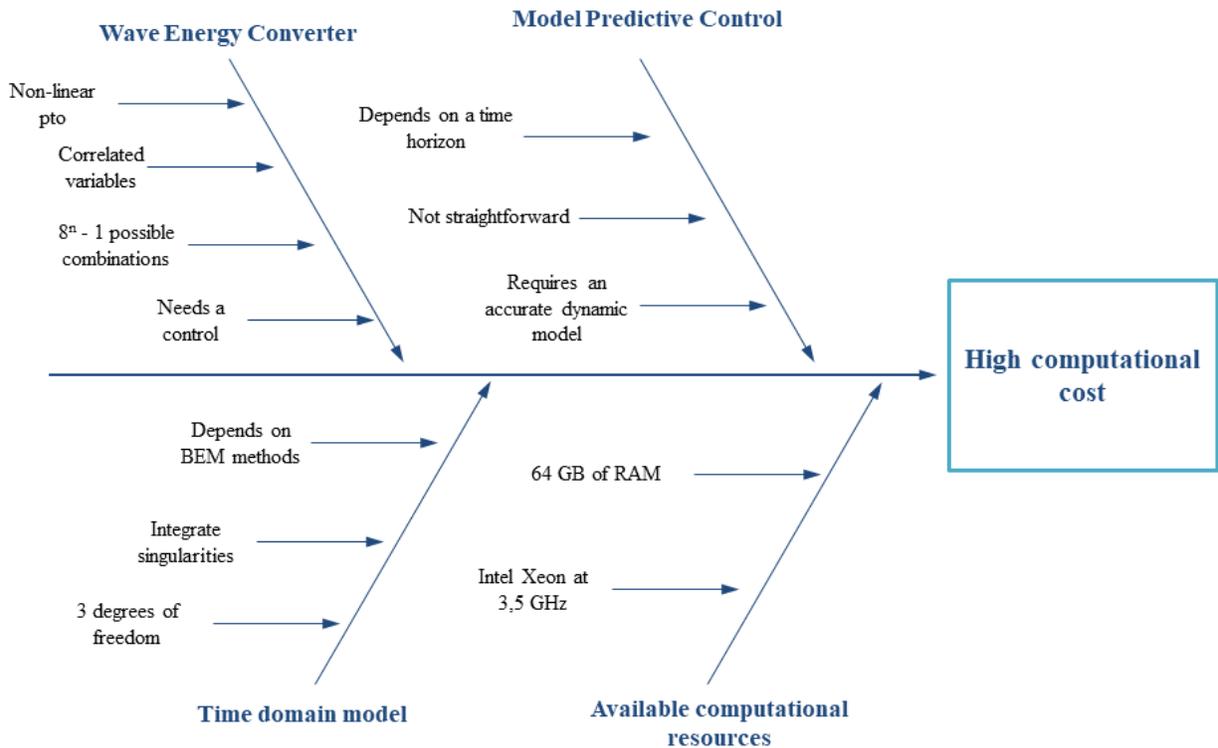


Figure 3.1: Ishikawa diagram with the causes of the computational cost of controlling the current time-domain model.

### 3.2 Problem Synthesis

The Ocean Grazer WEC needs a controller to maximize the extracted energy, and its implementation is not straightforward, for that reason, a low computationally demanding control strategy is required.

### 3.3 User Requirements

Each buoy on the OG WEC blanket can enable up to three pistons with different areas, creating eight different possible combination of pistons per buoy. Thus, the control system has to analyse the incident wave and determine which is the best combination of pistons for all buoys in the blanket that ensures the maximum absorbed energy. Given the above, the user requirements asked by the OG team for the solution are:

- The developed solution response must be as reliable as possible, using the pH model presented in [18] as base.
- The developed solution must be less computationally demanding than the model proposed by [15].
- The developed solution must be compatible with the OG WEC.
- The solution must be implemented in an open source language.

### 3.4 System Requirements

Given the user requirements presented before, the system requirements are:

- The developed solution must use port-Hamiltonian model, base on the model presented in [18].
- The developed solution must give the optimal piston arrangement in less than 24 hours when executed in a computer with 64 GB of RAM and an Intel Xeon processor.
- The developed solution must ensure that the energy loss, compared with the model presented in [10], considered as maximum theoretical energy, is less than 10 %.
- The developed solution must be able to use the information from the system sensors, and the output data has to be able to control the pistons.
- The solution must be implemented in an open source language.

### 3.5 Solution Approach

The time domain model of the Ocean Gracer WEC presented by [15] takes into consideration a high amount of variables, including the hydrodynamic coefficients and three degree-of-freedom (DoF), ensuring a high fidelity time domain model [16]. However, the complexity of the model implies that each simulation requires a significant computational cost (computation time and required hardware), which is counterproductive at the moment of implementing a control strategy, because it is not straight forward, and, in control implementations, computation time is as crucial as the response itself. Next, the solution criteria will be detailed to obtain an optimal solution.

## 3.6 Control Strategy Selection Criteria

The first step to implement a control strategy is to define which is going to be used. So far, several control strategies have been studied by the OG team and the preliminary results show a high computational cost due to the number of buoys in the structure, causing up to  $8^n$  possible piston combination. For that reason, a new strategy or approach to an already tested method must be found. To compare the possible control strategy, the rubric showed in Table 3.1 was developed, considering the following criteria:

- **Previous implementations in wave energy:** answers the question: “*Is there any background about the use of that control strategy on wave energy in the literature?*” Besides the importance of develop new strategies and techniques, a review of the literature can simplify the design process to avoid or foreseen common issues and their solutions during the design process.
- **Previous work done by the OG team:** answers the question: “*Has the OG team already analysed the strategy to control the WEC?*” The OG team has already considered several control strategies for the models they have developed, thus, the previous information can be used as starting point or even benchmark.
- **Strategies requirements:** answers the question: “*What resources the strategy requires to be implemented?*” It’s related to any additional requirement (*e.g.* hardware, software or data) that the control strategy requires.

## 3.7 Considered Control Strategies

Given the criteria presented in Section 3.6, several control strategies are considered to compare different approaches for the control implementation, in order to chose the most appropriate for the solution development.

- **Proportional-integral-differential controller (PID):** a widely used strategy for industrial processes because its good performance and simplicity, its based on three terms: a proportional gain associated with the current value, an integral gain associated with past values and a derivative gain associated with future values, [51].
- **Model predictive control (MPC):** widely used control technique for multivariable and non-linear control problems, [37]; first, the algorithm uses a model to predict the outputs of the system in a determined set of future instants, called horizon. Then, it calculates the control sequence by minimizing an objective function, called cost function; and receding the sequence each sampling instant, displacing the horizon to the future, [37], [38] and [39].

Rubric	5	3	1
<b>Previous implementations in wave energy</b>	There are successful implementations of the control method in wave energy registered in the literature.	There are attempts to implement the control method in wave energy in the literature.	There is no information about the control method been implemented in wave energy in the literature.
<b>Previous work done by the OG team</b>	The OG team has done preliminary studies about the implementation of the control method in the WEC.	In the literature created by the OG team the control method is mentioned as a possible control strategy for the WEC.	The control method is not mentioned in the literature created by the OG team.
<b>Strategy requirements</b>	All the control method requirements ( <i>e.g.</i> hardware, software or data) already exists in the OG database.	Some of the control method requirements ( <i>e.g.</i> hardware, software or data) doesn't exist on the OG databases, but can be obtained without any additional expense.	At least one of the control method requirements ( <i>e.g.</i> hardware, software or data) are not in the OG databases an an additional expense is required.

Table 3.1: Criteria to compare the possible control strategies to chose the optimal proposal.

- **Artificial neural networks (ANN)**: is a computational paradigm that simulates the biological neural system, that allows to approximate any real function, [52]. Some applications for ANNs are mentioned by [53] and include: forecasting, classification and pattern recognition.
- **Evolutionary computing (EC)**: inspired in Darwin's theory of evolution, a "population" of individual possible solutions is created and combine between them to create new possible solutions. The process includes random changes in some individuals ("mutation"), until a satisfactory solution is achieved. Some of the applications for Evolutionary computation are: combinational optimization, fault diagnosis, and scheduling, [52]

### 3.7.1 Control Strategies Evaluation

Taking into consideration the criteria shown in Table 3.3, each considered language was scored in Table 3.2.

	PID	MPC	ANN	EC
Previous implementations in wave energy	3	5	3	0
Previous work done by the OG team	0	5	0	0
Strategy requirements	5	5	5	5
<b>Total</b>	8	15	8	5

Table 3.2: Evaluation of the considered control strategies.

According with the results in Table 3.2, MPC is the best option to develop the control strategy, since its background not only in wave energy in general, as shown in the investigation done by [43], [44] and [45], but also by the OG team itself, as shown in the work of [16] and [54]. With respect to the requirements, MPC only needs a model, in this case, the model presented by [18] is provided by the OG team; besides, the software requirements need only optimization tools, but the software selection will be detailed in the following Sections.

Despite its frequent use in the industry, the PID controller is not considered due the difficulties it presents when controlling non-linear models, like the WEC, as shown by [10], [15] and [18]. On the other side, recent research in wave energy is considering several artificial intelligence techniques, as shown in [55] and [56], however, the work in [22] done by the OG team gives a less optimistic perspective about its application on the WEC due its heuristic nature.

### 3.8 Programming Language Selection Criteria

To determine the optimal solution, the chosen proposal must fulfil the requirements presented before in the sections 3.3 and 3.4. Thus, the design must be done thinking in the computational cost of the solution. Also, the solutions should be developed according to the resources the OG team already has. Therefore, the models, simulations and experimental data obtained previously by the OG team will be used as benchmark. On the other hand, the solution must be developed in an open source language, therefore, several options can be considered. To compare the possible open source languages, the rubric shown in Table 3.3 was developed considering the following criteria:

- **Parallel computing:** answers the question: “*Does this language allows any parallel computing strategy (e.g. multithreading)?*” Since the MPC strategies require a model to predict the future behaviour of the model, do multiple calculations at same time would reduce the computation time.

- **MPC background:** answers the question: “*Is there any previous MPC implementation with the language documented already?*” In research it’s important to develop new strategies and techniques, however, a review of the state-of-the-art can simplify the design process. For that reason, previous work must be considered to analyse common issues and solutions that would eventually happen in the design process.
- **Existent documentation:** answers the question: “*How formal is the existent documentation about the current functions and toolboxes?*” At the moment to use new functions or toolboxes, proper documentations about them can facilitate the design and programming process.
- **Compatibility:** answers the question: “*How easily the language deal with external inputs and outputs?*” Even though the Ocean Grazer is still on the designing stage, the future implementation in the real system and in the current and future prototypes must be considered. Therefore, the language must be able to work with external inputs and outputs (i/o) to control the real system.

Rubric	5	3	1
<b>Parallel computing</b>	It’s possible to implement any strategy of parallel computing, <i>e.g.</i> multi-threading.	-	It’s not possible to implement any strategy of parallel computing.
<b>MPC background</b>	Non-linear implementations using the language were found.	Only linear MPC implementations using the language were found.	No previous MPC implementations using the language were found.
<b>Existent documentation</b>	Information can be found from users and the official website.	There is only documentation from users or the official website about the functions and packages, but not both.	There is no documentation about the language functions and packages.
<b>Compatibility</b>	It’s possible to manage external i/o directly.	It’s possible to manage external information but with an intermediate file, <i>e.g.</i> .csv files.	It’s not possible to manage external information.

Table 3.3: Criteria to compare the possible open source languages to chose the optimal proposal.

## 3.9 Considered Languages

As a requirement, the solution must be developed in an open source method, however, the current models are in Matlab files. One solution is to replicate the current Matlab code on the selected open source language. Another solution is that the open source language used for the MPC toolbox is also able to run and obtain data from a Matlab code, so it won't be necessary to modify or replicate the model, however, the final implementation should be developed in open source.

Besides the fact that lower-level languages tend to be more efficient in terms of computational cost, programming with them is more difficult and, thereby, time consuming. For that reason, only high-level languages are considered to implement the solution. Also, since it's an engineering solution, it needs to be easy to understand and modify to make eventual improvements or corrections in the code, so a higher-level language would facilitate those tasks.

The languages considered are:

**Python:** is a well known open source programming language with a big variety of research and industrial applications, with both official and third party modules.

**Octave:** with a similar environment as Matlab, is the alternative GNU offers to numerical computations, including control algorithms.

**Julia:** under an MIT license, is a relatively new language for numerical computing with a more than 1700 registered packages, including non-linear control optimization tools.

### 3.9.1 Considered Languages Evaluation

Taking into consideration the criteria shown in Table 3.3, each considered language was scored in Table 3.4.

	Python	Octave	Julia
Parallel computing	5	5	5
MPC background	5	5	5
Existent documentation	5	3	3
Compatibility	5	3	3
<b>Total</b>	20	16	16

Table 3.4: Evaluation of the considered open source languages.

As can be seen, Python is the optimal choice to implement the solution. Besides the fact it can communicate with simultaneous Matlab codes, it's the best documented option, having existent optimization packages and, also, is the most compatible with external inputs and outputs, what is necessary at the moment of the implementation on the real Ocean Grazer. Octave has the inconvenience that can't run all Matlab codes, compromising the model. Moreover, even though it has an MPC package, the development of the tool is discontinued, [57]. Finally, Julia presents a promising possibility, however, the lack of experience on the language, and the limited background, make it less attractive than Python to develop the solution.

## 3.10 Possible Solutions

To develop the solution, two important parts must be considered: the solution method itself, and the language that will be used to implement the solution. As was mentioned in the requirements in Section 3.3, an open source must be used to implement the solution, mainly for two reasons: in first place, the license of common softwares is around several thousand dollars per year, money that can be saved, or invested in the equipment or maintenance for example, if an open source is used, and second, it's easier to modify the functions or create them in an open source code, making it more flexible. To have a group of several solution methods, a brainstorm was made and each considered solution method is described in Subsections 3.10.1 and 3.10.2. Those proposed solutions aims to stablish the way the MPC strategy will be implemented on Python.

### 3.10.1 A Script with All the Functions Needed

This proposal consist in the creation of a script which contains all the necessary functions to implement the control strategy.

#### Advantages

- It's only one file, therefore, all the information is together.

#### Disadvantages

- Due the control strategy implementation can turn into complex calculation of several parameters plus the documentation, the script would be too long.
- Debugging can become difficult, since all the functions are together, thus, individual tests can't be done easily.
- Each time the model or the parameters change, the script must be modified.

### 3.10.2 A Set of Functions as a Toolbox

This proposal consist in the creation of a set of the functions needed to implement the control strategy, saved on individual files or grouped in individual files.

#### Advantages

- Since each function is analysed individually, each file would have less lines, even with the function documentation.
- If each function is written on an individual file, or grouped properly, debugging would be easier
- A general script can be made to call the functions and the model.
- Can be easily loaded in any script.
- If it's make general enough, can be used not only for the WEC, but for other optimization processes.

#### Disadvantages

- Due the control strategy implementation can turn into several complex calculation, if each function is saved in an individual file, it may result in numerous individual files.

## 3.11 Chosen Solution

About the possible solutions, taking into consideration the advantages and disadvantages of each proposal, the toolbox represent the best choice, because, even though it would imply to develop individual functions, they can be arranged in groups according to their tasks and all the functions can be imported at once when importing the toolbox when running the optimization algorithm. Likewise, only one script would be a too large document, and if improvements are needed, it would require more time to modify the code that if a toolbox is developed, because variables are independent between functions.

## 3.12 Viability Criteria

When the proposed solution is developed, it's necessary to verify if it is viable to be implemented, therefore, a series of parameters must be considered. The relative priority of the criteria are shown in a Pareto chart in Figure 3.2. As can be seen, the most important point to consider the solution as valid is the computational cost, since the objective is to reduce the computation time needed to solve the model to implement a control strategy suitable for the real process and extract as much energy as possible. In second place, the solution must be compatible with the current prototypes to be able process experimental data. Finally, the solution must be accurate, however, since it's response depends directly on the model, and it

can be improved afterwards if the solution is promising, its relative importance if compared with the computational cost or the compatibility is less.

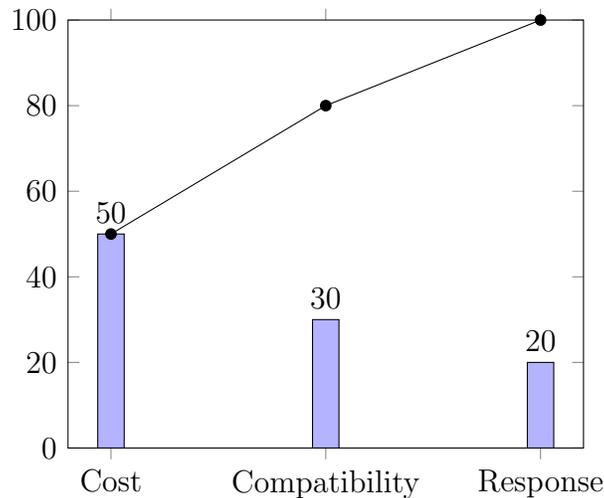


Figure 3.2: Pareto chart summarizing the criteria for the solution validation.

Several simulations will be made with regular and irregular waves to analyse the behaviour of the developed control strategy. The developed solution must fulfil the following conditions to define it as viable:

- The developed solution must have a shorter computing time than the current time domain model, while using the same hardware (RAM and processor), ideally the same computer.
- The developed solutions must not need additional data than the provided from the OG prototype, *i.e.* the same data that the current model uses or less.
- The extracted energy due the resulting combination of pistons given by the developed solution must be at least the 90 % of the theoretical maximum energy, in average.

### 3.13 Concluding Remarks

In this Chapter, the causes that conclude in the high computational cost of the implementation of a control strategy using the current models are given, including the computational resources and the complexity of the real system as a whole and the current WEC models. Then, the problem was synthesised, the user requirements announced and the system requirements determined. Given the above, is necessary to determine a control strategy for the WEC, based on the model proposed by [18] and the selection of an open source programming language.

Then, the methodology used for the solution design was detailed. Several control strategies and programming languages were evaluated using specific rubrics to compare them objectively. The results shown that the MPC is the best control strategy since its viability has already analysed, being the computation cost the main obstacle on its implementation. However, since its computational requirements are according to the model used, a different model can be used to decrease the computation time. On the other hand, due the requirement of use an open source language, Python resulted in the selected candidate since its high level, the amount of existing documentation and previous successful MPC implementations,

# Chapter 4

## Model Description and Control Strategy

The present Chapter aims to introduce the port-Hamiltonian model that is going to be used to implement the MPC strategy. First, in Section 4.1, the base Matlab model presented by [18] will be described, explaining each of its parameters. However, that model doesn't consider the hydrodynamics of the piston, plus, the pH modelling doesn't allow a simple control implementation. For that reason, Section 4.2 describes a series of modifications to the model that will allow the implementation of a control strategy.

Section 4.3 is dedicated to the control approach that will be used, describing the developed control algorithm. The control variable is determined and its influence on the system is demonstrated. Also, the cost function is defined as function of the control variable. Finally, Section 4.4 is dedicated to the Python implementation algorithm of the control in the model described in Section 4.2.

### 4.1 Previous port-Hamiltonian Model Analysis

As mentioned in Section 2.4, the general port-Hamiltonian representation of a system is presented as follows

$$\Sigma = \begin{cases} \dot{\mathbf{x}} = (\mathbf{J} - \mathbf{R}) \frac{\partial \mathbf{H}(x)}{\partial \mathbf{x}} + \mathbf{G} \mathbf{u}(x) \\ \mathbf{y} = \mathbf{G}^T \frac{\partial \mathbf{H}(x)}{\partial \mathbf{x}} \end{cases} . \quad (4.1)$$

In this case, the pH model of the WEC presented by [18] is summarized as a multiple mass-spring-damper system, as shown in Figure 4.1, where its force analysis is

$$m\ddot{q} + f_b + f_{pto} = f_{ex} + f_r , \quad (4.2)$$

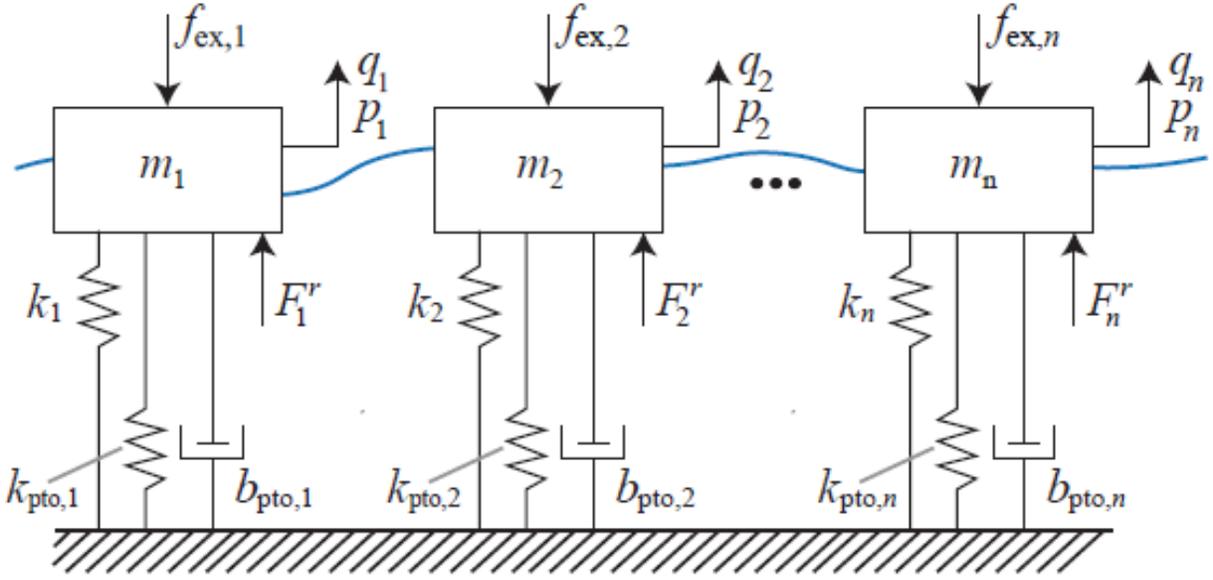


Figure 4.1: Diagram of the multi-floater system used in [18].

following the Newton's Second Law, including the power take-off forces due to the stiffness and damping of the system,  $f_{pto}(t)$ , as described in Subsection 2.4.1, the excitation forces produced by the waves  $f_{ex}(t)$ , the buoyancy force over the buoys

$$f_b(t) = \rho g A_b q(t) , \quad (4.3)$$

where  $q$  is the displacement of the buoy's center of mass, and the radiation forces produced by the movement of the other buoys, that according with [58] can be calculated as

$$f_r(t) = -m_\infty \ddot{q}(t) - \int_0^t \varphi(t - \tau) \dot{q}(\tau) d\tau , \quad (4.4)$$

where  $\ddot{q}$  is the acceleration of the buoy,  $\dot{q}$  the velocity of buoy and  $\varphi$  is a convolution kernel, which depends on the geometry of the buoys, usually calculated using hydrodynamic numerical tools, [59]. In the base model used by [18], the tool NEMOH is used to approximate with a radiation IRF with an order of 9.

In the Subsection 2.4.1, a simple mass-spring-damper system is modelled in a pH representation. In that case, the states corresponds to the position and the momentum of the

mass. However, in this system, there are several masses (the buoys), which movement in the water creates radiation forces that induce movement from one buoy to the others and over themselves. For that reason the buoys interfere indirectly in the movement of the others, and its effect can be seeing in the  $(\mathbf{J}-\mathbf{R})$  matrix.

Given the above, the  $(\mathbf{J}-\mathbf{R})$  matrix of the WEC pH model is composed by four individual blocks that can be individually analysed, as shown next,

$$(\mathbf{J}-\mathbf{R}) = \begin{bmatrix} (\mathbf{J}-\mathbf{R})_{q,p} & (\mathbf{J}-\mathbf{R})_O \\ -(\mathbf{J}-\mathbf{R})_O^T & (\mathbf{J}-\mathbf{R})_r \end{bmatrix}, \quad (4.5)$$

were the first diagonal block  $(\mathbf{J}-\mathbf{R})_{q,p}$  of  $2n \times 2n$  elements correspond to the classical mechanical description demonstrated in Section 2.4.1, been  $n$  the number of buoys. The second diagonal  $2no \times 2no$  block  $(\mathbf{J}-\mathbf{R})_r$  corresponds to the effect of the radiations forces between the buoys, calculated using a hydrodynamic tool, were  $o$  corresponds to the order of radiation approximation. The first  $2n \times 2o$  off-diagonal block  $(\mathbf{J}-\mathbf{R})_O$  is mainly composed by zeros, however, every  $(io, n+1+i)$  is a non-zero element, in this case 8, were  $i$  is an index from 0 to  $n-1$ .

The Hamiltonian represents the energy of the system in any moment. In this system is given as

$$H(x) = \sum_{i=1}^n \frac{k_i q_i^2}{2} + \frac{p_i^2}{2m_i} + \frac{z_i r_i^2}{2}, \quad (4.6)$$

where  $k_i$  is the stiffness,  $m_i$  is the mass,  $q_i$  the displacement and  $p_i$  is the momentum associated to each buoy sub-system  $i$ ;  $z_i$  and  $r_i$  are components associated with the radiation produced by the buoys movement. The derivatives of (4.6) with respect to the position, momentum and radiation components are presented, respectively, as

$$\frac{\partial H(x)}{\partial q_i} = k_i q_i, \quad (4.7)$$

$$\frac{\partial H(x)}{\partial p_i} = m_i^{-1} p_i, \quad (4.8)$$

and

$$\frac{\partial H(x)}{\partial r_i} = z_i r_i. \quad (4.9)$$

In the port-Hamiltonian WEC model presented by [18], the spring constant can be calculated analytically as

$$k = A_b \rho_s g, \quad (4.10)$$

where  $A_b$  is the basal area of the buoy,  $\rho$  is the density of the sea water and  $g$  is the gravitational acceleration. The damping coefficient, however, is a constant defined to guarantee the system response corresponds to a reference value. On the other hand, the mass corresponds to the sum of the mass of the buoy, the added mass induced by the radiation components in the infinite frequency, defined as

$$\mathbf{M}_\infty = \begin{bmatrix} m_{\infty 1,1} & m_{\infty 2,1} \\ m_{\infty 1,2} & m_{\infty 2,2} \end{bmatrix}, \quad (4.11)$$

where the masses  $m_{\infty i,j}$  are the added mass produced by the buoy  $i$  over the buoy  $j$  in the infinite frequency. Finally, the mass of the system is given as follows:

$$\mathbf{M} = \mathbf{M}_b + \mathbf{M}_\infty. \quad (4.12)$$

To simulate the model previously exposed, a series of Matlab codes were done, which depend on a series of parameters defined by the user, including: the wave height and period, the mass of the buoy, the stiffness of the system, the damping coefficient of the system, the simulation time and the initial conditions, dividing the process in four basic steps. Once the parameters were defined, the excitation forces are calculated. It is important to mention that the current Matlab code can only calculate the excitation data for regular waves.

After the calculation of the excitation forces, the **(J-R)** matrix is calculated, based on the parameters and the calculation of the radiation components. The **G** matrix, on the other side, is obtained directly from the model. Once those matrices are obtained, using the initial conditions, the next state  $\mathbf{x}(k+1)$  is calculated, using a ODE45 solver and saved into an array. If the calculated step is not the final desired step, the **(J-R)** matrix is updated and the next step is calculated in the same way. The previous process is repeated until the desired number of steps is calculated. A summary of the algorithm behind the Matlab code is shown in the flowchart presented in Figure 4.2.

As mentioned in [18], for the simulation were used two identical cuboid floaters with a square base of  $49 \text{ m}^2$  and 2 m of height, and a radiation impulse response function (IRF) approximation of order  $\sigma=9$ . The the parameters and constants used are shown in Table 4.1.

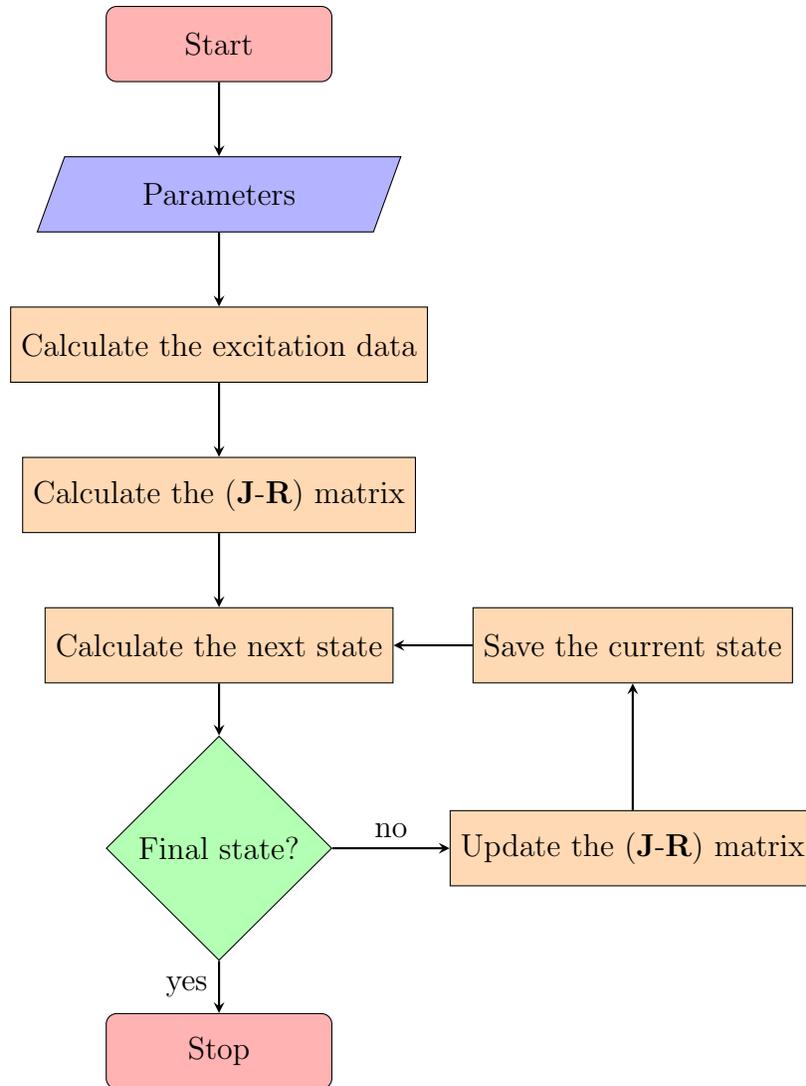


Figure 4.2: Flowchart of the Matlab code of the pH model of the WEC in open loop.

In this case, for both buoys, all the parameters used for the pH modelling are the same, but they can be different.

Given those parameters, a simulation of 400 seconds was done with the parameters shown in Table 4.1, during around 1 minutes and 50 seconds on a computer with an Intel Core i7 Processor at 2,2 GHz and 16 GB of RAM. A representative sample of the simulation is shown in Figure 4.3. When comparing the results with the WEC-Sim tool as reference, a maximum error of 3,67 mm in the first buoy and 3,24 mm in the second was obtained.

Parameter	Symbol	Value
Added mass produced by the radiation over the buoy itself	$m_{\infty,i,i}$	105 447,92 kg
Added mass produced by the radiation between the buoys	$m_{\infty,i,j}$	11 326,83 kg
Density of the sea water	$\rho_s$	1025 kg/m <sup>3</sup>
Gravitational acceleration	$g$	9,81 m/s <sup>2</sup>
Mass of the buoy	$m_b$	45 000 kg
Stiffness	$k$	$4,9271 \times 10^5$ N/m
Piston height	$h_p$	0,1 m
Piston radius	$R_p$	0,1 m
Piston-cylinder separation	$S_p$	400 $\mu$ m
PTO Stiffness	$k_{pto}$	0 N/m
PTO damping coefficient	$b_{pto}$	$11,53 \times 10^6$ kg/s
Viscosity of the working fluid	$\mu$	0,0734 Pa-s
Wave height	$H$	4 m
Wave period	$T$	5 s

Table 4.1: Parameters and constants used in the Matlab simulation, [18].

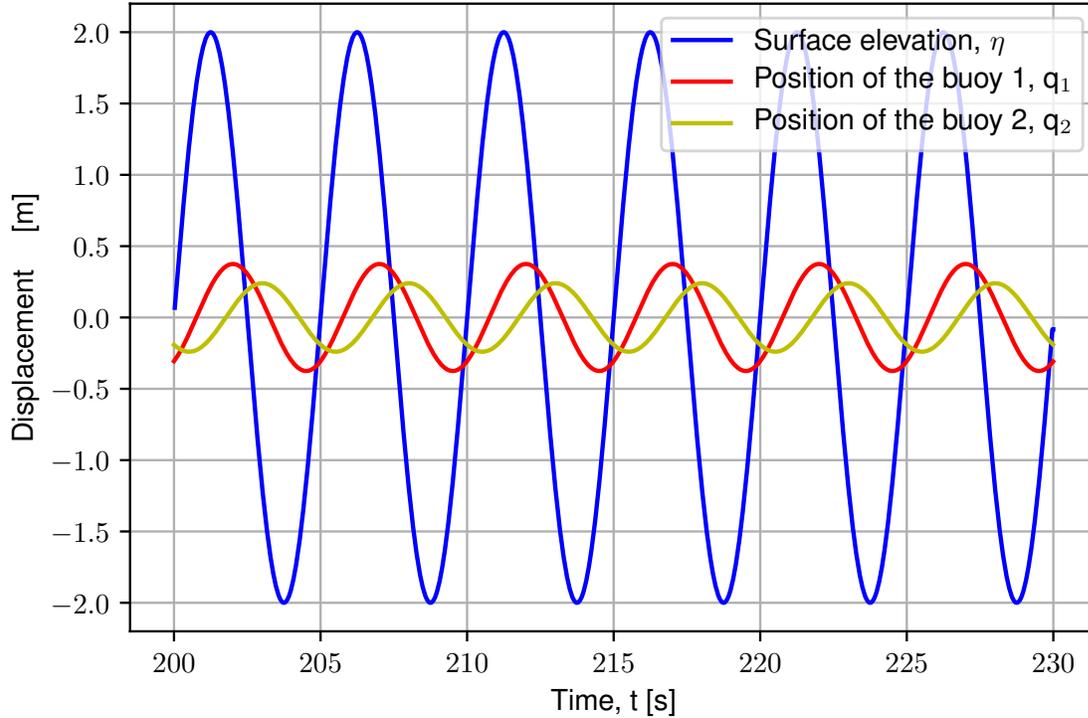


Figure 4.3: Response of the pH model with the Matlab code using a regular wave with 4 m of height and a period of 5 s, and the parameters shown in Table 4.1.

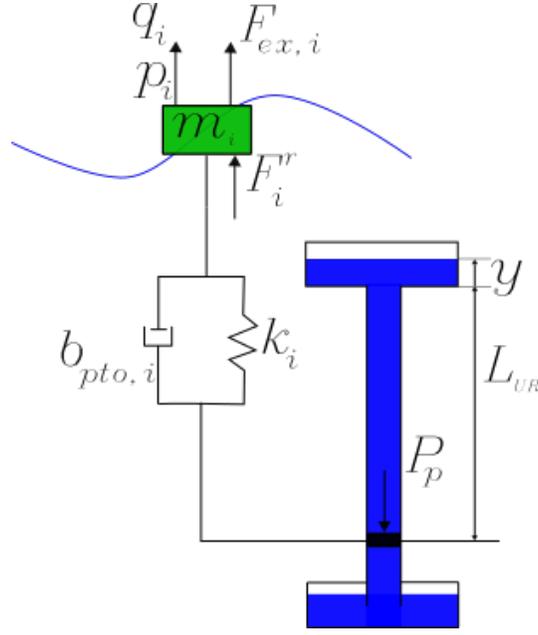


Figure 4.4: Diagram of the proposed model, considering the hydrodynamics of the piston.

## 4.2 Proposed Model

The model presented in [18] doesn't consider the effects of the hydrodynamics of the piston over the system. For that reason, the proposed model no longer considers the mass-spring-damper system attached to the seabed but to the piston, as shown in Figure 4.4.

In this case, however, the damping coefficient won't have the same value it has in the model presented by [18], instead, the equation and values used in [6] and [10] are used. The damping coefficient is now associated with the friction between the piston and the cylinder that can be calculated as

$$b = \frac{2\pi R_p H_p \mu}{S_p}, \quad (4.13)$$

where  $R_p$  is the radius of the piston,  $H_p$  is the height of the piston,  $\mu$  is the viscosity of the working fluid and  $S_p$  is the separation between the piston and the cylinder.

Furthermore, the effect of the pressure among the pistons must be considered. It can be calculated as

$$P_p = P_{UR} + \rho g(L_c + y - q) + \frac{\rho \dot{q}^2}{2}, \quad (4.14)$$

where  $P_{UR}$  is the pressure in the upper reservoir,  $\rho$  is the density of the working fluid,  $L_c$  is the distance between the piston and the upper reservoir base during a calm sea,  $y$  is the level of the water in the upper reservoir,  $q$  is the position of the piston and  $\dot{q}$  is the velocity of the piston.

Likewise, the water in the upper reservoir stores energy, adding a extra term in the Hamilton calculation done by [18] and expressed as (4.6), turning into

$$H(x) = \sum_{i=1}^n \frac{k_i q_i^2}{2} + \frac{p_i^2}{2m_i} + \frac{z_i r_i^2}{2} + \frac{C_{UR} P_{p,i}^2}{2}, \quad (4.15)$$

where  $C_{UR}$  is the capacitance of the upper reservoir, *i.e.*, the resistance that a liquid in a reservoir offers to change its pressure due an inlet flow, [6], and can be calculated as

$$C_{UR} = \frac{A_{UR}}{\rho g}, \quad (4.16)$$

where  $A_{UR}$  is the area of the upper reservoir and  $\rho$  is the density of the working fluid. Then, the derivative of the Hamiltonian with respect to the pressure is

$$\frac{\partial H(x)}{\partial P_i} = C_i P_i. \quad (4.17)$$

Given the above, the capacitance's effect of the upper reservoir over the piston must be taking into account. As can be implied, the pressure among the piston will produce a force against the movement of the buoy during the up-stroke, but in favour of its movement during the down-stroke, thus, the force of the column of water over the piston is expressed as:

$$f_w = P_p A_p, \quad (4.18)$$

where  $P_p$  is the pressure among the piston and  $A_p$ . Therefore, (4.2) turns into:

$$m\ddot{q} + f_b + f_{pto} = f_{ex} + f_r + f_w. \quad (4.19)$$

Then, a new block, related to the effect of the pressure in the system appears in the  $(\mathbf{J}-\mathbf{R})$  matrix in (4.5), turning it into

$$(\mathbf{J}-\mathbf{R}) = \begin{bmatrix} (\mathbf{J}-\mathbf{R})_{q,p} & (\mathbf{J}-\mathbf{R})_O & (\mathbf{J}-\mathbf{R})_P \\ -(\mathbf{J}-\mathbf{R})_O^T & (\mathbf{J}-\mathbf{R})_r & \mathbf{0}_{2no \times 1} \\ -(\mathbf{J}-\mathbf{R})_P^T & \mathbf{0}_{1 \times 2no} & \mathbf{0}_{1 \times 1} \end{bmatrix}, \quad (4.20)$$

where

$$(\mathbf{J}-\mathbf{R})_P = \begin{bmatrix} \mathbf{0}_{n \times 1} \\ -\left( \frac{\mathbf{A}}{\mathbf{C}_{UR}} \right)_{n \times 1} \end{bmatrix}. \quad (4.21)$$

In this case, the capacitance of the fluid appears in the denominator of the non-zero elements because the force is independent of the capacitance, however, the derivative of the Hamiltonian with respect to the pressure depends on it, thus, the factor must be cancelled.

Finally, the height of the water in the upper reservoir will increase only during the up-strokes and will remain during the down-stroke. The change in the level of the working fluid can be calculated as follows:

$$\Delta y_{UR} = \begin{cases} \sum_{i=1}^n \frac{A_p \Delta q_i}{A_{UR}} & \forall \dot{q} > 0 \\ 0 & \forall \dot{q} \leq 0 \end{cases}, \quad (4.22)$$

where  $n$  is the number of buoys,  $A_p$  is the current area of the piston,  $A_{UR}$  is the area of the upper reservoir and  $\Delta q$  is the change in the height of the piston.

### 4.2.1 Assumptions and Simplifications

To keep the proposed model as simple as possible, a series of assumptions related to the model previously described are detailed next.

- The working fluid is laminar and non-compressive.
- The movement of the buoy is the same of the piston, i.e. the elasticity of the connection is not considered.
- The movement of the buoy is limited to only 1 degree of freedom (1 DoF).

- Leaks of working fluid from the upper reservoir to the lower reservoir between the piston and the pipe are not considered.
- The check valves that avoid the working fluid to return from the upper reservoir to the lower reservoir during the down-stroke are assumed ideal.
- The mass associated to each buoy will be assumed as constant and equal to the sum of the masses of each of the piston of the SPP plus the mass of the floater, using the values considered by [10].

$$m_b = m_f + m_p \quad (4.23)$$

- The height of the buoy is big enough to avoid a full submersion.

$$\frac{H_b}{2} \geq q_i \quad \forall \quad i = 1, 2, \dots, n \quad (4.24)$$

- The pressure among both pistons will be the same for the calculations, since the change of their position is small in comparison of the column of water over the piston, *i.e.*

$$q_i \ll L_{UR} \implies P_1 \approx P_2 \approx \dots \approx P_n \quad \forall \quad i = 1, 2, \dots, n \quad (4.25)$$

- During the down-stroke, the area of the piston used to calculate the pressure among the pistons will be considered as 0, since the valves are open.
- The hydrodynamic pressure in the piston is not considered since is too small in comparison with the hydrostatic pressure, *i.e.*

$$\frac{\rho \dot{q}_i^2}{2} \ll \rho g (L_{UR} + y) \implies P_p \approx P_{UR} + \rho g (L + y) \quad \forall \quad i = 1, 2, \dots, n \quad (4.26)$$

- The compression of the air in the upper reservoir is not considered, *i.e.* the pressure in the upper reservoir will remain constant and will be the atmospheric pressure.

$$P_{UR} = P_{atm} \quad (4.27)$$

### 4.3 Control Strategy

As mentioned in Subsection 2.6.1, the WEC operation principle can be summarized as the transformation of potential energy from the working fluid reservoir to electrical energy through a turbine. Taking that into consideration, the maximum theoretical amount of energy that can be extracted in an up-stroke is the potential energy of the working fluid itself gained when moved from the lower reservoir to the higher reservoir due the work of the piston, as shown

$$E_{max} = mgh_p, \quad (4.28)$$

therefore, as can be seen, the control strategy can be focused on maximizing the increment of the potential energy stored in the higher reservoir when pumping the fluid from the lower.

The amount of working fluid that is pumped from one reservoir to the other can be modelled as a fluid column, that is present only during the up-stroke. It is important to remark that the volume of the fluid column moved is proportional to the area of the piston and also to the square of the buoy's movement amplitude.

Besides, the energy absorbed by the WEC is directly proportional to the mass of the working fluid that is been moved, thus, to the area of the piston, for that reason and due to its physical meaning on the WEC, the area of the piston is chosen as control variable, since it consists on a Multiple-Piston Pump (MPP). However, in the present project a Single-Piston Pump (SPP) approach is used, modelling the multiple pistons as one with variable area.

Given the above, for control purposes, the general pH representation presented in (4.1) is modified to obtain a more explicit matrix, as function of the area of the piston. As can be deduced from (4.7), (4.8), (4.9) and (4.17), the derivative of the Hamiltonian can be expressed as the multiplication of an auxiliary matrix  $\mathbf{N}$  and the current state, resulting as follows:

$$\frac{\partial \mathbf{H}(x)}{\partial \mathbf{x}} = \begin{bmatrix} \mathbf{K} & 0 & 0 & 0 \\ 0 & \mathbf{M}^{-1} & 0 & 0 \\ 0 & 0 & \mathbf{Z} & 0 \\ 0 & 0 & 0 & \mathbf{C}_{UR} \end{bmatrix} \mathbf{x} = \mathbf{N} \mathbf{x}, \quad (4.29)$$

thus, (4.1) takes the form

$$\begin{cases} \dot{\mathbf{x}} = [\mathbf{J}(x) - \mathbf{R}(x)] \mathbf{N} \mathbf{x} + \mathbf{G} \mathbf{u}(x) \\ \mathbf{y} = \mathbf{G}^T \mathbf{N} \mathbf{x} \end{cases}. \quad (4.30)$$

Moreover, (4.30) has a similar form than the standard State-Space representation. So, to simplify it, the matrices  $\mathbf{A}_{pH}$ ,  $\mathbf{B}_{pH}$  and  $\mathbf{C}_{pH}$  will be defined, respectively, as shown

$$\mathbf{A}_{pH}(x) = [\mathbf{J}(x) - \mathbf{R}(x)] \mathbf{N}, \quad (4.31)$$

$$\mathbf{B}_{pH} = \mathbf{G}(x) \quad (4.32)$$

and

$$\mathbf{C}_{pH} = \mathbf{G}^T \mathbf{N}, \quad (4.33)$$

then, (4.30) becomes into

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}_{pH}(x) \mathbf{x} + \mathbf{B}_{pH} u(x) \\ \mathbf{y} = \mathbf{C}_{pH} \mathbf{x} \end{cases}. \quad (4.34)$$

On the other hand, the objective is to maximize the extracted energy, which is directly proportional to the volume of the fluid column (area of the piston times the height of the column) and the height of column itself. Therefore, the extracted energy is proportional to the area of the piston and to the height of the column squared. For that reason, the cost function will be defined as function of the potential energy. Due to the optimizing tools of the ScyPy library, the cost function must be minimized, and since the energy is pretended to be maximized, the cost function will be defined as the opposite of the energy, as shown next

$$J(x, A_p) = - \sum_{i=1}^n \sum_{K=1}^{H_p} [\rho g A_{p,K}(x) h_{p,K}^2(x)]_i, \quad (4.35)$$

where  $x$  is the state,  $A_p$  is the area of the piston,  $n$  the number of buoys,  $K$  is the instant of the peak in the displacement of each buoy,  $H_p$  is the prediction horizon and  $h_p$  is the height of the buoy during the up-stroke.

Since the control variable is the area of the piston, the parameter in the model that will be adjusted is the matrix  $\mathbf{A}_{pH}$ , because it's the only dependant on the area of the piston. Also, some constraints must be considered in the optimization problem. In this case, the area of the piston corresponds to a set of discrete values presented in [13], therefore, the damping coefficients are also a discrete set of values if (4.13), is rewritten as function of the area, resulting in

$$b = \frac{2\mu H_p \sqrt{\pi A_p}}{S_p} \quad (4.36)$$

The resulting set of values when evaluating each area is presented in Table 4.2. Since the case with all the pistons disabled won't move water, is discarded as possible minimum, *i.e.* the combination  $\{0,0\}$  is not considered as solution. It is important to mention that the optimization process will generate a continuous set of values between the constraints, so, the

discrete value that will be assumed is the closest to the resulting value.

Piston configuration	Area of the piston [m <sup>2</sup> ]	Damping coefficient [kg/s]
{0,0,0}	0	0
{0,0,1}	0,06	15,93
{0,1,0}	0,18	27,6
{0,1,1}	0,24	31,9
{1,0,0}	0,30	35,6
{1,0,1}	0,36	39,0
{1,1,0}	0,48	45,1
{1,1,1}	0,54	47,8

Table 4.2: Set of equivalent areas, [13], and damping coefficients for the possible piston configurations.

## 4.4 Control Algorithm Design

In Section 2.5 was mentioned that the MPC strategies are based on the calculation of the system response during a future time window called prediction horizon. For that reason, the first step in the control algorithm must be the calculation of an appropriate horizon, represented as a time window. In this case, the sampling time for the steps in the control strategy are not going to be constant, instead, the steps are going to be measured from one peak to the next in the measure data, as shown in Figure 4.5.

For the control strategy it's assumed that during the up-stroke and the down-stroke, the wave can be analysed as individual sections of regular waves, *i.e.* the time window of irregular waves can be seen as sequence of segments of regular waves with different period, height or both, method used also by [15]. Thus, the period of each segment of wave is going to be the double of the time between the valley and the peak, or vice versa; analogously, the height of the segment wave will be measured from valley to the peak.

Likewise, the possibility of small waves between bigger ones must be considered, because they may cause unnecessary changes in the piston configuration during small periods, introducing noise or instabilities in the system, without adding a significant amount of energy. For that reason, the theoretical energy per unit length (in kJ/m) of the up-stroke segment of wave will be used as filter. The energy per unit length can be obtained when the Equation (2.5) is integrated with respect to time during the up-stroke, resulting in

$$E_w = \frac{\rho g^2 H_s^2 T_M^2}{512\pi}. \quad (4.37)$$

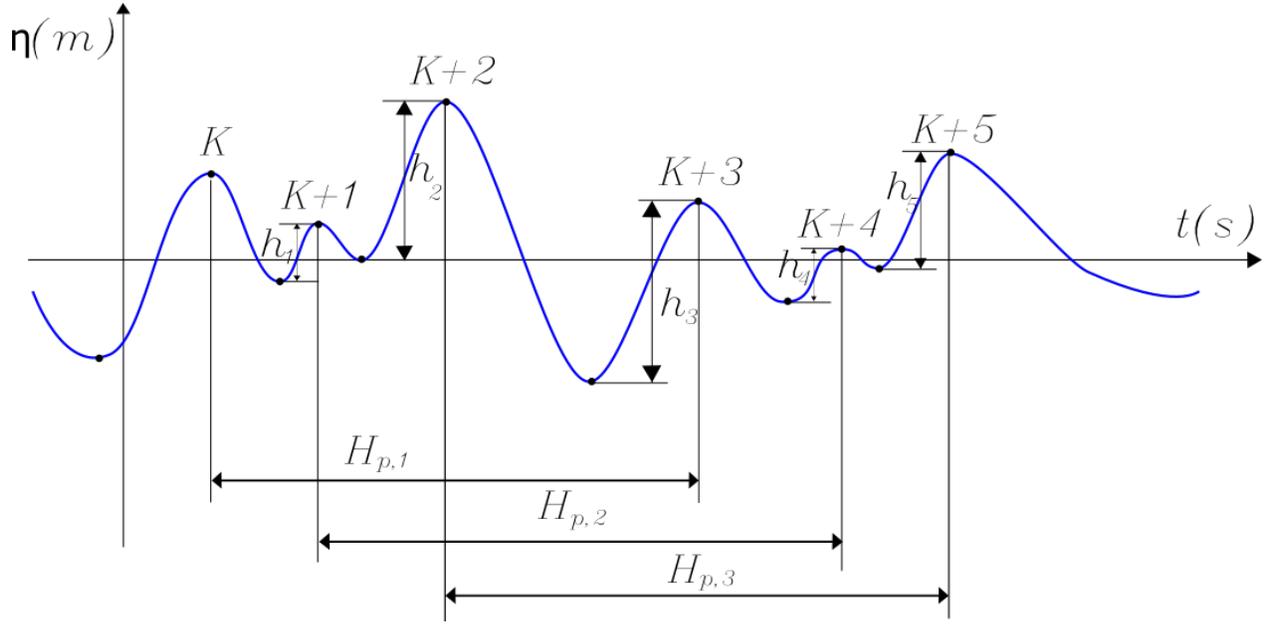


Figure 4.5: Representation of the prediction horizon time window with irregular waves, where  $h_n$  is the height of each one of the up-strokes,  $H_n$  is the prediction horizon and  $K$  is the peak instant.

Thereby, the input will be analysed until find enough local maximums according to the prediction horizon. For each up-stroke, the energy will be calculated and using a filtering factor  $c$ , the maximums associated with low height up-strokes will be discarded and the time window will be updated until find enough local maximums to fulfil the prediction horizon, as shown in Figure 4.6.

The filtering factor  $c$  is defined as a percentage of the most energetic up-stroke, thus, if the energy of an up-stroke in the time window is smaller than the energy associated with the most energetic up-stroke multiplied by the filtering factor, or a minimum defined value, the corresponding maximum will be discarded and a new maximum will be added to the time window, creating a smoother transition from one configuration to the next.

For the purposes of this project, the filtering factor will be assumed as a designer criteria, which must be different for each buoy. However, in the implementation, the filtering factor can be defined as a function of the total cost the change in the piston configuration would imply, resulting in a more objective selection criteria. Alike, during the simulation, the data for the time windows will be read from a file with the excitation data; nevertheless, in the real system, the data will be obtained trough real time measurements.

With respect to the optimization, as mentioned previously, the cost function is defined as the opposite of the potential energy the column of working fluid will add to the system when moved by the piston to the upper reservoir. That means that the cost function depends only on the maximum height of the buoy during the up-stroke, and because of that, for the optimization, only that value of height is needed to determine the best piston configuration,

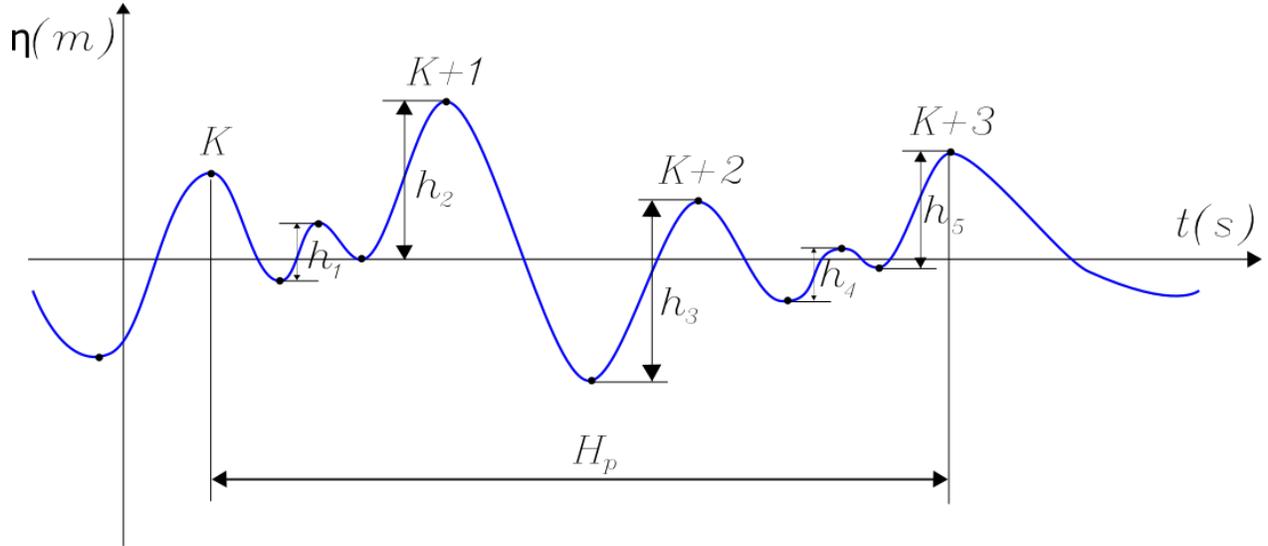


Figure 4.6: Representation of the prediction horizon time window, filtering the low energy peaks in irregular waves, where  $h_n$  is the height of each one of the up-strokes,  $H_n$  is the prediction horizon and  $K$  is the sampling instant.

calculated from valley to peak on the instant  $K$ . On the other hand, for the model, a smaller sampling time  $k$  is needed, therefore, the sampling time for the model and for the optimization must be different, where  $K > k$ .

Once the best piston combination for each buoy is found through the optimization for each step of the prediction horizon, only the values for the first up-stroke will be used, and, if necessary, the change in the configuration will be done just after the transition from up-stroke to down-stroke, *i.e.* just after the velocity of the piston change from positive to negative values, and updating the  $A_{pH}$  matrix.

It is also possible to use the combination found for more than one step, reducing the computational cost since the algorithm is executed less frequently, nevertheless, due high variability in irregular waves, it could reduce the accuracy. Finally, after adjusting the configuration of the pistons (if necessary), the algorithm will return to the update of the time window and repeat the whole process, as shown in the flowchart in Figure 4.7.

## 4.5 Concluding Remarks

In this Chapter, the base port-Hamiltonian model presented by [18] was described, including the MATLAB algorithm used to simulate the model and the resulting data. Then, the modifications of the model were proposed, including the hydrodynamics of the piston and adding the pressure among the pistons as a state variable. Finally, the assumptions and simplifications for the simulation of the model are established, allowing the develop of a more straightforward control strategy.

Furthermore, the last step before the control strategy validation was described, indicating the control variable and its effect on the model. In this case the area of the piston is chosen as control variable, thus, the matrix  $\mathbf{A}_{pH}$  will be the factor that will change in the proposed model for control purposes. Then, the cost function was defined as the opposite of the potential energy that the column of water add to the system storage when is moved from the lower to the upper reservoir; the opposite is used due the Python's available optimization tools. Finally, the control algorithm is presented as a flowchart, showing its individual sections and interconnections.

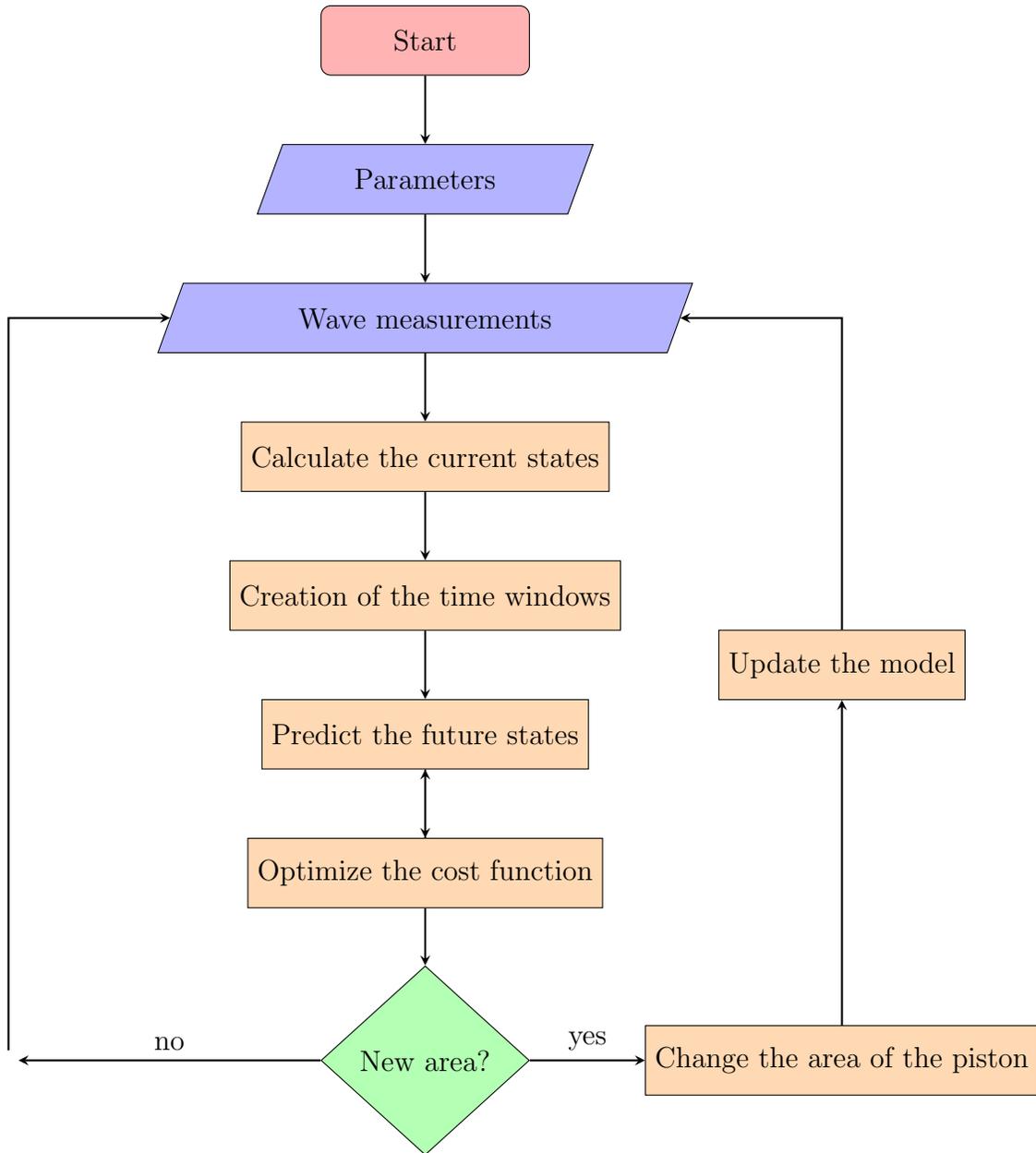


Figure 4.7: Flowchart of the control algorithm.

# Chapter 5

## Results Validation

The present Chapter presents the obtained results and describes the process used to validate the simulation results of the proposed model in open loop and with the control implementation. Section 5.1 describes the algorithm used in the Python code of the proposed model, including the modifications with respect to the model presented by [18] to improve the performance, using the results presented in [10] as reference. Then, the control strategy is implemented, using the optimal founded settings for the Euler method. The Chapter ends with a discussion section, where all the results of the present research are described and analysed.

### 5.1 Design of the Python Algorithm

One of the requirements for the present project asked from the OG team is the use of an open source language to develop the solution, and as shown in Section 3.11, Python resulted as the best candidate. Even though it was considered the idea of executing the current Matlab model through a Python code, due to the complexity of the way the model was programmed (a code separated in about 30 different .m files) and documented, use it for a control strategy turns more complicated than recreate the code.

Given the above, a Python version of the MATLAB model was developed, however, since it will be implemented on a MPC strategy, the new Python is a simplified but still reliable version of the MATLAB code. The purpose of simplifying the model is to reduce the computational cost of the model itself. Therefore, the optimization of the cost function in the MPC will be less computationally demanding.

Several tests were done with the Matlab code, changing the wave parameters (height and period), to analyse the behaviour of the model matrices, specially in the radiation components. The results showed no change at all in the (**J-R**), **N** or **G** matrices, thereby, calculate them every time the code is executed is considered inefficient due its computational cost to calculate always the same values. For that reason, the values of those matrices were stored in .csv files that are uploaded only once in the Python code when it's executed.

Other aspect in the model that can be optimized is the amount of variables, which is directly proportional to: the number of buoys, that can't be modified for this purpose, and the order of the IRF radiation approximation. In [18] is mentioned that the best results will be obtained using 9 as IRF order of approximation. However, several tests were done, and the maximum error obtained when comparing the pH MATLAB model against the WEC-Sim tool using any order between 1 and 9 is still three orders of magnitude smaller than the wave height, as can be seen in Figure 5.1. Even when using waves with a frequency higher than 1 Hz, the error obtained between the MATLAB model and the WEC-Sim has the same order of magnitude.

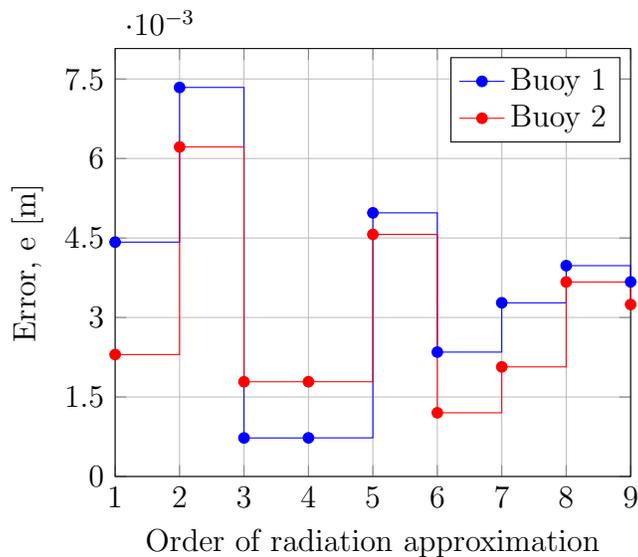


Figure 5.1: Error between the MATLAB model and the WEC-Sim results in the pH calculations with respect to the radiation approximation order, when using a regular wave with 4 m of height and a period of 5 s.

As can be seen, the difference in the results when using an IRF approximation of 9 or 1 is almost negligible, and showing even better results in the second buoy when using 1. On the other hand, the difference in the computational cost is considerable, because when using an order 9, the system depends on 40 states. That means, for instance, that only the  $(\mathbf{J}-\mathbf{R})$  matrix has 1600 elements, against the 64 that it has when using an order 1 for the approximation. For that reason, the number of operations to be solved in the system considerable less, therefore, the computational cost decreases.

Besides the change in the order of the IRF approximation and loading the model matrices instead of calculating them, the algorithm of the model in Python is similar to the Matlab one, as can be seen in the flowchart presented in Figure 5.2.

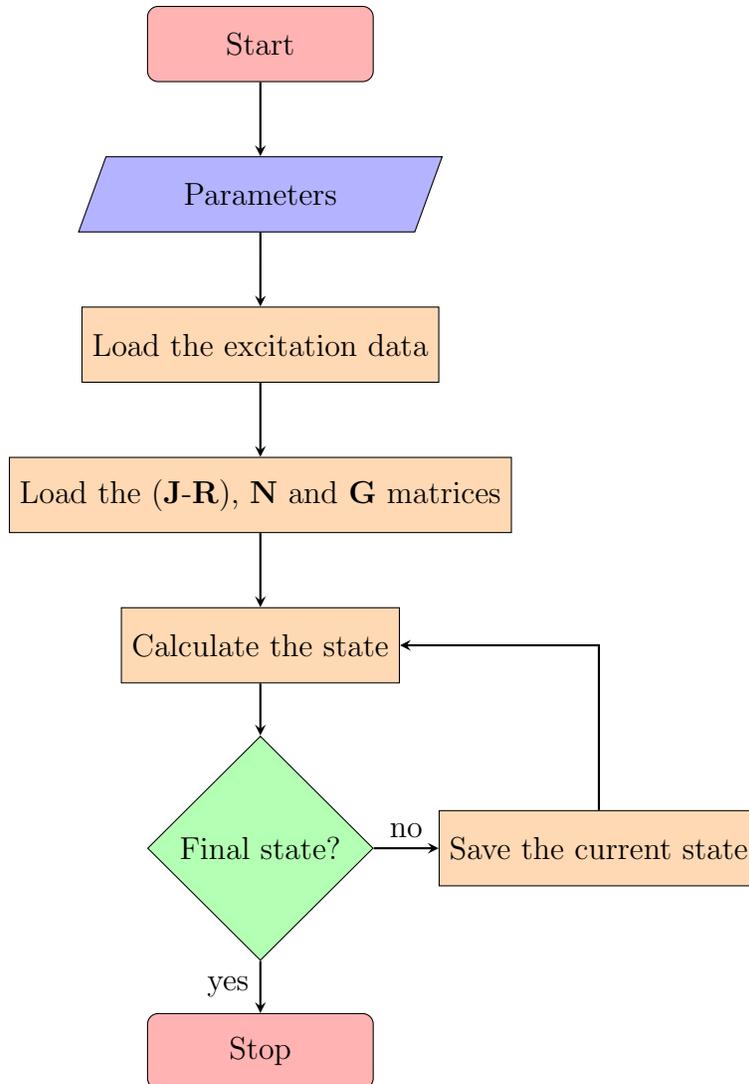
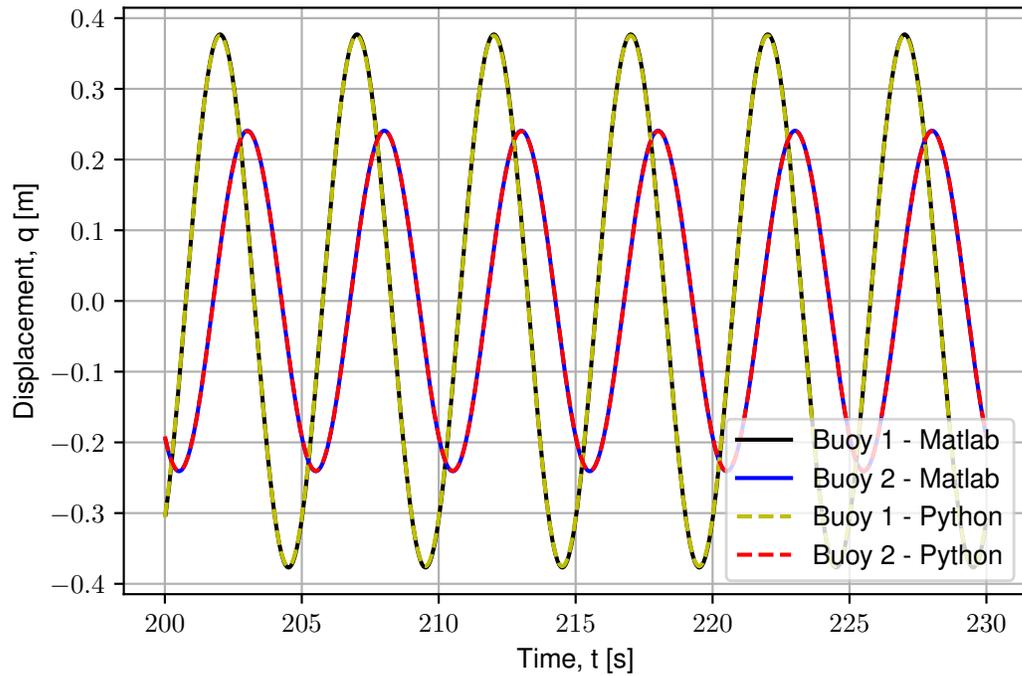


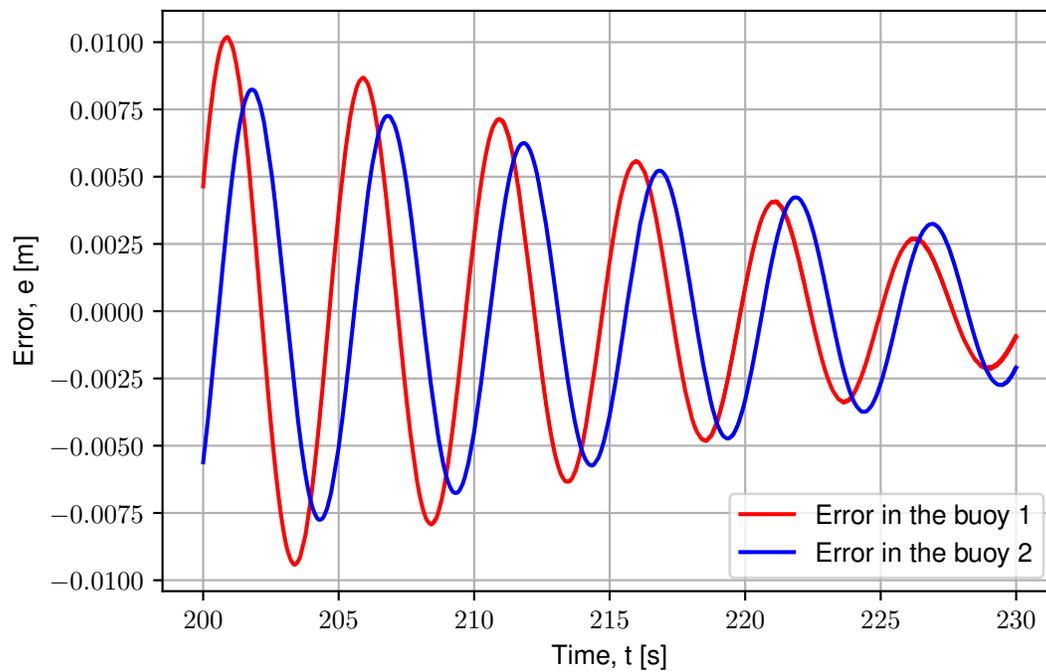
Figure 5.2: Flowchart of the Python code of the pH model of the WEC in open loop.

### 5.1.1 Open Loop Testing of Python’s Equivalent of the MATLAB Model

As mentioned before, the Python and Matlab algorithms are very similar on its structure, but when the performance is analysed, the Python algorithm presents a big advantage. Meanwhile the Matlab algorithm takes around 1:50 minutes to be executed, the Python algorithm takes around 0,6 s. Even when only the solution of the system to calculate the states is analysed, the Python code presents an advantage, elapsing around 0,4 s, against the 1:11 min in Matlab. Likewise, the results are also almost identical, as can be seen in Figure 5.3, where are shown the response of the system with MATLAB and Python, using the paramiters also by [18].



(a) Response of the system in Matlab and Python.



(b) Error in the response of the system between the Matlab and Python.

Figure 5.3: Comparison of results of the previous pH model in Python against the Matlab model.

### 5.1.2 Open Loop Testing in Python Using the Proposed Model

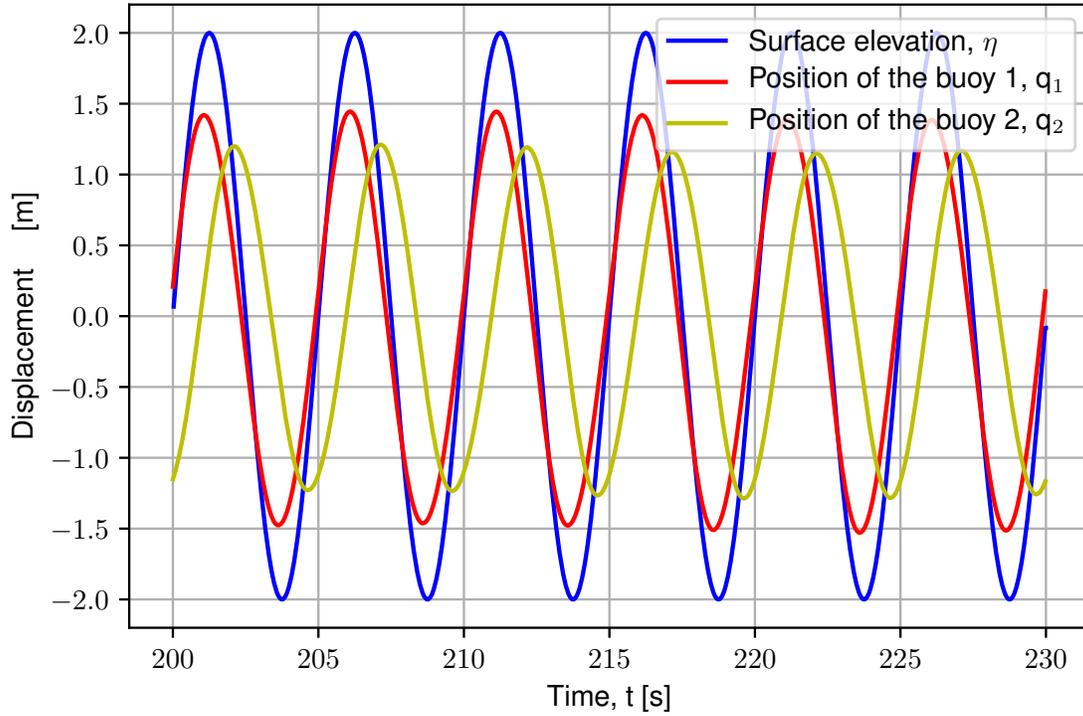
The model presented by [18] doesn't consider the hydrodynamics on the piston, for that reason, the modifications detailed in Section 4.2 were done, changing the parameters of the previous model to the proposed one, as shown in Table 5.1. The results when simulating the model using in the first piston the same area used by [10] and zero in the second can be seen in the Figure 5.4, where the displacement of the buoys is shown in the first graph, and the level in the upper reservoir in the second. Furthermore, it's important to mention that the algorithm showed in Figure 5.2 is independent of the modifications in the model.

Parameter	Symbol	Value
Added mass produced by the radiation over the buoy itself	$m_{\infty i,i}$	105 447,92 kg
Added mass produced by the radiation between the buoys	$m_{\infty i,j}$	11 326,83 kg
Area of the upper reservoir	$A_{UR}$	33,33 m <sup>2</sup>
Capacitance of the upper reservoir	$C_{UR}$	$3,40 \times 10^{-3}$
Density of the working fluid	$\rho$	1000 kg/m <sup>3</sup>
Density of the sea water	$\rho_s$	1025 kg/m <sup>3</sup>
Distance between the piston and the upper reservoir	$L_{UR}$	115 m
Gravitational acceleration	$g$	9,81 m/s <sup>2</sup>
Mass of the floater	$m_f$	1 500 kg
Mass of the piston	$m_p$	150 kg
Stiffness	$k$	$4,9271 \times 10^5$ N/m
Piston height	$h_p$	0,1 m
Piston-cylinder separation	$S_p$	400 $\mu$ m
Pressure in the upper reservoir	$P_{UR}$	101 325 Pa
PTO stiffness	$k_{pto}$	0 N/m
PTO damping coefficient	$b_{pto}$	11,53 kg/s
Viscosity of the working fluid	$\mu$	0,0734 Pa-s
Wave height	$H$	4 m
Wave period	$T$	5 s

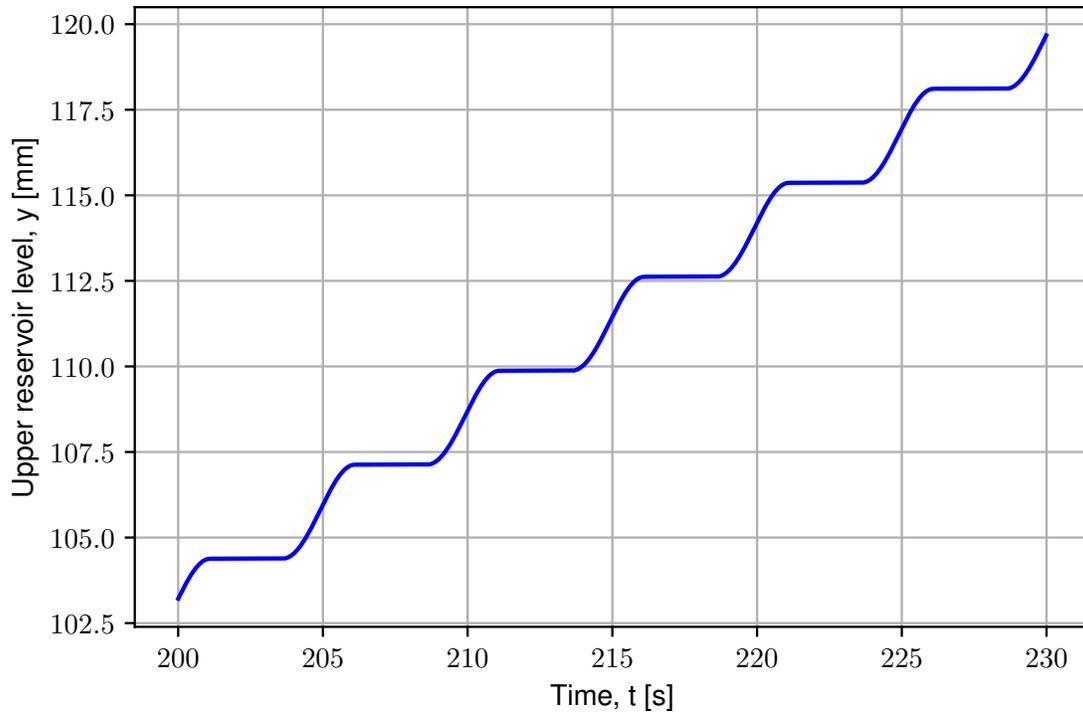
Table 5.1: Parameters and constants used in the Python simulation with the modified model.

The results can be compared against the results presented in [10], where is indicated that each up-stroke increases the level of the working fluid in the upper reservoir about 5 mm when a 4 m height and a period of 10 s create the movement, in this case, when using a wave of the same height but a period of 5 s and considering the effect of the radiation components between buoys, the increase in the level is approximately 2,6 mm when a piston radius of 0,1 m is used in the first piston and the second remains open.

Likewise, as can be seen in Figure 5.5, the point equilibrium of the buoys' movement was displaced below zero when increasing the piston area. The phenomena is due the force produced by the hydraulic head over the pistons that pulls the piston down, and with it, the buoy. Then, the fact that the hydraulic head is the same for both buoys, but its excitation



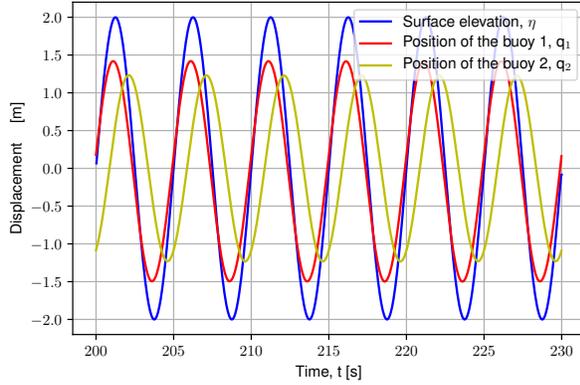
(a) Response of the proposed model in Python, with  $A_{p1}=0,0314 \text{ m}^2$  and  $A_{p2}=0$ .



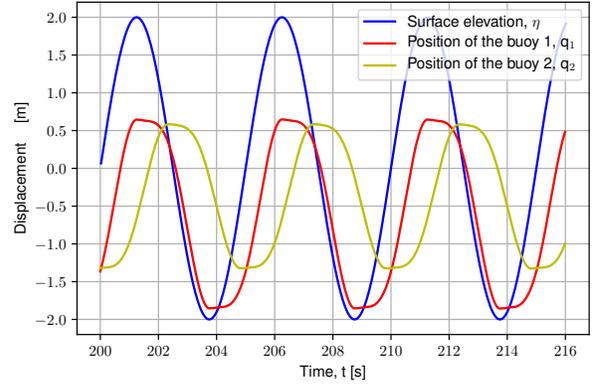
(b) Increase of the level in the upper reservoir using the proposed model in Python, with  $A_{p1}=0,0314 \text{ m}^2$  and  $A_{p2}=0$ .

Figure 5.4: Response of the proposed model with the Python code using a regular wave with 4 m of height and a period of 5 s.

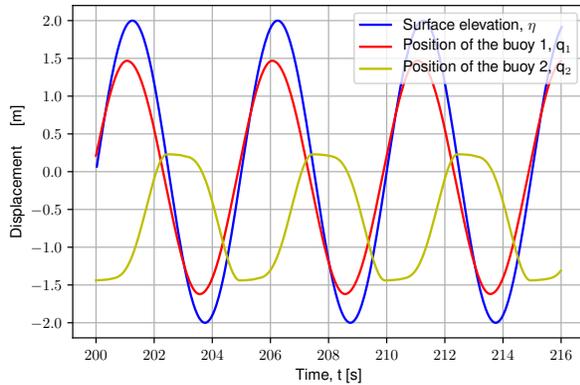
force is different, reduced from the first buoy to the next, not only the center of mass of the second buoy will be displaced down, but also, its amplitude will be smaller than the first in the case them both have the same area.



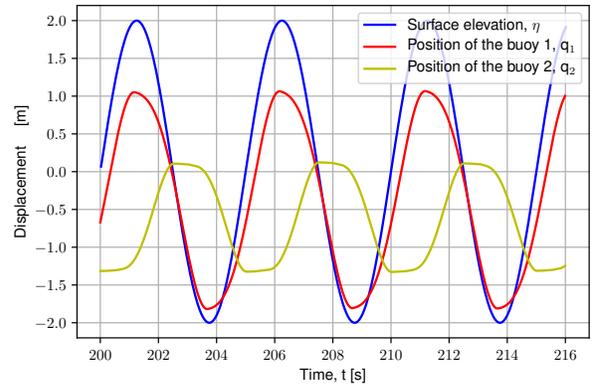
(a) Response of the proposed model, with  $A_{p1}=0,48 \text{ m}^2$  and  $A_{p2}=0,06 \text{ m}^2$ .



(b) Response of the proposed model, with  $A_{p1}=0,48 \text{ m}^2$  and  $A_{p2}=0,030 \text{ m}^2$ .



(c) Response of the proposed model, with  $A_{p1}=0,06 \text{ m}^2$  and  $A_{p2}=0,48 \text{ m}^2$ .



(d) Response of the proposed model, with  $A_{p1}=0,30 \text{ m}^2$  and  $A_{p2}=0,48 \text{ m}^2$ .

Figure 5.5: Response of the proposed model with the Python code, using a regular wave with 4 m of height and a period of 5 s, with different piston area configurations.

On the other hand, the use of areas bigger than  $0,01 \text{ m}^2$  caused a divergence in the results due to the Euler method used to solve the differential equation system of the port-Hamiltonian representation and calculate the states in the model if a  $dt=0,1 \text{ s}$  is used. To solve this issue, the input data was interpolated, allowing to use a smaller interval in the Euler method, nonetheless, the computation time increases when the time interval is decreased.

The behaviour of the computation time when simulating the system for 100 s can be seen in Figure 5.6, where the computation time of 10 individual tests for each time interval were averaged, with an average relative standard deviation of 2,39 %. As can be seen, the computation time is a negative potential function of the time interval, thus, time intervals

smaller than 10 ms will take too long, nevertheless, time intervals bigger than 40 ms will result in a divergence in the results.

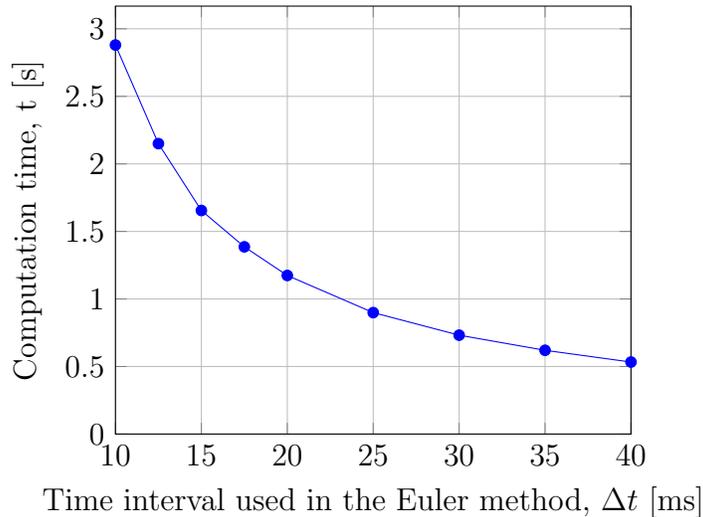


Figure 5.6: Computation time of the proposed model when simulating a 100 s interval with regular waves, with respect to the time interval used in the Euler method.

## 5.2 Implementation of the MPC

The last step before implement the MPC is to develop an appropriate optimization of the model to obtain the area that guarantee the maximum energy absorption. However, since the optimization tools are based on numerical methods, the highest is the desired accuracy, the more time it will take for the method to solve the problem, therefore, a good combination computation time against accuracy needs to be found.

In this case, the sequential least squares programming (SLSQP) method was used in the Python’s optimization tool, since it’s the most accurate method available that can handle boundaries, in this case, the range of possible areas per piston. At the same time, and as mentioned before, instead of maximizing the cost function, it was minimized.

In general, the WEC has  $8^n - 1$  possible piston combinations if the  $\{0,0\}$  case is discarded, where  $n$  is the number of buoys, becoming inefficient to calculate the individual effect for each combination and then compare then to obtain the best combination, as concluded by [22]. When the two floater case presented in [18], 64 possible piston combinations can be reach, and in specific wave conditions. Using the irregular wave presented in Figure 5.7 it is possible to graph the level of the working fluid after the interval, as shown in Figure 5.8.

The wave presented in Figure 5.7 was, then, used to feed the optimization tool. As known, the resulting piston configuration and, therefore, the amount of energy extracted, depends

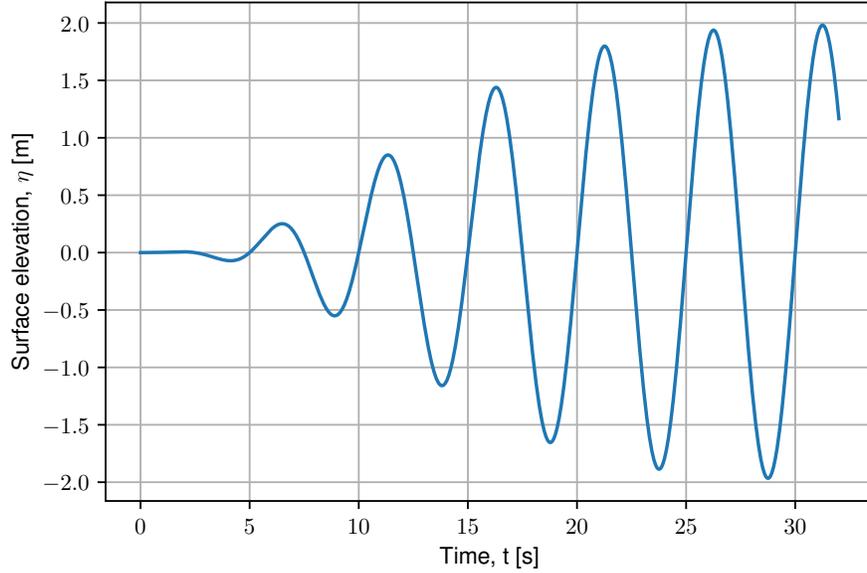


Figure 5.7: Irregular wave used as testing conditions.

on the time interval used in the Euler method. In the Table 5.2 are shown the resulting configuration of the piston for each of the intervals used in the optimization tests.

Time interval (ms)	1 <sup>st</sup> up-stroke	2 <sup>nd</sup> up-stroke	3 <sup>rd</sup> up-stroke	4 <sup>th</sup> up-stroke
10	{0.497, 0.514}	{0.497, 0.514}	{0.497, 0.5144}	{0.497, 0.514 }
15	{0.54, 0.54}	{0.54, 0.54}	{0.54, 0.54}	{0.54, 0.54}
20	{0.54, 0.54}	{0.54, 0.54}	{0.54, 0.54}	{0.54, 0.54}
25	{0.54, 0.54}	{0.54, 0.54}	{0.54, 0.54}	{0.54, 0.54}
30	{0.342, 0.422}	{0.342, 0.422}	{0.362, 0.472}	{0.54, 0.54 }

Table 5.2: Resulting piston combination from the optimization strategy of the proposed model when the wave presented in Figure 5.7 is used.

In Figure 5.9 are shown the computation time and the increase of the level in the upper reservoir as function of the time interval used in the Euler method. The presented values for the computation time are the average of 10 individual tests for each time interval, with an average relative standard deviation of 3,21 %. The values of the upper reservoir level are the theoretical values using directly the resulting areas from the optimization.

As can be seen, the negative potential behaviour of the computation time against the time intervals used in the Euler method is shown also during the optimization. On the other hand, the theoretical level of the working fluid on the upper reservoir shows a convex behaviour. *A priori*, it appears that the higher the interval, the less computation time without affecting considerably the increase in the level of the upper reservoir, however, a big interval is more likely to produce divergences in the model.

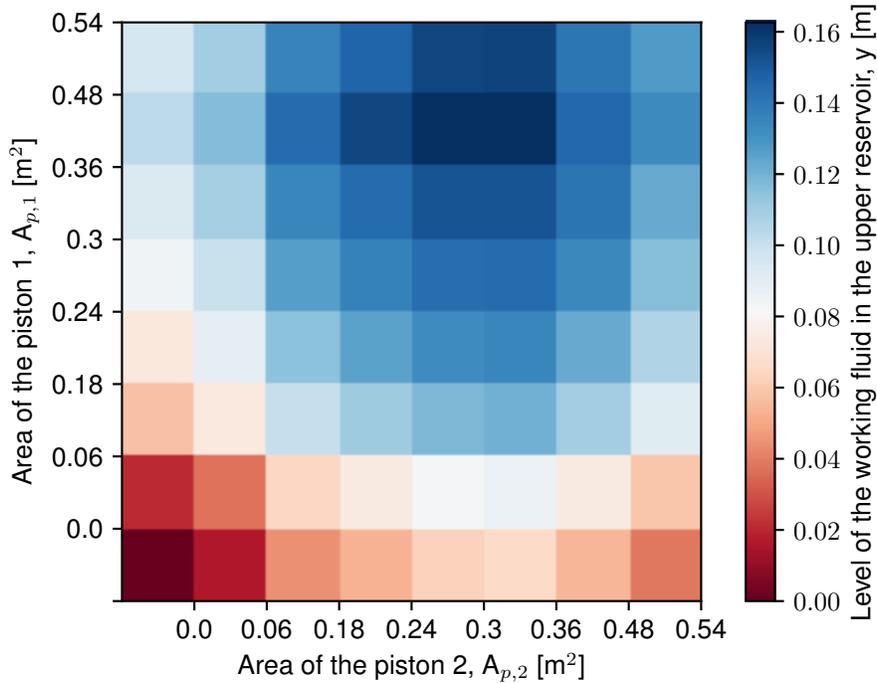
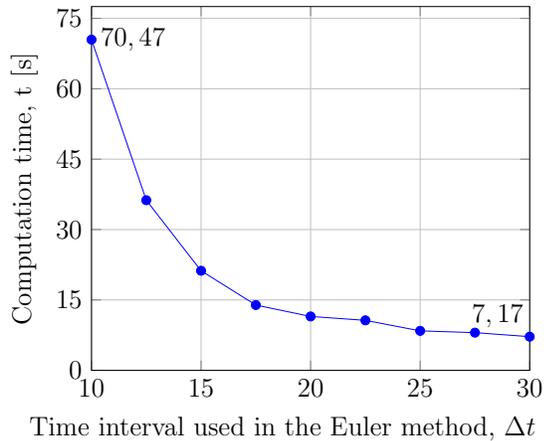


Figure 5.8: Level of the working fluid in the upper reservoir against the piston combination after the irregular wave shown in Figure 5.7. The maximum value is 0.1626 m, with the combination  $\{0.48, 0.30\}$ .

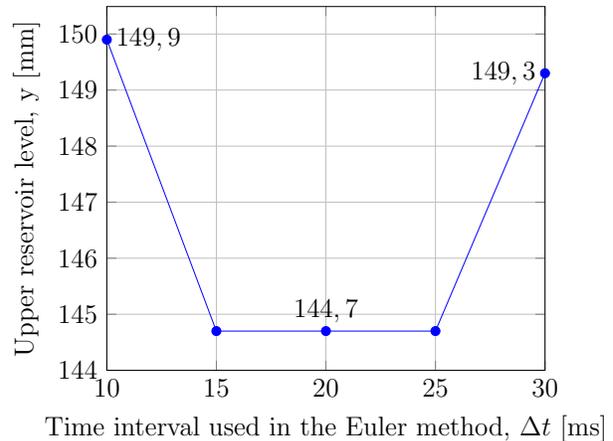
For that reason, and even though it was established that intervals over 40 ms will produce divergences, a safety buffer was used for the MPC implementation and the maximum interval used for the Euler method was 30 ms, which, as shown in the graphs in Figure 5.9, appears to be the best cost-benefit combination according with the data used when using a prediction horizon of 3, as recommended by [16] and [44]. Nonetheless, the level presented on the graph is the theoretical level, in the case that the piston would have a continuous set of values, however, one of the possible combinations must be chosen.

To chose the best possible piston combination, several possibilities can be considered. For instance, rounding to the closest possible area. In that case, the resulting combination for the first two up-strokes is  $\{0.36, 0.48\}$ , resulting in 144,2 mm in the upper reservoir level, against the theoretical 149,3, *i.e.* result in an error of 3,43 %. However, if the second area can be rounded to 0,36 m<sup>2</sup> instead of 0,48 m<sup>2</sup>, since the relation between the area and the increase in the level is not linear. If that round is taken into account, it would result in a level of 158,1 mm, 6,19 % bigger than the theoretical value associated with the combination given by the optimizer. For the third considerable up-stroke, the areas were round to  $\{0.36, 0.48\}$  since the results showed better results. The behaviour of the buoys and the level in the upper reservoir is shown in Figure 5.10

The fact that the optimization didn't actually achieved the best value can be due numerical errors associated with the method itself. If the discrete set of values could be considered



(a) Computation time for the optimization process with each time interval.



(b) Obtained level in the upper reservoir with each time interval used in the Euler method.

Figure 5.9: Results of the optimization of the proposed model with the Python code, using the irregular wave shown in Figure 5.7, and a filtering factor to neglect up-strokes smaller than 0,5 m for the buoy 1 and 0,2 m for buy 2.

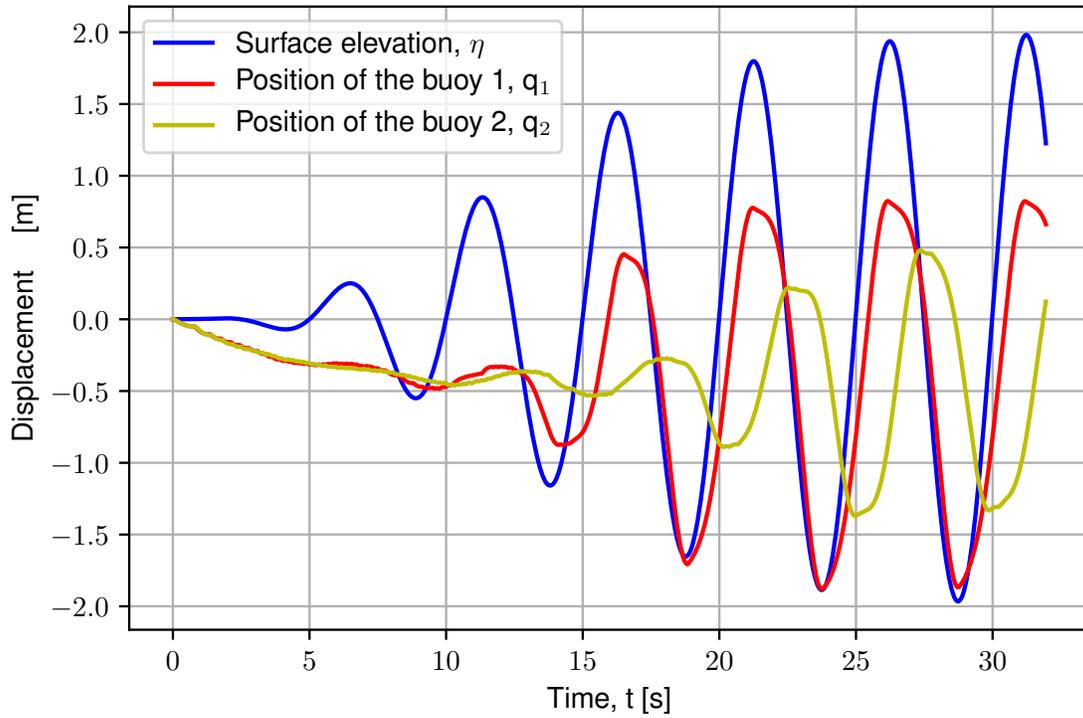
instead, is likely that the error wouldn't appear, however, Python tools doesn't allow a discrete optimization. Given the above, the errors when the level is compared against the maximum theoretical value presented in the Figure 5.8 of the previously mentioned combinations, according with the considered roundings, are: 12,57 % and 3,04 % respectively.

### 5.3 Discussion

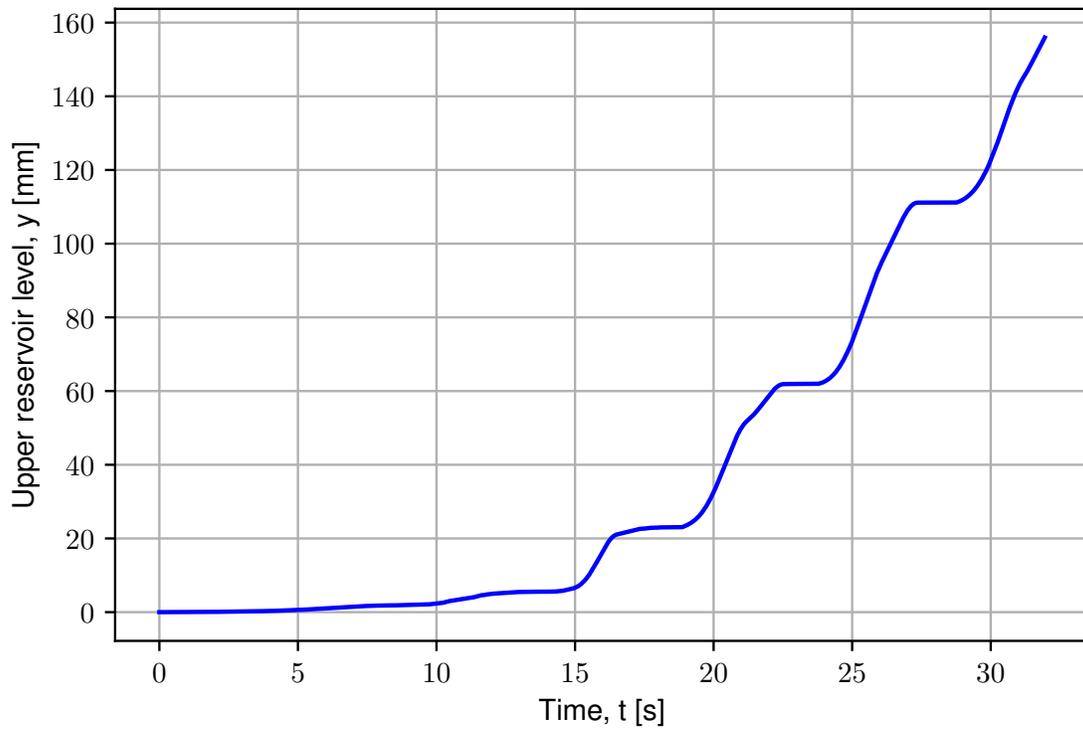
The problem to be solved with the present work is mainly the high computational cost needed to implement a control strategy on the OG WEC in time domain. Previous works required up to one day of computation only to model the system, and for that reason, a port-Hamiltonian approach was presented in order to reduce the time required for the model. Thereby, the main goal of this project was to develop a control strategy, using as base the pH model presented by [18] as requirement, to obtain the piston configuration that guarantee the maximum possible extracted energy.

To fulfil the main goal, several specific objectives were defined, starting with the selection of a control strategy in time domain for the WEC. As shown in Chapter 3.5, several control strategies were compared based on a selection criteria detailed in Section 3.6 and evaluating using the rubric shown in Table 3.1, resulting MPC the best option, since previous work has been done by the OG team showing promising results, [16]. On the other hand, a PSO was used without showing optimistic results, [22].

With respect to the programming language, in Section 3.8 the criteria used to select between Python, Octave and Julia was described, which rubric is presented in Table 3.3,



(a) Position of the buoys after four effective up-strokes.



(b) Increase of the level in the upper reservoir after four effective up-strokes.

Figure 5.10: Response of the proposed model with the Python code against the irregular wave shown in Figure 5.7, using a filtering factor to neglect up-strokes smaller than 0,5 m for the buoy 1 and 0,2 m for buy 2.

resulting Python the best candidate due its background in MPC implementations, existing documentation and packages. Finally, two implementation strategies were proposed in Section 3.10, resulting more promising the develop of a set of functions as a toolbox for its advantages, specially in easy update and scalability. Therefore, the control strategy is designed as a MPC toolbox on Python.

The second specific objective aims to develop the control algorithm for the WEC. Due a MPC strategy is being used, first, the model must be validated. The model used in [18] is described in Section 4.1 and the modifications that lead to the proposed model in Section 4.2. The assumptions used to simplify the model are mentioned in Subsection 4.2.1.

Considering the proposed model, the control strategy to be implemented was described in Chapter 4.3. There, the effect of the area of the piston was demonstrated in the model matrices, allowing to use it directly as control variable. Then, the cost function was defined as the opposite of theoretical potential energy gained when pumping the working fluid from the lower reservoir to the upper reservoir, since the available optimization tools in Python allows only to minimize a function. Finally, a filtering factor was defined with the theoretical energy the wave can add to the system, fixing a value as threshold to avoid unnecessary changes that may include noise on the system, represented as oscillations in the displacement or velocity of the pistons.

Given the above, Chapter 5 is dedicated to show the simulation results obtained with the different codes. First, an equivalent of the MATLAB code used in [18] was done to validate the base model. The results in Figure 5.3 showed an accurate response, with an maximum absolute error of around 1 mm in the buoys displacement, which is one order of magnitude smaller than the amplitude of the movement. Then, the proposed model results were compared against the results reported in [10]. As can be seen in Figure 5.4, where the order of the answer is comparable, showing similar behaviour on the buoys, and the increase per up-stroke in the upper reservoir of the same order of magnitude.

On the other hand, a delay between the first buoy and the wave appears when the area of the piston increases. This is because of the model used; in this case, when the up-stroke begins, the check valves are closed suddenly to start the rising movement of the piston, however, when the area of the piston is bigger than zero, the pressure over the piston creates a force that pulls down the buoy, so the when the difference between the profile of the wave and the center of mass of the buoy create a buoyancy force big enough to lift the buoy it starts to move, but it is not an immediate movement.

During the down-stroke, a similar effect happens, since the area of the piston is zero, the pressure of the water column effect disappears, and the buoy starts to submerge by its own weight and due the added mass produced by the radiation components, creating a different movement in the displacement of the piston during the up-stroke and the down-stroke.

With respect to the MPC implementation, the irregular wave presented in Figure 5.7 was used as input, and the MPC was tested using a prediction horizon of 3 as recommended by

[15]. The behaviour in the level of the working fluid in the upper reservoir for each one of the combinations is shown in Figure 5.8, where a brute force method is used to obtain the maximum theoretical energy that can be extracted per combination to have a reference to validate the results of the control strategy.

Likewise, the computation time and the response of the optimization of the cost function in the MPC depends on the interval used in the Euler method when calculating the states. For that reason, a sensibility analysis was done, using Figure 5.8 as reference. The results are shown in Figure 5.9 and the piston combination per up-stroke is shown in Table 5.2.

As expected, some of the areas obtained after the optimization doesn't correspond to any of the available set of values, therefore, they need to be rounded. In this case, for example, with an interval of 30 ms, the best combination was obtained, showing that during the first two down-strokes was  $\{0.342, 0.422\}$ , that can be rounded upside or downside. If they are rounded to its absolute closest area, the energy obtained is less than if the second piston is rounded downside, despite its closest value is rounding upside, resulting in an error of 3,04 % against the best combination obtained through brute force, indicating that there are numerical error in the strategy when the time is limited, but achieving an approximation inside the allowed range.

The previous phenomenon can be explained using Figure 5.8, because the relation between the level of the working fluid in the upper reservoir is a non-linear relation with the area, thus, a method to determine the best rounding possibility is needed. In this case, the possible rounding combinations were calculated and the best of these possible combination were used. However, if this method is used for  $n$  buoys, there is a maximum range of  $2^n$  possible rounding combinations, that can be decreased if the resulting area of two or more buoys are between the same possible areas, nevertheless, the method can become too computationally demanding, so, statistic methods can be used, based on charts as the one presented in Figure 5.8.

Finally, the developed strategy showed promising results when compared with the code used in [18], since the equivalent model is around 180 times faster (0,6 s against 1:50 minutes, in average) in a computer with an Intel i7 processor at 2,2 GHz and with 16 GB of RAM; the proposed model computation time as function of the time interval used for the Euler method is shown in Figure 5.6.

The computational cost of the proposed model, however, can't be directly compared with the time domain model presented in [15], because it consists on a  $10 \times 1$  instead of a  $2 \times 1$  array, and the radiation components weren't yet available for that combination. Nonetheless, the time domain model took around 24 hours to simulate 200 s in a computer with an Intel Xeon at 3,4 GHz and 64 GB of RAM, without considering the interaction between buoys, as done in the present model, against the 0,6 s to simulate 100 s in the proposed model, with the conditions mentioned before.

Given the above, if it is considered a model based control strategy, as MPC, the compu-

tation time the model is a crucial factor in the computation time of the optimization process. In this case, the optimization process took 2,4 s (in a computer with the conditions mentioned before) to determine the piston combination for each incoming wave, giving an optimistic perspective about the extrapolation of this strategy.

# Chapter 6

## Conclusions, Limitations and Recommendations

The present Chapter aims to analyse the main results in Section 5.3 and the conclusions obtained in Section 6.1. Then, the limitations of the project are pointed in Section 6.2 and, base on them, a series of recommendations are presented in Section 6.3 to eventually be taken into consideration in future works.

### 6.1 Conclusions

Several control strategies were considered as possible implementation methods. They were compared using the rubric shown in Table 3.1, resulting MPC as the best candidate in this specific case. The main reasons is the existing background of the OG research team in MPC implementations, based on the current state-of-the-art of control strategies for WECs. As shown in the present research, MPC is effective strategy to optimize the amount of energy extracted from a WEC.

A port-Hamiltonian approach was used to create a time domain model, which allows to implement a control algorithm for the pumping system. Under regular waves condition, the results of the developed control strategy are comparable with the results reported by the OG team in [10], despite all the simplifications considered and showed in Subsection 4.2.1, but considering the interaction between buoys, demonstrating that port-Hamiltonian model is a promising approach due to its simplicity and adaptability. As well, when using irregular waves conditions, the control strategy successfully obtained a piston combination that maximizes the amount of energy extracted by the WEC, when compared with a brute force method that analyses the extracted energy by the system, according with the piston combination.

At the same time, the proposed port-Hamiltonian model requires less computation time than the models presented by [10], [15], [16] and [18]. As mentioned, modelling 400 s of a  $2 \times 1$  system takes around 0,6 s in a computer with less resources (processor Intel Core i7, 16 GB of RAM and 2,2 GHz), time that will decrease considerably if the computer used to

simulate the time domain model is used (processor Intel Xeon, 64 GB of RAM and 3,4 GHz). It must be considered that the time domain model system is a  $10 \times 1$  array, and the direct comparison wasn't possible because the radiation components for that configuration weren't available yet. However, given the results, it is expected that the computation time under the same conditions will be less.

Given the above, a model predictive control strategy with less computation cost than the time domain model was successfully implemented in Python, indicating that open source languages are a useful tool that can reduce future implementation costs. However, some of the functions available on Python packages might not be as complete as those available in Matlab/Simulink environments, so, further works can include the creation of functions to improve the performance of the toolbox developed in this project.

Furthermore, the analysis of the model presented in [18] showed that decreasing the order of the IRF for the radiation approximation will reduce considerably the computation cost by reducing the number of operations needed to calculate the states, without affecting negatively the response of the model. At the same time, the optimization process computation time is also reduced, since the model is simpler.

Besides, the results confirmed that the relation between the area of the pistons and the increase in the water head is not linear. Therefore, if a continuous optimizer is used instead than a discrete optimizer, a rounding method is needed, since choosing the absolute closest available piston area won't guarantee the maximum possible energy extraction.

## 6.2 Limitations

The current project considered several assumptions to simplify the model and being able to obtain the best possible piston configuration in the shortest time possible. For that reason, several limitations exist and must be considered for future works:

- To reduce the computation time, the parameters of the pH model proposed were assumed as fixed values, except those which equations were described as part of the control strategy. Since the provided model corresponds to a single line of two consecutive floaters, the current project is limited to that configuration, specially the radiation components are crucial to include more elements in the model. However, that information wasn't provided by the OG team during the period established for the fulfilment of this project. Nevertheless, since the strategy was meant to be scalable, most of its functions won't require mayor modifications when different configurations are tested.
- In general, the MPC strategies require a high fidelity model to obtain reliable feedback for the control, for that reason, the accuracy on the results depends mostly on the model used. At the same time, the most accurate the model is, the more computationally demanding it turns.

- A SPP approximation is used, however, the real system is a MPP. Therefore, the scaling in several factors need to be considered. In this case, one aspect that can directly influence the model is the mass of the pistons, because it was assumed a fixed value for the mass on the piston, regardless the area used, but, since the change in the area is related to the use of one or more pistons, its total mass depends on the area.
- The area of the valves in the piston is considered zero during the down-stroke. However, the valves opening area is not the same that the area of the pipe, thus, the force of the column of water over the piston during the down-stroke is not considered.
- Due the limited resources on Python in the solution of the simultaneous differential equations, the Euler method was used to calculate the states in the model. However, it may be inaccurate or even diverge under specific circumstances, *e.g.* when the time interval is not small enough, driving to the need to interpolate the input data to avoid divergences.
- A filtering factor was use to avoid unnecessary changes in the configuration of the pistons. In this project, it was determined according with the data provided by the OG team to test the model and the control system. Therefore, an objective method to calculate it is required.
- Even though Python has several function to optimize scalar and vectorial non-linear systems, despite of been in the range considered as acceptable in both response and computation time, the results obtained weren't completely accurate when finding the best solution.
- The optimization method gives a continuous solution, however, the nature of the problem is discrete. For that reason, a rounding method is required to determine the best area because the increase in the level of the working fluid in the upper reservoir has a non-linear relation with the area of the piston, thus, rounding to the closest area is not necessary the best choice.

## 6.3 Recommendations for Future Research

Based on the limitations presented in the previous Section, a series of recommendations based on the current project are proposed to future research:

- In Subsection 4.2.1, the assumptions to simplify the system were announced. Even though they are based on previous results obtained by the OG team and the current project, some of them can actually be considered in further research to obtain a more realistic model. However, a sensibility analysis is recommended to be done to determine if their consideration in the model represent a significant improvement in the results, without a negative effect on the computation time.
- Since the SPP approach was used, the mass associated to each buoy was considered as constant, however, in the real MPP system, the mass will change according with the

configuration of the piston, thus, in eventual modification of the presented model or future models, this effect can be considered.

- During the down-stroke, the area of the pistons are considered as 0. Nevertheless, the valves have an open area close, but not equal, to the total area of the pipe during the down-stroke, and the difference between the total area of the pipe and the open area of the valve is not a fixed proportional value, thus, a function to calculate this value during the down-stroke can be added to the Python code of the model.
- One of the main differences between the Python code and the MATLAB code is the method used to solve the set of differential equations in the port-Hamiltonian model. In the present project, the Euler method was used because of the limited options Python offers. Even though it has some ODE tools, they can't be directly applied on the developed Python code, for that reason, an ODE solver can be built to ensure more accurate solutions for the system and avoid the need for interpolate the data to prevent divergences in the model.
- In the creation of the time windows, a filtering factor was chosen as a design criteria according with the data used for simulations and the obtained results. However, several more objective considerations can be used to determine its value, for instance, calculate the energy required to change the configuration of the pistons and use it as a minimum threshold.
- With the optimization tools available in Python it's possible to obtain a solution in the acceptable range. However, it doesn't guarantee the best combination. For that reason, a new optimizing function can be developed to improve the results obtained in this project, or its application in further projects.
- Because of the continuous nature of the optimization tool, the discrete nature of the problem and the non-linear relation between the increase in the level of the working fluid in the upper reservoir and the area of the piston, a rounding method must be determined to chose the most appropriate area for each piston.

# Bibliography

- [1] J. Sachs, *The Age of Sustainable Development*. New York: Columbia University Press, 2015.
- [2] N.Stern. *Te Economics of Climate Change - The Stern Review*, Cambridge University Press, 2007.
- [3] EIA, *International Energy Outlook 2016*, The U.S. Energy Information Administration, [Online]. Available: [https://www.eia.gov/outlooks/ieo/pdf/0484\(2016\).pdf](https://www.eia.gov/outlooks/ieo/pdf/0484(2016).pdf). [Accessed 26 January, 2018] 2016.
- [4] A. S. Bahaj, “Generating electricity from the oceans”, *Renewable and Sustainable Energy Reviews*, vol. 15, pp. 3399-3416, 2011.
- [5] H. Meijer, “Simulation of a piston-type hydraulic pump for the Ocean Grazer”, M.Sc. thesis, University of Groningen, The Netherlands. 2014.
- [6] J. Martí Saumell, “Dynamical modelling, analysis and control design of a distributed sea wave energy storage system”, M.Sc. thesis, Universitat Politècnica de Catalunya, Spain, University of Groningen, The Netherlands. 2013.
- [7] A, Cruz Gispert, “Analysis and dynamical modeling of a piston valve for a wave energy converter”, M.Sc. thesis, University of Groningen, The Netherlands. 2015.
- [8] M. A. Hassink, “Design of experimental measurements to obtain performance characteristics of a multiple ball check valve”, M.Sc. thesis, University of Groningen, The Netherlands. 2016.
- [9] R.M. Zaharia, “Understanding of the Single-Piston Pump inside the Ocean Grazer”, M.Sc. thesis, University of Groningen, The Netherlands. 2018.

- [10] A. I. Vakis & J. S. Anagnostopoulos, “Mechanical design and modelling of a single-piston pump for the novel power take-off system of a wave energy converter”, *Renewable Energy*, vol. 96, pp. 531-547, 2016.
- [11] Z. Yu, “Hydrodynamic analysis of the floater blanket in the frequency domain”, M.Sc. thesis, University of Groningen, The Netherlands. 2017.
- [12] M. Mohr, “Calculation of power absorption from irregular waves by the floater blanket of the Ocean Grazer”, B.Sc. thesis, University of Groningen, The Netherlands. 2018.
- [13] B. Galván García, “Nonlinear control design for wave energy converter”, M.Sc. thesis, Universitat Politècnica de Catalunya, Spain, University of Groningen, The Netherlands. 2014.
- [14] T. Dijkstra, “Maximizing revenue of electricity generated by the Ocean Grazer - Implementation of a model predictive control on the storage reservoir of the Ocean Grazer-”, M.Sc. thesis, University of Groningen, The Netherlands. 2016.
- [15] Y. Wei, J.J. Barradas-Berglind, M. van Rooij, W.A. Prins, B. Jayawardhana & A.I. Vakis, “Investigating the adaptability of the multi-pump multi-piston power take-off system for a novel wave energy converter”, *Renewable Energy*, vol. 111, pp. 598-610, 2017.
- [16] J. J. Barradas-Berglind, H. Meijer, M. van Rooij, S. Clemente-Pinol, B. Galvan-García, W. Prins, A. Vakis & B. Jayawardhana, “Energy capture optimization for an adaptive wave energy converter”, in: Proceedings of *Second International Conference on Renewable Energies Offshore*, CRC Press, Taylor and Francis Group, 2016, pp. 171-178.
- [17] A. Fernández Vuelta, “Modular modelling for the power take-off system of a wave energy converter”, M.Sc. thesis, University of Groningen, The Netherlands. 2017.
- [18] M.Z. Almuzakki, J.J. Barradas-Berglind, Y. Wei M. Muñoz-Arias, A.I. Vakis & B. Jayawardhana. “A port-Hamiltonian Approach to Cummins’ Equation for Floater Arrays with Linear Power-Take Off Systems”, 2018.
- [19] M. Melikoglu, “Current status and future of ocean energy sources: A global review”, *Ocean Engineering*, vol. 1148 563–573, 2018.

- [20] A. F. O. Falcão, “Wave energy utilization: A review of the technologies”, *Renewable and Sustainable Energy Reviews*, vol. 14, pp. 899-918, 2010.
- [21] Observ’ER, *The state of renewable energies in Europe*. Paris: EuroObserv’ER Report, 2013.
- [22] C. Hoogerbrugge, “Investigating the effect of varying the distribution of damping coefficients inside a floater blanket on the power extraction for a novel wave energy converter”, BSc. thesis, University of Groningen, The Netherlands. 2018.
- [23] L. H. Holthuijsen, *Waves in Oceanic and Coastal Waters*, 1st. ed. Cambridge, 2007.
- [24] B. Kamranzad, V. Chegini & A. Etemad-Shahidi. “Temporal-spatial variation of wave energy and nearshore hotspots in the Gulf of Oman based on locally generated wind wave”, *Renewable Energy*, vol. 94, pp 341-352, 2016.
- [25] European Marine Energy Centre, “Press Release: Emec Seeks Feedback From Industry for P2-002”, EMEC, February, 2016, [Online]. Available: <http://www.emec.org.uk/press-release-emec-seeks-feedback-from-industry-for-p2-002/>. [Accessed 17 January, 2018].
- [26] A. Jha, “Making waves: UK firm harnesses power of the sea in Portugal”, *The Guardian*, September, 2008, [Online]. Available: <https://www.theguardian.com/technology/2008/sep/25/greentech.alternativeenergy>. [Accessed 17 January, 2018].
- [27] E. Friis-Madsen, H.C. Soerensen & S. Parmeggiani, “The development of a 1,5 MW Wave Dragon North Sea Demonstrator”, in: *Fourth International Conference on Ocean Energy*, 2012, [Online]. Available: <http://stateofgreen.com/files/download/612>. [Accessed 26 January, 2018].
- [28] T. Bruce & J. O’Callaghan, “Oscillating Water Column Wave Energy Converters at the Coast: A Review and Forward Look”, in: *Proceedings of Sixth International Conference on Coastal Structures*, World Scientific, pp. 345-356, 2011.
- [29] T. Helth, “The Construction, Commissioning and Operation of the LIMPET Wave Energy Collector”, Wavegen, [Online]. Available: <https://web.archive.org/web/20110626204435/http://www.wavegen.co.uk/pdf/Construction%20commission%2026%20operation%20of%20LIMPET.pdf>. [Accessed 17 January, 2018].

- [30] P. Frigaard, J. P. Kofoed & W. Knapp, “Wave Power Plant using low-head turbines”, Aalborg University, 2004, [Online]. Available: [http://vbn.aau.dk/files/57370067/Wave\\_Dragon\\_wave\\_power\\_plant\\_using\\_low\\_head\\_turbines\\_Presentation.pdf](http://vbn.aau.dk/files/57370067/Wave_Dragon_wave_power_plant_using_low_head_turbines_Presentation.pdf). [Accessed 26 January, 2018].
- [31] J. Salton, “Oyster - the world’s largest working hydro-electric wave energy device”, New Atlas, November 26th, 2009, [Online]. Available: <https://newatlas.com/oyster-hydro-electric-wave-energy-device/13461/>. [Accessed 26 January, 2018].
- [32] L. Cameron, R. Doherty, A. Henry, K. Doherty, J. Van’t Hoff, D. Kaye, D. Naylor, S. Bourdier & T. Whittaker, “Design of the Next Generation of the Oyster Wave Energy Converter ”, in: *Third International Conference on Ocean Energy*, 2010.
- [33] S. Parsons, “‘Oyster’ System is a New Way to Harness the Power of Waves”, inhabitat, August 5th, 2009, [Online]. Available: <https://inhabitat.com/oyster-generates-electricity-from-waves/>. [Accessed 26 January, 2018].
- [34] A. van der Schaft, D. Jeltsema, *Port-Hamiltonian Systems Theory: An Introductory Overview*, Foundations and Trends in Systems and Control, vol. 1(2-3), pp- 173-378, 2014. [Online]. Available: <http://www.math.rug.nl/arjan/DownloadVarious/PHbook.pdf>. [Accessed 4 March, 2018].
- [35] M. Muñoz-Arias, “Energy-based control design for mechanical systems, applications of the port-Hamiltonian approach”, PhD thesis, University of Groningen, The Netherlands. 2015.
- [36] A. van der Schaft, *Port-Hamiltonian Systems: An Introductory Survey*, University of Groningen, The Netherlands, 2018. [Online]. Available: <http://www.math.rug.nl/arjan/DownloadPublicaties/ICMvanderSchaft.pdf>. [Accessed 4 March, 2018].
- [37] D. E. Seborg, T. F. Edgar & D. A. Mellichamp, *Process Dynamics and Control*, 2nd. ed. Wiley, 2004.
- [38] E. F. Camacho & C. Bordons, *Model Predictive Control*, 2nd. ed. London: Springer, 2007.

- [39] J. M. Maciejowski, *Predictive Control with Constraints*, 1st. ed. Prentice Hall, 2000.
- [40] W. R. Sultana, S. K. Sahoo, S. Sukchai, S. Yamuna & D. Venkatesh, “A review on state of the art development of model predictive control for renewable energy applications”, *Renewable and Sustainable Energy Reviews*, vol. 76(2017), pp. 391-406, 2017.
- [41] Y. S. Qudaih, M. Bernard, Y. Mitani and T. H. Mohamed, “Model Predictive Based Load Frequency Control Design in the Presence of DFIG Wind Turbine”, *Proceedings of the 2nd International Conference on Electric Power and Energy Conversion Systems (EPECS<sup>o</sup> 11)*, Sharjah, UAE, 15-17 Nov, 2011.
- [42] M. Z. Bernard, T. H. Mohamed, R. Ali, Y. Mitani & Y. S. Qudaih, “PI-MPC Frequency Control of Power System in the presence of DFIG Wind Turbines”, *Engineering*, vol 5(9B), pp. 43-50, 2013. [Online]. Available in: <http://www.scirp.org/journal/PaperInformation.aspx?PaperID=37707>
- [43] S. Zhan, W. He, G. Li, “Robust Feedback Model Predictive Control of Sea Wave Energy Converters”, *IFAC PapersOnLine*, vol 50(1), pp. 141-146, 2017.
- [44] M. N. Soltani, M. T. Sichani, M. Mirzaei, “Model Predictive Control of Buoy Type Wave Energy Converter”, *Proceedings of the 19th World Congress of The International Federation of Automatic Control*, Cape Town, South Africa, 24-29 Aug, 2014.
- [45] S. Zou, O. Abdelkhalik, R. Robinett, G. Bacelli, D. Wilson, “Optimal control of wave energy converters”, *Renewable Energy*, vol 103(2017), pp. 217-225, 2017.
- [46] Ocean Grazer, “Ocean Grazer”, [Online]. Available: <http://www.oceangrazer.com/prev/technology/ocean-grazer>. [Accessed 11 January, 2018].
- [47] Ocean Grazer, “Images”, [Online]. Available: <http://www.oceangrazer.com/prev/media-m/images>. [Accessed 11 January, 2018].
- [48] L. Gaul, M. Kögl, & M. Wagner, *Boundary Element Methods for Engineers and Scientist: An Introductory Course with Advanced Topics*, 1st. ed. Springer-Verlag Berlin Heidelberg, 2003.
- [49] J. J. Pérez-Gavilán, *Introducción a los Elementos de Frontera*, México, Consejo Nacional de Ciencia y Tecnología, 2006.

- [50] J. Maciejowski, class lecture, topic: “Model Predictive Control - Lecture 1”, Engineering Department, Cambridge University, Cambridge.
- [51] R. C. Dorf & R. H. Bishop, *Modern Control Systems*, 11th. ed. Pearson Prentice Hall, 2008.
- [52] A. P. Engelbrecht, *Computational Intelligence*, 2nd. ed. West Sussex: Wiley, 2007.
- [53] J. L. Crespo, “Redes Neuronales Artificiales”, class notes for MT8004-Artificial Intelligence, Mechatronics Engineering Academic Area, Costa Rica Institute of Technology, First semester, 2017.
- [54] T. Dijkstra, “Maximizing revenue of electricity generated by the Ocean Grazer”, M.Sc. thesis, University of Groningen, The Netherlands, 2016.
- [55] Z. Ying, L. Mengxin & D. Zaili, “Application of BP Artificial Neural Networks in Modelling Energy Obtained by Wave Energy System”, *IEEE International Conference on Intelligent Computing and Intelligent Systems*, 2009.
- [56] E. Anderlini, D. I. M. Forehand, P. Stansell, Q. Xiao, & M. Abusara, “Control of a Point Absorber Using Reinforcement Learning”, *IEEE Transactions on Sustainable Energy*, vol 7-4, pp. 1681-1690, 2016.
- [57] O. Mikuláš, “A framework for nonlinear model predictive control”, M.Sc. thesis, Czech Technical University in Prague, Czech Republic, 2016.
- [58] W. Cummins, ”The impulse response function and ship motions”. David Taylor Model Basin Washington DC, Technical report, 1962.
- [59] A. Babarit & G. Delhommeau, “Theoretical and numerical aspects of the open source BEM solver NEMOHin: Proceedings of the *EWTEC2015 Conf*, 2015.